SIAM J. OPTIM.
Vol. 10, No. 1, pp. 1–21

TWO METHODS FOR SOLVING OPTIMIZATION PROBLEMS ARISING IN ELECTRONIC MEASUREMENTS AND ELECTRICAL ENGINEERING[*]

YA. D. SERGEYEV[†], P. DAPONTE[‡], D. GRIMALDI[§], AND A. MOLINARO[§]

Abstract. In this paper we introduce a common problem in electronic measurements and electrical engineering: finding the first root from the left of an equation in the presence of some initial conditions. We present examples of electrotechnical devices (analog signal filtering), where it is necessary to solve it. Two new methods for solving this problem, based on global optimization ideas, are introduced. The first uses the exact a priori given global Lipschitz constant for the first derivative. The second method adaptively estimates local Lipschitz constants during the search. Both algorithms either find the first root from the left or determine the global minimizers (in the case when the objective function has no roots). Sufficient conditions for convergence of the new methods to the desired solution are established in both cases. The results of numerical experiments for real problems and a set of test functions are also presented.

Key words. global optimization, numerical algorithms, electronic measurements, electrical engineering

AMS subject classifications. 90C30, 94A12

PII. S1052623496312393

1. Introduction.

Let us consider a device whose behavior depends on a characteristic $f(x)$, $x \in [a, b]$; $f(x)$ may be, for instance, an electrical signal and $[a, b]$ a time interval. The device functions correctly while $f(x) > 0$. At the initial moment $x = a$, we have $f(a) > 0$. It is supposed that the function $f(x)$ is multiextremal and the Lipschitz condition is satisfied for its first derivative; that is,

$$|f'(x) - f'(y)| \leq L|x - y|, \quad x, y \in [a, b], \tag{1}$$

where the constant L, $0 < L < \infty$, is the Lipschitz constant. Generally, L is unknown. The problem we deal with in this paper is determining a time interval $[a, x^*]$ where the device works correctly. This problem is equivalent to the problem of finding the root of f that is first from the left, that is, finding

$$x^* = \min\{x : f(x) = 0\}, \tag{2}$$

subject to conditions:

$$x \in [a, b], \qquad f(a) > 0. \tag{3}$$

This problem very often arises in electronic measurements [1], [6], [22] and electrical engineering [3], [5], [11], [12], [13], [14], [17], and in section 2 we present two concrete

[*] Received by the editors October 15, 1996; accepted for publication (in revised form) October 6, 1998; published electronically October 20, 1999. This research was partially supported by Russian Foundation of Fundamental Research grant 95-01-01073.

http://www.siam.org/journals/siopt/10-1/31239.html

[†] ISI-CNR c/o DEIS, Universitá della Calabria, 87036 Rende (CS), Italy, and Nizhni Novgorod State University, pr. Gagarina 23, Nizhni Novgorod, Russia (yaro@si.deis.unical.it).

[‡] Dipartimento di Ingegneria dell'Informazione ed Ingegneria Elettrica, via Ponte Don Melillo 1, 84084 Fisciano (SA), Italy (daponte@nadis.dis.unina.it).

[§] Dipartimento di Elettronica, Informatica e Sistemistica, Universita' della Calabria, 87036 Rende (CS), Italy (grimaldi@ccusc1.unical.it).

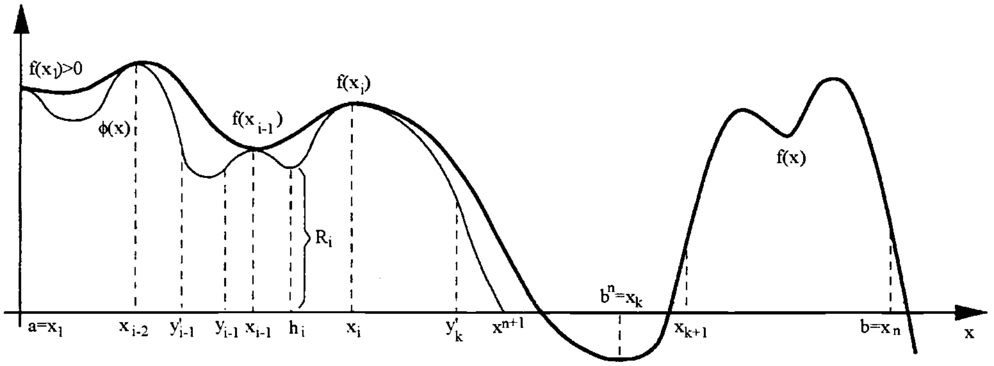


FIG. 1. *Auxiliary support function $\phi(x)$ for the function $f(x)$.*

applications as examples. Usually it is difficult to solve the problem (2), (3) in an analytical way and numerical methods are used to find a σ-approximation x_σ of the point x^* such that

$$0 \le f(x_\sigma), \qquad |x_\sigma - x^*| < \sigma. \tag{4}$$

Two approaches are currently used by engineers to solve the problem (1)–(4). The first one uses standard local techniques for finding equation roots in order to achieve a rapid convergence to the point x^*. The drawback of this method is that convergence is not assured since $f(x)$ is a multiextremal function on $[a, b]$, and it may diverge or converge to a local minimum greater than zero (see [17]). Moreover, if the objective function $f(x)$ has more than one root (and this is usually the case; see Figure 1), different choices of the initial conditions can produce different solutions of the equation $f(x) = 0$.

The second approach is based on the use of any simple grid technique which produces a dense mesh starting from the left margin of the interval and going on by σ until the value $f(x)$ becomes less than zero (see [3]). This approach is very reliable but the number of evaluations of $f(x)$ is too high.

In this paper we propose two new numerical algorithms for solving the problem (1)–(4) in order to find a point x_σ from (4). The methods are based on geometric ideas of the global optimization technique [20]. The new algorithms either find the point x_σ from (4) or determine a σ approximation x'_σ of the global minimizer x' of $f(x)$ and the corresponding value $f(x'_\sigma)$ in the case

$$f(x) > 0, \qquad x \in [a, b]. \tag{5}$$

The first algorithm uses the given constant L from (2). Since in practice it is difficult to know this value a priori, the problem of estimating it in the course of the search arises. The second method presented here estimates the local Lipschitz constants for subintervals of the search region in the course of the search. Using local estimates instead of the global one accelerates the search significantly.

The new techniques are described in section 3. Convergence conditions are established in section 4. Section 5 contains results of numerical experiments executed both with objective functions derived from the applications presented in section 2 and with a series of test functions. Finally, section 6 concludes the paper.

2. Filters as examples of electronic devices where the problem arises. The problem (1)–(4) arises very often in applications (see [1], [3], [5], [6], [11], [12], [13], [14], [17], [22]). In fact, the objective function $f(x)$ can be considered as, for example, a reliability characteristic of a device or a mathematical model. While $f(x) > 0$ the model is reliable, but for $x \geq x^*$ it is not.

Here we consider two concrete examples that illustrate the problem (1)–(4). Both of them deal with electrical filters. *Filters* are basic electronic components utilized in many fields such as power conversion circuits, electronic measurement instruments and communications systems. In particular, *electrical filters* can be found in telephones, televisions, radios, radar, and sonar. A filter is a device that modifies in a predetermined way the input signal that passes through it. Electrical filters may be classified as *analog* filters, used to process analog or continuous-time signals, or *digital* filters, used to process digital signals (discrete-time signals).

Let us consider a signal $s(x)$, where x is time. If a signal $s(x)$, composed of a sum of signals $s_1(x), s_2(x), \ldots, s_n(x)$ so that

$$s(x) = s_1(x) + s_2(x) + \cdots + s_n(x),$$

is the input of an analog filter, the output signal is obtained from the input one by suppressing certain components $s_k(x), k \in \{1, \ldots, n\}$. Let us define for the signal $s(x)$ its *frequency* θ as the number of times that the signal repeats itself in unit time and the *pulse* $\omega = 2\pi\theta$. Below we refer to θ or ω simply as frequency. If a signal $s(x)$ has a certain frequency θ, it may be represented by a rotant vector having angular speed equal to ω and the amplitude equal to the maximum amplitude of $s(x)$. As all the vectors with the same angular speed can be represented in a complex plane (see [5]), since we can represent the signal $s(x)$ in the frequency domain instead of the time domain, we can write $s(x)$ as $s(\omega)$. If $Y(\omega)$ is the filter output and $X(\omega)$ is its input into the frequency domain, the function

$$|H(\omega)| = \frac{|Y(\omega)|}{|X(\omega)|}$$

is called the *transfer function* in the frequency domain (see [11], [12]). From the value of $|H(\omega)|$ we can evaluate the answer of the filter, that is, how far the output signal is from the input signal. The *cutoff frequency* ω_c is defined as the frequency where the transfer function is equal to $\sqrt{0.5}$ of its maximum amplitude. This implies that ω_c can be calculated from the following equation:

$$|H(\omega)| = \sqrt{0.5}H_{\max},$$

or

$$|H(\omega)|^2 = 0.5H_{\max}^2,$$

where

$$H_{\max} = \max\{H(\omega) : \omega \in [0, \infty)\}.$$

The *passband* is the width of the interval $[\omega_{c1}, \omega_{c2}]$ in which

$$|H(\omega)|^2 \geq 0.5H_{\max}^2,$$

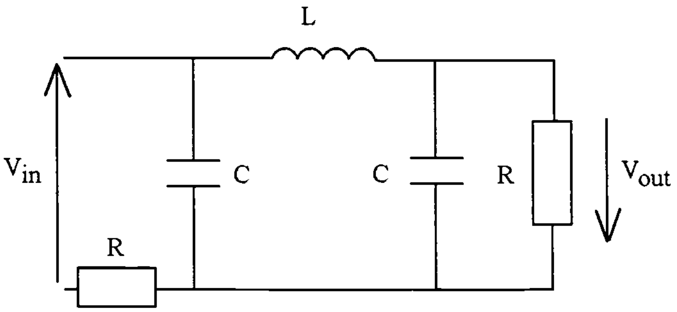


FIG. 2. *A circuit realization of the Chebyshev filter.*

where ω_{c1} is called the *lower cutoff frequency* and ω_{c2} is called the *higher cutoff frequency.* If $\omega_{c1} = 0$ the filter is a *low-pass filter*; if $\omega_{c2} \to \infty$ the filter is a *high-pass filter*; finally, if ω_{c1} and ω_{c2} are finite values, the filter is a *passband filter.* In general, an electrical filter is designed to separate one component of the input signal from the others. If we want to separate a particular frequency ω_p and reject all other frequencies, for technical reasons we cannot build a filter that will pass only ω_p, but a set of frequencies $\omega_p \in [\omega_{c1}, \omega_{c2}]$. As an example, let us consider a radio or television receiver. The transmission station is assigned an interval of frequencies called the *band of frequencies* or *channel frequencies*, in which it must transmit its signal. Ideally, the receiver should accept and process any signal in the assigned channel and completely exclude signals at all other frequencies. Thus the simplest specifications on the magnitude of the transfer function of the receiver are

$$|H(\omega)| = \begin{cases} H_{\max} & \text{for } \omega_{c1} \le \omega \le \omega_{c2}, \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $\omega_{c1} \le \omega \le \omega_{c2}$ is the channel of the signal to be received. However, no circuits can produce such a transfer function exactly. In practice, filters are not required to meet the extremely stringent requirements such as those of (6), and some filters with a transfer function approximating (6) have been found to be consistently satisfactory. A filter that has a uniform approximation property in the passband is the Chebyshev filter (see [11], [12]), which is the first example that will be described in what follows.

An electrical circuit that realizes a Chebyshev filter is shown in Figure 2, where voltage $V_{in}(\omega)$ is the input and voltage $V_{out}(\omega)$ is the output. The transfer function $F(\omega)$, obtained by applying Kirchhoff laws to the circuit of Figure 2, has the following expression:

$$F(\omega) = \left| \frac{V_{out}(\omega)}{V_{in}(\omega)} \right| = \frac{1}{\sqrt{1 + R^2C^2\omega^2}} \cdot \frac{1}{\sqrt{(2 - \omega^2 LC)^2 + \omega^2 L^2/R^2}}, \tag{7}$$

where the values R, C, and L are introduced in Figure 2 and $|\cdot|$ is the length of a complex vector. This function suppresses the frequency components beyond the cutoff frequency ω_c and is characterized by ripples in the passband. The number of maxima and minima in the ripple passband defines the filter order. In our case, the filter order is $n = 3$. The cutoff frequency can be found as the first root from the left for the function

$$f(\omega) = F(\omega)^2 - 0.5F_{\max}^2,$$

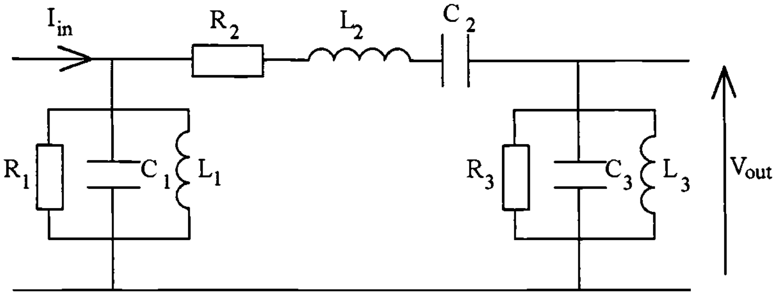


FIG. 3. *A circuit realization of the passband filter.*

where $F_{\max}$ is the maximum of the function

$$F(\omega), \qquad \omega \in (0, \infty).$$

Let us consider the second example. In Figure 3 the electrical circuit of a bandpass filter [11] is shown. The transfer function of this filter is given by

$$F(\omega) = \left| \frac{V_{out}(\omega)}{I_{in}(\omega)} \right| = \frac{\omega L_1 R_1}{\sqrt{(Z_1^2 + Z_2^2)^2 \cdot Z_3}}, \tag{8}$$

where

$$Z_1 = -\omega^3 R_1 L_1 L_2 + \omega R_1 L_2 + \omega R_1 L_1 C_1 / C_2 - R_1/(\omega C_2) + 2\omega L_1 R_1 + \omega L_1 R_2,$$

$$Z_2 = \omega^2 L_1 L_2 + \omega^2 R_1 R_2 L_1 C_1 - R_1 R_2 - L_1/C_2,$$

$$Z_3 = (\omega L_1)^2 + (\omega^2 R_1 L_1 C_1 - R_1)^2.$$

This result can be obtained by applying Kirchhoff laws to the circuit in Figure 3. The transfer function tends to zero for $\omega \to 0$ and $\omega \to \infty$. The cutoff frequency can be found as the first zero crossing for the function

$$f(\omega) = -(F(\omega)^2 - 0.5 F_{\max}^2).$$

The values of the circuit parameters may be changed, thus varying the flatness of the transfer function.

3. Theoretical background and the algorithms. New algorithms presented here for solving the problem (1)–(4) are based on the following idea. Let us suppose that the objective function $f(x)$ and its first derivative $f'(x)$ have already been calculated at n trial points $x^i, 1 \le i \le n$. We can reorder these points by subscripts in such a way that

$$a = x_1 < x_2 < \cdots < x_i < \cdots < x_n = b.$$

Thus, dealing with the trial points, we use two numerations. The record x^i means that this point has been produced during the ith iteration. The record x_i, $i = i(n)$, means that this point has the ith position in the string above during the nth iteration. We designate the results of trials as $z_i = f(x_i)$, $z_i' = f'(x_i)$, $1 \le i \le n$.

For every interval $[x_{i-1}, x_i]$, $1 < i \le n$, we construct an auxiliary function $\phi_i(x)$ in such a way that $\phi_i(x) \le f(x)$, $x \in [x_{i-1}, x_i]$. Knowing the structure of the auxiliary function, we can find the first function $\phi_i(x)$ such that for some $x \in [x_{i-1}, x_i]$ it follows that

$$\phi_i(x) = 0,$$

and we can determine the first root from the left of this equation. Adaptively improving the set of functions $\phi_i(x)$, $1 < i \le n$, by adding new points $x^{n+1}, x^{n+2}, \ldots$ we will improve our approximation of $f(x)$ and of the point x^*. This geometric approach is widely used in global optimization (see [7], [10]), applying functions $\phi_i(x)$ with different structures (see [8], [15], [16] for methods using only the values of objective functions and [2], [4], [9], [20], [24], [25], [26], [27] for algorithms where the first derivatives are also taken into consideration).

In the two algorithms presented here we use the following three ideas to provide a fast localization of the points x_σ from (4):

—using smooth auxiliary functions from [20], where they demonstrated good performance in global optimization;

—constructing auxiliary functions only for intervals $[x_{i-1}, x_i]$, $1 < i \le k$, where

$$k = \min\{\{n\} \cup \{i : f(x_i) < 0, 1 < i \le n\}\}; \tag{9}$$

—adaptively estimating of the *local* Lipschitz constant L_i for every interval $[x_{i-1}, x_i]$ (in the second algorithm).

Let us discuss these ideas one after another. First, there exist three types of support function elaborated to solve global optimization problems: piecewise linear (see [8], [15], [16], [18]), nonsmooth quadratic (see [4], [9], [24], [26]), and smooth quadratic (see [20], [25], [27]). We use the last construction because (see [20], [25], [27]) it best approximates the objective function.

Suppose that we have an estimate m_i of the constant L_i such that

$$m_i \ge L_i. \tag{10}$$

In this case it is possible to construct a support function $\phi_i(x)$ for $f(x)$ over $[x_{i-1}, x_i]$ (see [20]) as follows:

$$\phi_i(x) = \begin{cases} z_{i-1} + z'_{i-1}(x - x_{i-1}) - 0.5m_i(x - x_{i-1})^2, & x \in [x_{i-1}, y'_i], \\ 0.5m_i x^2 + b_i x + c_i, & x \in (y'_i, y_i], \\ z_i - z'_i(x_i - x) - 0.5m_i(x_i - x)^2, & x \in (y_i, x_i], \end{cases} \tag{11}$$

where

$$y_i = \frac{x_i - x_{i-1}}{4} + \frac{z'_i - z'_{i-1}}{4m_i} + \frac{z_{i-1} - z_i + z'_i x_i - z'_{i-1}x_{i-1} + 0.5m_i(x_i^2 - x_{i-1}^2)}{m_i(x_i - x_{i-1}) + z'_i - z'_{i-1}}, \tag{12}$$

$$y'_i = -\frac{x_i - x_{i-1}}{4} - \frac{z'_i - z'_{i-1}}{4m_i} + \frac{z_{i-1} - z_i + z'_i x_i - z'_{i-1}x_{i-1} + 0.5m_i(x_i^2 - x_{i-1}^2)}{m_i(x_i - x_{i-1}) + z'_i - z'_{i-1}}, \tag{13}$$

$$b_i = z'_i - 2m_i y_i + m_i x_i, \tag{14}$$

$$c_i = z_i - z'_i x_i - 0.5m_i x_i^2 + m_i y_i^2. \tag{15}$$

An illustration of the functions $f(x)$, $\phi_i(x)$ is presented in Figure 1. The function $\phi_i(x)$ has been constructed using the Taylor formula for $f(x)$ about the point x_{i-1} (see the first line in (11)) and the point x_i (see the third line in (11)). The second line of (11) has been obtained as the quadratic which agrees with the $f(x)$ curvature at the interval extremes. Note that the first derivative $\phi_i'(x)$, for all $x \in (x_{i-1}, x_i)$ exists.

It will be useful for us to find the point

$$h_i = \operatorname{argmin}\{\phi_i(x) : x \in [x_{i-1}, x_i]\} \tag{16}$$

and the corresponding value

$$R_i = \phi_i(h_i) = \min\{\phi_i(x) : x \in [x_{i-1}, x_i]\}. \tag{17}$$

We will call the value R_i the *characteristic* of the interval $[x_{i-1}, x_i]$. Let us consider two cases. If $\phi_i'(y_i') < 0$ and $\phi_i'(y_i) > 0$, then

$$h_i = \operatorname{argmin}\{f(x_{i-1}), \phi_i(\widehat{x}_i), f(x_i)\}, \tag{18}$$

where

$$\widehat{x}_i = 2y_i - z_i' m_i^{-1} - x_{i-1}. \tag{19}$$

The point $\widehat{x}_i$ is determined from the equation $\phi_i'(x) = 0$, $x \in [y_i', y_i]$. It follows from (11) that

$$\phi_i(\widehat{x}_i) = c_i - 0.5 m_i \widehat{x}_i^2. \tag{20}$$

In the second case there is no point $\widehat{x}_i \in [y_i', y_i]$ such that $\phi_i'(\widehat{x}_i) = 0$ and

$$h_i = \operatorname{argmin}\{f(x_{i-1}), f(x_i)\}. \tag{21}$$

The algorithms work by constructing the function $\phi_i(x)$ from (11) from left to right taking the intervals one after another and calculating their characteristics. If in a step $R_j < 0$ has been found, this means that there exists a point $\tilde{x} \in [x_{j-1}, x_j]$ such that $\phi_j(\tilde{x}) = 0$.

In this case we determine the new trial point $x^{n+1} = \tilde{x}$ (all possible locations of x^{n+1} are shown in Figures 4, 5, 6) and evaluate $f(x^{n+1})$ and $f'(x^{n+1})$. If $f(x^{n+1}) < 0$, then there is no need to consider the interval $(x^{n+1}, b]$ because the solution x_σ is in $(a, x^{n+1}]$ (here a, b are from (3)). Then we increment n and restart the procedure. If (5) takes place, then the algorithm will function to find an approximation x_σ' of the point x'.

The last points we discuss before describing the algorithms are obtaining values m_i such that (10) holds and the influence on the speed (and correctness) of the algorithm. The first method for obtaining the values m_i is to take $m_i = L$, where L is from (1). But in real problems, it is often difficult to know the exact value of L. In this case a fixed estimate of L is taken and used in the course of the search. This strategy is applied in global optimization methods (for different auxiliary functions $\phi_i(x)$) in [2], [4], [15], [27]. We use it in our first algorithm A1.

The second approach to determining the values m_i is to estimate L in the course of the search using information obtained from evaluating $f(x)$, $f'(x)$ at trial points. The way of estimating L is under intensive investigation (see [23]), and many global optimization algorithms do it in different manners (see [8], [9], [16], [21]).

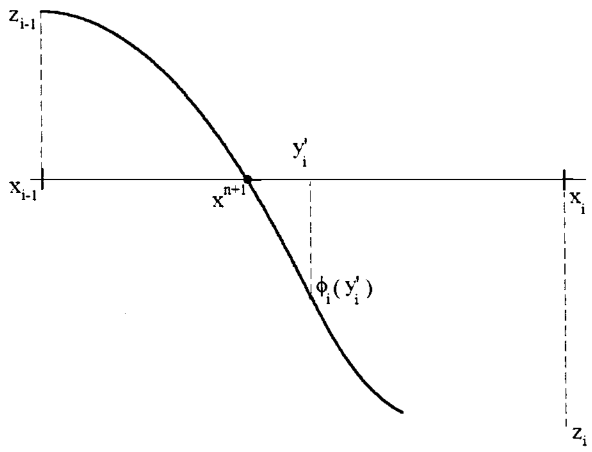


FIG. 4. *Possible locations for the roots of the equation* $\phi_i(x) = 0$*: the case when* x^{n+1} *is calculated by the formula* (29).

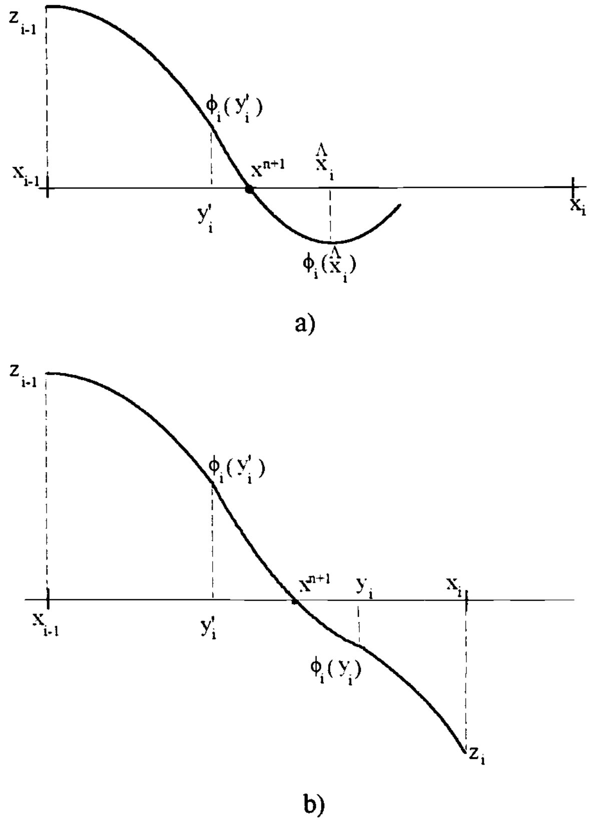


FIG. 5. *Possible locations for the roots of the equation* $\phi_i(x) = 0$*: the case when* x^{n+1} *is calculated by the formula* (30).

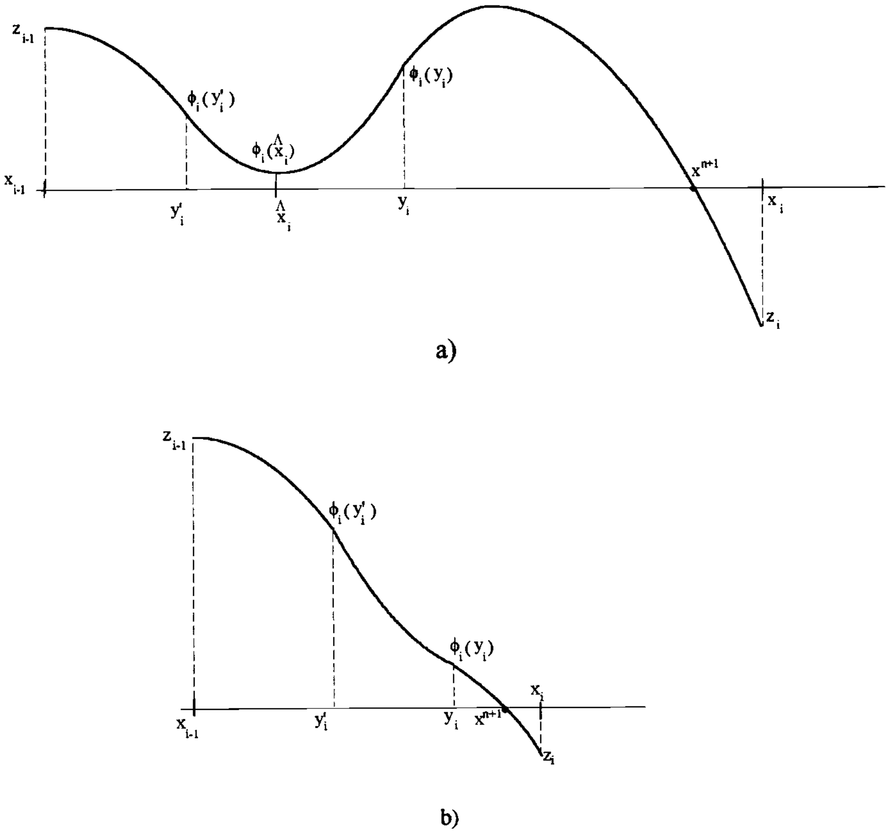


FIG. 6. *Possible locations for the roots of the equation* $\phi_i(x) = 0$: *the case when* x^{n+1} *is calculated by the formula* (31).

The main drawback of both approaches is the following. The global Lipschitz constant L gives very poor information about the behavior of $f(x)$ in every small interval $[x_{i-1}, x_i]$. That is why, in our second algorithm A2, we estimate local Lipschitz constants L_i for every interval $[x_{i-1}, x_i]$, $1 < i \leq k$. This strategy has been successfully applied in global optimization techniques [18], [19], [20], [24], [25], [26] for different classes of problems.

Now we are ready to describe the methods. We present only A2, since the A1 scheme can be obtained easily from it by eliminating Step 2, below, and using $m_i = K$, $L \leq K < \infty$, in all subsequent steps.

Let us suppose that n trials, with $n \geq 2$, of the algorithm have already been carried out at points $x^1, \ldots, x^n$. The $(n+1)$th trial point x^{n+1} is chosen according to the following procedure.

Step 1 (ordering trial points). Among the trial points $x^1, \ldots, x^n$ of the previous n iterations, form the subset $X^{k(n)}$ such that

$$X^{k(n)} = \{x_1, x_2, \ldots, x_k\},$$

where k is as defined in (9). We let $b^n = x_k$, the right margin of the search interval during the nth iteration (see Figure 1).

Step 2 (computing m_i). Calculate estimates m_i for the local Lipschitz constants L_i for the intervals

$$[x_{i-1}, x_i], \qquad 1 < i \leq k,$$

as follows:

$$m_i = r \cdot \max\{\lambda_i, \gamma_i, \xi\}, \tag{22}$$

where $\xi > 0$ and $r > 1$ are parameters of the method.

The values λ_i and γ_i spy on changes of local and global information, respectively, obtained in the course of the search. The value λ_i is calculated as

$$\lambda_i = \max\{v_j : 1 < j \leq k, i-1 \leq j \leq i+1\}, \tag{23}$$

where

$$v_j = \frac{|2(z_{j-1} - z_j) + (z'_j + z'_{j-1})(x_j - x_{j-1})| + d_j}{(x_j - x_{j-1})^2} \tag{24}$$

and

$$d_j = \sqrt{[2(z_{j-1} - z_j) + (z'_j - z'_{j-1})(x_j - x_{j-1})]^2 + (z'_j - z'_{j-1})^2(x_j - x_{j-1})^2}.$$

The second component γ_i from (22) is calculated as

$$\gamma_i = m(x_i - x_{i-1})/X^{\max}, \tag{25}$$

where m estimates the global Lipschitz constant L from (1):

$$m = \max\{v_i : 1 < i \leq k\} \tag{26}$$

and

$$X^{\max} = \max\{x_i - x_{i-1} : 2 \leq i \leq k\}.$$

Step 3 (calculating characteristics R_i of the intervals). Initialize the index sets $I = \emptyset$, $Y = \emptyset$, $Y' = \emptyset$. Set the index of the current interval $i = 2$.

Step 3.0 (organizing the main cycle). If $i > k$, then go to Step 4; otherwise compute the values y_i, y'_i according to (12) and (13). If $\phi'_i(y'_i) \cdot \phi'_i(y_i) < 0$, then go to Step 3.2; otherwise, go to Step 3.1.

Step 3.1 (computing R_i if $\phi'_i(y'_i) \cdot \phi'_i(y_i) \geq 0$). Calculate $R_i = \phi_i(h_i)$, where h_i is from (21). If $h_i = x_i$, then include i in Y; else include i in Y'. Go to Step 3.3.

Step 3.2 (computing R_i if $\phi'_i(y'_i) \cdot \phi'_i(y_i) < 0$). Calculate $R_i = \phi_i(h_i)$, where h_i is from (18). Include i in I. Go to Step 3.3.

Step 3.3 (verifying the sign of R_i). If $R_i \leq 0$, then go to Step 5; otherwise set $i = i + 1$ and go to Step 3.0.

Step 4 (computing the new trial point if $R_j > 0, 1 < j \leq k$). Find an interval i with the minimal characteristic, that is,

$$i = \operatorname{argmin}\{R_j : 1 < j \leq k\}, \tag{27}$$

and define the new trial at the point x^{n+1} as follows:

$$x^{n+1} = \begin{cases} y_i' & \text{if } i \in Y', \\ \widehat{x}_i & \text{if } i \in I, \\ y_i & \text{if } i \in Y. \end{cases} \tag{28}$$

Go to Step 6.

Step 5 (computing the new trial point if $R_i \leq 0$). If $\phi_i(y_i') \leq 0$, then go to Step 5.1. Otherwise go to Step 5.2.

Step 5.1 (choosing the new trial point if $\phi_i(y_i') \leq 0$). Calculate

$$x^{n+1} = x_{i-1} + \frac{1}{m_i}(z_{i-1}' + \sqrt{z_{i-1}'^2 + 2m_i z_{i-1}}), \tag{29}$$

that is, the right root of the equation

$$z_{i-1} + z_{i-1}'(x - x_{i-1}) - 0.5m_i(x - x_{i-1})^2 = 0$$

obtained from the first line of (11) (see Figure 4), and go to Step 6.

Step 5.2 (choosing the new trial point if $\phi_i(y_i') > 0$ and $\phi_i'(y_i')\phi_i'(y_i) < 0$). If $\phi_i'(y_i') \cdot \phi_i'(y_i) \geq 0$, then go to Step 5.3. Otherwise, if $\phi_i(\widehat{x}_i) > 0$, then compute

$$x^{n+1} = x_i + \frac{1}{m_i}(z_i{}' + \sqrt{z_i'^2 + 2m_i z_i}), \tag{30}$$

that is, the right root of the equation

$$z_i - z_i'(x_i - x) - 0.5m_i(x_i - x)^2 = 0$$

obtained from the third line of (11) (see Figure 5(a)) and go to Step 6.

If $\phi_i(\widehat{x}_i) \leq 0$, then x^{n+1} is calculated following the formula (this situation is presented in Figure 6(a))

$$x^{n+1} = \frac{-b_i - \sqrt{b_i^2 - 2m_i c_i}}{m_i} \tag{31}$$

obtained from the second line of (11) as the left root of the equation

$$0.5m_i x^2 + b_i x + c_i = 0;$$

then go to Step 6.

Step 5.3 (choosing the new trial point if $\phi_i'(y_i') \cdot \phi_i'(y_i) \geq 0$). If $\phi_i(y_i) > 0$, then calculate x^{n+1} using (30) (see Figure 5(b)) and go to Step 6. Otherwise use (31) for calculating x^{n+1} (see Figure 6(b)).

Step 6 (the stopping rule). If the stopping rule $|x_i - x_{i-1}| \leq \sigma$, where σ is from (4), is fulfilled, then Stop. Otherwise calculate the value $f(x^{n+1})$ and go to Step 7, setting $b^{n+1} = x^{n+1}$ if $f(x^{n+1}) < 0$.

Step 7 (adjusting the search information). Calculate the value $f'(x^{n+1})$. Set $n = n + 1$ and go to Step 1.

After fulfillment of the stopping rule the following situations can take place:

i. $b^{n+1} \neq b$. This means that we can take $x_\sigma = x_{k-1}$ because this is the last trial point such that $f(x_{k-1}) > 0$.

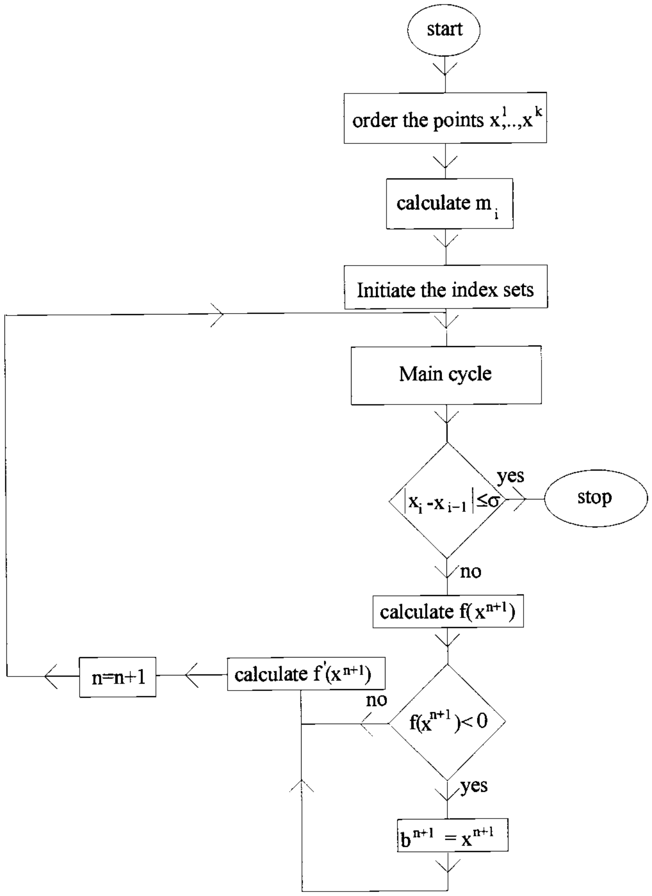


FIG. 7. *Flowchart for the algorithms.*

ii. $b^{n+1} = b$ and $R_i > 0$ for all $i, 1 < i \leq k$. This means that no root has been found in the interval $[a, b]$ and we can continue our investigation taking a new interval $[a^1, b^1]$, where $a^1 = b$. The point

$$x_\sigma^n = \operatorname{argmin}\{f(x_j) : 1 \leq j \leq n\}$$

can be taken as a σ–approximation of the global minimizer x' over $[a, b]$ and the value $f(x_\sigma^n)$ can be used as an estimate of reliability of our device over the interval $[a, b]$.

iii. $b^{n+1} = b$ and there exists an interval j such that its characteristic $R_j \leq 0$. This situation means that it is necessary to take new $\sigma^1 < \sigma$ because the algorithm stops within the interval $[x_{j-1}, x_j]$ with properties $z_{j-1} > 0$, $z_j > 0$, and $R_j \leq 0$ and cannot proceed because $x_{j-1} - x_j \leq \sigma$. For a better understanding of the algorithms' logic we present their flowchart in Figure 7.

Let us say a few words about parameters of the second method. The parameter ξ is a small number reflecting our supposition that $f'(x)$ is not constant over the interval $[x_{i-1}, x_i]$ and $r > 1$ is a reliability parameter of the method. Increasing r means that we suppose that the information obtained during the search is not sufficiently reliable and the objective function behavior is worse than is seen from the

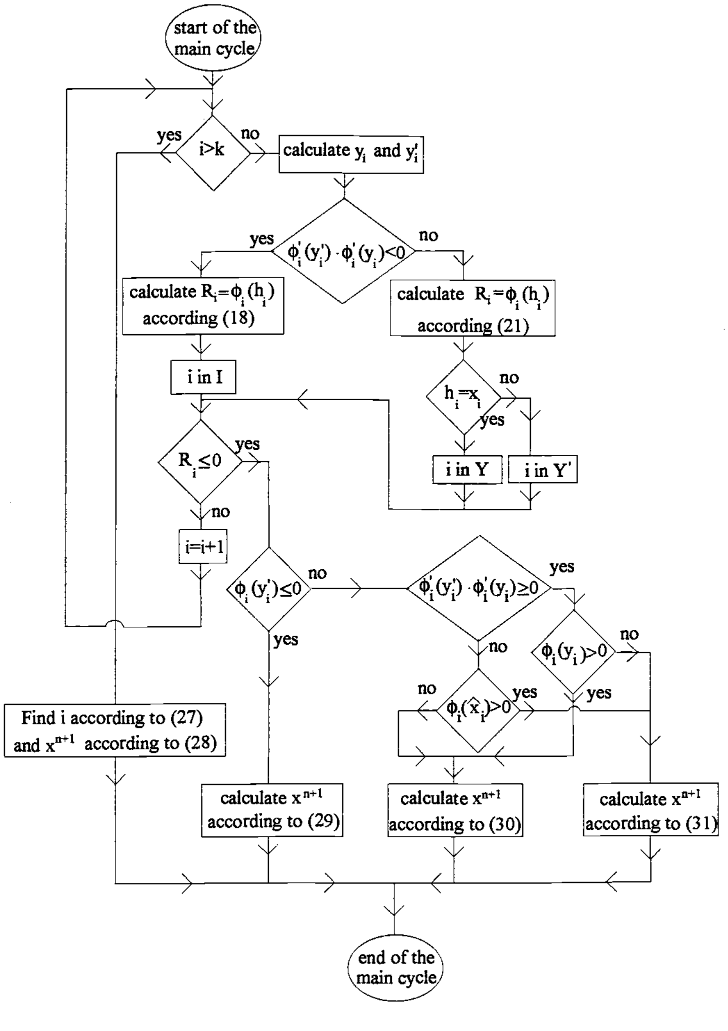

start of the main cycle
i>k
yes
no
calculate y_i and y_i'
$\phi_i'(y_i')\cdot\phi_i'(y_i)<0$
yes
no
calculate $R_i=\phi_i(h_i)$ according (18)
calculate $R_i=\phi_i(h_i)$ according (21)
i in I
$h_i=x_i$
no
yes
i in Y
i in Y'
$R_i\leq 0$
yes
no
i=i+1
$\phi_i(y_i')\leq 0$
no
yes
$\phi_i'(y_i')\cdot\phi_i'(y_i)\geq 0$
yes
no
$\phi_i(y_i)>0$
no
yes
$\phi_i(\hat{x}_i)>0$
no
yes
Find i according to (27) and x^{n+1} according to (28)
calculate x^{n+1} according to (29)
calculate x^{n+1} according to (30)
calculate x^{n+1} according to (31)
end of the main cycle

FIG. 7. *(Cont.).*

search. Our experience shows that taking $r \in [1.2, 2]$ gives good results both in terms of convergence and in terms of speed.

4. Convergence analysis. In this section we demonstrate that the infinite trial sequence $\{x^n\}$ generated by A1 or A2 in the case $\sigma = 0$ has as its limit points (points of accumulation):

—the point x^* from (2) if within $[a, b]$ there exists at least one root of the equation $f(x) = 0$;

—the global minimizer x' if (5) takes place.

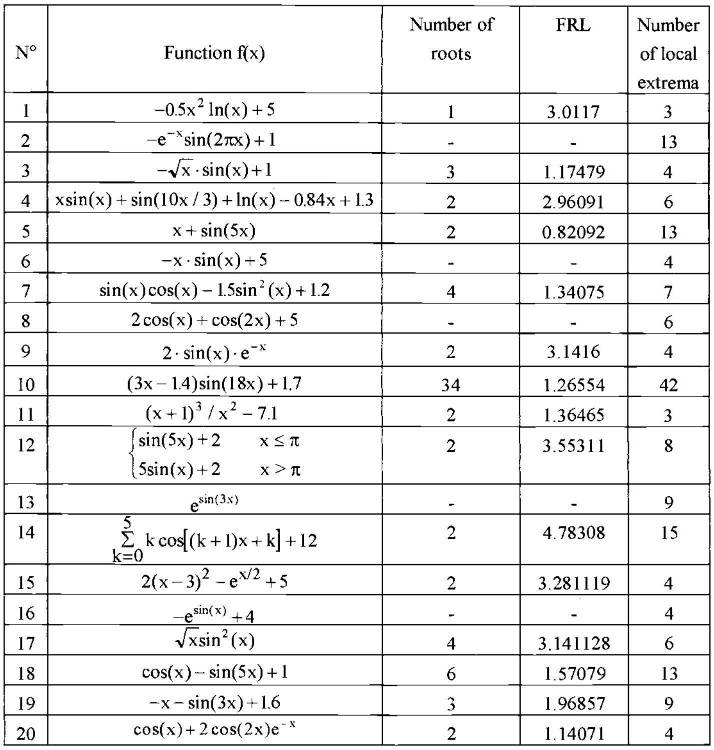
We start by establishing these results for A1.

THEOREM 4.1. *If there exists the point $x^* \in [a, b]$ from* (2), *then x^* will be the unique limit point of the sequence $\{x^n\}$ of trial points generated by* A1.

Proof. Since, due to the A1 scheme for all $n \geq 1$ we use

$$m_{i(n)} = K, \quad L \leq K < \infty, \quad 1 < i \leq n,$$

TABLE 1
Characteristics of the functions utilized for numerical experiments.



N°	Function f(x)	Number of roots	FRL	Number of local extrema
1	$-0.5x^2\ln(x)+5$	1	3.0117	3
2	$-e^{-x}\sin(2\pi x)+1$	-	-	13
3	$-\sqrt{x}\cdot\sin(x)+1$	3	1.17479	4
4	$x\sin(x)+\sin(10x/3)+\ln(x)-0.84x+1.3$	2	2.96091	6
5	$x+\sin(5x)$	2	0.82092	13
6	$-x\cdot\sin(x)+5$	-	-	4
7	$\sin(x)\cos(x)-1.5\sin^2(x)+1.2$	4	1.34075	7
8	$2\cos(x)+\cos(2x)+5$	-	-	6
9	$2\cdot\sin(x)\cdot e^{-x}$	2	3.1416	4
10	$(3x-1.4)\sin(18x)+1.7$	34	1.26554	42
11	$(x+1)^3/x^2-7.1$	2	1.36465	3
12	$\begin{cases}\sin(5x)+2 & x\le\pi\\ 5\sin(x)+2 & x>\pi\end{cases}$	2	3.55311	8
13	$e^{\sin(3x)}$	-	-	9
14	$\sum\limits_{k=0}^{5} k\cos[(k+1)x+k]+12$	2	4.78308	15
15	$2(x-3)^2-e^{x/2}+5$	2	3.281119	4
16	$-e^{\sin(x)}+4$	-	-	4
17	$\sqrt{x}\sin^2(x)$	4	3.141128	6
18	$\cos(x)-\sin(5x)+1$	6	1.57079	13
19	$-x-\sin(3x)+1.6$	3	1.96857	9
20	$\cos(x)+2\cos(2x)e^{-x}$	2	1.14071	4

then the auxiliary functions $\phi_i(x)$ from (11) constructed by the algorithm will be the support ones for all i, $1 < i \le k(n)$, where k is from (9); that is,

$$f(x) \ge \phi_i(x), \quad x \in [x_{i-1}, x_i], \quad 1 < i \le k(n).$$

Denote by $t = t(n)$ the number of an interval $[x_{t-1}, x_t]$ such that

$$x^* \in [x_{t-1}, x_t] \tag{32}$$

in the course of the nth iteration. Due to (16), (17) its characteristic R_t is such that

$$R_t < 0. \tag{33}$$

Since $K < \infty$ there exists an infinite sequence of iteration numbers $\{d\}$ such that

$$R_j > 0, \quad 1 < j < t(d), \quad d \in \{d\}. \tag{34}$$

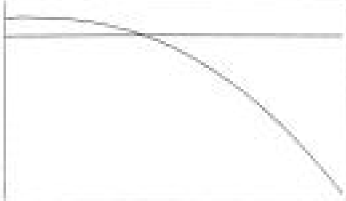
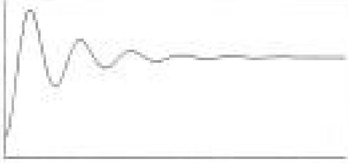
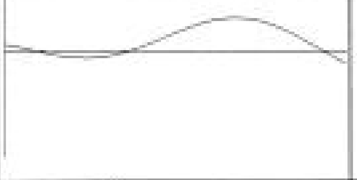
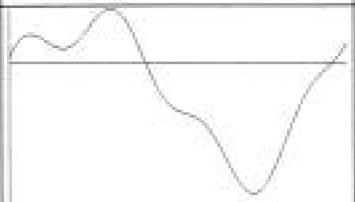
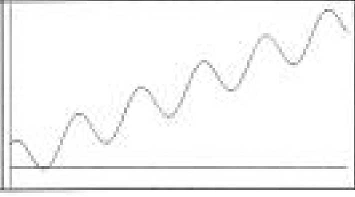
This means that every trial with the number $d+1$, $d \in \{d\}$, will fall in the interval $[x_{t-1}, x_t]$. Using again the inequality $L \le K < \infty$ and (29)–(31) we obtain that

$$\lim_{d\to\infty} x^{d+1} = x^*.$$

To conclude the proof we show that x^* is the unique accumulation point of $\{x^n\}$. The presence of another limit point $\overline{x}$ on the right of x^* is impossible because of Step 1 of A1 and (9). The situation $\overline{x} < x^*$ cannot take place for the following reason.

Table 2

Comparison between grid technique and the methods using smooth auxiliary functions.

N°	Function	Grid	A1	A2
1		4135	5	5
2		10000	31	34
3		1295	6	5
4		4060	12	7
5		5470	7	11

Let $\overline{x} \in [x_{c(n)-1}, x_{c(n)}]$ in the course of the nth iteration. Then, if $\overline{x}$ is a limit point of $\{x^n\}$, the characteristic $R_{c(n)}$ of the interval $[x_{c(n)-1}, x_{c(n)}]$ should be less than 0 infinitely many times, but this is impossible because of (2) and the limitness of K. □

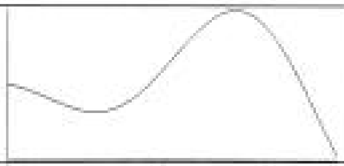
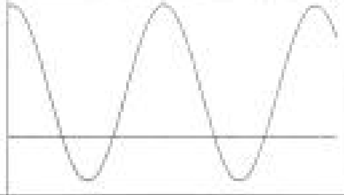
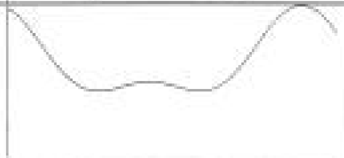
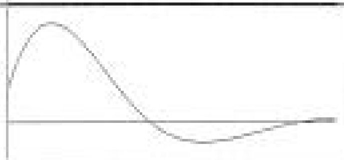
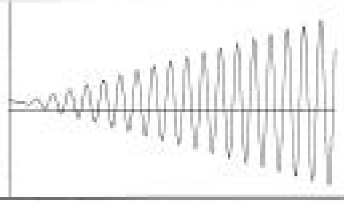
Theorem 4.2. *If* (5) *takes place, then all global minimizers will be limit points of the trial sequence* $\{x^k\}$ *generated by* A1.

Proof. Due to (5) and the fact that the constant $K < \infty$ there is an iteration number p such that

$$R_j > 0, \qquad 1 < j \leq p. \tag{35}$$

This means that for $n > p$ Step 5 will never be executed and A1 functions as the global optimization method 1 from [20], where the corresponding convergence results are given. □

Table 2
(Cont.).

N	Function	Grid	A1	A2
6		10000	10	9
7		1678	5	6
8		10000	36	24
9		4326	15	10
10		1567	55	12

Let us consider now the performance of the algorithm A2. First, note that for a correct functioning of the method it is necessary to choose m_i in accordance with the information obtained from the trials executed at the points x_{i-1}, x_i. If m_i is underestimated it is possible to obtain $y_i, y'_i \notin [x_{i-1}, x_i]$.

Proposition 4.1. *The choice of m_i by the formula* (22) *provides that $y_i, y'_i \in [x_{i-1}, x_i]$.*

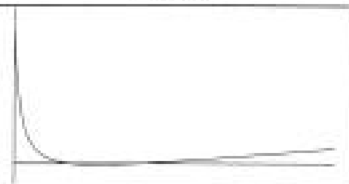
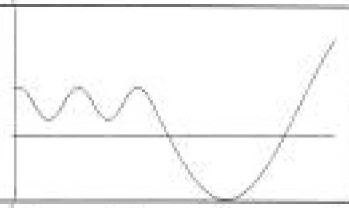
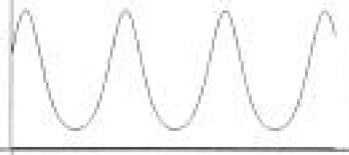
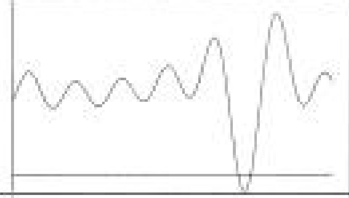
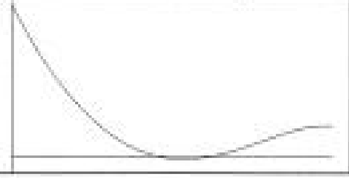
Proof. The accordance of the choice of m_i with the local information is done by the presence of λ_i in (22). This value is determined by (23), (24). A complete discussion of this result can be found in [20]. □

Theorem 4.3. *Let L_t be the local Lipschitz constant of $f(x)$ over the interval $[x_{t-1}, x_t] \ni x^*$, $t = t(n)$, during the nth iteration of* A2. *If there exists an iteration number n^* such that for all $n > n^*$ the inequality*

$$m_t \geq L_t \tag{36}$$

holds, then the point x^* will be the unique limit point of the trial sequence $\{x^n\}$ generated by A2.

Table 2
(Cont.).

N	Function	Grid	A1	A2
11		1713	69	60
12		4931	13	6
13		10000	99	39
14		6740	23	18
15		4531	9	9

Proof. As the values $m_j, 2 \leq j < t$, are bounded (see (22)) thusly,

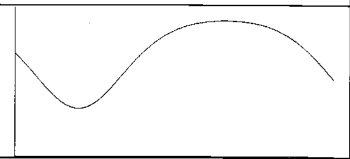
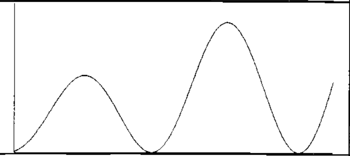
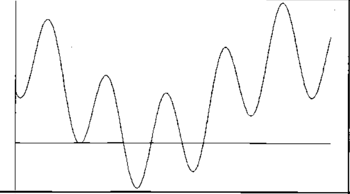
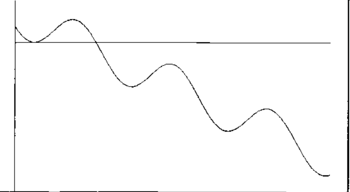
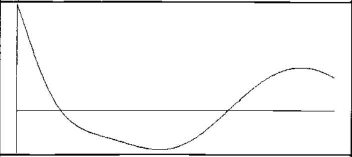
$$r\xi \leq m_j \leq r \cdot \max\{\xi, L\}, \qquad 2 \leq j < t, \tag{37}$$

then there exists an iteration number $\overline{n}$ after which a sequence $\{d\}$ from (34) will exist and (34) will take place. Thus, considering iterations with numbers $n > \{n^*, \overline{n}\}$, we obtain that both (33) and (34) hold, and the theorem is proved following the proof of Theorem 4.1. □

Remark 1. *Note that to have convergence to the point x^* it is not necessary to estimate the global Lipschitz constant correctly over the whole region* $[a, b]$. *It is enough to do it only for the local constant L_i for the subinterval* $[x_{t-1}, x_t]$. *This condition is significantly weaker than the corresponding convergence results for the methods using estimates of Lipschitz constants (see* [8], [9], [16], [21]*).*

Theorem 4.4. *Let* (5) *be true and L_t be the local Lipschitz constant of $f(x)$ over the interval $[x_{t-1}, x_t] \ni x'$, where x' is a global minimizer and there exists a number n' such that* (36) *takes place. Then x' will be the limit point of the trial sequence generated by* A2.

Table 2
(Cont.).

N	Function	Grid	A1	A2
16		10000	7	12
17		4325	20	17
18		2016	11	10
19		2601	12	12
20		7413	6	6
Average		5119.15	22.55	16.17

Proof. It follows from (37) and (5) that there exists a number p from (35) such that (35) holds. From (36) we obtain

$$\phi_t(x) \leq f(x), \qquad x \in [x_{t-1}, x_t].$$

Thus, from the iteration number $\widehat{x} = \max\{p, n'\}$, A2 functions as the global optimization algorithm 3 from [20] and (36) is its sufficient condition of convergence to the point x'. □

5. Numerical experiments. In this section we present the results of numerical experiments carried out in order to demonstrate the performance of the new algorithms and to compare them with the grid technique mainly used by engineers to solve the problem (1)–(4).

In the first series of experiments 20 test functions were chosen over the interval [0.2, 7]. Their analytic expressions and characteristics are given in Table 1. We denote by FRL the first root from the left. The functions are reported in Table 2 and concern general real cases that can be found in many different applications, such as filtering

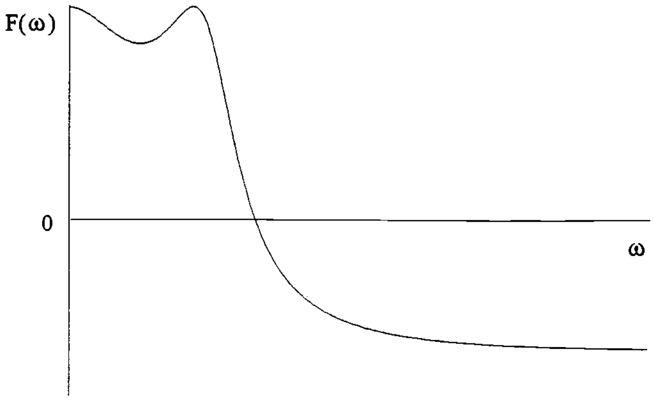


Fig. 8. *Plot of function* (38) *for determining the cutoff frequency for the Chebyshev filter.*

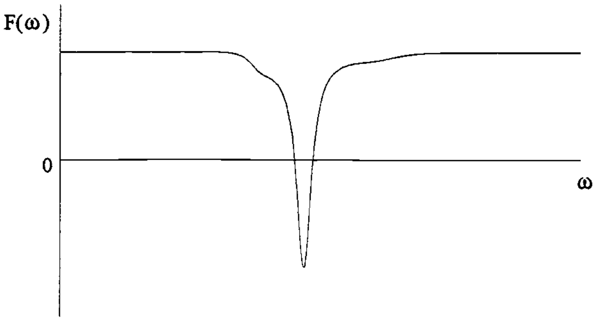


Fig. 9. *Plot of function* (39) *for determining the cutoff frequency for the passband filter.*

and harmonic analysis in electrical or electronic systems, image processing, wavelet theory, and so on (see [1], [3], [5], [6], [11], [12], [13], [14], [17], [22]).

The parameters of the algorithms have been chosen as follows: $\xi = 10^{-6}$, $r = 1.2$ for the algorithm A2 and $\sigma = 10^{-4}(b-a)$ for the algorithms A1 and A2 and for the grid method. We used exact Lipschitz constants for $f'(x)$ in A1 in all the experiments. Table 2 contains the numbers of trials required by A1, A2, and the grid method working with the step σ before satisfaction of the stopping rule.

In the second part of the experiments we solved practical electrotechnical problems by finding the cutoff frequency for the filters presented in section 2. The parameters for the Chebyshev filter were the following: $R = 1\Omega$; $L = 2H$; $C = 4F$. The cutoff frequency was found as the first zero crossing for the function

$$f(\omega) = F(\omega)^2 - 0.5F_{\max}^2 \tag{38}$$

(see Figure 8), where $F(\omega)$ is from (7) and was found in the point $\omega = 0.8459 rad/s$. This result was obtained in 2745 iterations by the grid method, in 11 iterations by the algorithm A1, and in 10 iterations by the algorithm A2.

The second filter is a passband filter (see section 2). The parameters for this filter have been chosen as follows: $R_1 = 3108\Omega$, $L_1 = 40e^{-3}H$, $C_1 = 1e^{-6}F$, $R_2 = 477\Omega$, $L_2 = 350e^{-2}H$, $C_2 = 0.1e^{-6}F$. The cutoff frequency was found as the first zero crossing for the function

$$f(\omega) = -(F(\omega)^2 - 0.5F_{\max}^2) \tag{39}$$

(see Figure 9), where $F(\omega)$ is from (8) and was found in the point $\omega = 4824.43 rad/s$. This result was obtained in 4474 iterations by the grid method, in 44 iterations by the algorithm A1, and in 27 iterations by the algorithm A2.

6. Conclusions. In this paper we have considered the problem of finding the first root from the left of an equation $f(x) = 0$, where $f(x)$ satisfies condition (1). This problem very often arises in practice, and we have presented two applications from signal filtering.

To solve this problem we have proposed two methods. The first one uses the exact a priori given Lipschitz constant L. When L is not known a priori the second method solves the problem using adaptive estimation of the local Lipschitz constant in the course of the search. It uses the obtained estimates in order to accelerate the search.

Numerical experiments executed with real problems and with a set of test functions demonstrate good performance of the new techniques in comparison with the method usually used by engineers. Comparing numerically the first algorithm with the second, we can see that, on the set of functions considered in the experiments, the use of local estimates accelerates the search.

Acknowledgment. We would like to thank the first referee for helpful suggestions.

REFERENCES

[1] G. Antonelli, F. Binasco, G. Danese, and D. Dotti, *Virtually zero cross-talk dual frequency eddy current analyzer based on personal computer*, IEEE Trans. Instrumentation and Measurement, 43 (1994), pp. 463–468.

[2] W. Baritompa, *Accelerations for a variety of global optimization methods*, J. Global Optim., 4 (1994), pp. 37–45.

[3] D. Bedrosian and J. Vlach, *Time domain analysis of network with internally controlled switches*, IEEE Trans. Circuits Syst., CAS-39(3) (1992), pp. 192–212.

[4] L. Breiman and A. Cutler, *A deterministic algorithm for global optimization*, Math. Programming, 58 (1993), pp. 179–199.

[5] L. O. Chua, C. A. Desoer, and E. S. Kuh, *Linear and Non Linear Circuits*, McGraw–Hill, Singapore, 1987.

[6] L. D. Cosart, L. Peregrino, and A. Tambe, *Time domain analysis and its practical application to the measurement of phase noise and jitter*, in Proc., IEEE Instrumentation and Measurement Technology Conference, Brussels, Belgium, IEEE, Piscataway, NJ, 1996, pp. 430–1435.

[7] C. A. Floudas and P. M. Pardalos, *State of the Art in Global Optimization*, Kluwer, Dordrecht, the Netherlands, 1996.

[8] E. A. Galperin, *The alpha algorithm and the application of the cubic algorithm in case of unknown Lipschitz constant*, Comput. Math. Appl., 25 (1993), pp. 71–78.

[9] V. P. Gergel, *A global search algorithm using derivatives*, in Systems Dynamics and Optimization, A. V. Sergievskij, ed., N. Novgorod University Press, Nizhni Novgorod, Russia, 1992, pp. 161–178.

[10] R. Horst and P. M. Pardalos, *Handbook of Global Optimization*, Kluwer, Dordrecht, the Netherlands, 1995.

[11] D. E. Johnson, *Introduction to Filter Theory*, Prentice–Hall, Englewood Cliffs, NJ, 1976.

[12] H. Y.-F. Lam, *Analog and Digital Filters-Design and Realization*, Prentice–Hall, Englewood Cliffs, NJ, 1979.

[13] A. M. Luciano and A. G. M. Strollo, *A fast time-domain algorithm for the simulation of switching power converters*, IEEE Trans. Power Electron., PE–5 (1990), pp. 363–370.

[14] S. Mallat, *Zero-crossing of a wavelet transform*, IEEE Trans. Inform. Theory, 37 (1991), pp. 1019–1033

[15] S. A. Pijavskii, *An algorithm for finding the absolute extremum of a function*, USSR Math. Math. Physics, 12 (1972), pp. 57–67.

[16] J. Pinter, *Global Optimization in Action*, Kluwer, Dordrecht, the Netherlands, 1996.

[17] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge University Press, Cambridge, UK, 1986.

[18] Ya. D. Sergeyev, *A one-dimensional deterministic global minimization algorithm*, Comput. Math. Math. Phys, 35 (1995), pp. 705–717.

[19] Ya. D. Sergeyev, *An information global optimization algorithm with local tuning*, SIAM J. Optim., 5 (1995), pp. 858–870.

[20] Ya. D. Sergeyev, *Global one-dimensional optimization using smooth auxiliary functions*, Math. Programming, 81 (1998), pp. 127–146.

[21] R. G. Strongin, *Numerical Methods on Multiextremal Problems*, Nauka, Moscow, 1978.

[22] P. Turcza, R. Sroka, and T. Zielinski, *Implementation of an analytic signal method of instantaneous phase detection in real-time on digital signal processor*, in Proc., 7th International Symposium, TC-4 IMEKO Modern Electrical and Magnetic Measurement, V. Maasz, D. Mejtmanová, and M. Sedláček, eds., ELEKTRA, Prague, 1995, pp. 482–486.

[23] G. C. Wood and B. P. Zhang, *Estimation of the Lipschitz constant of a function*, J. Global Optim., 8 (1996), pp. 91–103.

[24] Ya. D. Sergeyev, *A global optimization algorithm using derivatives and local tuning*, ISI–CNR Report 1, 1994, Rende, Italy.

[25] Ya. D. Sergeyev, *Global optimization algorithms using smooth auxiliary functions*, ISI–CNR Report 5, 1994, Rende, Italy.

[26] Ya. D. Sergeyev, *A method using local tuning for minimizing functions with Lipschitz derivatives*, in Developments in Global Optimization, E. Bomze, T. Csendes, R. Horst, and P. M. Pardalos, eds., Kluwer, Dordrecht, the Netherlands, 1994, pp. 199–216.

[27] D. MacLagan, T. Sturge, and W. Baritompa, *Equivalent methods for global optimization*, in State of the Art in Global Optimization, C. A. Floudas and P. M. Pardalos, eds., Kluwer, Dordrecht, the Netherlands, 1996, pp. 201–212.

SIAM J. Optim.
Vol. 10, No. 1, pp. 22–42

COMPUTING THE MINIMUM COST PIPE NETWORK INTERCONNECTING ONE SINK AND MANY SOURCES[*]

GUOLIANG XUE[†], THEODORE P. LILLYS[‡], and DAVID E. DOUGHERTY[§]

Abstract. In this paper, we study the problem of computing the minimum cost pipe network interconnecting a given set of wells and a treatment site, where each well has a given capacity and the treatment site has a capacity that is no less than the sum of all the capacities of the wells. This is a generalized Steiner minimum tree problem which has applications in communication networks and in groundwater treatment. We prove that there exists a minimum cost pipe network that is the minimum cost network under a full Steiner topology. For each given full Steiner topology, we can compute all the edge weights in linear time. A powerful interior-point algorithm is then used to find the minimum cost network under this given topology. We also prove a lower bound theorem which enables pruning in a backtrack method that partially enumerates the full Steiner topologies in search for a minimum cost pipe network. A heuristic ordering algorithm is proposed to enhance the performance of the backtrack algorithm. We then define the notion of k-optimality and present an efficient (polynomial time) algorithm for checking 5-optimality. We present a 5-optimal heuristic algorithm for computing good solutions when the problem size is too large for the exact algorithm. Computational results are presented.

Key words. minimum cost pipe network, generalized Steiner minimum tree problem, bounding theorem, backtrack, interior-point methods, k-optimal

AMS subject classifications. 68Q20, 68Q25, 90B10, 90B12

PII. S1052623496313684

1. Introduction.

In the Euclidean Steiner minimum tree (ESMT) problem [21, 13], we are seeking a minimum cost network interconnecting a set of given points on the Euclidean plane, where the cost of a network is measured as the sum of edge lengths. Note that additional points joining the line segments may be added in order to reduce network cost. These added points are called *Steiner points* and the given points are called *regular points*. There is a large literature on the ESMT problem [6, 7, 31]. We refer readers to the survey paper [16] and the book [17].

In this paper, we study a generalization of the ESMT problem, which arises from communication networks [13] and groundwater treatment in civil and environmental engineering. Here we are also given a set of points on the Euclidean plane and are seeking the minimum cost network interconnecting these given points. In this problem, the cost of the network is the sum of the costs of all the edges in the network, but the cost of an edge is defined differently. One of the given points is a *sink* (or *service center*) and the rest of the given points are *sources* (or *clients*). For each source, there is a given positive capacity. The sink has a capacity that is no less than the sum of the capacities of the sources. For a given network, there is a flow on each edge, and the cost of that edge is the product of the length of the edge with the *cost per*

[*] Received by the editors December 11, 1996; accepted for publication (in revised form) September 22, 1998; published electronically October 20, 1999.
http://www.siam.org/journals/siopt/10-1/31368.html

[†] Department of Computer Science, University of Vermont, Burlington, VT 05405 (xue@cs.uvm.edu). This research was supported in part by U.S. Army grant DAAH04-96-1-0233 and by NSF grants ASC-9409285 and OSR-9350540.

[‡] Department of Civil and Environmental Engineering, University of Vermont, Burlington, VT 05405. Current address: HydroGeoLogic, Inc., 1155 Herndon Parkway, Suite 900, Herndon, VA 20170 (tpl@hgl.com). This research was supported in part by NSF grant OSR-9350540.

[§] Department of Civil and Environmental Engineering, University of Vermont, Burlington, VT 05405 (ddougher@emba.uvm.edu). This research was supported in part by NSF grant OSR-9350540.

unit length (CPUL) of that edge, which is a function of the flow of that edge. For example, a larger diameter pipe is needed for an edge with a bigger flow and a smaller diameter pipe is needed for an edge with a smaller flow. In most cases, the CPUL is a monotonically nondecreasing function of the flow, which is zero for zero flow and positive for positive flows. Related problems have been studied [3, 19, 36], where algorithms for computing a suboptimal solution were proposed. In this paper, we present exact and heuristic algorithms for solving this generalized Steiner minimum tree problem. We will use groundwater treatment as a motivating example. For applications in communication networks, we refer the readers to Gilbert [13].

The rest of this paper is organized as follows. In section 2, we present the physical problem and define the mathematical model. The notion of *topology* is also introduced in that section. In section 3, we present the notion of full Steiner topology and prove that there exists a minimum cost pipe network, which is the minimum cost network under a full Steiner topology. In section 4, we show that for a given full Steiner topology, we can compute the CPUL for all the edges in the topology in linear time. Then the minimum cost network under the given topology is solved using a powerful interior-point method [34]. In section 5, we prove a bounding theorem which enables the application of a backtrack algorithm that partially enumerates the full Steiner topologies in search of the minimum cost pipe network. A *max-min* heuristic ordering algorithm is proposed to enhance the performance of the backtrack algorithm. In section 6, we define the notion of k-optimality and present an efficient algorithm for checking 5-optimality. When a network is found to be not 5-optimal, we can easily change the topology to one with a lower cost. This leads to a heuristic algorithm for computing 5-optimal networks when the problem size is too large for the backtrack algorithm. We present some preliminary computational results in section 7 and conclude the paper in section 8.

2. The physical problem and its mathematical model. To treat contaminated groundwater in a given region, several wells are constructed in that region and contaminated water from within these wells is piped to a treatment site. Each well has a known location and a known flow-rate (or capacity). The treatment site has a known location and a known capacity, which is no less than the sum of the capacities of all wells. We need to build a pipe network to transport water from the wells to the treatment site. Given the type of materials to be used, it is desirable to build a pipe network that has the minimum cost. In contrast to many water supply problems, service reliability is not a significant requirement here, so no redundant arcs are needed in this problem. Depending on different flow-rates through different edges of the network, we need to use pipes of different sizes for edges with different flow-rates. For a given type of material, the CPUL of the pipe is a function of the flow-rate of that pipe. In many situations, this function, denoted by $f(\bullet)$, is *monotonically nondecreasing* with the flow-rate and is also *concave* (i.e., $f(\frac{x+y}{2}) \geq \frac{1}{2}(f(x)+f(y))$ for any x, y in the domain of the function [28]). In addition, $f(0) = 0$ and $f(x) > 0$ for any $x \in (0, \infty)$. We will assume this property for $f(\bullet)$ throughout this paper. The cost of an edge in the network is the product of the length of the edge with the CPUL of that edge. The cost of the network is the sum of the costs of all the edges. We are interested in computing a minimum cost pipe network.

In the following, we will present a precise mathematical model for this problem. Let $n \geq 3$ be a given integer. Let $P = \{p_1, p_2, \ldots, p_n\}$ be a set of n points on the Euclidean plane R^2, where p_1 represents the location of the treatment site and $p_2, p_3, \ldots, p_n$ represent the locations of the $n-1$ wells, respectively. Let $c(p_1)$, $c(p_2)$,

$\ldots, c(p_n)$ be n positive numbers such that $c(p_1) \geq \sum_{k=2}^{n} c(p_k)$. One may think of $c(p_1)$ as the capacity of the treatment site and of $c(p_2), c(p_3), \ldots, c(p_n)$ as the capacities of the $n-1$ wells, respectively.

Since we need to transport water from all the wells to the treatment site, the pipe network will form a *connected graph* $G = (V, E)$, where E is the set of edges that correspond to the pipes and $V = \{v_1, \ldots, v_n, v_{n+1}, \ldots, v_{n+m}\}$ is the set of vertices such that $v_1, v_2, \ldots, v_n$ correspond to the points $p_1, p_2, \ldots, p_n$, respectively. The m additional vertices $v_{n+1}, \ldots, v_{n+m}$ correspond to some possible *Steiner points* in the pipe network where two or more pipes meet. For the moment, we assume that m can be any nonnegative integer. In section 3, we will prove that m can be restricted to nonnegative integers less than or equal to $n-2$. The previous discussion leads to the following definition.

DEFINITION 2.1. *A* topology $T(P)$ *for a given point set* $P = \{p_1, p_2, \ldots, p_n\}$ *is an undirected connected graph* $G = (V, E)$, *where* E *is the set of edges and* $V = \{v_1, \ldots, v_n, v_{n+1}, \ldots, v_{n+m}\}$ *is the set of vertices such that* $v_1, v_2, \ldots, v_n$ *correspond to the points* $p_1, p_2, \ldots, p_n$, *respectively, and* $v_{n+1}, \ldots, v_{n+m}$ *correspond to* m *Steiner points. A* realization $R(T, P, S, A, X)$ *of* $T(P)$ *is given by the tuple* (S, A, X), *where* S *is a set of points* $\{p_{n+1}, \ldots, p_{n+m}\} \in R^2$, A *is a set of* arcs *that is a subset of* $\{(i,j), (j,i) | \{i,j\} \in E\}$, *and* X *is a set of* flows $\{x(i,j) \geq 0 | (i,j) \in A\}$ *satisfying the following conditions:*

$$\sum_{(i,1)\in A} x(i,1) - \sum_{(1,j)\in A} x(1,j) \leq c(p_1), \tag{2.1}$$

$$\sum_{(k,j)\in A} x(k,j) - \sum_{(i,k)\in A} x(i,k) = c(p_k), \quad k = 2, 3, \ldots, n, \tag{2.2}$$

$$\sum_{(k,j)\in A} x(k,j) - \sum_{(i,k)\in A} x(i,k) = 0, \quad k = n+1, n+2, \ldots, n+m. \tag{2.3}$$

The cost of realization R *is given by*

$$F(R) = \sum_{(i,j)\in A} f(x(i,j))||p_i - p_j||, \tag{2.4}$$

where $|| \bullet ||$ *stands for the Euclidean norm. The* cost of topology T *is given by*

$$F(T) = \min\{F(R) | R \textit{ is realization of } T\}. \tag{2.5}$$

A realization of a topology for P *is called a* pipe network *interconnecting* P.

Note that in the above definition, the number of vertices in a topology for point set P may be any integer greater than or equal to $|P|$. We assume implicitly that the vertices $v_1, v_2, \ldots, v_n$ correspond to the points $p_1, p_2, \ldots, p_n$. In a realization, the vertices $v_1, \ldots, v_n$ are always fixed at points $p_1, \ldots, p_n$ and the vertices $v_{n+1}, \ldots, v_{n+m}$ are fixed at some points $p_{n+1}, \ldots, p_{n+m}$. A realization also specifies the way the water is transported to the treatment site. Condition (2.1) says that the net flow into the treatment site is at most $c(p_1)$. Condition (2.2) says that the net flow out of the kth well is $c(p_k)$. Condition (2.3) says that the flow into any Steiner point equals the flow out of that Steiner point. For any two vertices i and j, we use $\{i, j\}$ to represent the (undirected) edge interconnecting i and j and use (i, j) to represent the (directed) arc from i to j. For graph notations not defined in this paper, we refer readers to [15].

It is clear that, given a realization R of a topology for a point set, we can obtain another realization R_1 of the same topology for the point set by deleting all the zero flows and arcs with zero flows in R. Realization R_1 has the property that every arc has a positive flow and that the cost of R_1 is less than or equal to that of R. This observation will help us in reducing the number of topologies to be considered.

Now we can state the minimum cost pipe network problem formally as follows.

PROBLEM 2.1. *Given the point set P containing $p_1, p_2, \ldots, p_n \in R^2$, the positive constants $c(p_1), c(p_2), \ldots, c(p_n)$ where $c(p_1) \geq \sum_{k=2}^{n} c(p_k)$, and the function $f(\bullet)$ which is concave, monotonically nondecreasing such that $f(0) = 0$ and $f(x) > 0$ for any $x \in (0, \infty)$, compute a minimum cost pipe network interconnecting P.*

Note that Problem 2.1 becomes the ESMT problem [21] when the cost function $f(\bullet)$ always equals 1. Since the ESMT problem is NP-hard [12], polynomial time algorithms for solving Problem 2.1 do not seem to exist.

3. Full Steiner topologies.

DEFINITION 3.1. *Given a point set P containing n points $\{p_1, \ldots, p_n\}$ on the Euclidean plane, a* Steiner topology *for P is a topology for P that is a tree graph such that every vertex corresponding to a Steiner point has degree* 3.

We will prove that there is a minimum cost pipe network that is a realization of a Steiner topology. The following notation is needed in the proof.

DEFINITION 3.2. *Given a path in a pipe network, the flow of this path is the minimum of the flows on the arcs of the path.*

THEOREM 3.1. *For any given topology $G = (V, E)$ for P, there exists a Steiner topology T for P such that*

1. *T is a subgraph of G;*
2. *The cost of T is no greater than the cost of G.*

Proof. Let R be a minimum cost realization of G. As discussed in section 2, we may assume that every arc of R has a positive flow, without loss of generality. For any edge $\{i, j\}$ of G, at most one of (i, j) and (j, i) may be an arc of R, because otherwise we can reduce the network cost by reducing the flows on (i, j) and (j, i).

Let E' be the set of all edges $\{i, j\}$ such that either (i, j) or (j, i) is an arc in R. Let $G' = (V', E')$ be the subgraph of G induced by E'. It is clear that G' is also a topology for P and that R is a realization of both G and G'. If the arcs of R constitute a directed tree (with v_1 as the unique sink), then G' is a Steiner topology for P and the cost of G' is no greater than the cost of R (which is the cost of G) and the theorem is proved.

In the rest, we will prove that there exists a minimum cost realization of G whose arcs constitute a directed tree (with v_1 as the unique sink).

It follows from the definition of a realization that there is at least one directed path from u to v_1 for any vertex $v_1 \neq u$ of R. If the u–v_1 path is unique for any $u \neq v_1$, the arcs of R constitute a directed tree (with v_1 as the unique sink) and there is nothing that needs to be proved.

Suppose that for some k, there are two paths from v_k to v_1 with positive flows. Let the two paths be

$$\pi_1 = (v_{i_0}, v_{i_1}, v_{i_2}, \ldots, v_{i_{m_1}}) \quad \text{and} \quad \pi_2 = (v_{j_0}, v_{j_1}, v_{j_2}, \ldots, v_{j_{m_2}}),$$

where $i_0 = j_0 = k$ and $i_{m_1} = j_{m_2} = 1$. Let the flow of π_1 be r_1 and the flow of π_2 be r_2. Let $r = \min\{r_1, r_2\}$. Since both r_1 and r_2 are positive, r is also positive. Let the flow on arc $(v_{i_{t-1}}, v_{i_t})$ be $\alpha_t + r$ $(t = 1, 2, \ldots, m_1)$ and the flow on arc $(v_{j_{t-1}}, v_{j_t})$ be $\beta_t + r$ $(t = 1, 2, \ldots, m_2)$. Then the α_t's and the β_t's are all nonnegative. Now consider the function

$$F(z) = \sum_{t=1}^{m_1} ||v_{i_t} - v_{i_{t-1}}|| f(\alpha_t + r - z) + \sum_{t=1}^{m_2} ||v_{j_t} - v_{j_{t-1}}|| f(\beta_t + r + z) \tag{3.1}$$

defined for $z \in [-r, r]$. Since $f(\bullet)$ is a concave function, $F(\bullet)$ is also concave. It follows from the property of concave functions [28] that

$$F(0) \geq \frac{1}{2}(F(-r) + F(r)) \geq \min\{F(-r), F(r)\}. \tag{3.2}$$

However, for any $z \in [-r, r]$, $F(z) - F(0)$ is the increase in the network cost caused by shifting z amount of the flow from π_1 to π_2. Inequality (3.2) says that we can shift r amount of flow from π_1 to π_2 without increasing the cost of the network when $F(r) = \min\{F(-r), F(r)\}$ and shift r amount of flow from π_2 to π_1 without increasing the cost of the network when $F(-r) = \min\{F(-r), F(r)\}$. Whenever we perform such a shift, there is at least one edge of G whose corresponding flow will be changed from a nonzero value to zero. Furthermore, if an edge of G has zero flow before the shift, it will still have zero flow after the shift because we are shifting some flow to a path that has a positive flow. Therefore, after a finite number of such shifts, we can eliminate all the duplicate paths, without increasing the cost of the network. Therefore, there exists a minimum cost pipe network R' such that there is a unique path from v_k to v_1 in N that has a positive flow for every $k = 2, 3, \ldots, n$. □

DEFINITION 3.3. *Given a point set P containing n points $\{p_1, \ldots, p_n\}$ on the Euclidean plane, a* full Steiner topology *for P is an undirected connected graph $G = (V, E)$, where the vertex set V contains $2n - 2$ vertices $\{v_1, \ldots, v_n, v_{n+1}, \ldots, v_{2n-2}\}$ and the edge set E contains exactly $2n - 3$ edges such that the degree of each of the first n vertices is 1 and the degree of each of the last $n - 2$ vertices is 3. In other words, a full Steiner topology is a tree whose leaves correspond to the points in P and whose interior vertices (which correspond to Steiner points) all have degree 3.*

DEFINITION 3.4. *Let T be a Steiner topology. An edge is called a* Steiner-regular edge *if exactly one end point of this edge corresponds to a Steiner point (the other corresponds to a regular point). Let $e = \{u, v\}$ be a Steiner-regular edge where u corresponds to a regular point p; we can* shrink *this edge by deleting the edge e and replacing the two vertices u and v by a new vertex that corresponds to the regular point p. Any vertex other than u and v is connected to this new vertex if and only if it was connected to u or v before the shrinking operation. Notice that a Steiner topology changes to another Steiner topology when a Steiner-regular edge is shrunk, provided that the degree of the resulting new vertex is no more than 3. A Steiner topology T' is called a* degeneracy *of T if T' can be obtained from T using zero or more shrink operations. We use $D(T)$ to denote the set of Steiner topologies that are degeneracies of T.*

In the rest of this section, we will prove that there exists a full Steiner topology that has a realization that is a minimum cost pipe network interconnecting P. In other words, any Steiner topology T is a degeneracy of some full Steiner topology. Therefore, we only need to consider pipe networks that are realizations of full Steiner topologies. This is largely due to the fact that realizations permit all Steiner topologies to be handled *as if* they were full, by allowing zero-length edges. Note that the number of full Steiner topologies for n regular points is $\frac{(2n-4)!}{(n-2)!2^{n-2}}$, while the number of Steiner topologies is $\sum_{s=0}^{n-2} \binom{n}{s+2} \frac{(n+s-2)!}{s!}$. Therefore, the number of Steiner topologies is much larger than the number of full Steiner topologies [17, 32].

THEOREM 3.2. *There is a minimum cost pipe network interconnecting a point set P that is a realization of a full Steiner topology for P.*

Proof. From Theorem 3.1, we know that there exists a minimum cost pipe network interconnecting P that is a realization of a Steiner topology T. Assume that $P = \{p_1, p_2, \ldots, p_n\}$ and that the vertices of the topology are $\{v_1, v_2, \ldots, v_n,$

$v_{n+1}, \dots, v_{n+m}\}$, where the first n vertices correspond to the points $\{p_1, p_2, \dots, p_n\}$. The rest of the proof is composed of four parts.

1. *v_i $(i = 1, 2, \dots, n)$ is a leaf vertex in T.* Let v_i be the interior vertex with the smallest index. If $i > n$, there is nothing to be proved. Now assume that $i \leq n$. Let $b_1, b_2, \dots, b_k$ be all the vertices that are adjacent to v_i in T. We can change the name of vertex v_i to a new name v_{n+m+1}, then add a new vertex v_i and an edge $\{v_i, v_{n+m+1}\}$ to obtain a new Steiner topology T_1. We note that for any realization of T, there is a realization of T_1 with the same cost (obtained by forcing $p_{n+m+1} = p_i$). Repeating this process, we can obtain a Steiner topology in which the first n vertices are all leaf vertices.

2. *v_j $(j > n)$ is an interior vertex in T.* In any realization, the sum of flows into a Steiner point equals the sum of flows out of that Steiner point. Therefore, having a leaf Steiner vertex does not reduce the cost of the minimum cost realization of the topology. Hence, if v_j is a leaf vertex in T and $j > n$, we may delete v_j from T to obtain a new topology whose cost is no greater than the cost of T.

3. *Degree-2 interior vertices can be removed without increasing the cost.* This follows from the fact that the shortest connection between points is the straight line segment connecting them.

4. *A degree-k interior vertex can be split into $k - 2$ degree-3 interior vertices without increasing the network cost $(k \geq 4)$.* Let k be an integer greater than or equal to 4. Let a be an interior vertex of degree k. Assume that the neighbors of a are $b_1, b_2, \dots, b_k$. We may split the vertex a into a_1 and a_2, which are connected by an edge $\{a_1, a_2\}$, and replace the edges $\{a, b_1\}$ and $\{a, b_2\}$ by $\{a_1, b_1\}$ and $\{a_1, b_2\}$, replace the edges $\{a, b_j\}$ by $\{a_2, b_j\}$ for $j = 3, \dots, k$. The resulting topology has a cost that is no greater than the cost of the previous topology. Repeating the above process, we can obtain a topology in which every interior vertex has degree exactly 3. This completes the proof of part 4 and also the proof of the theorem. $\square$

4. Minimum cost network under a given topology.

In this section, we show that the minimum cost pipe network under a given full Steiner topology can be computed efficiently. We will first show that the flows in the minimum cost realization of a given full Steiner topology can be computed in $O(n)$ time, without knowing the optimal locations of the Steiner points. We then show that computing the minimum cost network under a full Steiner topology is a special case of the well-studied problem of *minimizing a sum of Euclidean norms* [1, 2, 4, 5, 8, 24, 9, 34]. In [34], Xue and Ye presented a primal-dual interior-point algorithm for computing an ϵ-optimal solution [22] to the problem of minimizing a sum of Euclidean norms and proved that their algorithm requires $O(n^{1.5}(\log(n) + \log(\frac{\bar{c}}{\epsilon})))$ arithmetic operations if the problem has a tree structure, where $\bar{c}$ is a constant dependent on the input. We will show that this algorithm is also a polynomial time approximation scheme (PTAS) [18] for computing the minimum cost network under a given full Steiner topology that computes a $(1+\epsilon)$-approximation in $O(n^{1.5}(\log(n) + \log(\frac{1}{\epsilon})))$ time.

4.1. Computing the flows of the minimum cost network.

Suppose that we are given a full Steiner topology. We may consider the tree to be rooted at v_1, which corresponds to the treatment site p_1. The root vertex has one child. Every other interior vertex has exactly two children. The leaf vertices $v_2, v_3, \dots, v_n$ correspond to the wells $p_2, p_3, \dots, p_n$. Clearly, in a minimum cost realization of the given topology, the flow from vertex v_k to its parent vertex must be $c(p_k)$ for $k = 2, 3, \dots, n$. The flow from any interior vertex to its parent vertex must equal the sum of the flows into this vertex from its two children. Therefore, we can use a dynamic programming algorithm to compute the flows on all edges in linear time.

4.2. Approximating the optimal locations of the Steiner points. Once the flows of the minimum cost network under a given Steiner topology are computed, the problem of finding the optimal locations of the Steiner points becomes a special case of the following problem of minimizing a sum of Euclidean norms [24].

PROBLEM 4.1. *Let $c_1, c_2, \ldots, c_M \in R^2$ be column vectors in the Euclidean 2-space and $A_1, A_2, \ldots, A_M \in R^{N\times 2}$ be N-by-2 matrices each having full column rank. Find a point $u \in R^N$ such that the following sum of Euclidean norms is minimized:*

$$\begin{array}{ll} \min & \sum_{i=1}^{M} ||c_i - A_i^T u|| \\ \text{s.t.} & u \in R^N. \end{array} \tag{4.1}$$

This problem has been studied by Calamai and Conn [4, 5] and Overton [24], where second-order methods were proposed to solve the problem. Recently, Andersen [1], Conn and Overton [8], and Andersen and Christiansen [2] proposed computationally effective interior-point algorithms for solving (4.1). Xue and Ye [34] also proposed an interior-point algorithm for solving this problem and proved that their algorithm produces an ϵ-optimal solution in polynomial time.

DEFINITION 4.1. *Consider a minimization problem. Let ϵ be a positive number. A $(1+\epsilon)$-approximation is a feasible solution whose corresponding objective function value is no more than the* product *of the optimal objective function value and $(1+\epsilon)$. An ϵ-optimal solution is a feasible solution whose corresponding objective function value is no more than the* sum *of the optimal objective function value and ϵ.*

The notion of $(1+\epsilon)$-approximations can be found in [18], and the notion of ϵ-optimal solutions can be found in [22, 23, 34, 35].

In [34], a primal-dual interior-point algorithm was presented that computes an ϵ-optimal solution to problem (4.1) in polynomial time. They also proved that when the instance of problem (4.1) is obtained from a Euclidean multifacility location problem with a tree structure (as in the case of computing the minimum cost pipe network under a given tree topology), the total number of arithmetic operations required is $O(N^{1.5}(\log(N) + \log(\frac{\overline{c}}{\epsilon})))$, where $\overline{c} = \max_{1\leq i\leq M} ||c_i||$. We will show that this same algorithm is a PTAS that computes a $(1+\epsilon)$-approximation to the minimum cost network under a given full Steiner topology using $O(n^{1.5}(\log(n) + \log(\frac{1}{\epsilon})))$ arithmetic operations.

Let $x(v_i, v_j)$ be the flow on arc (v_i, v_j), so that $f(x(v_i, v_j))$ is the CPUL for the pipe interconnecting p_i and p_j. Therefore, optimal locations for the Steiner points can be determined by solving the following optimization problem:

$$\min \mathcal{F}(p_{n+1}, p_{n+2}, \ldots, p_{n+n-2}) = \sum_{v_j \text{ is the parent of } v_i} f(x(v_i, v_j))||p_i - p_j||. \tag{4.2}$$

The function in (4.2) is a nonsmooth, continuous, convex function. It is a special case of (4.1), where $N = 2n-4$, $M = 2n-3$; $c_1 = f(x(\text{child}(v_1), v_1))p_1$; $c_i = f(x(v_i, \text{parent}(v_i)))p_i$ for $i = 2, 3, \ldots, n$; and $c_i = 0$ for $i = n+1, n+2, \ldots, n+n-2$. The parent and child notations are those used in section 4.1.

Since problem (2.1) does not change if we *translate* the locations of the treatment site and all the wells by the same vector, we may assume, without loss of generality, that p_1 is at the origin, i.e., $p_1 = 0$. For $i = 2, 3, \ldots, n$, $||c_i|| = f(x(v_i, \text{parent}(v_i)))||p_i||$ is the cost of transporting the water from p_i to the treatment site via a straight line segment between p_i and p_1. Therefore, $\max_{1\leq i\leq n+n-2} ||c_i||$ is less than or equal to

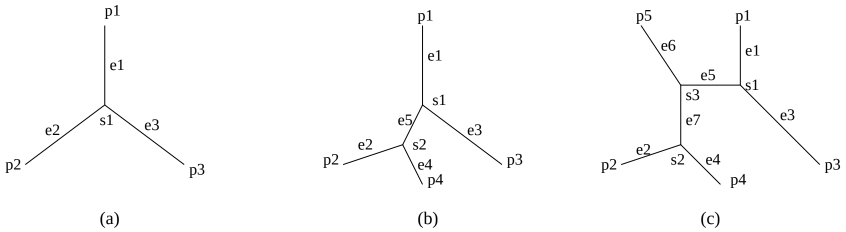


FIG. 1. *The correspondence between $n-3$ vectors and full Steiner topologies.*

the cost of the minimum cost network under the given topology (note that $p_1 = 0$ and $c_i = 0$, $i = n+1, n+2, \ldots, n+n-2$). Therefore, a $\bar{c}\epsilon$-optimal solution is also a $(1+\epsilon)$-approximation for the minimum cost network under a given full Steiner topology. To summarize, we have proved the following theorem.

THEOREM 4.1. *For any given full Steiner topology, a $(1+\epsilon)$-approximation to the minimum cost pipe network under this topology can be computed in $O(n^{1.5}(\log(n) + \log(\frac{1}{\epsilon})))$ time.* □

5. Partially enumerating the topologies. In [29], Smith proved the following theorem, which establishes a one-to-one correspondence between the set of full Steiner topologies on n regular points and a special set of $(n-3)$-element vectors.

THEOREM 5.1. *There is a one-to-one correspondence between full Steiner topologies on $n \geq 3$ fixed points, and $(n-3)$-element vectors $t = (t_1, t_2, \ldots, t_{n-3})$, whose ith entry t_i is an integer in the range $1 \leq t_i \leq 2i+1$. Therefore, the number of full Steiner topologies on n fixed points is $1 \cdot 3 \cdots (2n-5)$.* □

Figure 1 illustrates the one-to-one correspondence for the case of $n = 5$ and $t = (2, 5)$. Figure 1(a) illustrates the topology for three fixed points where the three edges are labeled e_1, e_2, and e_3. The only moving point is labeled s_1. To add the fourth fixed point into the network, we connect p_4 to an interior point on the edge labeled e_2 (since $t_{4-3} = 2$ in the topological vector). This point then becomes the second moving point s_2. Edge e_2 is broken into two parts, one part still labeled e_2 and the other labeled e_5 $(= 2 \times 4 - 3)$. The edge interconnecting p_4 and s_2 is labeled e_4 $(= 2 \times 4 - 4)$. After the above process, we obtain the topology for the first four fixed points, which is illustrated in Figure 1(b). Similarly, Figure 1(c) illustrates the topology for the first five fixed points, which is obtained by breaking the edge labeled e_5 $(= t_{5-3})$.

A brute-force algorithm for solving problem (2.1) is to compute the minimum cost network under a full Steiner topology for every full Steiner topology interconnecting the n fixed points. Since there are $1 \times 3 \times \cdots \times (2n-5)$ different full Steiner topologies for a set of n fixed points, a complete enumeration method is very expensive, even with the aid of the efficient algorithm of [34].

One way to tackle this problem is to use a backtrack algorithm that enumerates only part of the topologies. We need a bounding theorem that can be used to prune hopeless branches. Such a bounding theorem will be proved in the next subsection.

5.1. A bounding theorem.

THEOREM 5.2. *For any $k = 3, 4, \ldots, n-1$ and any $t_{k-2} \in \{1, 2, \ldots, 2k-3\}$, the cost of a full Steiner topology (interconnecting the points $p_1, p_2, \ldots, p_k$) with the topological vector $(t_1, t_2, \ldots, t_{k-3})$ is no greater than the cost of the full Steiner topology (interconnecting the points $p_1, p_2, \ldots, p_{k+1}$) with the topological vector $(t_1, t_2, \ldots, t_{k-3},$*

t_{k-2}). *Note that we assumed that the topological vector for $p_1, p_2, \ldots, p_k$ is a prefix of the topological vector for $p_1, p_2, \ldots, p_{k+1}$.*

Proof. This is a generalization of Theorem 4 in [29], which deals with flow-independent minimum cost networks. Suppose we have found a minimum cost realization of the topology for $p_1, p_2, \ldots, p_{k+1}$. Delete the leaf vertex corresponding to p_{k+1} from the rooted tree. This will reduce the CPUL for all the arcs along the path from the parent vertex of p_{k+1} to the root p_1, due to the assumption that $f(\bullet)$ is a monotonically nondecreasing function. In addition, the parent vertex of p_{k+1} can now be removed (since it is now a degree-2 interior vertex) to shorten the network. Optimizing the modified network will further reduce the cost. □

5.2. The backtrack algorithm. A backtrack algorithm for computing the minimum cost pipe network is given as Algorithm 1. It follows from Theorem 5.2 that Algorithm 1 correctly computes the minimum cost feasible network. However, for the algorithm to be practically efficient, we need to have a good initial upper bound and a good ordering of the regular points so that the backtrack algorithm will generate only a small portion of the whole tree.

Algorithm 1. Partially enumerating the topologies by backtracking.

```
Step_1 Compute an upper bound $UB$ or set $UB := \infty$. Set $k := 4$ and $t_1 := 3$.
Step_2 Compute the minimum cost network under the topology with topological
       vector $(t_1, t_2, \ldots, t_{k-3})$. Let $C$ be the cost of the current network.
Step_3 if $C \geq UB$ then goto Step_4 else goto Step_5 endif
Step_4 if $t_{k-3} > 1$ then
           $t_{k-3} := t_{k-3} - 1$
           goto Step_2
       elseif $k = 4$ then
           stop ; $UB$ is the minimum cost and $(bt_1, bt_2, \ldots, bt_{n-3})$ is the optimal
           topological vector.
       else
           $k := k - 1$
           goto Step_4
       endif
Step_5 if $k = n$ then
           Set $UB := C$ and save the current topological vector in
           $(bt_1, bt_2, \ldots, bt_{n-3})$.
           goto Step_4.
       else
           $k := k + 1$
           $t_{k-3} := 2k + 1$
           goto Step_2
       endif
```

5.3. Initial upper bound. In order for the backtrack algorithm to be effective, we also need a good initial upper bound. In the flow-independent case, one can always use the cost of the minimum spanning tree as the initial upper bound. In the flow-dependent case, even such an initial upper bound is not available. In this section, we present a min-min heuristic algorithm for computing the initial upper bound. The min-min heuristic builds up a tree network in the following way. Initially, only the treatment site p_1 is on the tree. The cost of this tree is zero. For each point $p \in P$, we

Algorithm 2. The min-min heuristic algorithm for initial upper bound.

```
Step_1 Let $q_1 := p_1$ and $c(q_1) := c(p_1)$. Let $Q := \{q_1\}$. Delete $p_1$ from $P$.
Step_2 for each $p \in P$ do
           compute $w(p) = c(p)||q_1 - p||$
       endfor
       Let $q_2 := p_j$ and $c(q_2) := c(p_j)$, where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\min\{w(p)|p \in P\}$. Add $q_2$ to $Q$. Delete $p_j$ from $P$.
Step_3 for each $p \in P$ do
           compute the minimum cost network interconnecting $q_1$, $q_2$, and $p$.
           Let $w(p)$ be the minimum cost.
       endfor
       Let $q_3 := p_j$ and $c(q_3) := c(p_j)$, where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\min\{w(p)|p \in P\}$. Add $q_3$ to $Q$. Delete $p_j$ from $P$.
       Let $T_3$ be the empty topological vector for $Q$.
Step_4 Let $k := |Q|$.
       for each $p \in P$ do
           Let $q_{k+1} := p$ and $c(q_{k+1}) := c(p)$.
           for $i := 1$ to $2k - 3$ do
               Let $f(p, i)$ be the cost of the topology $(T_k, i)$ interconnecting
               $Q \cup \{q_{k+1}\}$.
           endfor
           Let $w(p) := \min\{f(p, i)|1 \leq i \leq 2k - 3\}$.
       endfor
       Let $q_{k+1} := p_j$ and $c(q_{k+1}) := c(p_j)$, where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\min\{w(p)|p \in P\}$. Add $q_{k+1}$ to $Q$. Delete $p_j$ from $P$.
       Let $T_{k+1}$ be the topological vector for $Q$ which has a cost of $f(q_{k+1}, i) :=$
       $w(q_{k+1})$ which is generated in the (appropriate) inner for loop.
Step_5 if $P = \emptyset$ then stop ; otherwise goto Step_4.
```

can add p to the current tree by directly interconnecting p with p_1. The associated (minimum) cost of adding this point to the current tree is given by $c(p)||p_1 - p||$. We select the point whose associated cost is minimum and add it to the current tree. This point is then deleted from P. Suppose that the current tree has k vertices and we are trying to add another point from P to the current tree, which has $2k - 3$ edges. Let p be a point in P. To connect p to the current tree, we may select an edge in the current tree and connect p to this edge (creating a new Steiner point). There are $2k-3$ such choices. Each such choice corresponds to a full Steiner topology interconnecting the regular points in the current tree and the point p. For each of these full Steiner topologies, there is an associated cost of the topology. We define the minimum of the costs of these topologies as the cost of point p. In the min-min heuristic, the point with minimum cost is selected to be the next point to be added to the current tree. This process continues until we have a tree interconnecting all the regular points.

Our min-min heuristic can be considered as a generalization of Prim's algorithm [26] for computing a minimum spanning tree; i.e., we are always selecting the next vertex whose addition to the tree will result in the smallest additional cost. Our motivation, however, comes from the incremental optimization heuristic of Dreyer and Overton [9].

The heuristic of Dreyer and Overton requires $O(n^2)$ local minimizations (solving an instance of Problem 4.1). Our min-min heuristic is more expensive: it requires $O(n^3)$ local minimizations. To see why this is so, we note that there are $2k - 3$

edges in the tree interconnecting k regular points and that there are $n-k$ regular points left to be selected to join the tree. Therefore, we need to perform $(n-k) \times (2k-3)$ local minimizations to select the $(k+1)$th point to join the tree. Therefore, the min-min heuristic requires $O(n^3)$ local minimizations. This time complexity is affordable, compared with the exponential time required by the backtrack algorithm. Our limited computational experiments show that this heuristic produces better initial upper bounds than using a random topology or the max-min heuristic to be discussed in the next section.

5.4. A heuristic ordering. In the backtrack algorithm, a branch is cut off only if its associated cost is no less than the cost of the current incumbent. If we use the ordering determined by the min-min heuristic, a cut-off is not likely to happen high in the search tree, because the cost of the current tree could be relatively small and a small mistake may not lead to a cost greater than the cost of the current incumbent.

ALGORITHM 3. The max-min heuristic ordering algorithm.

```
Step_1 Let $q_1 := p_1$ and $c(q_1) := c(p_1)$. Let $Q := \{q_1\}$. Delete $p_1$ from $P$.
Step_2 for each $p \in P$ do compute $w(p) = c(p)||q_1 - p||$ endfor
       Let $q_2 := p_j$ and $c(q_2) := c(p_j)$ where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\max\{w(p)|p \in P\}$. Add $q_2$ to $Q$. Delete $p_j$ from $P$.
Step_3 for each $p \in P$ do
           compute the minimum cost network interconnecting $q_1$, $q_2$, and $p$.
           Let $w(p)$ be the minimum cost.
       endfor
       Let $q_3 := p_j$ and $c(q_3) := c(p_j)$, where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\max\{w(p)|p \in P\}$. Add $q_3$ to $Q$. Delete $p_j$ from $P$.
       Let $T_3$ be the empty topological vector for $Q$.
Step_4 Let $k := |Q|$.
       for each $p \in P$ do
           Let $q_{k+1} := p$ and $c(q_{k+1}) := c(p)$.
           for $i := 1$ to $2k-3$ do
               Let $f(p,i)$ be the cost of the topology $(T_k, i)$ interconnecting
               $Q \cup \{q_{k+1}\}$.
           endfor
           Let $w(p) := \min\{f(p,i)|1 \le i \le 2k-3\}$.
       endfor
       Let $q_{k+1} := p_j$ and $c(q_{k+1}) := c(p_j)$, where $p_j$ is a point in $P$ and $w(p_j) :=$
       $\max\{w(p)|p \in P\}$. Add $q_{k+1}$ to $Q$. Delete $p_j$ from $P$.
       Let $T_{k+1}$ be the topological vector for $Q$ which has a cost of $f(q_{k+1}, i) :=$
       $w(q_{k+1})$ which is generated in the (appropriate) inner for loop.
Step_5 if $P = \emptyset$ then stop ; otherwise goto Step_4.
```

In this section, we present a max-min heuristic ordering such that cut-offs are likely to happen high in the search tree. In the max-min heuristic, we also start with the treatment site. For each point p in P, the associated cost is defined in the same way as in the min-min heuristic. However, we select the point that has the *maximum* cost as the next point to be included in the tree. This heuristic ordering algorithm is given as Algorithm 3.

The max-min heuristic algorithm also requires $O(n^3)$ local minimizations. Our computational results show that the initial upper bound produced by the max-min heuristic is usually not as good as that produced by the min-min heuristic. However,

Table 1
Number of full Steiner topologies $T(n)$ as a function of the number of regular points n.

n	$T(n)$	n	$T(n)$
3	1	8	10395
4	3	9	135135
5	15	10	2027025
6	105	11	34459425
7	945	12	654729075

the max-min ordering reduces dramatically the number of topologies to be considered in the backtrack algorithm.

6. A 5-optimal heuristic algorithm.

Because the minimum cost pipe network problem is NP-hard, the exact algorithm described in the previous section is too expensive to be practical when n is large. For n regular points, the number of different full Steiner topologies is $1 \times 3 \times 5 \times \cdots \times (2n-5)$. These values for $3 \leq n \leq 12$ are illustrated in Table 1. Although we can compute the minimum cost network under a given full Steiner topology very efficiently, the backtrack algorithm is impractical for large n, limited by the huge number of topologies.

In this section, we introduce the notion of k-optimality of a pipe network and present efficient algorithms for checking whether a given pipe network is 5-optimal. In case the given pipe network is not k-optimal, our algorithm also provides a pipe network that has a lower cost. Our notion of k-optimality for pipe networks is motivated by the notion of k-optimality for ESMTs discussed in a private communication of Overton and Xue [25] and the ideas in [11].

Definition 6.1. *Let T be a full Steiner topology interconnecting the points in P. Let k be an integer such that $3 \leq k \leq n$. A size-k component (C) of T is a subtree of T which has k leaf vertices (in C) and no degree-2 vertices (in C).*

Definition 6.2. *Let T be a full Steiner topology interconnecting the points in P. Let R be the minimum cost network under topology T. A size-k component C of T defines a minimum cost pipe network problem with k regular points, where the point in R that corresponds to the unique vertex in C that is the ancestor of all the other vertices in C is the* new treatment site *and every other point in R that corresponds to a leaf vertex of C is a* new well *whose* new capacity *is defined to be the amount of flow that needs to be transmitted from its corresponding vertex to v_1 in T. T is said to be a k-optimal topology if for any size-k component C of T, its corresponding leaf vertices are connected optimally in the minimum cost realization R of T.*

Note that, given a full Steiner topology, the optimal flows (flows in the minimum cost network under the given topology) on the edges can be computed without knowing the locations of the Steiner points. Therefore, given locations of the leaf vertices of a component C in R, we can formulate another minimum cost pipe network problem where the leaf vertices of C are treated as regular points located at the corresponding locations given in R. In order to check for k-optimality, we need to be able to select all the size-k components and to be able to find the minimum cost pipe network for k given points efficiently. In the following, we illustrate how to check for 5-optimality efficiently.

Figure 2 illustrates a size-5 component. If the vertices a, b, c, d, e are treated as regular points, then the vertices x, y, z are the Steiner points. Note that such a size-5 component is uniquely determined by the *center vertex y* and the vertex c. For any Steiner point y, there are at most three choices of vertex c (since the degree of y is

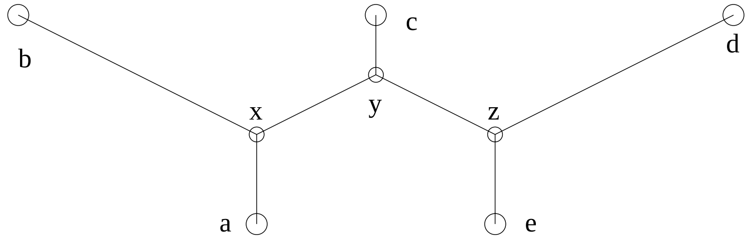


FIG. 2. *A size-5 component.*

3). Once y and c are chosen, we can find the *set* of vertices $\{a, b, d, e\}$ if the other two neighbor vertices of y are both Steiner points. Since there are $n-2$ Steiner points in a full Steiner topology interconnecting n regular points, there are at most $n-2$ choices for y. Therefore, there are at most $3n-6$ size-5 components in a full Steiner topology interconnecting n regular points. Since there are fifteen possible full Steiner topologies interconnecting five regular points, we need to solve at most $45n-90$ minimum cost networks under a full Steiner topology interconnecting five points. Therefore, we have proved the following theorem.

THEOREM 6.1. *Let R be the minimum cost realization of a full Steiner topology T interconnecting P. We can check that T is 5-optimal or find a topology whose cost is lower than the cost of T after solving at most $45n-90$ minimum cost networks under a full Steiner topology interconnecting five points.* □

Now we can present our heuristic algorithm for computing a 5-optimal pipe network interconnecting n regular points. This algorithm is presented as Algorithm 4.

ALGORITHM 4. A 5-optimal heuristic algorithm.

```
Step_1 Run either the min-min heuristic or the max-min heuristic to obtain a topol-
       ogy $T$ and its minimum cost realization $R$.
Step_2 **if** running out of time **then**
           Output the current topology $T$ and its realization $R$; **stop**
       **endif**
Step_3 **for** each size-5 component $C$ of $T$ **do**
           **if** $C$ is not optimally connected in $T$ **do**
               Change topology $T$ by replacing the connections of $C$ by
               its optimal connection;
               Compute the realization $R$ of this new topology $T$.
               **goto** Step_2.
           **endif**
       **endif**
Step_4 The topology $T$ is 5-optimal. Output $T$ and its realization $R$.
```

We do not know the time complexity of Algorithm 4 because we do not know how many changes of topology are needed to reach a 5-optimal topology. All we know is the following. Step_1 requires $O(n^3)$ local minimizations. After that, we need to perform at most $45n-90$ local minimizations (for five-point instances) to either find a *better* topology or confirm that the current topology is 5-optimal. Our computational results show that this algorithm finds an optimal solution or a good suboptimal solution in much less time, compared with the backtrack algorithm.

7. Computational results. In this section, we present preliminary computational results for both the backtrack algorithm and the 5-optimal heuristic algorithm.

Table 2
Constants used in the computation of the CPUL.

ϵ	g	S_f	ν	α	β	γ
0.000015	32.2	0.003	0.0000166	4.7156213	40.406146	1.0727788

Table 3
The eight regular points in test problem 1.

i	$x(i)$	$y(i)$	Capacity	i	$x(i)$	$y(i)$	Capacity
01	1647846.62	432550.91		05	1648180.00	433210.00	0.0278
02	1646685.00	432430.00	0.1114	06	1648440.00	433470.00	0.0278
03	1647540.00	433210.00	0.0278	07	1648960.00	432950.00	0.0278
04	1647545.00	432690.00	0.1114	08	1649220.00	431650.00	0.0836

Both algorithms were implemented in F77 and run on a 100-MHz SGI Indy workstation with MIPS R4000 CPU and 1-MB secondary cache. In all cases, the duality gap tolerance was chosen as 0.00001.

In our first three test problems, we have used data from an application to a groundwater remediation problem at the Lawrence Livermore National Laboratory in Livermore, CA. The locations of the wells were the result of a particular management policy imposed upon an optimization method presented by Rizzo and Dougherty [27]. Treatment sites were located as the weighted center of each of three well fields. To further test the performances of the backtrack algorithm and the 5-optimal heuristic algorithm, we combine the well fields of test problems 2 and 3 to form a field of 18 wells. We then apply the algorithms to the first 15 and first 16 regular points in this list. The fifth test problem was randomly generated to test the 5-optimal heuristic algorithm. For all but one of the cases in the first four test problems, the 5-optimal heuristic that starts with the min-min heuristic found the optimal solution. The 5-optimal heuristic that starts with the max-min heuristic found the optimal solution in all of the first four test problems. Based on our computational result, we estimate that the run time of the backtrack algorithm on the fifth problem is about 800 years of CPU time on our SGI workstation. Therefore, we did not apply that backtrack algorithm on this problem. As a result, we do not know if the result of the 5-optimal heuristic algorithm is optimal or suboptimal.

For all test problems, the CPUL is computed in the following way [30]. For a given flow-rate q, the diameter d of the pipe is computed by

$$d(q) = 0.66 \left[\epsilon^{1.25} \left(\frac{q^2}{gS_f} \right)^{4.75} + \nu q^{9.4} \left(\frac{1}{gS_f} \right)^{5.2} \right]^{0.04}, \tag{7.1}$$

and the CPUL is $f(q)$, which is defined as

$$f(q) = \alpha + \beta d(q)^{\gamma}, \tag{7.2}$$

where the constants are given in Table 2.

Those constants are determined by the energy gradient, the material properties of the pipe and fluid, the cost of digging a trench, etc. For more details on those engineering properties, we refer readers to Lillys [20].

The input data for test problem 1 are given in Table 3. For each regular point, the table shows its index, its x-coordinate, its y-coordinate, and its capacity. Note that the capacity entry for the first regular point is empty. This means that the first regular point is the treatment site, whose capacity can be computed as the sum of the capacities of the other regular points. We will use the same format for all the other problems.

Table 4
Test results for test problem 1.

	Initial bound	Final func	Time (sec)	Search perc
Original ordering	87015.910155	73314.693982	230.56	0.10E+01
max-min ordering	84202.713809	73314.693982	21.57	0.84E-01
min-min 5-opt	84235.450263	73314.693982	2.81	
max-min 5-opt	84202.713809	73314.693982	3.70	

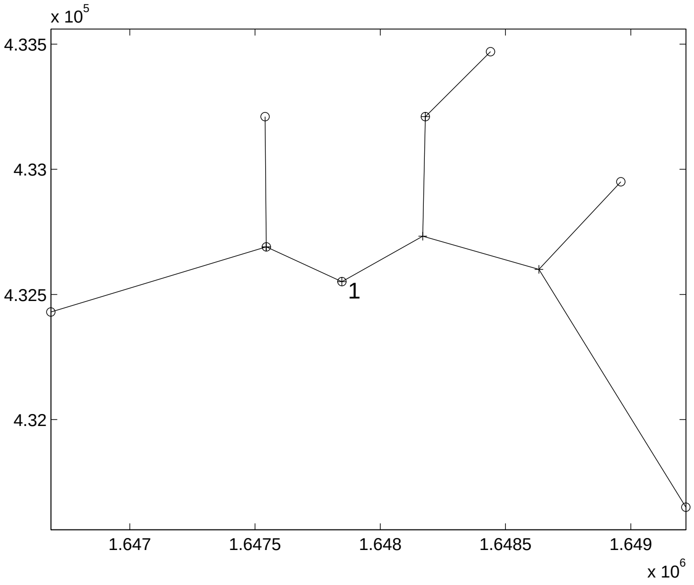


Fig. 3. *The optimal pipe network for test problem* 1.

Computational results for the first test problem are presented in Table 4. The first row of the table reports the result of the backtrack algorithm without any ordering heuristic, i.e., using the ordering given in the input. The initial upper bound 87015.910155 is obtained from the topological vector whose ith entry is $2i + 1$ ($i = 1, 2, \ldots, n - 3$). The final cost 73314.693982 (which is the optimal cost in this case) can be found in the column "Final func." The run time of the algorithm is 230.56 seconds. The last column shows the percentage of the total number of topologies searched. In this case, we have searched all the different topologies. The second row of the table reports the result of the backtrack algorithm with the max-min heuristic. Note that the number of topologies searched is only 8.4% of the total number of topologies. As a result, the run time required is only 21.57 seconds. The third row of the table reports the result of the 5-optimal heuristic algorithm. The last entry is left empty because it is hard to compare the minimization of a size-5 component with the minimization of a size-n component. For this problem, the 5-optimal heuristic algorithm using the min-min heuristic found the optimal solution in 2.81 seconds. The 5-optimal heuristic algorithm using the max-min heuristic found the optimal solution in 3.70 seconds. The minimum cost pipe network for this problem is illustrated in Figure 3, where a regular point is denoted by an o and a Steiner point is denoted by a +. To keep the picture clean, we only labeled the first regular point by the label 1 to the right of the regular point.

Table 5
The nine regular points in test problem 2.

i	$x(i)$	$y(i)$	Capacity	i	$x(i)$	$y(i)$	Capacity
01	1652303.48	434129.13		06	1651820.00	434250.00	0.0334
02	1652340.00	433730.00	0.0278	07	1652860.00	434250.00	0.0278
03	1652860.00	433730.00	0.0278	08	1652080.00	434510.00	0.0334
04	1651560.00	433990.00	0.0334	09	1652600.00	434510.00	0.0278
05	1652600.00	433990.00	0.0278				

Table 6
Test results for test problem 2.

	Initial bound	Final func	Time (sec)	Search perc
Original ordering	46876.355882	36139.255833	485.59	0.16E+00
max-min ordering	42078.166784	36139.255833	121.63	0.44E-01
min-min 5-opt	42115.487371	36649.223822	5.39	
max-min 5-opt	42078.166784	36139.255833	3.68	

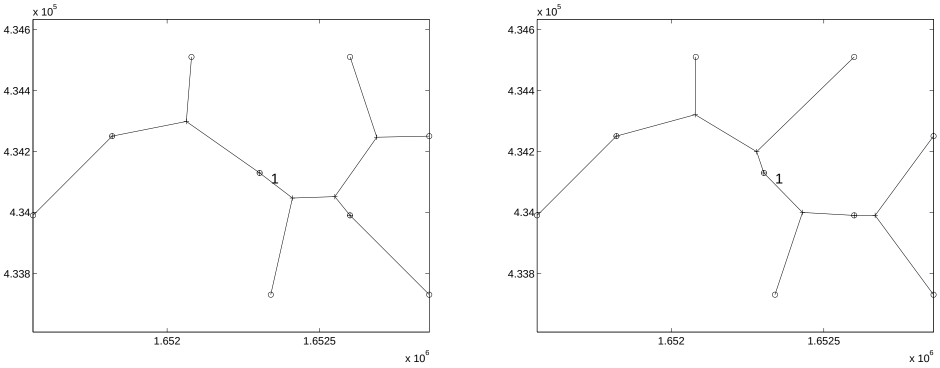


Fig. 4. *The optimal (left) and suboptimal (right) pipe network for test problem* 2.

The input data for test problem 2 are given in Table 5. The computational result for this problem is given in Table 6. For this problem, the backtrack algorithm without heuristic ordering found the optimal solution in 485.59 seconds, searching 16% of the total number of topologies. The backtrack algorithm with the max-min heuristic ordering found the optimal solution in 121.63 seconds, searching 4.4% of the total number of topologies. The 5-optimal heuristic algorithm using the min-min heuristic spent 5.39 seconds finding a suboptimal solution whose cost is 1.014 times the optimal cost. The 5-optimal heuristic algorithm using the max-min heuristic spent 3.68 seconds. It successfully found the optimal solution.

The minimum cost pipe network (left) and the suboptimal pipe network (right) for test problem 2 are illustrated in Figure 4. Note that the 5-optimal heuristic algorithm using the min-min heuristic failed to connect the two regular points at the top-right corner with a Steiner point. This is as expected, because the 5-optimal heuristic only checks for size-5 components. A k-optimal heuristic algorithm with a larger k might produce better solutions, at the cost of longer run time.

Table 7 illustrates the input data and the computational result for test problem 3. For this problem, the backtrack algorithm without heuristic ordering found the optimal solution in 29309.47 seconds, searching 3.2% of the total number of topolo-

Table 7
The input data (top) and test results (bottom) for test problem 3.

i	$x(i)$	$y(i)$	Capacity	i	$x(i)$	$y(i)$	Capacity
01	1652204.14	432529.22		07	1652860.00	432690.00	0.0334
02	1652340.00	431650.00	0.0334	08	1652600.00	432950.00	0.0334
03	1652860.00	431650.00	0.0334	09	1651820.00	433210.00	0.0334
04	1651040.00	431910.00	0.0500	10	1652340.00	433210.00	0.0334
05	1651560.00	432430.00	0.0334	11	1652860.00	433210.00	0.0334
06	1652340.00	432690.00	0.0334				

	Initial bound	Final func	Time (sec)	Search perc
Original ordering	88021.645072	66484.340380	29309.47	0.32E-01
max-min ordering	86697.786286	66484.340380	340.99	0.41E-03
min-min 5-opt	79166.346702	66484.340380	9.72	
max-min 5-opt	86697.786286	66484.340380	9.39	

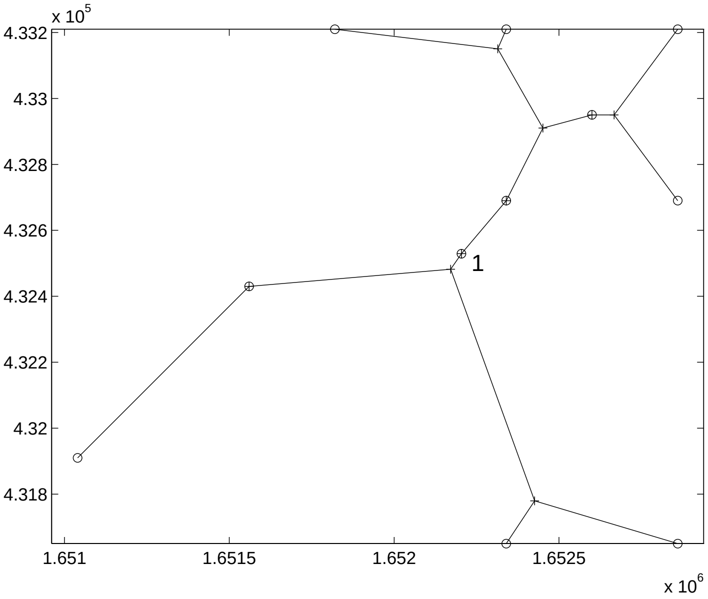


Fig. 5. *The optimal pipe network for test problem* 3.

gies. The backtrack algorithm with the max-min heuristic ordering found the optimal solution in 340.99 seconds, searching only 0.041% of the total number of topologies. The 5-optimal heuristic algorithm using the min-min heuristic found the optimal solution in 9.72 seconds. The 5-optimal heuristic algorithm using the max-min heuristic found the optimal solution in 9.39 seconds. Figure 5 illustrates the minimum cost pipe network for this problem.

The regular points and associated capacities for test problem 4 are given in Table 8. We have applied the algorithms to the first 15 regular points and the first 16 regular points, respectively. Table 9 illustrates the computational results. In both cases, the 5-optimal heuristic algorithm found the optimal solutions within a minute. The backtrack algorithm spent 3.36 hours for the 15-point case and 7.82 hours for the 16-point case. This shows that the run time of the backtrack algorithm is doubled with an additional regular point. Therefore, to apply the backtrack algorithm to the

Table 8
The 16 *regular points in test problem* 4.

i	$x(i)$	$y(i)$	Capacity	i	$x(i)$	$y(i)$	Capacity
01	1652296.67	433253.33		09	1652340.00	432690.00	0.0334
02	1651040.00	431910.00	0.0500	10	1652340.00	433210.00	0.0334
03	1651560.00	432430.00	0.0334	11	1652340.00	433730.00	0.0278
04	1651560.00	433990.00	0.0334	12	1652600.00	432950.00	0.0334
05	1651820.00	433210.00	0.0334	13	1652600.00	433990.00	0.0278
06	1651820.00	434250.00	0.0334	14	1652600.00	434510.00	0.0278
07	1652080.00	434510.00	0.0334	15	1652860.00	431650.00	0.0334
08	1652340.00	431650.00	0.0334	16	1652860.00	432690.00	0.0334

Table 9
Test results for test problem 4.

$n = 15$	Initial bound	Final func	Time (sec)	Search perc
max-min ordering	157470.028030	98133.591436	12127.08	0.50E-07
min-min 5-opt	147002.048885	98133.591436	31.96	
max-min 5-opt	157470.028030	98133.591436	36.15	
$n = 16$	Initial bound	Final func	Time (sec)	Search perc
max-min ordering	166726.146322	103061.764655	28141.68	0.41E-08
min-min 5-opt	157363.223440	103061.764655	43.00	
max-min 5-opt	166726.146322	103061.764655	47.37	

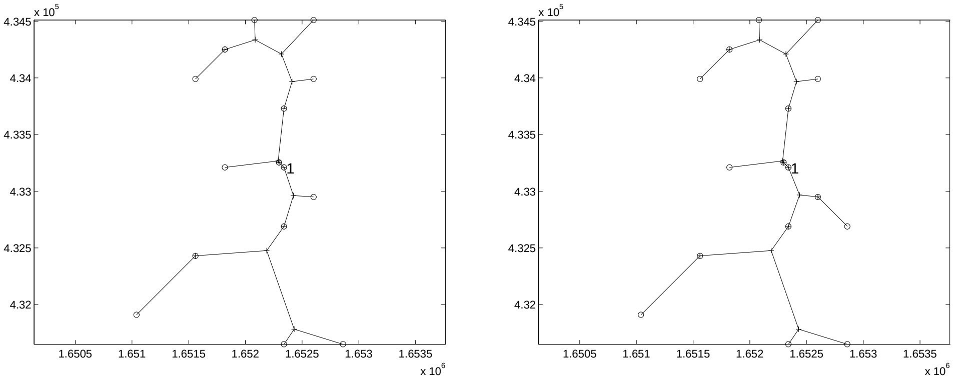


Fig. 6. *The optimal pipe networks for* 15 *points (left) and* 16 *points (right).*

36-point case is not realistic with the current computing power. On one hand, our computational results show that 17 or 18 regular points are about the limit for computing an optimal solution using the exact algorithm in one workstation CPU day. On the other hand, these results show that the 5-optimal heuristic is very practical. The optimal pipe network for both cases in test problem 4 are illustrated in Figure 6.

In test problem 5, we used 36 regular points which are selected from a huge set of grid points [20]. The coordinates and capacities of the regular points are illustrated in Table 10. For this problem, we applied the 5-optimal heuristic algorithm with the min-min heuristic and produced a network with a cost of 693567.504664. The 5-optimal heuristic found a network whose cost is 416116.527856, after making 84 changes in the topology. Note that this is a 40% reduction in the initial cost. The total run time is 1042.43 seconds. We suspect that the result is suboptimal but did not verify it because the estimated run time of the exact algorithm is about 800 years on a workstation. The resulting network is illustrated in Figure 7.

TABLE 10
The 36 *regular points in test problem* 5.

i	$x(i)$	$y(i)$	Capacity	i	$x(i)$	$y(i)$	Capacity
01	1647400	432950		19	1651040	433730	0.032801
02	1651820	432170	0.071895	20	1649220	433990	0.050551
03	1651560	433210	0.036717	21	1652860	431650	0.070822
04	1648960	434250	0.105160	22	1650520	433210	0.128993
05	1648960	433990	0.121502	23	1651040	431910	0.187886
06	1647400	433730	0.119121	24	1648700	434510	0.189312
07	1647140	431910	0.064512	25	1652080	432170	0.144190
08	1651040	434250	0.011117	26	1648960	434510	0.045043
09	1647400	432170	0.066358	27	1646880	431650	0.038324
10	1650000	433730	0.115583	28	1651820	432950	0.136873
11	1649220	434250	0.069011	29	1646880	431910	0.127853
12	1649480	433990	0.198850	30	1651820	434250	0.138132
13	1648700	433210	0.166491	31	1650520	434510	0.009700
14	1649480	432430	0.156950	32	1652600	433470	0.080098
15	1652340	433210	0.063542	33	1648440	432950	0.085227
16	1647400	432430	0.040603	34	1648180	432170	0.038722
17	1651300	433210	0.035788	35	1648700	433990	0.130234
18	1649220	432170	0.034979	36	1651820	433990	0.174764

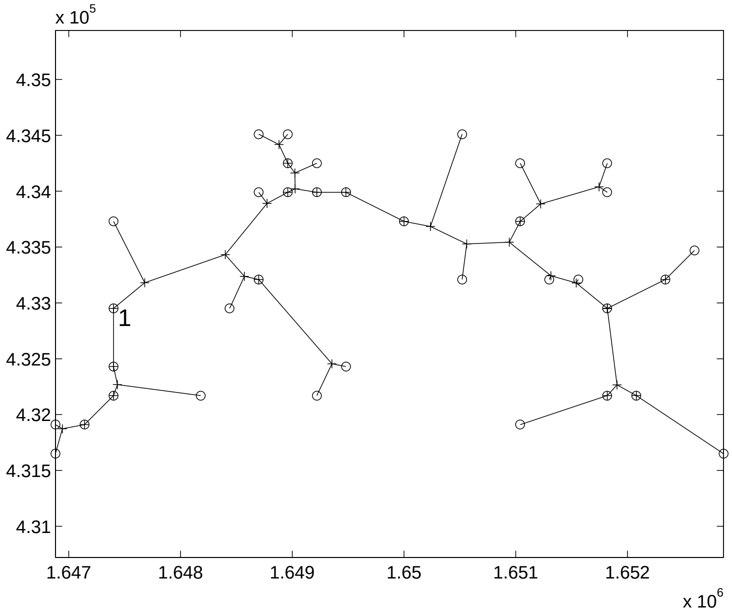


FIG. 7. *The* 5-*optimal network for the* 36 *points in test problem* 5.

8. Conclusions. We have presented a backtrack algorithm for computing the minimum cost pipe network interconnecting a single sink and many sources. This algorithm, when used with the max-min heuristic ordering, can solve problems with 11 regular points on a workstation in several minutes. It can also solve a problem with 15 regular points on a workstation in an hour. For larger problems, the algorithm becomes too expensive. In order to provide *good* suboptimal solutions to larger problems, we have also presented a 5-optimal algorithm for computing 5-optimal pipe networks. This algorithm, which starts from a min-min heuristic, can change to a *better* pipe network in $O(n)$ time if the current one is not 5-optimal. Our computational results show that this algorithm is very fast and often finds optimal or close to optimal solutions. Several interesting problems remain unsolved. We mention a few of them to conclude this paper.

The first one concerns the quality of the initial upper bound. For the ESMT problem, the cost of the minimum spanning tree is within a factor of $\frac{2}{\sqrt{3}}$ of the cost of the Steiner minimum tree [10, 14]. For the problem considered in this paper, there is no known polynomial time algorithm that can compute a solution whose cost is within a constant factor of the optimal cost. We suspect that both the min-min heuristic and the max-min heuristic produce 2-approximations to the minimum cost pipe network. Our computational results support this conjecture. However, we have not been able to arrive at a proof.

Although the number of full Steiner topologies is much smaller than the number of Steiner topologies, it is still superexponential in n. Therefore, it is very important to limit the number of topologies considered. For the ESMT problem, Winter and Zachariasen [32, 33] have developed a very effective technique of enumerating *equilateral points* (eq-points for short). They can rule out the vast majority of full topologies that cannot possibly have a degeneracy corresponding to the optimal topology. Generalizing the concept of eq-points to the pipe network problem is an intriguing topic for further research.

When there are several treatment sites and several wells, the problem becomes harder and more interesting. We are currently investigating this problem. Also, the 5-optimal algorithm can be implemented in parallel because the checking of different size-5 components can be done independently. Results on parallel implementations of the 5-optimal heuristic algorithm will be reported in a separate paper.

Acknowledgment. The authors would like to thank two referees and the associate editor for their helpful comments and remarks on the first version of the paper.

REFERENCES

[1] K.D. Andersen, *An efficient Newton barrier method for minimizing a sum of Euclidean norms*, SIAM J. Optim., 6 (1996), pp. 74–95.

[2] K.D. Andersen and E. Christiansen, *A Symmetric Primal-Dual Newton Method for Minimizing a Sum of Norms*, Manuscript, Odense University, Denmark, 1995.

[3] S. Bhaskaran and F.J.M. Salzborn, *Optimal design of gas pipeline network*, J. Oper. Res. Soc., 30 (1979), pp. 1047–1060.

[4] P.H. Calamai and A.R. Conn, *A stable algorithm for solving the multifacility location problem involving Euclidean distances*, SIAM J. Sci. Stat. Comput., 1 (1980), pp. 512–526.

[5] P.H. Calamai and A.R. Conn, *A projected Newton method for l_p norm location problems*, Math. Programming, 38 (1987), pp. 75–109.

[6] E.J. Cockayne and D.E. Hewgill, *Exact computation of Steiner minimal trees in the plane*, Inform. Process. Lett., 22 (1986), pp. 151–156.

[7] E.J. Cockayne and D.E. Hewgill, *Improved computation of plane Steiner minimal trees*, Algorithmica, 7 (1992), pp. 219–229.

[8] A.R. Conn and M.L. Overton, *A Primal-Dual Interior Point Method for Minimizing a Sum of Euclidean Distances*, manuscript, 1994.

[9] D.R. Dreyer and M.L. Overton, *Two heuristics for the Steiner tree problem*, J. Global Optim., 13 (1998), pp. 95–106.

[10] D.Z. Du and F.K. Hwang, *An approach for proving lower bounds: solution of Gilbert-Pollak conjecture on Steiner ratio*, in Proceedings of 31st IEEE Foundations of Computer Science, 1990, pp. 76–85.

[11] D.Z. Du, Y.J. Zhang, and Q. Feng, *On better heuristic for Euclidean Steiner minimum trees*, in Proceedings of 32nd IEEE Foundations of Computer Science, 1991, pp. 431–439.

[12] M.R. Garey, R.L. Graham, and D.S. Johnson, *The complexity of computing Steiner minimal trees*, SIAM J. Appl. Math., 32 (1977), pp. 835–859.

[13] E.N. Gilbert, *Minimum cost communication networks*, Bell System Tech. J., 46 (1967), pp. 2209–2227.

[14] E.N. Gilbert and H.O. Pollak, *Steiner minimal trees*, SIAM J. Appl. Math., 16 (1968), pp. 1–29.

[15] F. Harary, *Graph Theory*, Addison-Wesley, New York, 1969.

[16] F.K. Hwang, *A primer of the Euclidean Steiner problem*, Ann. Oper. Res., 33 (1991), pp. 73–84.

[17] F.K. Hwang, D.S. Richard, and P. Winter, *The Steiner Tree Problem*, Ann. Discrete Math. 53, North-Holland, Amsterdam, 1992.

[18] E. Horowitz and S. Sahni, *Fundamentals of Computer Algorithms*, Computer Science Press, Rockville, MD, 1978.

[19] D.H. Lee, *Low cost drainage networks*, Networks, 6 (1976), pp. 351–371.

[20] T.P. Lillys, *Optimal Piping Networks for Sub-Surface Remediation Designs*, M.S. thesis, Department of Civil and Environmental Engineering, University of Vermont, Burlington, 1997.

[21] Z.A. Melzak, *On the problem of Steiner*, Canad. Math. Bull., 4 (1961), pp. 143–148.

[22] Yu. E. Nesterov and A. Nemirovskii, *Interior Polynomial Algorithms in Convex Programming,* SIAM, Philadelphia, 1994.

[23] Yu. E. Nesterov and M. J. Todd, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.

[24] M.L. Overton, *A quadratically convergent method for minimizing a sum of Euclidean norms*, Math. Programming, 27 (1983), pp. 34–63.

[25] M.L. Overton and G.L. Xue, *private communications,* February 1996.

[26] R.C. Prim, *Shortest connection networks and some generalizations*, Bell System Tech. J., 36 (1967), pp. 1389–1401.

[27] D.M. Rizzo and D.E. Dougherty, *Design optimization for multiple management period groundwater remediation*, Water Resources Research, 32 (1996), pp. 2549–2562.

[28] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[29] W.D. Smith, *How to find Steiner minimal trees in Euclidean d-space*, Algorithmica, 7 (1992), pp. 137–177.

[30] P.K. Swamee and K.J. Akalank, *Explicit equations for pipe-flow problems*, J. Hydraulic Engineering, ASCE, 102 (1976), pp. 657–664.

[31] P. Winter, *An algorithm for the Steiner problem in the Euclidean plane*, Networks, 15 (1985), pp. 323–345.

[32] P. Winter and M. Zachariasen, *Large Euclidean Steiner Minimum Trees in an Hour*, Technical report 96/34, http://www.diku.dk/~pawel/publications.html.

[33] P. Winter and M. Zachariasen, *Large Euclidean Steiner minimum trees: An improved exact algorithm*, Networks, 30 (1998), pp. 149–66.

[34] G.L. Xue and Y.Y. Ye, *An efficient algorithm for minimizing a sum of Euclidean norms with applications*, SIAM J. Optim., 7 (1997), pp. 1017–1036.

[35] Y.Y. Ye, *Interior-point algorithms for quadratic programming*, in Recent Developments in Mathematical Programming, S. Kumar, ed., Gordon and Breach Science, New York, 1991, pp. 237–262.

[36] J.Z. Zhang and D.T. Zhu, *A bilevel programming method for pipe network optimization*, SIAM J. Optim., 6 (1996), pp. 838–857.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 43–55

DETECTION AND REMEDIATION OF STAGNATION IN THE NELDER–MEAD ALGORITHM USING A SUFFICIENT DECREASE CONDITION*

C. T. KELLEY†

Abstract. The Nelder–Mead algorithm can stagnate and converge to a nonoptimal point, even for very simple problems. In this note we propose a test for sufficient decrease which, if passed for all iterations, will guarantee convergence of the Nelder–Mead iteration to a stationary point if the objective function is smooth and the diameters of the Nelder–Mead simplices converge to zero. Failure of this condition is an indicator of potential stagnation. As a remedy we propose a new step, which we call an oriented restart, that reinitializes the simplex to a smaller one with orthogonal edges whose orientation is determined by an approximate descent direction from the current best point. We also give results that apply when the objective function is a low-amplitude perturbation of a smooth function. We illustrate our results with some numerical examples.

Key words. Nelder–Mead algorithm, sufficient decrease, stagnation

AMS subject classifications. 65K05, 65K10, 90C30

PII. S1052623497315203

1. Introduction.

In this paper we consider the Nelder–Mead [16] direct search algorithm for the unconstrained minimization of a possibly nonconvex or even discontinuous function f. The problem is

$$\min_{x \in R^N} f(x). \tag{1.1}$$

As in [10], we also consider objective functions that are small perturbations of smooth and easy-to-minimize functions.

The Nelder–Mead algorithm maintains a simplex of approximations to an optimal point. We assume throughout that the vertices $\{x_j\}_{j=1}^{N+1}$ are sorted according to the objective function values

$$f(x_1) \le f(x_2) \le \cdots \le f(x_{N+1}). \tag{1.2}$$

We will refer to x_1 as the best vertex and x_{N+1} as the worst. The algorithm attempts to change the worst vertex x_{N+1} to a new point of the form

$$x(\delta) = (1+\delta)\overline{x} - \delta x_{N+1}, \tag{1.3}$$

where $\overline{x}$ is the centroid of the sequence $\{x_i\}_{i=1}^{N}$,

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i. \tag{1.4}$$

A typical sequence [14] of candidate values for δ is

$$\left\{\delta_r, \delta_e, \delta_{oc}, \delta_{ic}\right\} = \left\{1, 2, \frac{1}{2}, -\frac{1}{2}\right\},$$

*Received by the editors January 20, 1997; accepted for publication (in revised form) September 11, 1998; published electronically October 20, 1999. This research was supported by National Science Foundation grant DMS-9321938.

http://www.siam.org/journals/siopt/10-1/31520.html

†Center for Research in Scientific Computation and Department of Mathematics, North Carolina State University, Box 8205, Raleigh, NC 27695-8205 (Tim_Kelley@ncsu.edu).

corresponding to the reflection, expansion, outside contraction, and inside contraction steps of the Nelder–Mead iteration. In general, we require that

$$-1 < \delta_{ic} < 0 < \delta_{oc} < \delta_r < \delta_e.$$

Algorithm 1.1 (nm) is a formal description of the iteration, with a termination condition on small differences in the function values at the vertices. The input is the initial simplex, the objective, and a tolerance for termination based on the difference between the best and worst values. We require that the initial simplex be nondegenerate (i.e., that the vectors $\{x_1 - x_j\}_{j=1}^N$ be linearly independent).

```
ALGORITHM 1.1. nm($S, f, \tau$).
1. Sort the vertices of $S$ so that (1.2) holds.
2. While $f(x_{N+1}) - f(x_1) > \tau$
   a. Compute $\overline{x}$, (1.4), $x(\delta_r)$, (1.3), and $f_r = f(x(\delta_r))$.
   b. Reflect: If $f(x_1) \le f_r < f(x_N)$, replace $x_{N+1}$ with $x(\delta_r)$ and go to
      step 2g.
   c. Expand: If $f_r < f(x_1)$ compute $f_e = f(x(\delta_e))$. If $f_e < f_r$ replace $x_{N+1}$
      with $x(\delta_e)$; otherwise replace $x_{N+1}$ with $x(\delta_r)$. Go to step 2g.
   d. Outside Contraction: If $f(x_N) \le f_r < f(x_{N+1})$ compute $f_c =$
      $f(x(\delta_{oc}))$. If $f_c \le f_r$ replace $x_{N+1}$ with $x(\delta_{oc})$ and go to step 2g; other-
      wise go to step 2f.
   e. Inside Contraction: If $f_r \ge f(x_{N+1})$ compute $f_c = f(x(\delta_{ic}))$. If
      $f_c < f(x_{N+1})$ replace $x_{N+1}$ with $x(\delta_{ic})$ and go to step 2g; otherwise go
      to step 2f.
   f. Shrink: For $2 \le i \le N+1$: $x_i = x_1 + \frac{x_i - x_1}{2}$; compute $f(x_i)$.
   g. Sort: Sort the vertices of $S$ so that (1.2) holds.
```

The sort step is not precisely defined by Algorithm 1.1. One could simply use a sort from one's computing environment, or specify a sort algorithm with a tie-breaking rule. Our results require only that each simplex satisfy (1.2), so we can follow [16] and accept any sort. One specific tie-breaking rule was proposed in [14].

As one can see from the algorithm, if reflection, expansion, and the two types of contractions do not succeed, the simplex is reduced in size, keeping only the vertex x_1 with the lowest objective function value. This last scenario, the shrink step, is rare, and the analysis we present will assume that shrinks do not occur.

We regard the simplex, and not just the best point, as the state of the iteration. If a shrink step is not taken, then the worst vertex is replaced with the new, better, vertex; the vertices are resorted; and the average objective function value

$$\underline{f} = \frac{1}{N+1} \sum_{j=1}^{N+1} f(x_j) \tag{1.5}$$

has been decreased.

Unlike the pattern search algorithms [5], [8], [11], [20], [21] that maintain the shape of the simplex, or the hybrid algorithm from [22], the Nelder–Mead algorithm can stagnate and converge to a nonoptimal point [15], [9], [25], [14], [26], even for very simple, smooth, and convex objective functions. Our results and analysis are simplex oriented and assume, even if f is discontinuous or nonsmooth, an underlying smooth structure of the objective that, while not present in the general case, is present in many applications.

In section 2 we define our notation and state a few simple lemmas. In particular, we define a *simplex gradient*, which we use to monitor the performance of the Nelder–Mead iteration and for our modification of the shrink step. We use those ideas in section 3, where we describe our condition for sufficient decrease (3.1), and we show in section 3.1 that if the objective function f is sufficiently smooth, the Nelder–Mead iterates satisfy this sufficient decrease condition, and the simplex diameters converge to zero in a certain way, then any limit point of the simplex vertices is stationary. Our decrease condition is simpler than the one in [22], reflecting our interest in high-frequency low-amplitude perturbations of smooth functions and in exploiting the one function evaluation per iteration cost of the Nelder–Mead algorithm as aggressively as safely possible.

In section 3.2 we propose an alternative to the shrink step that is to be used when (3.1) fails to hold. This new step, which we call an *oriented restart*, reinitializes the simplex to a smaller one with orthogonal edges which contains a difference approximation to the steepest descent step from the current best point.

Throughout the paper we present results that apply when the objective function is a low-amplitude perturbation of a smooth function. Finally, in section 4 we show how a modified Nelder–Mead algorithm that incorporates our ideas performs on the examples in [15].

2. Notation.

In this paper $\| \cdot \|$ will denote the l^2 norm or the induced matrix norm. We consider algorithms that maintain a simplex S of potential optima with vertices $\{x_j\}_{j=1}^{N+1}$ that satisfy (1.2).

We let V (or $V(S)$) denote the $N \times N$ matrix of *simplex directions* by

$$V(S) = (x_2 - x_1, x_3 - x_1, \ldots, x_{N+1} - x_1) = (v_1, \ldots, v_N),$$

and $\mathrm{diam}(S)$, the *simplex diameter*, by

$$\mathrm{diam}(S) = \max_{1 \le i,j \le N+1} \|x_i - x_j\|.$$

We will refer to the l^2 condition number $\kappa(V)$ of V as the *simplex condition*. We let $\delta(f, S)$ denote the vector of objective function differences

$$\delta(f : S) = (f(x_2) - f(x_1), f(x_3) - f(x_1), \ldots, f(x_{N+1}) - f(x_1))^T.$$

We will not use the simplex diameter directly in our estimates or algorithms. Rather, we will use two *oriented lengths*

$$\sigma_+(S) = \max_{2 \le j \le N+1} \|x_1 - x_j\| \quad \text{and} \quad \sigma_-(S) = \min_{2 \le j \le N+1} \|x_1 - x_j\|.$$

Clearly,

$$\sigma_+(S) \le \mathrm{diam}(S) \le 2\sigma_+(S).$$

DEFINITION 2.1. *Let S be a simplex with vertices $\{x_j\}_{j=1}^{N+1}$ ordered so that* (1.2) *holds and $V(S)$ is nonsingular. The* simplex gradient *$D(f : S)$ is*

$$D(f : S) = V^{-T}\delta(f : S).$$

This definition of simplex gradient is motivated by the following first-order estimate.

LEMMA 2.2. *Let S be a simplex with vertices ordered so that* (1.2) *holds. Let ∇f be Lipschitz continuous in a neighborhood of S with Lipschitz constant $2K_f$. Then there exists $K > 0$, depending only on K_f, such that*

$$\|\nabla f(x_1) - D(f : S)\| \leq K\kappa(V)\sigma_+(S). \tag{2.1}$$

Proof. Our smoothness assumptions on f and Taylor's theorem imply that for all $1 \leq j \leq N$,

$$\begin{aligned}\left\| f(x_1) - f(x_j) + \frac{\partial f(x_1)}{\partial v_j} v_j \right\| &= \|f(x_1) - f(x_j) + v_j^T \nabla f(x_1)\| \\ &\leq K_f\|v_j\|^2 \leq K_f\sigma_+(S)^2.\end{aligned}$$

Hence

$$\|\delta(f : S) - V^T\nabla f(x_1)\| \leq N^{1/2}K_f\sigma_+(S)^2$$

and, setting $K = N^{1/2}K_f$,

$$\|\nabla f(x_1) - D(f : S)\| \leq K\|V^{-T}\|\sigma_+(S)^2.$$

The conclusion follows from the fact that $\sigma_+(S) \leq \|V\|$. □

Objective functions of the form

$$f(x) = g(x) + \phi(x), \tag{2.2}$$

where g is to be thought of as a smooth and easy-to-optimize function and ϕ as a low-amplitude perturbation, arise naturally in applications [2], [3], [6], [10], [18], [19], [23], [24]. In this paper we will assume only that ϕ is everywhere defined and bounded in R^N. Algorithms that use difference approximations to the gradient of f have been proposed for bound-constrained [18], [10] and unconstrained [27] problems as a way to avoid entrapment in local minima caused by the perturbation. The implicit filtering algorithm described in [10] and [18] also attempts to obtain superlinear convergence in the terminal phase of the iteration if $\phi(x) \to 0$ as x tends to the optimal point. Like the pattern search algorithms, these difference methods require $O(N)$ function evaluations per iteration and, therefore, may be much less efficient in the initial phase of the iteration than a simplex algorithm like Nelder–Mead that requires only $O(1)$ evaluations or iterations.

One purpose of this paper is to apply simplicial algorithms that use fewer than $O(N)$ function evaluations per iteration to such objective functions. One would hope that the varying sizes of the simplices during the iteration would help avoid local minima. We will need to measure the perturbations on each simplex. To that end, we define for a simplex S

$$\|\phi\|_S = \max_{1\leq j\leq N+1}\|\phi(x_j)\|.$$

A first-order estimate also holds for the simplex gradient of an objective function that satisfies (2.2).

Lemma 2.3. *Let S be a simplex with vertices ordered so that* (1.2) *holds. Let f satisfy* (2.2) *and let ∇g be Lipschitz continuous in a neighborhood of S with Lipschitz constant $2K_g$. Then, there exists $K > 0$, depending only on K_g, such that*

$$\|\nabla g(x_1) - D(f : S)\| \le K\kappa(V)\left(\sigma_+(S) + \frac{\|\phi\|_S}{\sigma_+(S)}\right). \tag{2.3}$$

Proof. Lemma 2.2 (applied to g) implies

$$\|\nabla g(x_1) - D(g : S)\| \le K_g N^{1/2}\kappa(V)\sigma_+(S).$$

Now, since $\|\delta(\phi : S)\| \le 2\sqrt{N}\,\|\phi\|_S$, and $\sigma_+(S) \le \|V\|$,

$$\begin{aligned}\|D(f : S) - D(g : S)\| &\le \|V^{-T}\|\|\delta(f : S) - \delta(g : S)\| = \|V^{-T}\|\|\delta(\phi : S)\| \\ &\le 2N^{1/2}\|V^{-T}\|\|\phi\|_S \le 2N^{1/2}\kappa(V)\frac{\|\phi\|_S}{\sigma_+(S)}.\end{aligned}$$

This completes the proof with $K = N^{1/2}K_g + 2N^{1/2}$. □

The constants K in (2.1) and (2.3) depend on S only through the Lipschitz constant of f (or g) in a neighborhood of S. We will express that dependence as $K = K(S)$ when needed.

We will denote the vertices of the simplex S^k at the kth iteration by $\{x_j^k\}_{j=1}^{N+1}$. We will simplify notation by suppressing explicit mention of S^k in what follows by defining

$$V^k = V(S^k),\ \delta^k f = \delta(f : S^k),\ K^k = K(S^k), \quad \text{and} \quad D^k f = D(f : S^k).$$

If V^0 is nonsingular, then V^k is nonsingular for all $k > 0$ [14]. Hence, if V^0 is nonsingular, $D^k f$ is defined for all k.

Our assumptions on the sequence of simplices are modest.

Assumption 2.1.

- V^0 is nonsingular.
- The vertices satisfy (1.2).
- For each k, $\underline{f}^{k+1} < \underline{f}^k$.

Assumption 2.1 is satisfied by the Nelder–Mead sequence if no shrink steps are taken and the initial simplex directions are linearly independent. Assumption 2.1 need not be satisfied by the pattern search methods considered in [21], where only conditions on the best value are enforced. One way to illustrate the difference between pattern search methods and methods like Nelder–Mead is to note that the pattern search methods require that the simplices be geometrically similar (so the simplex condition is bounded) and require improvement in the best point. Nelder–Mead demands that the average function value improve, but no control is possible on which value is improved, and the simplex condition number can become unbounded.

3. Sufficient decrease and the oriented restart. Motivated by conventional line search decrease criteria for optimization and nonlinear equations [1], [7], [12], [17], we will ask that the $(k + 1)$st iteration satisfy

$$\underline{f}^{k+1} - \underline{f}^k < -\alpha\|D^k f\|^2. \tag{3.1}$$

Here $\alpha > 0$ is a small parameter. Our choice of sufficient decrease condition is motivated by the smooth case, where the sufficient decrease condition for the steepest descent direction is

$$f(x_c + \lambda \nabla f(x_c)) - f(x_c) < \alpha_0 \lambda \|\nabla f(x_c)\|^2,$$

where λ is a line search parameter. In the smooth case, one typically obtains a lower bound λ_- for λ and arrives at a smooth analogue of (3.1),

$$f(x_c + \lambda \nabla f(x_c)) - f(x_c) < \alpha_0 \lambda_- \|\nabla f(x_c)\|^2.$$

Unlike the smooth case, however, we have no descent direction and must incorporate λ_- into α. This leads to the possibility that if the simplex diameter is much smaller than $\|D^k f\|$, (3.1) could fail on the first iterate. We address this problem with the scaling

$$\alpha = \alpha_0 \frac{\sigma_+(S^0)}{\|D^0 f\|}.$$

A typical choice in line search methods, which we use in our numerical results, is $\alpha_0 = 10^{-4}$. We propose to use failure of (3.1) as a test for impending stagnation at a nonminimizer.

3.1. Convergence results. The convergence result for smooth functions follows easily from Lemma 2.2.

THEOREM 3.1. *Let a sequence of simplices satisfy Assumption* 2.1 *and let the assumptions of Lemma* 2.2 *hold, with the Lipschitz constants K^k bounded. Assume that $\{\underline{f}^k\}$ is bounded from below. Then if* (3.1) *holds for all but finitely many k and the product $\sigma_+(S^k)\kappa(V^k) \to 0$, then any accumulation point of the simplices is a critical point of f.*

Proof. The boundedness from below of $\{\underline{f}^k\}$ and (3.1) implies that $\{\underline{f}^k\}$ converges to a constant. Assumption 2.1 and (3.1) imply that $\lim_{k\to\infty} D^k f = 0$. Hence (2.1) implies

$$\lim_{k\to\infty} \|\nabla f(x_1^k)\| \le \lim_{k\to\infty} \left(K^k \kappa(V^k)\sigma_+(S^k) + \|D^k f\|\right) = 0.$$

Hence, if x^* is any accumulation point of the sequence $\{x_1^k\}$, then $\nabla f(x^*) = 0$. This completes the proof, since $\kappa(V^k) \ge 1$ implies that $\sigma_+(V^k) \to 0$ and, hence, the vertices have a common accumulation point. □

Note that the conclusion of Theorem 3.1 also holds [17] if the sufficient decrease condition (3.1) is replaced by

$$\underline{f}^{k+1} - \underline{f}^k < -\Phi(\|D^k f\|), \tag{3.2}$$

where Φ is a monotonically increasing function on $[0, \infty)$ with $\Phi(0) = 0$.

The result for the noisy functions that satisfy (2.2) with g smooth reflects the fact that the resolution is limited by the size of ϕ; hence the assumption (3.3), which implies that $\|\phi\|_{S^k} \to 0$ sufficiently rapidly. To see why this is important for the theory, assume that $\|\phi\|_{S^k} \ge \epsilon$ for all k. In this case, once $\sigma(S^k) \ll \epsilon$, then the noise is larger than the variation in g over S^k and one should terminate the iteration. If

$\sigma(S^k) \ll \epsilon^{1/2}$, the error term $\frac{\|\phi\|_{S^k}}{\sigma(S^K)}$ on the right side of (2.3) is dominant and the simplex gradient of f is no longer nicely related to the gradient of g.

THEOREM 3.2. *Let a sequence of simplices satisfy Assumption* 2.1 *and let the assumptions of Lemma* 2.3 *hold with the Lipschitz constants* K_g^k *uniformly bounded. Assume that* $\{\underline{f}^k\}$ *is bounded from below. Then if* (3.1) *holds for all but finitely many* k *and if*

$$\lim_{k\to\infty} \kappa(V^k)\left(\sigma_+(S^k) + \frac{\|\phi\|_{S^k}}{\sigma_+(S^k)}\right) = 0, \tag{3.3}$$

then any accumulation point of the simplices is a critical point of g.

Proof. Our assumptions, as in the proof of Theorem 3.1, imply that $D^k f \to 0$. Lemma 2.3 implies that

$$\|D^k g\| \le \|D^k f\| + K^k \kappa(V^k)\left(\sigma_+(S^k) + \frac{\|\phi\|_{S^k}}{\sigma_+(S^k)}\right), \tag{3.4}$$

and the sequence $\{K^k\}$ is bounded. Hence, by (3.3), $D^k g \to 0$ as $k \to \infty$. □

We close this subsection with a partial converse of Theorem 3.2 that shows how the connection between the condition number, simplex size, and amplitude of the noise leads to a necessary condition for convergence to a critical point that is different from that for smooth problems. In the smooth case, if $x_k \to x^*$ and x^* is a critical point of f, then $\nabla f(x^k) \to 0$. In the present case, we must require that the entire simplex converge to x^* rapidly enough to control potential growth in the condition number and also demand that the noise die out near the critical point.

THEOREM 3.3. *Let* $\{S^k\}$ *be a sequence of simplices with the assumptions of Lemma* 2.3 *holding for each* k. *Assume that*

$$\lim_{k\to\infty} \kappa(V^k)\left(\sigma_+(S^k) + \frac{\|\phi\|_{S^k}}{\sigma_+(S^k)}\right) = 0$$

and that $x_1^k \to x^*$, *a critical point of* g. *Then*

$$\lim_{k\to\infty} D^k f = 0.$$

Proof. The result follows from (2.3) and the fact that $\nabla g(x_1^k) \to 0$. □

3.2. Oriented restarts. One can monitor a simplex-based iteration to see if (3.1) holds. However, unlike the case of a gradient-based line search method, simply reducing the size of the simplex (for example, a shrink step in Nelder–Mead) will not remedy the problem. We propose performing an *oriented restart* when (3.1) fails but $\underline{f}^{k+1} - \underline{f}^k < 0$. This means replacing the current simplex with vertices $\{x_j\}_{j=1}^{N+1}$, ordered so that (1.2) holds, with a new, smaller simplex having vertices (before ordering!) $\{y_j\}_{j=1}^{N+1}$ with $y_1 = x_1$ and

$$y_j = y_1 + \beta_{j-1} e_{j-1} \quad \text{for} \quad 2 \le j \le N+1, \tag{3.5}$$

where e_l is the kth coordinate vector,

$$\beta_l = \frac{1}{2}\begin{cases} \sigma_-(S^k)\mathrm{sign}((D^k f)_l), & (D^k f)_l \ne 0, \\ \sigma_-(S^k), & (D^k f)_l = 0, \end{cases}$$

and $(D^k f)_l$ is the lth component of $D^k f$. If $D^k f = 0$ we assume that the Nelder–Mead iteration would have been terminated at iteration k because of no difference between best and worst values.

So, before ordering, the new simplex has the same first point as the old. The diameter of the new simplex is $\frac{\sigma_-(S^k)}{\sqrt{2}}$. Therefore, after reordering, $\sigma_+(S^{k+1}) \le \sigma_-(S^k)$. As for κ, after the oriented shrink, but before reordering, $\kappa(V) = 1$. After reordering, of course, the best point may no longer be x_1. If the best point is unchanged, V^{k+1} is a diagonal matrix with entries ± 1. If the best point has been changed, then, up to row permutation and multiplication by the scalar $\frac{\pm\sigma_-(S^k)}{2}$, V^{k+1} is given by the upper triangular matrix

$$V^{k+1} = (V^{k+1})^{-1} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 0 & -1 & 0 & \dots & 0 \\ \vdots & 0 & -1 & \ddots & \vdots \\ 0 & & \ddots & \ddots & 0 \\ 0 & \dots & \dots & 0 & -1 \end{pmatrix}.$$

Hence either $V^{k+1} = I$ and $\kappa(V) = 1$ or the l^1 condition number is $\kappa_1(V^{k+1}) = \|V^{k+1}\|_1^2 = 4$. The l^2 condition number can be estimated by

$$\kappa(V^{k+1}) = \|V^{k+1}\|^2 \le (1 + \sqrt{N})^2.$$

In any case, the new simplex is well conditioned.

The new orientation of the simplex is intended to compensate for the kind of stagnation that was exhibited in [15], in which the best vertex, not a minimizer, remained unchanged throughout the entire iteration, and the simplices converged to that vertex. The expectation, at least partially realized in our examples, is that while the simplex gradient may be a poor approximation of the gradient, especially when the simplex condition is large, the simplex diameter is small, and the gradient either is nonzero or does not exist (all of which happen in the examples), there is enough information in the simplex gradient to determine an orthant in which to locate the new simplex. This is why we use only the signs of the components of the simplex gradient.

The reduction in the simplex size, like the reduction in steplength in a line search method, should, for smooth problems, make it easier to satisfy (3.1), especially when the new simplex has orthogonal edges. While there is nothing special about the factor of $\frac{1}{2}$ in (3.5), there is no reason to expect that more elaborate line search schemes based on polynomial models, such as those presented in [7], would be effective in the context in which Nelder–Mead and related algorithms are used.

This approach improves the robustness of Nelder–Mead, but it does not solve all the problems. It is possible that (3.1) may fail even after several restarts. In fact, in the results reported in section 4, that happens for one example. Our implementation terminates with failure after three such unsuccessful restarts. One can envision hybrid algorithms that combine Nelder–Mead with a more robust search algorithm, such as implicit filtering [10], multidirectional search [20], or the Hooke–Jeeves iteration [11]. These methods can all be analyzed with the simplex gradient [4], [13] to derive results like Theorem 3.3 without any concern for the growth of κ, which is bounded for all of these methods. However, the cost of $O(N)$ evaluations of f per iteration for these alternatives makes the Nelder–Mead algorithm attractive in those cases where it does not stagnate.

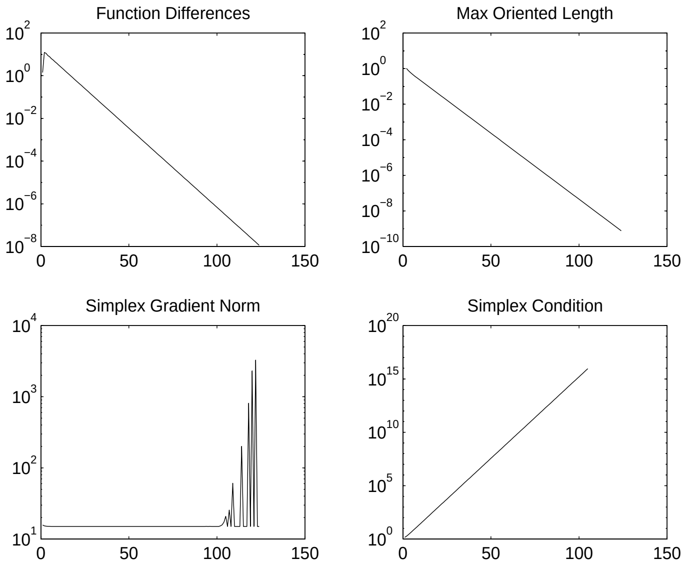


FIG. 4.1. *Unmodified Nelder–Mead,* $(\tau, \theta, \phi) = (1, 15, 10)$.

4. Numerical testing. We show how the detection of stagnation and the modification of the Nelder–Mead algorithm proposed in section 3 perform in the examples from [15]. Here $N = 2$ and

$$f(x, y) = \begin{cases} \theta\phi|x|^\tau + y + y^2, & x \le 0, \\ \\ \theta x^\tau + y + y^2, & x > 0. \end{cases}$$

The examples in [15] consider the parameter sets

$$(\tau, \theta, \phi) = \begin{cases} (3, 6, 400), \\ (2, 6, 60), \\ (1, 15, 10). \end{cases}$$

The initial simplex was

$$x_1 = (1, 1)^T, \quad x_2 = (\lambda_+, \lambda_-)^T, \quad x_3 = (0, 0)^T, \quad \text{where} \quad \lambda_\pm = \frac{1 \pm \sqrt{33}}{8}.$$

With this data, the Nelder–Mead iteration will stagnate at the origin, which is not a critical point for f.

We terminated the iteration when the difference between the best and worst function values was $< 10^{-8}$ or, in the modified algorithm, after three restarts, using this as a test for stagnation.

We illustrate the behavior of the unmodified Nelder–Mead algorithm in Figures 4.1, 4.3, and 4.5. In all the figures we plot, as functions of the iteration index, the difference between the best and worst function values, σ_+, the maximum oriented length, the norm of the simplex gradient, and the l^2 condition number of the matrix of simplex directions. In all three problems stagnation is evident from the behavior of the simplex gradients. Note also how the simplex condition number is growing rapidly.

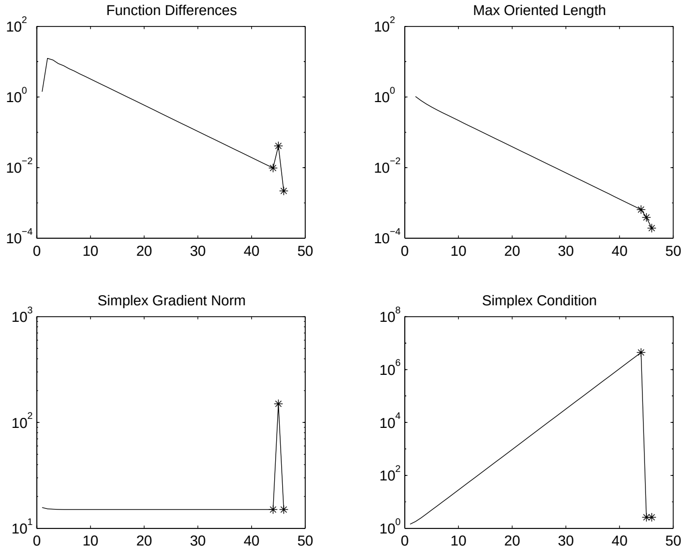


FIG. 4.2. *Modified Nelder–Mead,* $(\tau, \theta, \phi) = (1, 15, 10)$.

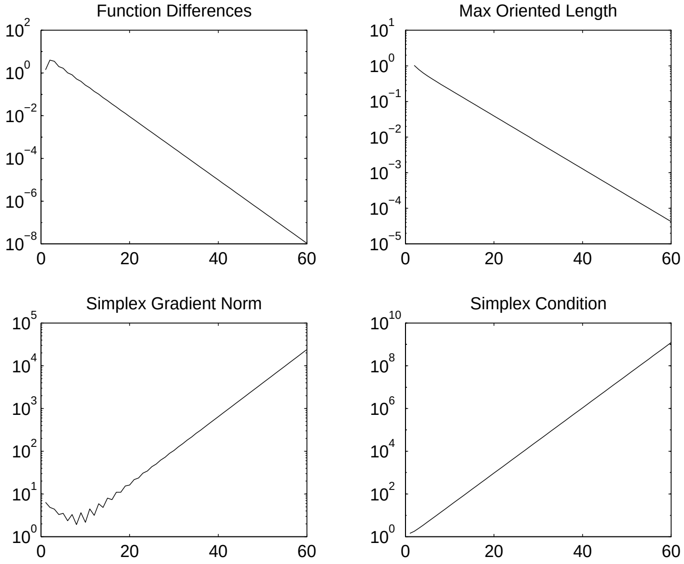


FIG. 4.3. *Unmodified Nelder–Mead,* $(\tau, \theta, \phi) = (2, 6, 60)$.

In Figures 4.2, 4.4, and 4.6 we present the same data for the modified algorithm, with stars on the graphs to indicate where oriented restarts were done.

Two of the examples in [15] are smooth. For the less smooth of these two, $(\tau, \theta, \phi) = (2, 6, 60)$, the modified form of Nelder–Mead took a single oriented restart at the 19th iteration. For the smoothest example, $(\tau, \theta, \phi) = (3, 6, 400)$, the modified form of Nelder–Mead took a single oriented restart at the 21st iteration. As one can see from Figures 4.4 and 4.6 the restart had an immediate effect on the simplex gradient norm and overcame the stagnation.

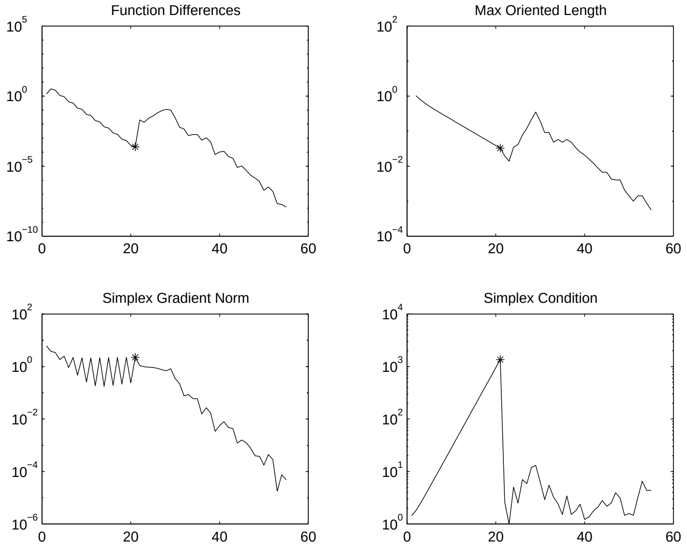


Fig. 4.4. *Modified Nelder–Mead,* $(\tau, \theta, \phi) = (2, 6, 60)$.

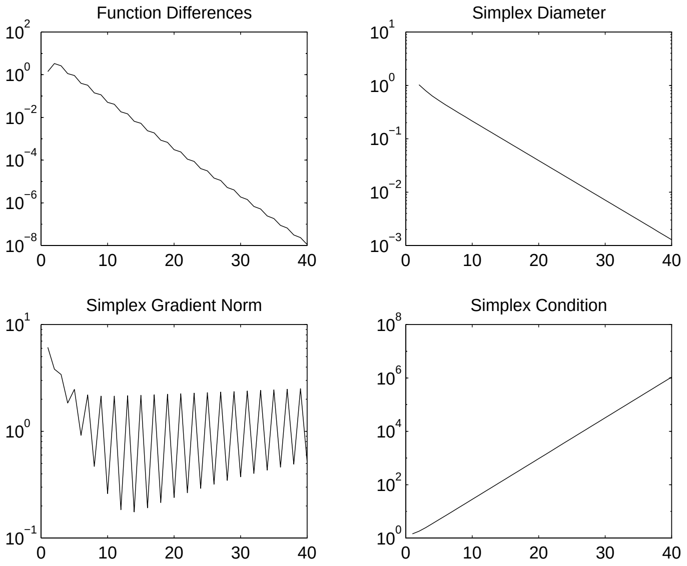


Fig. 4.5. *Unmodified Nelder–Mead,* $(\tau, \theta, \phi) = (3, 6, 400)$.

For the nonsmooth example, $(\tau, \theta, \phi) = (1, 15, 10)$, in Figure 4.1, the modified algorithm terminated with failure after restarting on the 44th, 45th, and 46th iterations. Since the objective is not smooth at the stagnation point, this is the best we can expect and is far better than the behavior of the unmodified algorithm, which stagnates with no warning of the failure.

All computations reported here were done using MATLAB 5.0 on a Sun Ultra Enterprise 1 running Sun Solaris v2.1.

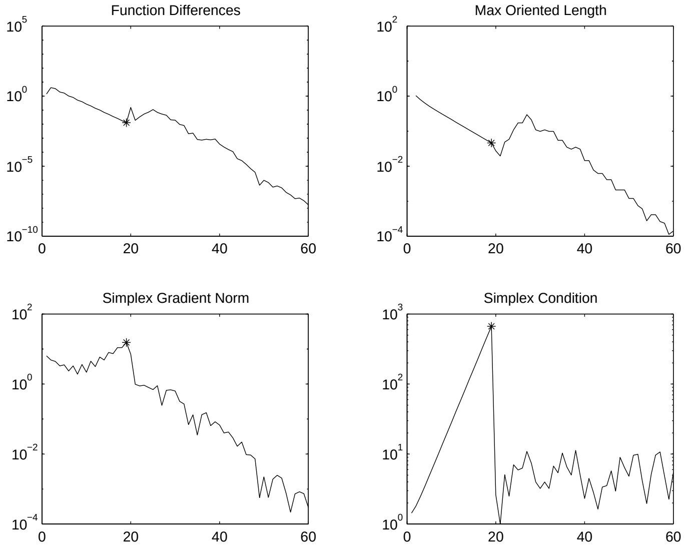


FIG. 4.6. *Modified Nelder–Mead,* $(\tau, \theta, \phi) = (3, 6, 400)$.

Acknowledgments. The author is grateful to Carl Meyer and Margaret Wright, for their assistance with several aspects of this work, and to an anonymous referee for some very thoughtful comments.

REFERENCES

[1] L. Armijo, *Minimization of functions having Lipschitz-continuous first partial derivatives*, Pacific J. Math., 16 (1966), pp. 1–3.

[2] K. R. Bailey and B. G. Fitzpatrick, *Estimation of Groundwater Flow Parameters Using Least Squares*, Tech. Report CRSC-TR96-13, Center for Research in Scientific Computation, North Carolina State University, Raleigh, April 1996.

[3] K. R. Bailey, B. G. Fitzpatrick, and M. A. Jeffries, *Least Squares Estimation of Hydraulic Conductivity from Field Data*, Tech. Report CRSC-TR95-8, Center for Research in Scientific Computation, North Carolina State University, Raleigh, February 1995.

[4] D. M. Bortz and C. T. Kelley, *The simplex gradient and noisy optimization problems*, in Computational Methods for Optimal Design and Control, Progr. System Control Theory 24, Birkhäuser, Boston, MA, 1998, p. 77–90.

[5] A. G. Buckley and H. Ma, *A Derivative-Free Algorithm for Parallel and Sequential Optimization*, Tech. Report, Computer Science Department, University of Victoria, BC, Canada, October 1994.

[6] J. W. David, C. T. Kelley, and C. Y. Cheng, *Use of an Implicit Filtering Algorithm for Mechanical System Parameter Identification*, SAE Paper 960358, Society of Automotive Engineers International Congress and Exposition Conference Proceedings, Model. CI and SI Engines, Detroit, MI, 1996, pp. 189–194.

[7] J. E. Dennis and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Classics Appl. Math. 16, SIAM, Philadelphia, 1996.

[8] J. E. Dennis, Jr., and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448–474.

[9] J. E. Dennis and D. J. Woods, *Optimization on microcomputers: The Nelder–Mead simplex algorithm*, in New Computer Environments: Microcomputers in Large-Scale Computing, A. Wouk, ed., Proc. Appl. Math. 27, SIAM, Philadelphia, 1987, pp. 116–122.

[10] P. Gilmore and C. T. Kelley, *An implicit filtering algorithm for optimization of functions with many local minima*, SIAM J. Optim., 5 (1995), pp. 269–285.

[11] R. Hooke and T. A. Jeeves, *"Direct search" solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–229.

[12] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, Frontiers Appl. Math. 16, SIAM, Philadelphia, 1995.

[13] C. T. Kelley, *Iterative Methods for Optimization*, Frontiers Appl. Math. 18, SIAM, Philadelphia, 1999.

[14] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, *Convergence properties of the Nelder–Mead simplex algorithm in low dimensions*, SIAM J. Optim., 9 (1999), pp. 112–147.

[15] K. I. M. McKinnon, *Convergence of the Nelder–Mead simplex method to a nonstationary point*, SIAM J. Optim., 9 (1999), pp. 148–158.

[16] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[17] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[18] D. Stoneking, G. Bilbro, R. Trew, P. Gilmore, and C. T. Kelley, *Yield optimization using a GaAs process simulator coupled to a physical device model*, IEEE Trans. Microwave Theory and Techniques, 40 (1992), pp. 1353–1363.

[19] D. E. Stoneking, G. L. Bilbro, R. J. Trew, P. Gilmore, and C. T. Kelley, *Yield optimization using a GaAs process simulator coupled to a physical device model*, in Proceedings, IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 374–383.

[20] V. Torczon, *On the convergence of the multidirectional search algorithm*, SIAM J. Optim., 1 (1991), pp. 123–145.

[21] V. Torczon, *On the convergence of pattern search algorithms*, SIAM J. Optim., 7 (1997), pp. 1–25.

[22] P. Tseng, *Fortified-Descent Simplicial Search Method: A General Approach*, Tech. Report, Department of Mathematics, University of Washington, Seattle, WA, 1995.

[23] T. A. Winslow, R. J. Trew, P. Gilmore, and C. T. Kelley, *Doping profiles for optimum class B performance of GaAs mesfet amplifiers*, in Proceedings, IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 188–197.

[24] T. A. Winslow, R. J. Trew, P. Gilmore, and C. T. Kelley, *Simulated performance optimization of GaAs MESFET amplifiers*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 393–402.

[25] D. J. Woods, *An Interactive Approach for Solving Multi-Objective Optimization Problems*, Ph.D. thesis, Rice University, Houston, TX, 1985.

[26] M. H. Wright, *Direct search methods: Once scorned, now respectable*, in Numer. Anal. 1995, Proceedings of the 1995 Dundee Bienneal Conference in Numerical Analysis, D. F. Griffiths and G. A. Watson, eds., Pitman Res. Notes Math. Ser. 344, Longman, Harlow, UK, 1996, pp. 191–208.

[27] S. K. Zavriev, *On the global optimization properties of finite-difference local descent algorithms*, J. Global Optim., 3 (1993), pp. 67–78.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 56–69

REDUCED STORAGE, QUASI-NEWTON TRUST REGION APPROACHES TO FUNCTION OPTIMIZATION*

LINDA KAUFMAN†

Abstract. In this paper we consider several algorithms for reducing the storage when using a quasi-Newton method in a dogleg–trust region setting for minimizing functions of many variables. Secant methods require $O(n^2)$ locations to store an approximate Hessian and $O(n^2)$ operations per iteration when minimizing a function of n variables. This storage requirement becomes impractical when n becomes large. Our algorithms use a BFGS update and require kn storage and $4kn + O(k^2)$ operations per iteration, but they may require more iterations than the standard trust region techniques. Typically k is between 10 and 100. Our dogleg–trust region strategies involve expressions with matrix products with both the inverse of this Hessian and with the Hessian itself. Our techniques for updating expressions for the Hessian and its inverse can be used to improve the performance of line search, limited memory algorithms.

Key words. quasi-Newton, trust region, limited memory

AMS subject classifications. 49M10, 65K10, 15A23

PII. S1052623496303779

1. Introduction.

Quasi-Newton methods are iterative methods for minimizing a function $f(x)$, where $x \in R^n$, using only first derivative and function information. If $f(x)$ is a quadratic function, quasi-Newton methods converge in at most n iterations in exact arithmetic with exact line searches. At each iteration a linear system is solved using a matrix B which is an approximation to the true Hessian G, which has components $g_{i,j} = \frac{\partial^2 f(x)}{\partial x_i \partial x_j}$. At each iteration the matrix B is updated to the matrix B_+ by a rank 1 or 2 update so that at the next iterate x^+, B_+ satisfies the quasi-Newton condition

$$B_+ s = y, \tag{1.1}$$

where $s = x^+ - x$ and $y = \nabla f(x^+) - \nabla f(x)$.

Quasi-Newton methods with reduced storage have been discussed by Nocedal [14], Buckley and LeNir [2], Liu and Nocedal [11], and Byrd, Nocedal, and Schnabel [4]. Their main idea is that by saving the s's and y's of the last k iterations and some of their inner products, one can easily generate a quasi-Newton search direction that is based on the previous k iterations. Thus one would choose a value of k based on the amount of storage or the expense of each iteration for the particular application, and for the first k iterations, the iterates from the reduced storage scheme and the quasi-Newton method would coincide, but on iteration $k + 1$, information from the first iteration would be discarded and the approximation would not necessarily agree with the $(k + 1)$st step of the traditional quasi-Newton approach.

Dogleg–trust region approaches, as in Dennis, Gay, and Welsch [6] and Dennis and Mei [7] for minimizing a function $f(x)$, determine a radius of trust τ which defines the region where one trusts the second-order model of the function to be minimized. The next iterate x^+ must satisfy

$$||x^+ - x||_2 \leq \tau. \tag{1.2}$$

*Received by the editors May 17, 1996; accepted for publication (in revised form) September 10, 1998; published electronically October 20, 1999.

http://www.siam.org/journals/siopt/10-1/30377.html

†Bell Laboratories, Room 2C-461, 700 Mountain Ave., Murray Hill, NJ 07974 (lck@lucent.com).

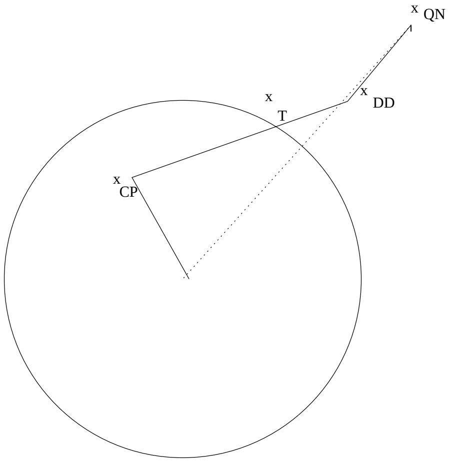


FIG. 1.1. *The double-dogleg path.*

In most trust region approaches, the person who wishes to minimize $f(x)$ initially chooses τ, and its value is gradually changed by the trust region algorithm as the method proceeds.

At each iteration, dogleg–trust region methods consider $\phi(d)$, a quadratic model of $f(x+d)$, of the form

$$\phi(d) = \frac{1}{2}d^T Bd + \nabla f^T d + f(x). \tag{1.3}$$

Here B is an approximate Hessian that has been updated according to condition (1.1). If the "quasi-Newton" point $x_{QN} = x - B^{-1}\nabla(fx)$ satisfies (1.2) it becomes a trial step; otherwise the trial step x_T in the double-dogleg strategy is taken as the largest step that satisfies (1.2) and lies on the polygonal line that runs from x to the Cauchy point, x_{CP} (the minimum of (1.3) along the steepest descent direction), to a point x_{DD} in the quasi-Newton direction, up to the x_{QN}, as shown in Figure 1.1. The point x_{DD} is chosen as in Dennis and Mei [7] so that ϕ is guaranteed to monotonically decrease along the polygonal line. If $f(x_T) > f(x)$, τ is decreased and the process is repeated.

The calculation of the dogleg–trust region step involves considering both the quasi-Newton direction and the steepest descent direction at each iteration. Determining the quasi-Newton direction requires the solution of a linear system with B. Adjusting the radius of trust and determining whether the Cauchy point along the steepest descent direction is within that region requires expressions with B.

For more than a decade, the quantum chemistry group at Bell Laboratories has been successfully using the double-dogleg strategy of Dennis and Mei [7], as implemented in MINOP, an early trust region code, which had been changed by the current author to work with the LDL^T decomposition of B updated according to the BFGS formula. The chemists had been drawn to this type of algorithm because their models could be trusted only locally; this paper was prompted by a chemist's reduced storage request.

In section 2 of this paper we construct an algorithm that combines the limited storage algorithms of Byrd, Nocedal, and Schnabel [4] with a dogleg–trust region approach in which a step s can be written as $s = \alpha\eta + \beta\nabla f(x)$, where η is the quasi-

Newton step and α and β are determined so that the step lies within the region of trust. Our algorithm requires at most $4nk + O(k^2)$ multiplications in overhead per iteration. The highest order term is the same as the highest order term required by a line search technique based on [4] that updates B^{-1}. If an algorithm requires B, the schemes given in [4] require $4nk + k^3/6 + O(k^2)$ multiplications. The updating techniques given in this paper can be combined with a limited memory line search approach to yield an algorithm that does not require $O(k^3)$ operations per iteration even when B is required. In section 3 we present some computational evidence that indicates that the limited memory dogleg approach is a viable alternative.

Burke and Weigmann [3] have recently proposed an algorithm for limited memory based on the trust region framework given in Moré and Sorensen [12], which does not use a double-dogleg step but is based on solving $(B + \mu I)d = \nabla f(x)$ for a prescribed value of μ. Their updating scheme also is based on [4] and sometimes has a $k^3/6$ component in the operation count for an iteration.

2. Algorithms. In dogleg–trust region methods the matrix B, the approximate quasi-Newton Hessian, and its inverse appear in several contexts. Given the current iterate x, its function value $f(x)$, the ∇g at x, and the current approximation of the Hessian B, each iteration of the algorithm of Dennis and Mei [7] as explained in Dennis and Schnabel [8] is essentially as follows.

```
1. Compute the quasi-Newton direction $\eta = -B^{-1}g$.
2. If the quasi-Newton step is within the trust region, i.e., if $||\eta||_2 \leq \tau$, then
       set $s = \eta$,
   else
       if the trust region includes a point between $DD$ and
       $QN$ in Figure 1.1, i.e., if $||t\eta|| \leq \tau$ where $c = ||g||^4/((g^T B^{-1} g)(g^T B g))$
       and $t = 0.2 + 0.8c$, then
           set $s = \tau/||\eta||\eta$,
       else
           if the Cauchy step, $p = -(||g||/g^T Bg)g$, is outside the radius
           of trust, i.e., $||p|| \geq \tau$, then
                 set $s = -(\tau/||g||)g$
           else
                 set $s = p + \theta w$, where $w = t\eta - p$,
                 $\theta = \sigma/(\phi + (\phi^2 + ||w||^2\sigma)^{1/2})$, $\sigma = \tau^2 - ||p||^2$, and $\phi = p^T w$.
3. Set the new $x$, $x^+ = x + s$. If $f(x^+) > f(x)$,
       set $\tau = \tau/2$ and go to step 2.
4. The radius $\tau$ can be updated as follows:
   The predicted function difference, $df_m$, is $g^T s + 0.5 s^T Bs$.
   If $(f(x) - f(x^+)) > 0.1 df_m$, then
       set $\tau = 0.5||s||$ and go to step 5
   else
       if $|df_m - (f(x) - f(x^+))| \leq 0.1(f(x) - f(x^+))$ or
       $(f(x) - f(x^+)) < .75 df_m$ or $g^T s \geq 2x^T \nabla f(x^+)$, then
           set $\tau = 2||s||$,
       else
           set $\tau = ||s||$.
5. Update the $B$ matrix.
```

From step 1 on the trust region double-dogleg algorithm we see that we need to compute

$$\eta = B^{-1}\nabla f(x). \tag{2.1}$$

From step 2 we might need

$$\nabla f(x)^T B \nabla f(x). \tag{2.2}$$

From step 4 we need to compute

$$s^T Bs. \tag{2.3}$$

The fact that one often can write

$$s = \alpha\eta + \beta\nabla f(x) \tag{2.4}$$

means that $Bs = -\alpha\nabla f(x) + \beta B\nabla f(x)$, so that the product Bs implied in (2.3) really does not have to be computed as long as $B\nabla f(x)$ itself is available or β is zero. In [6], $s = \alpha\eta + \beta D\nabla f(x)$ for some diagonal matrix D, but (2.2) is changed to $\nabla f(x)^T DBD\nabla f(x)$. Thus it appears that at each iteration one needs a representation of B^{-1} that can be inserted into (2.1), and if the quasi-Newton step is outside the trust radius, one needs a representation of B to insert into (2.2).

The ease with which the quantities in (2.1)–(2.3) are computed depends on one's representation of B. In [4] Byrd, Nocedal, and Schnabel suggest the following compact representation based on the BFGS update scheme for the approximate Hessian.

Let x_k represent the kth quasi-Newton iterate, $g(x_k) = \nabla f(x_k)$, $s_k = x_k - x_{k-1}$, and $y_k = g_k - g_{k-1}$. Let $p = \max(1, k - \hat{k})$, where $\hat{k}$ represents the number of iterates for which one has sufficient storage, and let $k' = \min(\hat{k}, k)$. Let

$$S_k = [s_p, \ldots, s_k], \qquad Y_k = [y_p, \ldots, y_k].$$

Let E_k be the $k' \times k'$ diagonal matrix

$$E_k = \operatorname{diag}[s_p^T y_p, \ldots, s_k^T y_k]$$

and let $Z^{(k)}$ be the $k' \times k'$ lower triangular matrix

$$z_{i,i'}^{(k)} = \begin{cases} s_{i+p-1}^T y_{i'+p-1} & \text{if } i > i', \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

Let C_k be the $2k' \times 2k'$ symmetric indefinite matrix

$$C_k = \begin{bmatrix} -E_k & Z^{(k)^T} \\ Z^{(k)} & \delta S_k^T S_k \end{bmatrix}. \tag{2.6}$$

Then, as Byrd, Nocedal, and Schnabel show in [4], the BFGS approximation B_k to the Hessian matrix can be written as

$$B_k = \delta I - [Y_k : \delta S_k]\, C_k^{-1} \begin{bmatrix} Y_k^T \\ \delta S_k^T \end{bmatrix}. \tag{2.7}$$

If we let

$$A_k = [Y_k : S_k],$$

then we may write (2.7) as

$$B_k = \delta I - A_k W_k A_k^T, \tag{2.8}$$

where

$$W_k = \left[\begin{array}{cc} I & 0 \\ 0 & \delta \end{array}\right] C_k^{-1} \left[\begin{array}{cc} I & 0 \\ 0 & \delta \end{array}\right].$$

As shown in [4], one can express C_k^{-1} as

$$C_k^{-1} = \left[\begin{array}{cc} -E_k^{1/2} & E_k^{-1/2} Z^{(k)^T} \\ 0 & J_k^T \end{array}\right]^{-1} \left[\begin{array}{cc} E_k^{1/2} & 0 \\ -Z^{(k)} E_k^{-1/2} & J_k \end{array}\right]^{-1}, \tag{2.9}$$

where J_k is a *lower* triangular matrix satisfying

$$J_k J_k^T = V_k = \delta S_k^T S_k + Z^{(k)} E_k^{-1} Z^{(k)^T}. \tag{2.10}$$

Because we are implementing a quasi-Newton method, we may assume that $s_k = B_k y_k$, which implies that $y_k^T s_k = y_k^T B_k y_k > 0$ since B is positive definite. Thus E is positive definite, and $E^{1/2}$ exists and can be written as a positive definite diagonal matrix. Also, V_k in (2.10) must be positive definite and J_k must exist.

Byrd, Nocedal, and Schnabel [4] show that for the BFGS update in (2.7), the matrix $B_k^{-1} = H_k$ can be expressed as

$$H_k = \delta^{-1} I + A_k \left[\begin{array}{cc} 0 & \delta^{-1} I \\ T^{(k)-T} & 0 \end{array}\right] \left[\begin{array}{cc} (E_k + \delta^{-1} Y_k^T Y_k) & -I \\ -I & 0 \end{array}\right] \left[\begin{array}{cc} 0 & T^{(k)^{-1}} \\ \delta^{-1} I & 0 \end{array}\right] A_k^T, \tag{2.11}$$

where

$$t_{i,i}^{(k)} = \begin{cases} s_{i+p-1}^T y_{i'+p-1} & \text{if } i \le i', \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

Since the matrix T is upper triangular, it is easy to apply T^{-1} to a vector by simply solving the appropriate upper triangular system.

Thus determining H_k does not require refactoring any matrix and is quite straightforward.

Given the representation in (2.11) one can compute $\eta = -H_k \nabla f(x_k)$ in (2.1) as follows.

```
ALGORITHM QN.
    1. Determine $u_k = A_k^T \nabla f(x_k)$.
    2. Partition $u$ as $\left[\begin{array}{c} u_1 \\ u_2 \end{array}\right] \begin{array}{c} k' \\ k' \end{array}$ .
    3. Solve $T^{(k)} w = u_2$
    4. Set $m = (E_k + \delta^{-1} Y_k^T Y_k) w - \delta^{-1} u_1$.
    5. Solve $T^{(k)^T} p = m$ .
    6. Set $\eta = -\delta^{-1} g_k - A_k \left[\begin{array}{c} -\delta^{-1} w \\ p \end{array}\right]$ .
```

Because E_k is diagonal and $T^{(k)}$ is triangular, Algorithm QN requires $4k'n + 2k'^2 + O(k') + O(n)$ multiplications. If u is available from a previous computation, then only $2k'n + 2k'^2 + O(k') + O(n)$ multiplications are necessary.

If the quasi-Newton step is not within the trust region, then one needs to compute $g_k^T B_k g_k$, which can be determined using the intermediate quantities of Algorithm QN as follows.

Algorithm CBG.

1. Solve $\left[\begin{array}{cc} E_k^{1/2} & 0 \\ -Z^{(k)}E_k^{-1/2} & J_k \end{array}\right] t = \left[\begin{array}{c} u_1 \\ \delta u_2 \end{array}\right]$.
2. Partition t as $\left[\begin{array}{c} t_1 \\ t_2 \end{array}\right] \begin{array}{c} k' \\ k' \end{array}$.
3. $\nabla f(x_k)^T B_k \nabla f(x_k) = \delta \nabla f(x_k)^T \nabla f(x_k) + t_1^T t_1 - t_2^T t_2$.

Because E_k is diagonal and $Z^{(k)}$ is triangular, the cost of Algorithm GBG is only $k'^2 + O(n) + O(k')$ multiplications. There is no nk' term in the operation count because we were able to use partial results from Algorithm QN. This same saving does not occur with the standard Cholesky factorization of B_k and is one of the major benefits of this approach.

Once a step has been determined one needs to adjust the trust radius τ and update the representations of B_k and H_k. The calculation of the trust region depends on (2.3). For (2.3), we get

$$s_{k+1}^T B_k s_{k+1} = -\alpha^2 \eta^T g_k + \beta^2 g_k^T B_k g_k - 2\alpha\beta g_k^T g_k, \tag{2.13}$$

since $B_k \eta = -g_k$. Computing the right-hand side of (2.13) requires at most $O(n)$ multiplications because either $\beta = 0$ or $g_k^T B_k g_k$ has been precomputed.

2.1. Updating B_k and H_k.

At the kth iteration the computation of H_{k+1} requires the matrix products $Y_k^T y_{k+1}$ and $S_k^T y_{k+1}$, i.e., $A_k^T y_{k+1}$. Since $y_{k+1} = \nabla f(x_{k+1}) - \nabla f(x_k)$, $A_k^T y_{k+1} = A_k^T \nabla f(x_{k+1}) - A_k^T \nabla f(x_k) = A_k^T \nabla f(x_{k+1}) - u_k$, where u_k was used and computed in Algorithm QN. If one computes $u_{k+1} = A_{k+1}^T \nabla f(x_{k+1})$ and saves it for the next application of Algorithm QN, then the work involved in Algorithm QN is reduced by $2nk'$ multiplications.

In general, to have a representation of C_{k+1}^{-1} in the formula for B_{k+1}, one needs $Z^{(k+1)}$ and $S_{k+1}^T S_{k+1}$, which in turn requires $Y_k^T s_{k+1}$ and $S_k^T s_{k+1}$, i.e., $A_k^T s_{k+1}$. Normally one would expect that the computation of $A_k^T s_{k+1}$ would require $2nk'$ multiplications. However, from (2.4), if $\alpha = 0$, we get $A_k^T s_{k+1} = \beta u_k$, which requires only $O(n)$ multiplications, and if $\alpha \neq 0$, from (2.11) and Algorithm QN, we see that

$$A_k^T s_{k+1} = (\beta - \alpha\delta^{-1}) u_k - \alpha A_k^T A_k \left[\begin{array}{c} -\delta^{-1} w \\ p \end{array}\right], \tag{2.14}$$

which costs $4k'^2$ multiplications because the matrix $A_k^T A_k$ is readily available.

We now turn our attention to the computation of J_k in (2.10). Byrd, Nocedal, and Schnabel [4] assume that k' is so small compared to n that the $O(k'^3)$ work involved with computing J_k from scratch each iteration will be dwarfed by the $O(nk')$ operations involved in multiplying $A_k^T \nabla f(x_k)$. In our work we will not make that assumption and we will try to avoid operations that involve $O(k'^3)$ multiplications.

If $k < \hat{k}$, at the next iteration, S_k will gain a column and $Z^{(k)}$, a strictly lower triangular matrix, will gain a row. Thus the matrix V_k in (2.10) will gain another row and column, and we find that V_{k+1} will have the form

$$V_{k+1} = \left[\begin{array}{cc} V_k & v \\ v^T & \beta \end{array}\right], \tag{2.15}$$

where $v^T = \delta s_{k+1}^T S_k + z^T E_k^{-1} Z^{(k)^T}$, $\beta = \delta s_{k+1}^T s_{k+1} + z^T z / e_{k+1}$, and z^T is the kth row of $Z^{(k)}$. Similarly, J_{k+1} has the form

$$J_{k+1} = \left[\begin{array}{cc} J_k & 0 \\ p^T & \gamma \end{array}\right], \tag{2.16}$$

where $p = J_k^{-1} v$ and $\gamma^2 = \beta - (p^T p)$. Updating J requires $k^2 + O(k)$ multiplications.

When $k > \hat{k}$, information must first be removed from S_k before the above algorithm is begun. Downdating essentially means that S_k would lose its first column and the lower triangular matrix $Z^{(k)}$ its first column, z_1. Removing the z_1 information from J entails forming the Cholesky factorization $\tilde{J}\tilde{J}^T$ of the matrix

$$\tilde{V} = V_k - e_1^{-1} z_1 z_1^T \tag{2.17}$$

for which five algorithms are given in [10]. In our implementation we have used the second one which requires $1.5k'^2 + O(k')$ multiplications.

An algorithm for removing the S information from J_k can be derived by noticing that J_k is the transpose of the R matrix in the QR decomposition of the matrix

$$F = \left[\begin{array}{c} \delta^{1/2} S_k \\ E_k^{-1/2} Z^{(k)^T} \end{array} \right], \tag{2.18}$$

since $V_k = F^T F$. Note that F initially has the structure

$$\left[\begin{array}{cccc} x & x & x & x \\ .. & . & . & . \\ x & x & x & x \\ - & - & - & - \\ & x & x & x \\ & & x & x \\ & & & x \end{array} \right]. \tag{2.19}$$

Since the QR decomposition of F is given by

$$F = Q_F \left[\begin{array}{c} J_k^T \\ 0 \end{array} \right], \tag{2.20}$$

eliminating the first column of S_k is equivalent to removing the first column of F in (2.19). This, in turn, is equivalent to deleting the first row from J_k, which is also the first row of $\tilde{J}$. This leaves us with a matrix $\tilde{J}$ of the form

$$\left[\begin{array}{ccccc} x & x & & & \\ x & x & x & & \\ .. & . & . & x & \\ .. & . & . & . & x \\ x & x & x & x & x \end{array} \right]. \tag{2.21}$$

Because $\tilde{J}^T$ is part of the QR decomposition of the F matrix, we are free to apply orthogonal transformations to its rows (i.e., to the columns of (2.21)) to return it to triangular form. Givens transformations applied successively to the columns of (2.21) in the planes $(1,2), (2,3), \ldots, (k', k'-1)$ can be used to return it to lower triangular form. Assuming four multiplications per Givens transformation, this part of the algorithm requires $2k'^2 + o(k')$ multiplications.

Table 2.1 below summarizes the operation count for the whole algorithm according to the type of step taken. In the rest of the paper we will call this approach ATA, indicating that it is based on $A^T A$.

Computing a decomposition of J from scratch each time rather than performing downdates and updates incurs a cost of $k'^2/2 + k'^3/6$ multiplications per iteration, which is efficient only if $k' < 18$ and there are many iterations.

Table 2.1
Multiplication counts of each iteration depending on the type of step.

Task	Quasi-Newton step	Not quasi-Newton step
Compute QN step	$2nk' + 2k'^2$	$2k'n + 2k'^2$
Compute g^TBg		k'^2
Updating A^TA	$2nk' + 4k'^2$	$2nk' + 4k'^2$
Downdating J	$3.5k'^2$	$3.5k'^2$
Updating J	k'^2	k'^2
Total	$4nk' + 10.5k'^2$	$4nk' + 11.5k'^2$

The line search quasi-Newton algorithms in [4] require at least $4k'n + O(k'^2)$ multiplications, so that the line search schemes and ATA have the same leading term. Those in [4] based on updating B compute a decomposition of V_k in (2.9) rather than updating the decomposition as detailed in (2.15)–(2.21), and thus incur an additional cost of $k'^3/6$ multiplications per iteration, which is consequential when, say, $k > 18$. Thus the algebraic overhead costs for the trust region approach per iteration are not greater than those for the line search approach. On any given problem any comparison between the two limited memory algorithms rests mainly on the effectiveness of the line search scheme versus the trust region mechanism. In our experience the line search approach usually requires more function evaluations per iteration but may require fewer iterations than the trust region method if the quadratic model used by the trust region approach is not "trustworthy." In the limited memory approach, where some information is discarded, the reliability of the model tends to be more sensitive to the value of δ than the nonlimited memory trust region algorithm and may require higher values of k than the limited memory line search approach.

2.2. Using the QR decomposition. Throughout our description of the algorithm we have encountered submatrices of $A_k^TA_k$. Rather than forming this matrix explicitly, one could take a hint from Nazareth [13] and form the QR decomposition of the $n \times 2k'$ matrix A_k of rank m, given by

$$A_k = Q_k \begin{bmatrix} R_k \\ 0 \end{bmatrix}, \tag{2.22}$$

where Q_k is an orthogonal matrix and R_k is an $m \times 2k'$ upper trapezoidal matrix. Rather than storing Q_k, one would store only its first m columns, which we will call $\hat{Q}_k$. The main advantage of using (2.22) is numerical stability. The A_k matrix itself would not have to be stored. Whenever it is needed for matrix-vector multiplication, (2.22) would be used. If one denoted the first k' columns of R_k as R_Y, then Y^TY in (2.11) could be written simply as $R_Y^TR_Y$. The product would never be formed, but to produce $z = Y^TYx$ one would set $v = R_Yx$ followed by $z = R_Y^Tv$. Actually, it would be more practical to interleave the columns of A_k to form the matrix $\tilde{A}_k$. At each iteration $\tilde{A}_k$ would gain two columns corresponding to s_k and y_k. Using the Gram–Schmidt procedure with reorthogonalization, updating the QR decomposition of $\tilde{A}_k$ would cost $2nm(l + 1) + O(n) + O(k')$ multiplications, where l represents the number of reorthogonalization steps [5]. If $m = 2k'$, this cost is about double the cost of updating $A_k^TA_k$.

The cost of updating and downdating the decomposition depends on the rank of the matrix $\tilde{A}_k$. We will show that for $k \leq 2\hat{k}$, the rank m_k of A_k satisfies $m_k \leq k+1$.

Our proof depends on the matrix Φ_k, which has the structure

$$\Phi_k = [s_1, g_1, s_2, g_2, \ldots, s_k, g_k]. \tag{2.23}$$

Theorem 1. *Assuming the search direction is a linear combination of the steepest descent direction and the quasi-Newton direction, then for $k < 2\hat{k}$ the rank m'_k of Φ_k satisfies $m'_k \leq k + 1$.*

Proof. The proof is by induction on k. For $k = 1$, the matrix Φ_k has at most two linearly independent columns.

Assume $k < 2\hat{k}$ and Φ_k has rank m'_k, where $m'_k \leq k + 1$. The vectors s_{k+1} and g_{k+1} are appended to Φ_k to form Φ_{k+1}. Now from (2.4) and the intermediate quantities of Algorithm QN, we see that

$$s_{k+1} = (\beta - \alpha\delta^{-1})g_k + \delta^{-1}\alpha Y_k w - \alpha S_k p. \tag{2.24}$$

Since $y_k = g_k - g_{k-1}$, the columns of Y_k are linear combinations of the columns of Φ_k. Thus s_{k+1} is a linear combination of the columns of Φ_k, the rank of $(\Phi_k | s_{k+1})$ is m'_k, and the rank of Φ_{k+1} is at most $m'_k + 1$. From our induction hypothesis we get that $m'_{k+1} \leq k + 2$, thus proving the theorem. □

From Theorem 1, we can prove the following theorem.

Theorem 2. *Assuming the search direction is a linear combination of the steepest descent direction and the quasi-Newton direction, then for $k < 2\hat{k}$ the rank m_k of $\tilde{A}_k$ satisfies $m_k \leq k + 1$.*

Proof. Consider the matrix Ψ_k a $2k \times 2k$ matrix that has the form

$$\Psi_k = \begin{bmatrix} 1 & \psi & & & & & & \\ & 1 & -1 & & & & & \\ & & 1 & & & & & \\ & & & 1 & & -1 & & \\ & & & & 1 & & & \\ & & & & & 1 & & -1 \\ & & & & & & 1 & \\ & & & & & & & 1 \end{bmatrix},$$

where $\psi = -||g_0||/\tau$. Note that $s_1 = \psi^{-1} g_0$. The matrix $\tilde{A}_k = \Phi_k \Psi_k$ and hence has the same rank as Φ_k, since Ψ_k is nonsingular. □

In practice one would work with and save $\hat{Q}_k$, the first m_k columns of the Q matrix from the QR of Φ_k. Since $\hat{Q}_k$ has at most $k + 1$ columns, $\hat{Q}_k$ occupies about half the space of A_k, which is why Theorems 1 and 2 consider $k < 2\hat{k}$ rather than $k \leq \hat{k}$. Applying A_k to a vector x using (2.22) requires $kn + k^2 + O(k) + O(n)$ multiplications rather than $2nk$ multiplications if A itself were used. Updating (2.22) requires $2nk(l+1) + O(n+k)$ multiplications, and, without reorthogonalization, about the same amount required to compute $A_k^T A_k$.

Downdating the Gram–Schmidt QR decomposition of $\tilde{A}_k$ is much more expensive than the downdating in (2.10) because, as shown in [5], here one has to apply transformations to $\hat{Q}_k$, which means that we will be dealing with $O(nk)$ multiplications rather than $O(k^2)$. Between iterations $\hat{k}$ and $2\hat{k}$, one can cite two reasons for delaying the downdating process. First, if $\hat{k}$ is large enough, one may converge to the solution of the optimization problem before iteration $2\hat{k}$, so that the downdating process might be superfluous. Second, it costs less to wait, assuming that shedding the first two columns each time does not decrease the rank of $\tilde{A}_k$. From Theorem 2 we gather

Table 2.2
Multiplication count with several options assuming "combination step."

	Determining step and radius	Updating matrices	Downdating matrices	Total
ATA$(k < \hat{k})$	$2k'n + 3k'^2$	$2k'n + 5k'^2$		$4k'n + 8k'^2$
QR $(k < \hat{k})$	$k'n + 5k'^2$	$2k'n + 2k'^2$		$3k'n + 7k'^2$
ATA$(k \geq \hat{k})$	$2k'n + 3k'^2$	$2k'n + 5k'^2$	$3.5k'^2$	$4k'n + 11.5k'^2$
QR $(\hat{k} \leq k < 2\hat{k})$	$kn + 3k'^2 +$ $2k^2$	$2kn + 2k^2$		$3kn + 3k'^2 +$ $4k^2$
QR $(k = 2\hat{k})$	$4k'n + 7k'^2$	$4k'n + 2k'^2$	$2k'^2n$	$2k'^2n + 8k'n +$ $9k'^2$
QR $(k > 2\hat{k})$	$4k'n + 7k'^2$	$4k'n + 2k'^2$	$10k'n + 13k'^2$	$18k'n + 22k'^2$

that the matrix $\tilde{R}_k$ (corresponding to $\tilde{A}$) has the structure of (2.25). Downdating each iteration requires a sequence of three-plane Householder transformations that require about five multiplications per transformation. By iteration $2\hat{k}$ one would need $\hat{k}^2$ such transformations, so that the total cost of applying these transformations to $\hat{Q}_k$ is about $5\hat{k}^2n$ multiplications. If one waited until iteration $2\hat{k}$ and eliminated the first $2\hat{k}$ columns of R_k and reduced the next $2\hat{k}$ columns to triangular form, then one would be using longer Householder transformations, and the application of these to $\hat{Q}$ would require at most $2\hat{k}^2n$ multiplications. Thus postponing is cost effective.

$$\begin{bmatrix} x & x & x & x & x & . & x & x & x \\ & x & x & x & x & . & x & x & x \\ & & & x & x & . & x & x & x \\ & & & & & . & x & x & x \\ & & & & & . & x & x & x \\ & & & & & & x & x & x \\ & & & & & & & & x \end{bmatrix}. \tag{2.25}$$

After iteration $2\hat{k}$ downdating costs $10k'n + 10k'^2$ multiplications to apply the transformations to $\hat{Q}_k$ and $\tilde{R}_k$. Referring to Table 2.2 one realizes that downdating swamps all the other algebraic computations. Moreover, the $k'n$ term in the downdating multiplication count refers to vector operations and not to matrix-vector operations, which could be implemented with fast BLAS [3]. For some problems it may make sense just to begin again, throwing out old information, recomputing δ, and taking a steepest descent step. In the next section we will call this option RSQR, for "restart using the QR decomposition."

Table 2.2 provides the cost of the A^TA-based and the QR-based algorithms, assuming both α and β in (2.4) are nonzero. The major lesson one gleans from the table is that downdating using the QR decomposition is expensive.

3. Numerical examples. The reduced storage algorithms defined in section 2 were inserted into Dennis and Mei's MINOP [7] code restricted to the BFGS update. These modified codes were applied to two problems to determine if in practice there were significant differences between the algorithms. The codes were run on a four-processor SGI machine and terminated when the gradient decreased to 1×10^{-6}.

The first problem is the journal bearing problem discussed in [1] without the nonnegativity constraints. In that problem we set the eccentricity ϵ to .1 and b to 1. For the journal bearing problem, function and gradient evaluations account for about

TABLE 3.1
Function evaluations and time (sec) for the journal bearing problem.

	$n = 400$		$n = 1600$		$n = 2500$	
	Fn. eval.	Time	Fn. eval.	Time	Fn. eval.	Time
MINOP, $\hat{k} = n$	35	6.7				
LBFGS, $\hat{k} = 50$	55	.54	96	3.6	111	8.4
RSQR, $\hat{k} = 50$	43	.57	70	3.7	87	7.2
ATA, $\hat{k} = 50$	43	.52	68	4.0	85	8.5
LBFGS, $\hat{k} = 25$	55	.55	96	3.9	111	7.2
RSQR, $\hat{k} = 25$	41	.48	117	5.3	166	11.1
ATA, $\hat{k} = 25$	43	.53	65	3.3	88	6.8
LBFGS, $\hat{k} = 10$	55	.50	102	3.6	111	6.0
RSQR, $\hat{k} = 10$	60	.64	163	6.4	238	15.1
ATA, $\hat{k} = 10$	42	.45	71	3.0	109	6.6

half the total computation time in the limited memory codes we tested.

In Table 3.1 we compare the original MINOP code algorithm to ATA and RSQR and LBFGS, a limited memory line search BFGS code discussed in [11]. For $n = 400$ the decrease in the number of linear algebra computations per iteration with the reduced storage schemes significantly reduces the overall time. This is definitely a situation in which one would want to use these types of schemes over the straight quasi-Newton codes.

From Table 3.1 one can make several observations. There was not a great deal of difference between the timings of the line search routine and ATA. The line search routine tended to take two function evaluations per iteration, while ATA usually took one, so that the number of function evaluations of LBFGS tended to be higher but fewer iterations were required. For this problem the trust region routines were more sensitive to the value of $\hat{k}$ and the smaller the value of $\hat{k}$, the more function evaluations were required. This was particularly true of the algorithm RSQR that restarted every $\hat{k}$ iterations.

The results for ATA and RSQR were very dependent on the choice of δ. We followed the suggestion of Shanno and Phua [16] and Oren and Spedicato [15] that δ be set initially to $y^T y/s^T y$, where $y = \nabla f(x_1) - \nabla f(x_2)$ and $s = x_1 - x_2$. For restarting RSQR, this formula was used. For this choice of δ, most of the steps were in the quasi-Newton direction. For the problem with $n = 2500$ and $\hat{k} = 25$, we multiplied the initial δ by 10 and found that most of the steps were combination steps, and for the same accuracy, 204 function evaluations were needed and the computation required 19 seconds. When we divided the initial δ by 10, again most of the steps were combination steps, the number of function evaluations increased to 333, and the computation required 22 seconds.

The second example was a small ($n = 167$) version of a molecular dynamics problem involving silicon and oxygen, which had prompted this research. The problem involved finding "unique" vectors z_i in 3-space, which minimized

$$f = \sum_i \sum_j a_{ij}/d_{ij} + b_{ij}/d_{ij}^6 + c_{ij}e^{(h_{ij}-d_{ij})}, \tag{3.1}$$

where $d_{ij} = ||z_i - z_j||_2^2$ and the a's, b's , c's, and h's were known constants. Unfortunately, there were some negative values of $b_{i,j}$, and for these values, if $z_i = z_j$, then $f = -\infty$. The application had many local minima, and different minima were found

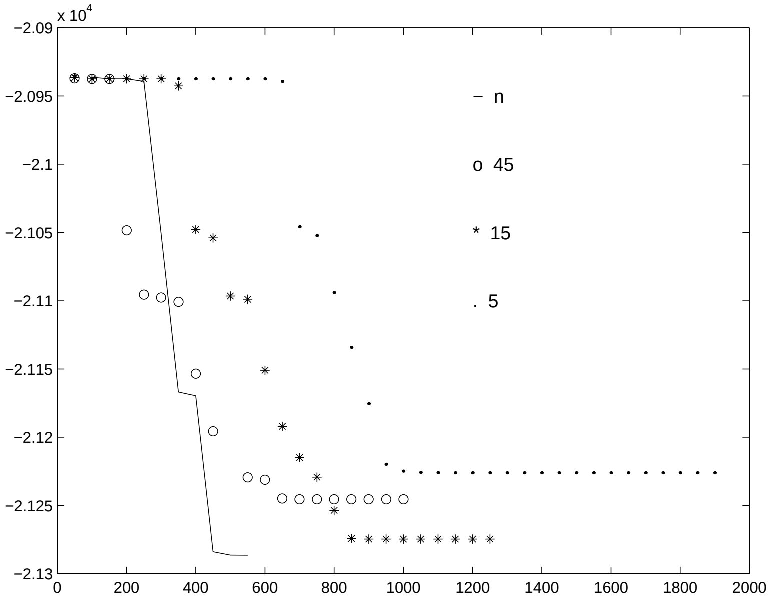


FIG. 3.1. *Number of iterations for various values of $\hat{k}$ for the silicon problem.*

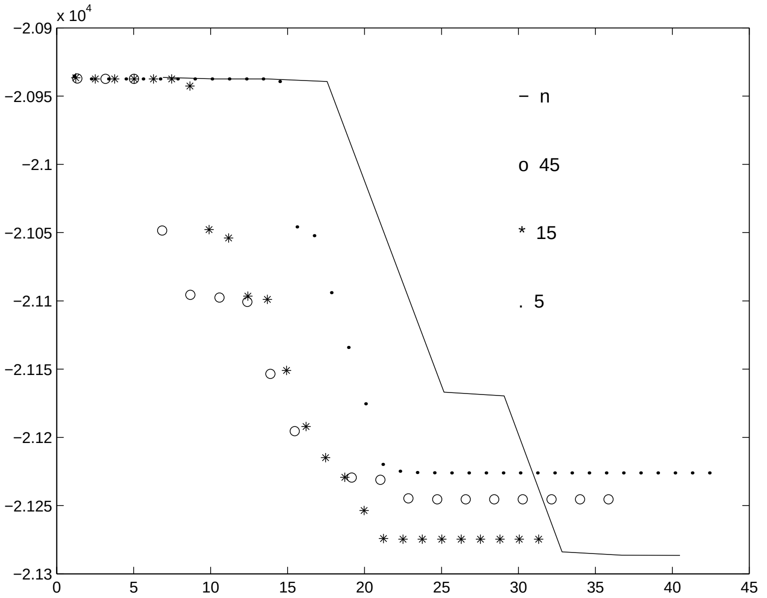


FIG. 3.2. *Times for various values of $\hat{k}$ for the silicon problem.*

as $\hat{k}$ and δ were varied. This made an exact comparison rather difficult.

Figure 3.1 plots the function values versus the number of function evaluations for the original nonreduced storage algorithm to the ATA scheme for several values of $\hat{k}$, and Figure 3.2 compares the time in seconds on our SGI machine.

Analytic gradients were computed along with the function values to take advantage of common subexpressions. The initial value of δ given above was also used here. As expected, the original scheme, which does not try to minimize storage, required the least number of iterations, and the smaller the value of $\hat{k}$, the larger the number of function evaluations. However, in terms of time, all the runs were similar. Increasing δ by a factor of 10 sometimes improved the results and sometimes led to a different minimum. The restarting scheme RSQR tended to require more iterations than the ATA scheme near the local minima. The function evaluation profile and time profile for RSQR with $\hat{k} = 45$ resembled that of ATA with $\hat{k} = 5$. The line search routine found the solution at $-\infty$ and balked whenever the user-specified bound on the line search step was shortened or a penalty term was added to (3.1) to prevent the d_{ij}'s from going to zero.

4. Conclusion. We have shown that one can construct a reduced storage dogleg–trust region quasi-Newton code that not only reduces the storage significantly, but can also reduce the total computation time over the traditional quasi-Newton dogleg–trust region methods on some problems. This scheme is competitive with the limited memory line search program LBFGS. Its algebraic overhead per iteration theoretically is either the same or less than limited memory line search program algorithms depending on how the line search routine is implemented. The same updating techniques that were described here could be introduced in some line search algorithms to lower their operation count. In practice, the reduced storage dogleg approach is very sensitive to the settings of some of its parameters and its performance is dependent on whether the model is "trustworthy."

Acknowledgment. I would like to thank the referees for their insightful comments and careful reading of the manuscript.

REFERENCES

[1] B. Averick, R. Carter, and J. Moré , *The Minpack-2 Test Problem Collection*, ANL-MCS-TM-150, Argonne National Laboratory, Argonne, IL, 1991.

[2] A. Buckley and A. LeNir *QN-like variable storage conjugate gradients*, Math. Programming, 27 (1983), pp. 155–175.

[3] J. Burke and A. Wiegmann, *Notes on limited memory BFGS updating in a trust-region framework*, SIAM J. Optim., submitted.

[4] R. Byrd, J. Nocedal, and R. Schnabel, *Representation of quasi-Newton matrices and their use in limited memory methods*, Math. Programming, 63 (1994), pp. 129–156.

[5] J. Daniel, W. Gragg, L. Kaufman, and G.W. Stewart, *Stable algorithms for updating the Gram–Schmidt QR factorization*, Math. Comp., 30 (1976), pp. 772–795.

[6] J. Dennis, D. Gay, and R. Welsch, *Algorithm 611 subroutines for unconstrained minimization using a model/trust-region approach*, ACM Trans. Math. Software, 9 (1983), pp. 503–524.

[7] J. Dennis and H. Mei, *An unconstrained optimization algorithm which uses function and gradient values*, J. Optim. Theory Appl., 28 (1979), pp. 455–480.

[8] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice–Hall, Englewood Cliffs, NJ, 1983; reprinted as Classics Appl. Math. 16, SIAM, Philadelphia, PA, 1996.

[9] J. Dongarra, J. DuCroz, S. Hammerling, and R. Hanson, *An extended set of Fortran basic linear algebra solvers*, ACM Trans. Math. Software, 14 (1989), pp. 1–17.

[10] P. Gill, G. Golub, W. Murray, and M. Saunders, *Factorized variable metric methods for unconstrained optimization*, Math. Comp., 30 (1976), pp. 796–811.

[11] D. Liu and J. Nocedal, *On the limited memory BFGS method for large scale optimization*, Math. Programming, 45 (1989), pp. 503–528

[12] J. Moré and D. Sorensen, *Computing a trust region step*, SIAM J. Sci. Stat. Comput., 4 (1983), pp. 553–572.

[13] L. Nazareth, *The method of successive affine reduction for nonlinear minimization*, Math. Programming, 35 (1986), pp. 97–109.

[14] J. Nocedal, *Updating quasi-Newton matrices with limited storage*, Math. Comp., 35 (1980), pp. 773–782.

[15] S. Oren and E. Spedicato, *Optimal conditioning of self-scaling and variable metric algorithms*, Math Programming, 10 (1976), pp. 70–90.

[16] D. Shanno and K. Phua, *Matrix conditioning and nonlinear optimization*, Math. Programming, 14 (1978), pp. 145–160.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 70–81

A POLYNOMIAL TIME ALGORITHM FOR SHAPED PARTITION PROBLEMS*

FRANK K. HWANG†, SHMUEL ONN‡, AND URIEL G. ROTHBLUM‡

Abstract. We consider the class of *shaped partition problems* of partitioning n given vectors in d-dimensional criteria space into p parts so as to maximize an arbitrary objective function which is convex on the sum of vectors in each part, subject to arbitrary constraints on the number of elements in each part. This class has broad expressive power and captures NP-hard problems even if either d or p is fixed. In contrast, we show that when both d and p are fixed, the problem can be solved in strongly polynomial time. Our solution method relies on studying the corresponding class of *shaped partition polytopes*. Such polytopes may have exponentially many vertices and facets even when one of d or p is fixed; however, we show that when both d and p are fixed, the number of vertices of any shaped partition polytope is $O(n^{d\binom{p}{2}})$ and all vertices can be produced in strongly polynomial time.

Key words. partition, cluster, optimization, convex, polytope, enumeration, polynomial time, separation, programming

AMS subject classifications. 05A, 15A, 51M, 52A, 52B, 52C, 68Q, 68R, 68U, 90B, 90C

PII. S1052623497344002

1. Introduction.

The *partition problem* concerns the partitioning of vectors $A^1, \ldots, A^n$ in d-space into p parts so as to maximize an objective function which is convex on the sum of vectors in each part; see [3]. Each vector A^i represents d numerical attributes associated with the ith element of the set $[n] = \{1, \ldots, n\}$ to be partitioned. Each ordered partition $\pi = (\pi_1, \ldots, \pi_p)$ of $[n]$ is then associated with the $d \times p$ matrix $A^\pi = \left[\sum_{i\in\pi_1} A^i, \ldots, \sum_{i\in\pi_p} A^i\right]$ whose jth column represents the total attribute vector of the jth part. The problem is to find an *admissible* partition π which maximizes an objective function f given by $f(\pi) = C(A^\pi)$, where C is a real convex functional on $\mathbb{R}^{d\times p}$. Of particular interest is the *shaped partition problem*, where the admissible partitions are those π whose *shape* $(|\pi_1|, \ldots, |\pi_p|)$ lies in a prescribed set Λ of admissible shapes. In this article we concentrate on this later situation.

The shaped partition problem has applications in diverse fields that include circuit layout, clustering, inventory, splitting, ranking, scheduling, and reliability; see [5, 9, 14, 15] and references therein. Further, as we demonstrate later, the problem has expressive power that captures NP-hard problems such as the max-cut problem and the traveling salesman problem, even when the number p of parts or attribute dimension d is fixed.

Our first goal in this article is to demonstrate constructively that a polynomial time algorithm for the shaped partition problem *does* exist when both p and d are fixed. This result is valid when the set Λ of admissible shapes and the function C are

[*] Received by the editors December 18, 1997; accepted for publication (in revised form) September 17, 1998; published electronically October 20, 1999. The research of the second author was supported in part by the Mathematical Sciences Research Institute at Berkeley, CA, through NSF grant DMS-9022140, by the N. Haar and R. Zinn Research Fund at the Technion, and by the Fund for the Promotion of Research at the Technion. The research of the third author was supported in part by the E. and J. Bishop Research Fund at the Technion and by ONR grant N00014-92-J1142.

http://www.siam.org/journals/siopt/10-1/34400.html

[†] Department of Applied Mathematics, Chiaotung University, Hsinchu, 30045, Taiwan (fhwang@math.nctu.edu.tw).

[‡] William Davidson Faculty of Industrial Engineering and Management, Technion—Israel Institute of Technology, 32000 Haifa, Israel (onn@ie.technion.ac.il, rothblum@ie.technion.ac.il).

presented by oracles. Our first result (formally stated and proved in section 4) is the following:

- *Theorem* 4.2: Any shaped partition problem is solvable in polynomial oracle time using $O(n^{dp^2})$ arithmetic operations and queries.

Our solution method is based on the observation that since C is convex, the shaped partition problem can be embedded into the problem of maximizing C over the *shaped partition polytope* $\mathcal{P}_A^\Lambda$ defined to be the convex hull of all matrices A^π corresponding to partitions of admissible shapes. The class of shaped partition polytopes is very broad and generalizes and unifies classical permutation polytopes such as Birkhoff's polytope and the permutohedron (see, e.g., [8, 19, 21]). Its subclass of *bounded* shaped partition polytopes with lower and upper bounds on the shapes was previously studied in [3], under the assumption that the vectors $A^1, \ldots, A^n$ are *distinct.* Therein a polynomial procedure for testing whether a given A^π is a vertex of $\mathcal{P}_A^\Lambda$ was obtained. This procedure was simplified and extended in [11]. A related but different generalization of classical permutation polytopes, arising when algebraic (representation-theoretical) constraints, rather than shape constraints, are imposed on the permutations involved, was studied in [19] and references therein.

Since a shaped partition polytope is defined as the convex hull of an implicitly presented set whose size is typically exponential in the input size even when both p and d are fixed, an efficient representation as the convex hull of vertices or as the intersection of half-spaces is not readily expected. Our second objective is to prove that, nevertheless, for fixed p and d, the number of vertices of shaped partition polytopes *is* polynomially bounded in n, and that it is possible to explicitly enumerate all vertices in polynomial time. Thus, our second result (formally stated and proved in section 4) is the following:

- *Theorem* 4.3: Any shaped partition polytope $\mathcal{P}_A^\Lambda$ has $O(n^{d\binom{p}{2}})$ vertices which can be produced in polynomial oracle time using $O(n^{d^2p^3})$ arithmetic operations and queries.

An immediate corollary of Theorem 4.3 is that, for fixed d, p, the number of facets of $\mathcal{P}_A^\Lambda$ is polynomially bounded as well and that all facets can be produced in polynomial oracle time (Corollary 4.4). Theorem 4.3 shows, in particular, that it is possible to compute the *number* of vertices efficiently. This might be extendable to the situation of variable d and p, where counting vertices is generally a hard task (cf. [16]), as is counting partitions with various prescribed properties (see [4, 10]). The vertex counting problem for variable d and p will be addressed elsewhere.

A special role in our development is played by *separable* partitions, defined as partitions where vectors in distinct sets are (weakly) separable by hyperplanes. In the special case $d = p = 2$, such partitions had been studied quite extensively (see, e.g., [2, 5, 7, 17]). The case $d = 3, p = 2$ has also been considered quite recently in [6]. Here we study such partitions for all d, p, as well as a class of *generic* partitions, and provide an upper bound on their number and an algorithm for producing them. In our recent related work [1], the precise extremal asymptotical behavior of such partitions is determined.

The embedding of the partition problem into the problem of maximizing the convex function C over the partition polytope is useful due to the optimality of vertices in the latter problem. When Λ consists of a single shape, the optimality of vertices holds for the more general class of asymmetric Schur convex functions, introduced in [13]; see [8]. All of our results apply with C as any asymmetric Schur convex function and Λ consisting of a single shape.

The article is organized as follows. In the next section we formally define the shaped partition problem and shaped partition polytope. We demonstrate the expressive power of this problem by giving four examples. For the first two examples, in which the parameters d, p are typically small and fixed, Theorem 4.2 provides a polynomial time solution. The last two examples show that the max-cut problem and traveling salesman problem can be modeled as shaped partition problems with fixed $p = 2$ and $d = 1$, respectively, and that the corresponding polytopes have exponentially many vertices. In section 3 we study separability properties of vertices of shaped partition polytopes and discuss separable and generic partitions. In the final section, section 4, we use our preparatory results of section 3 to establish Theorems 4.2 and 4.3 and Corollary 4.4.

2. Shaped partition problems and polytopes.

A *p-partition* of $[n] := \{1, \ldots, n\}$ is an ordered collection $\pi = (\pi_1, \ldots, \pi_p)$ of p disjoint sets (possibly empty) whose union is $[n]$. A *p-shape* of n is a tuple $\lambda = (\lambda_1, \ldots, \lambda_p)$ of nonnegative integers $\lambda_1, \ldots, \lambda_p$ satisfying $\sum_{i=1}^{p} \lambda_i = n$. The *shape of a p-partition* π is the p-shape of n given by $|\pi| := (|\pi_1|, \ldots, |\pi_p|)$. If Λ is a set of p-shapes of n, then a Λ-*partition* is any partition π whose shape $|\pi|$ is a member of Λ.

Let A be a real $d \times n$ matrix; for $i = 1, \ldots, n$, we use A^i to denote the ith column of A. For each p-partition π of $[n]$ we define the *A-matrix* of π to be the $d \times p$ matrix

$$A^\pi = \left[\sum_{i \in \pi_1} A^i, \ldots, \sum_{i \in \pi_p} A^i \right],$$

with $\sum_{i \in \pi_j} A^i := 0$ when $\pi_j = \emptyset$. We consider the following algorithmic problem.

Shaped Partition Problem. Given positive integers d, p, n, matrix $A \in \mathbb{R}^{d \times n}$, the nonempty set Λ of p-shapes of n, and the objective function on Λ-partitions given by $f(\pi) = C(A^\pi)$ with C convex on $\mathbb{R}^{d \times p}$, find a Λ-partition π^* that maximizes f and, specifically, satisfies

$$f(\pi^*) = \max\{f(\pi) : \ |\pi| \in \Lambda\}.$$

Of course, the complexity of this problem depends on the presentation of Λ and C, but we will construct algorithms that work in strongly polynomial time and can cope with minimal information on Λ and C. Specifically, we assume that the set of admissible p-partitions Λ can be represented by a membership oracle that, on query λ, answers whether $\lambda \in \Lambda$. The convex functional C on $\mathbb{R}^{d \times p}$ can be presented by an evaluation oracle that, on query A^π with π a Λ-partition, returns $C(A^\pi)$.

Since C is convex, the shaped partition problem can be embedded into the problem of maximizing C over the convex hull of A-matrices of feasible partitions, defined as follows.

Shaped Partition Polytope. For a matrix $A \in \mathbb{R}^{d \times n}$ and nonempty set Λ of p-shapes of n, we define the *shaped partition polytope* $\mathcal{P}_A^\Lambda$ to be the convex hull of all A-matrices of Λ-partitions, that is,

$$\mathcal{P}_A^\Lambda := \text{conv}\,\{A^\pi : \ |\pi| \in \Lambda\} \subset \mathbb{R}^{d \times p}.$$

We point out that for any A, the polytope $\mathcal{P}_A^\Lambda$ is the image of the shaped partition polytope P_I^Λ, with I the $n \times n$ identity, under the projection $X \mapsto AX$. In [12] this is exploited, for the situation where the function C is *linear* and $\Lambda = \{\lambda : \ l \leq \lambda \leq u\}$ is a set of *bounded* shapes, to solve the corresponding shaped partition problem for all n, d, p in polynomial time by linear programming over P_I^{Λ}.

We now demonstrate the expressive power of the shaped partition problem. In particular, we show that even if one of d or p is fixed, the shaped partition problem may be NP-hard, and the number of vertices of the shaped partition polytope may be exponential. Therefore, polynomial time algorithms for optimization and vertex enumeration are expected to (and, as we show, *do*) exist only when *both* d and p are fixed. We start with two examples in which it is natural to have d and p small and fixed.

EXAMPLE 2.1 (splitting). *The n assets of a company are to be split among its p owners as follows. For $j = 1, \ldots, p$, the jth owner prescribes a nonnegative vector $A_j = (A_{j,1}, \ldots, A_{j,n})$ with $\sum_{j=1}^{n} A_{i,j} = 1$, whose entries represent the relative values of the various assets to this owner. A partition $\pi = (\pi_1, \ldots, \pi_p)$ is sought which splits the assets among the owners and maximizes the l_q-norm $(\sum_{j=1}^{p} |\sum_{i\in\pi_j} A_{j,i}|^q)^{\frac{1}{q}}$ of the total value vector whose jth entry $\sum_{i\in\pi_j} A_{j,i}$ is the total relative value of the assets allocated to the jth owner by π. An alternative interpretation of the splitting problem concerns the division of an estate consisting of n assets among p inheritors having equal rights against the estate. With $p = 2$, the model captures a problem of a divorcing couple dividing their joint property* [5, 9].

FORMULATION: $n,\ d = p,\ A = (A_{j,i}),\ \Lambda = \{All\ p\text{-}shapes\},\ f(\pi) = C(A^{\pi})$ *with*

$$C : \mathbb{R}^{p\times p} \longrightarrow \mathbb{R} :\ M \mapsto \sum_{i=1}^{p} |M_{i,i}|^q.$$

For fixed p, Theorem 4.2 *asserts that we can find an optimal partition in polynomial time $O(n^{p^3})$, while the number p^n of Λ-partitions is exponential. We note that other (convex) functions C can be used within our framework. In particular, if C is linear on $\mathbb{R}_+^{p\times p}$, e.g., when $q = 1$, our results of* [12] *apply and yield a polynomial time solution even when p is variable.*

EXAMPLE 2.2 (balanced clustering). *Given are $n = pm$ objects represented by attribute vectors $A^1, \ldots, A^n \in \mathbb{R}^d$. The objects are to be grouped in p clusters, each containing m points, so as to minimize the sum of cluster variance of a partition π given by $\sum_{i=1}^{p} \left(\frac{1}{|\pi_i|}\sum_{j\in\pi_i} ||A^j - \bar{A}^{\pi_i}||^2\right)$, where $||\cdot||$ denotes the l_2-norm and $\bar{A}^{\pi_i} := \frac{1}{|\pi_i|}\sum_{j\in\pi_i} A^j$ is the barycenter of the ith cluster.*

FORMULATION: $n = pm,\ d,\ p,\ A = (A^1, \ldots, A^n),\ \Lambda = \{m^p = (m, \ldots, m)\},\ f(\pi) = C(A^{\pi})$ *with*

$$C : \mathbb{R}^{d\times p} \longrightarrow \mathbb{R} :\ M \mapsto ||M||^2 = \sum_{i=1}^{d}\sum_{j=1}^{p} M_{i,j}^2.$$

Here, we use the fact that $f(\pi) = \frac{1}{m^2}\sum_{i=1}^{n} ||A^i||^2 - \frac{1}{m^2}\sum_{j=1}^{p} ||\sum_{i\in\pi_j} A^i||^2$. For fixed d, p, by Theorem 4.2 *we can find an optimal balanced clustering in polynomial time $O(n^{dp^2})$, while the number of Λ-partitions is exponential $\Omega(p^n n^{\frac{1-p}{2}})$.*

The next two examples show that unless *both* d and p are fixed, the shaped partition problem may be NP-hard. The idea is simple: the formulation is such that every Λ-partition π gives a distinct vertex A^{π} of the shaped partition polytope $\mathcal{P}_A^{\Lambda}$. Then, *any* function f on Λ-partitions factors as $f(\pi) := C(A^{\pi})$ for suitable convex C

on $\mathcal{P}_A^\Lambda$, say, the one given by

$$C(X) := \inf\left\{\sum_{|\pi|\in\Lambda}\theta_\pi f(\pi):\ \sum_\pi \theta_\pi A^\pi = X,\ \sum_\pi \theta_\pi = 1,\ \theta_\pi \geq 0\right\}.$$

In the following examples, the membership oracle for Λ and the evaluation oracle for $f(\pi) := C(A^\pi)$, restricted to A-matrices, are easily polynomial time realizable from the natural data for the problem.

Example 2.3 (max-cut problem and unit cube). *Find a cut with maximum number of crossing edges in a given graph $G = ([n], E)$.*

Formulation: $n = d,\ p = 2,\ A = I_n,\ \Lambda = \{\textit{all } 2\textit{-shapes}\}$,

$$f(\pi) = \#\{e \in E:\ |e \cap \pi_1| = 1\}.$$

Here, the A-matrices of Λ-partitions are precisely all $(0,1)$-valued $n \times 2$ matrices with each row sum equal to 1; in particular, each such matrix is determined by its first column. It follows that the shaped partition polytope $\mathcal{P}_A^\Lambda$ has 2^n vertices that stand in bijection with Λ-partitions and is affinely equivalent to the n-dimensional unit cube by projection of matrices onto their first column. So, each A^π is a distinct vertex of $\mathcal{P}_A^\Lambda$ and there is a convex C on $\mathbb{R}^{d\times 2}$ such that $f(\pi) = C(A^\pi)$ for all π.

Example 2.4 (traveling salesman problem and permutohedron). *Find a shortest Hamiltonian path on n sites under a given symmetric nonnegative matrix D, where $D_{i,j}$ represents the distance between sites i and j.*

Formulation: $n = p,\ d = 1,\ A = (1, \ldots, n),\ \Lambda = \{1^n = (1, \ldots, 1)\}$,

$$f(\pi) = -\sum_{j=1}^{n-1} D_{\pi_j,\pi_{j+1}},$$

where we regard a partition simply as the corresponding permutation. The matrices A^π in this case are simply all permutations of A. The shaped partition polytope $\mathcal{P}_A^\Lambda$ has $n!$ vertices that stand in bijection with Λ-partitions, and is the so-called permutohedron. Since each A^π is a distinct vertex of $\mathcal{P}_A^\Lambda$, there is again a convex C on $\mathbb{R}^n$ such that $f(\pi) = C(A^\pi)$ for all π.

3. Vertices and generic partitions. In this section we show that every vertex of any shaped partition polytope $\mathcal{P}_A^\Lambda$ equals the A-matrix A^π of some *A-generic* partition, a notion that we introduce and develop below.

The convex hull of a subset U in $\mathbb{R}^d$ will be denoted $\operatorname{conv}(U)$. Two finite sets U, V of points in $\mathbb{R}^d$ are *separable* if there is a vector $h \in \mathbb{R}^d$ such that $h^T u < h^T v$ for all $u \in U$ and $v \in V$ with $u \neq v$; in this case, we refer to h as a *separating vector* of U and V. The proof of the following characterization of separability is standard and is left to the reader. It implies, in particular, that if U and V are separable, then $|U \cap V| \leq 1$.

Lemma 3.1. *Let U and V be finite sets of $\mathbb{R}^d$. Then U and V are separable if and only if their convex hulls either are disjoint or intersect in a single point that is a common vertex of both.*

Let A be a given $d \times n$ matrix. For a subset $S \subseteq [n]$, let $A^S = \{A^i :\ i \in S\}$ be the set of columns of A indexed by S (with multiple copies of columns identified). A p-partition $\pi = (\pi_1, \ldots, \pi_p)$ is *A-separable* if the sets A^{π_r} and A^{π_s} are separable for each pair $1 \leq r < s \leq p$, that is, if for each pair $1 \leq r < s \leq p$ there is a vector

$h_{r,s} \in \mathbb{R}^d$ such that $h_{r,s}^T A^i < h_{r,s}^T A^j$ for all $i \in \pi_r$ and $j \in \pi_s$ with $A^i \neq A^j$. We have the following lemma, which generalizes a result of [3] from matrices with no zero columns and no repeated columns.

LEMMA 3.2. *Let A be a matrix in $\mathbb{R}^{d\times n}$, let Λ be a nonempty set of p-shapes of n, and let π be a Λ-partition. If A^π is a vertex of $\mathcal{P}_A^\Lambda$, then π is an A-separable partition.*

Proof. The claim being obvious for $p = 1$, suppose that $p \geq 2$. Let A^π be a vertex of $\mathcal{P}_A^\Lambda$. Then there is a matrix $C \in \mathbb{R}^{d\times p}$ such that the linear functional on $\mathbb{R}^{d\times p}$ given by the inner product $\langle C, X\rangle = \sum_{i=1}^d \sum_{j=1}^p C_i^j X_i^j$ is uniquely maximized over $\mathcal{P}_A^\Lambda$ at A^π. Pick any pair $1 \leq r < s \leq p$, and let $h_{r,s} = C^s - C^r$. Suppose there are $i \in \pi_r$ and $j \in \pi_s$ with $A^i \neq A^j$ (otherwise A^{π_r} and A^{π_s} are trivially separable). Let π' be the Λ-partition obtained from π by switching i and j, i.e., taking $\pi'_r := \pi_r \cup \{j\} \setminus \{i\}$, $\pi'_s := \pi_s \cup \{i\} \setminus \{j\}$, and $\pi'_t := \pi_t$ for all $t \neq r, s$. Then $(A^{\pi'})^r = (A^\pi)^r + A^j - A^i \neq (A^\pi)^r$, and hence $A^{\pi'} \neq A^\pi$. By the choice of C, we have $\langle C, A^{\pi'}\rangle < \langle C, A^\pi\rangle$ and so

$$h_{r,s}^T(A^j - A^i) = (C^s - C^r)^T(A^j - A^i) = \sum_{t=1}^p (C^t)^T((A^\pi)^t - (A^{\pi'})^t) = \langle C, A^\pi - A^{\pi'}\rangle > 0.$$

This proves that A^{π_r}, A^{π_s} are separable for each pair $1 \leq r < s \leq p$, hence π is A-separable. □

We need some more terminology. Let $A \in \mathbb{R}^{d\times n}$. A p-partition $\pi = (\pi_1, \ldots, \pi_p)$ of $[n]$ is A-*disjoint* if $\operatorname{conv}(A^{\pi_r})$ and $\operatorname{conv}(A^{\pi_s})$ are disjoint for each pair $1 \leq r < s \leq p$. As the convex hulls of finite sets are disjoint if and only if the sets can be strictly separated by a hyperplane, we have that π is A-disjoint if and only if for each pair $1 \leq r < s \leq n$ there exists a vector $h_{r,s} \in \mathbb{R}^d$ such that $(h_{r,s})^T A^i < (h_{r,s})^T A^S$ for all $i \in \pi_r$ and $j \in \pi_s$. Of course, A-disjointness implies A-separability, and the two properties coincide when the columns of A are distinct.

For $v \in \mathbb{R}^d$ denote by $\bar{v} \in \mathbb{R}^{d+1}$ the vector obtained by appending a first coordinate 1 to v. For a matrix $A \in \mathbb{R}^{d\times n}$ and indices $1 \leq i_0 < \cdots < i_d \leq n$, denote

$$\operatorname{sign}_A(i_0, \ldots, i_d) := \operatorname{sign}(\det[\bar{A}^{i_0}, \ldots, \bar{A}^{i_d}]) \in \{-1, 0, 1\}.$$

A matrix A is *generic* if its columns are in affine general position, that is, if any set of $d + 1$ vectors or less from among $\{\bar{A}^i : i = 1, \ldots, n\}$ are linearly independent; in particular, if $n > d$ this is the case if and only if all signs $\operatorname{sign}_A(i_0, \ldots, i_d)$ for indices $1 \leq i_0 < i_1 < \cdots < i_d \leq n$ are nonzero. Also, the columns of a generic matrix are distinct.

We next provide a representation of the set of A-disjoint 2-partitions for generic matrices A. The case where $n \leq d$ is simple.

LEMMA 3.3. *Let $A \in \mathbb{R}^{d\times n}$ be generic, $p \leq 2$, and $n \leq d$. Then every p-partition of $[n]$ is A-disjoint.*

Proof. It suffices to consider the case $p = 2$. A standard result from linear algebra shows that as $\bar{A}^1, \ldots, \bar{A}^n$ are linearly independent, the range of $[\bar{A}^1, \ldots, \bar{A}^n]^T$ is $\mathbb{R}^n$. Hence, given a 2-partition π of $[n]$, there is a vector $\mu \in \mathbb{R}^{d+1}$ with $\mu^T A^i > 0$ for each $i \in \pi$ and $\mu^T A^j < 0$ for each $j \in \pi_2$; with C obtained from μ by truncating its first coordinate μ_1, we then have $C^T A^i > -\mu_1 > C^T A^j$ for all $i \in \pi_1$ and $j \in \pi_2$, proving that π is A-disjoint. □

Let $A \in \mathbb{R}^{d\times n}$ be generic with $n \geq d$. For any d-subset $I = \{i_1, \ldots, i_d\}$ of $[n]$ with $i_1 < \cdots < i_d$, define

$$I_A^- := \{i_0 \in [n] : \operatorname{sign}_A(i_0, i_1, \ldots, i_d) = -1\}, \quad I_A^+ := \{i_0 \in [n] : \operatorname{sign}_A(i_0, i_1, \ldots, i_d) = 1\}.$$

Of course, $\{I_A^-, I_A^+\}$ is a 2-partition of $[n]\backslash I$. Let $I \subseteq [n]$ be a d-set and (J^-, J^+) be a 2-partition of I. The 2-partitions of $[n]$ *associated* with A, I, and (J^-, J^+) are defined to be either of the two 2-partitions $\pi^- := (I_A^- \cup J^-, I_A^+ \cup J^+)$ and $\pi^+ := (I_A^+ \cup J^+, I_A^- \cup J^-)$.

LEMMA 3.4. *Let $A \in \mathbb{R}^{d\times n}$ be generic, with $n \geq d$. Then the set of A-disjoint 2-partitions is the set of all 2-partitions associated with A, d-sets $I \subseteq [n]$ and 2-partitions (J^-, J^+) of I.*

Proof. We will show that for each d-set $I \subseteq [n]$ and 2-partition (J^-, J^+) of I, the two 2-partitions associated with A, I, and (J^-, J^+) are A-disjoint and that each A-disjoint 2-partition is generated in this way.

First, let $I \subseteq [n]$ have d-elements, say, $i_1 < \cdots < i_d$, and let (J^-, J^+) be a 2-partition of I. Then $H := \{x \in \mathbb{R}^d : \det[\bar{x}, \bar{A}^{i_1}, \ldots, \bar{A}^{i_d}] = 0\}$ is a hyperplane that contains the columns of A indexed by I; this hyperplane can be written as $\{x \in \mathbb{R}^d : h^T x = \gamma\}$ for some $h \in \mathbb{R}^d$ and $\gamma \in \mathbb{R}$ such that $I_A^- = \{i \in [n] : h^T A^i < \gamma\}$ and $I_A^+ = \{i \in [n] : h^T A^i > \gamma\}$. Thus, $h^T A^i < h^T A^U < h^T A^j$ for all $i \in I_A^-, u \in I$, and $j \in I_A^+$. We next observe that $B = [A^{i_1}, \ldots, A^{i_d}]$ is generic, hence Lemma 3.3 ensures that the 2-partition $\{j : i_j \in J_-\}$, $\{j : i_j \in J_+\}$ of $[d]$ is B-disjoint. Thus, there exists a vector $d \in \mathbb{R}^d$ with $d^T A^i > d^T A^j$ for all $i \in J^-$ and $j \in J^+$. For sufficiently small positive t, we then have that $(C + td)A^i < (C + td)^T A^j$ for all $i \in I_A^- \cup J^-$ and $j \in I_A^+ \cup J^+$, proving that $(I_A^- \cup J^-, J_A^+ \cup J^+)$ is A-disjoint. It follows immediately that $(I_A^+ \cup J^+, I_A^- \cup J^-)$ is A-disjoint too, proving that the two 2-partitions of $[n]$ associated with A, I and the 2-partition (J_-, J_+) of I are A-disjoint.

Next assume that π is an A-disjoint 2-partition. Then there exists a hyperplane strictly separating A^{π_1} and A^{π_2}. Any such hyperplane can be perturbed to a hyperplane that is spanned by d columns of A and weakly separates A^{π_1} and A^{π_2} (the details of constructing such a perturbation are left to the reader). In particular, if $A^{i_1}, \ldots, A^{i_d}$ span the hyperplane and $1 \leq i_1 < \cdots \leq i_d \leq n$, then for $I := \{i_1, \ldots, i_d\}$ either $\pi_1^1 \subseteq I_A^- \cup I$ and $\pi_2 \subseteq I_A^+ \cup I$ or $\pi_1 \subseteq I_A^+ \cup I$ and $\pi_2 \subseteq I_A^- \cup I$. In the former case we have $\pi = (I_A^- \cup J^-, I_A^+ \cup J^+)$ for $J^- = \pi_1 \cap I$ and $J^+ = \pi_2 \cap I$, and in the latter case $\pi = (I_A^+ \cup J^+, I_A^- \cup J^-)$ for $J^+ = \pi_1 \cap I$ and $J^- = \pi_2 \cap J$. □

Let $p \geq 2$. With each list $[\pi^{r,s} = (\pi_1^{r,s}, \pi_2^{r,s}) :\ 1 \leq r < s \leq p]$ of $\binom{p}{2}$ 2-partitions of $[n]$ *associate a* p-tuple $\pi = (\pi_1, \ldots, \pi_p)$ of subsets of $[n]$ as follows: for $r = 1, \ldots, p$ put

$$\pi_r := \left(\cap_{j=r+1}^{p} \pi_1^{r,j}\right) \bigcap \left(\cap_{j=1}^{r-1} \pi_2^{j,r}\right).$$

Since $\pi_r \subseteq \pi_1^{r,s}$ and $\pi_s \subseteq \pi_2^{r,s}$ for all $1 \leq r < s \leq p$, the elements π_i of the p-tuple associated with the given list are pairwise disjoint. If $\cup_{i=1}^{p} \pi_i = [n]$ holds as well, then π is a p-partition that will be called the partition *associated* with the given list.

LEMMA 3.5. *For $A \in \mathbb{R}^{d\times n}$ and $p \geq 2$, the set of A-disjoint p-partitions equals the set of p-partitions associated with lists of $\binom{p}{2}$ A-disjoint 2-partitions.*

Proof. First, consider a p-partition π associated with a list of $\binom{p}{2}$ A-disjoint 2-partitions. Then, for each $1 \leq r < s \leq p$,

$$\operatorname{conv}(A^i : i \in \pi_r) \cap \operatorname{conv}(A^i : i \in \pi_s) \subseteq \operatorname{conv}(A^i : i \in \pi_1^{r,s}) \cap \operatorname{conv}(A^i : i \in \pi_2^{r,s}) = \emptyset,$$

so π is A-disjoint. Conversely, let $\pi = (\pi_1, \ldots, \pi_p)$ be an A-disjoint p-partition. Consider any pair $1 \leq r < s \leq p$. Since $\operatorname{conv}(A^{\pi_r})$ and $\operatorname{conv}(A^{\pi_s})$ are disjoint, there is a hyperplane $H_{r,s}$ that contains no column of A and defines two corresponding half-spaces $H_{r,s}^-$ and $H_{r,s}^+$ that satisfy $A^{\pi_r} \subset H_{r,s}^-$ and $A^{\pi_s} \subset H_{r,s}^+$. Let $\pi^{r,s} := (\pi_1^{r,s}, \pi_2^{r,s})$ be the A-disjoint 2-partition defined by $\pi_1^{r,s} := \{i \in [n] :\ A^i \in H_{r,s}^-\}$ and $\pi_2^{r,s} := \{i \in$

$[n] : A^i \in H^+_{r,s}\}$. Let π' be the p-tuple associated with the constructed $\pi^{r,s}$'s. Then the sets of π' are pairwise disjoint, and for $i = 1, \ldots, p$, we have

$$\pi_i \subseteq \left(\cap_{j=i+1}^{p} \pi_1^{i,j}\right) \bigcap \left(\cap_{j=1}^{i-1} \pi_2^{j,i}\right) = \pi'_i.$$

Since $[n] = \cup_{i=1}^{p} \pi_i \subseteq \cup_{i=1}^{p} \pi'_i$, it follows that $\pi = \pi'$ is the p-partition associated with the constructed list of $\binom{p}{2}$ A-disjoint 2-partitions. □

For each $\epsilon > 0$ define the *ϵ-perturbation $A(\epsilon) \in \mathbb{R}^{d\times n}$ of A* as follows: for $i = 1, \ldots, n$, let the ith column of $A(\epsilon)$ be $A(\epsilon)^i := A^i + \epsilon M_d^i$, where $M_d^i := [i, i^2, \ldots, i^d]^T$ is the image of i on the moment curve in $\mathbb{R}^d$. Consider any $1 \leq i_0 < \cdots < i_d \leq n$. Then the determinant

$$D(\epsilon) := \det[\bar{A}(\epsilon)^{i_0}, \ldots, \bar{A}(\epsilon)^{i_d}] = \sum_{j=0}^{d} D_j \epsilon^j$$

is a polynomial of degree d in ϵ, with D_d being the Van der Monde determinant $\det[\bar{M}_d^{i_0}, \ldots, \bar{M}_d^{i_d}]$, which is known to be nonzero. So for all sufficiently small $\epsilon > 0$, $\operatorname{sign}_{A(\epsilon)}(i_0, \ldots, i_d) = \operatorname{sign}(D(\epsilon))$ equals the sign of the first nonzero coefficient among $D_0, \ldots, D_d$ and is either -1 or 1 and independent of ϵ. We define the *generic sign* of A at $(i_0, \ldots, i_d)$, denoted $\chi_A(i_0, \ldots, i_d)$, as the common value of $\operatorname{sign}_{A(\epsilon)}(i_0, \ldots, i_d)$ for all sufficiently small positive ϵ.

Lemma 3.6. *Let $A \in \mathbb{R}^{d\times n}$ and $p \geq 1$. For all sufficiently small $\epsilon > 0$, $A(\epsilon)$ is generic and the set of $A(\epsilon)$-disjoint p-partitions is the same. Further, for every d-set $I \in [n]$, the sets $I^-_{A(\epsilon)}$ and $I^+_{A(\epsilon)}$ are independent of ϵ.*

Proof. By Lemma 3.5, the set of $A(\epsilon)$-disjoint p-partitions is entirely determined by the set of $A(\epsilon)$-disjoint 2-partitions. Thus, it suffices to consider only $p = 2$.

First assume that $n < d$. In this case augment A with $n + 1 - d$ zero vectors to obtain a matrix $A' \in \mathbb{R}^{d\times(d+1)}$. The above arguments show that for sufficiently small positive ϵ, $\det \bar{A}'(\epsilon)$ is nonzero, implying that $\bar{A}(\epsilon)^1, \ldots, \bar{A}(\epsilon)^n$ are linearly independent. From Lemma 3.3 it follows that for such ϵ, the set of $A(\epsilon)$-disjoint 2-partitions of $[n]$ is the set of all 2-partitions of $[n]$.

Next assume that $n > d$. As explained above, for all sufficiently small $\epsilon > 0$, $\operatorname{sign}_{A(\epsilon)}$ $(i_0, \ldots, i_d)$ equals the nonzero generic sign $\chi_A(i_0, \ldots, i_d)$ for all $1 \leq i_0 < \cdots < i_d \leq n$. It follows that for all sufficiently small ϵ, the matrix $A(\epsilon)$ is generic, and for every d-set I, the sets $I^-_{A(\epsilon)}$ and $I^+_{A(\epsilon)}$ are independent of ϵ. By Lemma 3.4, the set of $A(\epsilon)$-disjoint 2-partitions is the set of all pairs of 2-partitions of $[n]$ associated with A, d-sets $I \subseteq [n]$, and 2-partitions (J^-, J^+) of I; but each such pair depends only on $I^-_{A(\epsilon)}, I^+_{A(\epsilon)}$, J^-, and J^+. Hence the set of $A(\epsilon)$-disjoint 2-partitions is the same for all sufficiently small $\epsilon > 0$. □

Let $A \in \mathbb{R}^{d\times n}$. A p-partition of $[n]$ is *A-generic* if it is $A(\epsilon)$-disjoint for all sufficiently small $\epsilon > 0$. Denote by Π^p_A the set of A-generic p-partitions.

Lemma 3.6 shows that for all sufficiently small $\epsilon > 0$, the set of $A(\epsilon)$-disjoint partitions is the same and equals Π^p_A. The final lemma of this section links vertices of shaped partition polytopes with generic partitions.

Lemma 3.7. *Let $A \in \mathbb{R}^{d\times n}$ and let Λ be a nonempty set of p-shapes of $[n]$. Then every vertex of the polytope $\mathcal{P}^\Lambda_A$ has a representation as the A-matrix A^π of some A-generic Λ-partition.*

Proof. Let $B \in \mathbb{R}^{d\times p}$ be a vertex of $\mathcal{P}^\Lambda_A$ and let $C \in \mathbb{R}^{d\times p}$ be a matrix such that $\langle C, \cdot\rangle$ is uniquely maximized over $\mathcal{P}^\Lambda_A$ at B. Let $\Pi := \{\pi : |\pi| \in \Lambda\}$ be the set of

Λ-partitions and let $\Pi^* := \{\pi \in \Pi : A^\pi = B\}$. Then there is a sufficiently small $\epsilon > 0$ such that $\langle C, A(\epsilon)^{\pi^*}\rangle > \langle C, A(\epsilon)^{\pi}\rangle$ for all $\pi^* \in \Pi^*$ and $\pi \in \Pi \setminus \Pi^*$, and in addition, as guaranteed by Lemma 3.6, $A(\epsilon)$ is generic and the set of $A(\epsilon)$-disjoint p-partitions equals Π^p_A. For such ϵ, $\langle C, \cdot\rangle$ is maximized over the perturbed polytope $P^\Lambda_{A(\epsilon)}$ at a vertex of the form $A(\epsilon)^{\pi^*}$ for some $\pi^* \in \Pi^*$. By Lemma 3.2, π^* is $A(\epsilon)$-separable. Since $A(\epsilon)$ is generic it has distinct columns, and therefore π^* is also $A(\epsilon)$-disjoint. We conclude that π^* is A-generic, proving that π^* contains a generic partition. □

4. Optimization and vertex enumeration. We now use the facts established in the previous section to prove our main results. Our computational complexity terminology is fairly standard (cf. [20]). In all our algorithms, the positive integer n will be input in *unary* representation, whereas all other numerical data such as the matrix A will be input in *binary* representation. An algorithm is *strongly polynomial time* if it uses a number of arithmetic operations polynomially bounded in n, and runs in time polynomially bounded in n plus the bit size of all other numerical input.

LEMMA 4.1. *Let d, p be fixed. For any $A \in \mathbb{R}^{d\times n}$, the set Π^p_A of A-generic p-partitions has $|\Pi^p_A| = O(n^{d\binom{p}{2}})$. Further, there is an algorithm that, given $n \in \mathbb{N}$ and $A \in \mathbb{Q}^{d\times n}$, produces Π^p_A in strongly polynomial time using $O(n^{dp^2})$ arithmetic operations.*

Proof. If $n \le d$, the set of A-generic p-partitions is the set of all partitions, of which there are $p^n \le p^d$. Henceforth we assume that $n > d$. If $p = 1$, then $\Pi^p_A := \{([n])\}$ consists of the single p-partition $([n])$. Suppose now that $p \ge 2$. For each choice $1 \le i_0 < \cdots < i_d \le n$, compute the generic sign $\chi_A(i_0, \ldots, i_d)$ as follows. Evaluate the polynomial

$$D(\epsilon) := \det[\bar{A}(\epsilon)^{i_0}, \ldots, \bar{A}(\epsilon)^{i_d}] = \sum_{j=0}^{d} D_j \epsilon^j$$

at $\epsilon = 0, 1, \ldots, d$ to obtain $D(0), D(1), \ldots, D(d)$. Each evaluation involves the computation of the determinant of a matrix of order $d+1$ and can be done, say, by Gaussian elimination, using $O(d^3)$ arithmetic operations and, for rational A, in strongly polynomial time. Then, solve the following linear system of equations:

$$\sum_{j=0}^{d} \epsilon^j D_j = D(\epsilon), \quad \epsilon = 0, \ldots, d,$$

to obtain the indeterminates $D_0, \ldots, D_d$. This can be done by inverting the nonsingular Vandermonde matrix of coefficients of this system, again by Gaussian elimination. The generic sign $\chi_A(i_0, \ldots, i_d)$ is then the sign of the first nonzero D_i. So, for fixed d, the number of arithmetic operations needed to compute all $\binom{n}{d+1}$ generic signs is $O(\binom{n}{d+1}d^4) = O(n^{d+1})$.

By Lemma 3.6, for sufficiently small positive ϵ, for each d-set $I \subseteq [n]$, $I^-_{A(\epsilon)}$ and $I^+_{A(\epsilon)}$ are independent of ϵ. For a d-set $I \subseteq [n]$ and such ϵ, $I^-_{A(\epsilon)}$ and $I^+_{A(\epsilon)}$ are available from the above signs that determine $\det[\bar{A}^i, \bar{A}^{i_1}, \ldots, \bar{A}^{i_q}]$ for each $i \in [n] \setminus J$ (a permutation that puts $\bar{A}^i$ into the right location may be applied). Further, from Lemmas 3.6 and 3.4, Π^2_A equals the common set of $A(\epsilon)$-disjoint partitions for sufficiently small positive ϵ, and this set is the set of partitions of $[n]$ of the form $(I^-_{A(\epsilon)} \cup J^-, I^+_{A(\epsilon)} \cup J^+)$ or $(I^+_{a(\epsilon)} \cup J^+, I^-_{A(\epsilon)} \cup J^-)$, where I is a d-subset of $[n]$ and

(J^-, J^+) is a 2-partition of I. For each d-set $I \subseteq [n]$, the common 2-partitions $(I^-_{A(\epsilon)}, I^+_{A(\epsilon)})$ for sufficiently small positive ϵ have been determined; hence a list of the 2-partitions in Π^2_A is available (the construction may contain duplicates). As there are $\binom{n}{d}$ d-subsets I and 2^d 2-partitions (J^-, J^+) of each I, we have $|\Pi^2_A| \leq 2^{d+1}\binom{n}{d} = O(n^d)$ and all partitions in Π^2_A can be obtained from the generic signs, again using $O(n^{d+1})$ operations.

For sufficiently small positive ϵ, Π^p_A is the common set of $A(\epsilon)$-disjoint p-partitions and Π^2_A is the common set of $A(\epsilon)$-disjoint 2-partitions. It follows from Lemma 3.5 that Π^p_A is the set of all p-partitions associated with lists of $\binom{p}{2}$ 2-partitions from Π^2_A. This shows that

$$|\Pi^p_A| \leq |\Pi^2_A|^{\binom{p}{2}} = O(n^{d\binom{p}{2}}).$$

To construct Π^p_A, produce all such lists of $\binom{p}{2}$ 2-partitions from Π^2_A; for each list, form the associated p-tuple π and test if it is a partition (i.e., if $\cup^p_{i=1}\pi_i = [n]$). As there are $O(n^{d\binom{p}{2}})$ lists, all this work can be done easily using $O(n^{dp^2})$ arithmetic operations, which subsumes the work for computing the generic signs and constructing Π^2_A, and is the claimed bound. $\square$

We can now provide the solution of the shaped partition problem. The set of admissible p-partitions Λ can be represented by a membership oracle that, on query, λ answers whether $\lambda \in \Lambda$. The convex functional C on $\mathbb{R}^{d\times p}$ can be presented by an evaluation oracle that, on query A^π with π a Λ-partition, returns $C(A^\pi)$. The oracle for C will be called *M-guaranteed* if $C(A^\pi)$ is guaranteed to be a rational number whose absolute value is no larger than M for any Λ-partition π. The algorithm is then *strongly polynomial oracle time* if it uses a number of arithmetic operations and oracle queries polynomially bounded in n and runs in time polynomially bounded in n plus the bit size of A and M.

THEOREM 4.2. *For every fixed d, p, there is an algorithm that, given $n, M \in \mathbb{N}$, $A \in \mathbb{Q}^{d\times n}$, oracle-presented nonempty set Λ of p-shapes of n, and M-guaranteed oracle-presented convex functional C on $\mathbb{Q}^{d\times p}$, solves the shaped partition problem in strongly polynomial oracle time using $O(n^{dp^2})$ arithmetic operations and oracle queries.*

Proof. Use the algorithm of Lemma 4.1 to construct the set Π^p_A of A-generic p-partitions in strongly polynomial time using $O(n^{dp^2})$ arithmetic operations. Then test shapes of the partitions in the list to obtain the subset $\Pi^\Lambda := \{\pi \in \Pi^p_A : |\pi| \in \Lambda\}$ of A-generic Λ-partitions by querying the Λ-oracle on each of the $|\Pi^p_A| = O(n^{d\binom{p}{2}})$ partitions in Π^p_A. Since C is convex, it is maximized over the shaped partition polytope $\mathcal{P}^\Lambda_A$ at a vertex of $\mathcal{P}^\Lambda_A$. By Lemma 3.7, this vertex equals the A-matrix A^π of some partition in Π^Λ. Therefore, any $\pi^* \in \Pi^\Lambda$ achieving $C(A^{\pi^*}) = \max\{C(A^\pi) : \pi \in \Pi^\Lambda\}$ is an optimal solution to the shaped partition problem. To find such π^*, compute for each $\pi \in \Pi^\Lambda$ the matrix $A^\pi = [\sum_{i\in\pi_1} A^i, \dots, \sum_{i\in\pi_p} A^i]$, query the C-oracle for the value $C(A^\pi)$, and pick the best. The number of operations involved and queries to the C-oracle is again $O(n^{dp^2})$. The bit size of the numbers manipulated throughout this process is polynomially bounded in the bit size of M and A, and hence the algorithm is strongly polynomial oracle time. $\square$

Recall that the shaped partition polytope is defined as $\mathcal{P}^\Lambda_A = \text{conv}\{A^\pi : |\pi| \in \Lambda\}$. The number of matrices in the set $\{A^\pi : |\pi| \in \Lambda\}$ is typically exponential in n, even for fixed d, p. Therefore, although the dimension of $\mathcal{P}^\Lambda_A$ is bounded by dp, this polytope potentially can have exponentially many vertices and facets as well. Lemmas

3.7 and 4.1 yield the following theorem, which shows that, in fact, shaped partition polytopes are exceptionally well behaved.

THEOREM 4.3. *Let d, p be fixed. For any $A \in \mathbb{R}^{d\times n}$ and nonempty set Λ of p-shapes of n, the number of vertices of the shaped partition polytope $\mathcal{P}_A^\Lambda$ is $O(n^{d\binom{p}{2}})$. Further, there is an algorithm that, given $n \in \mathbb{N}$, $A \in \mathbb{Q}^{d\times n}$, and oracle-presented Λ, produces all vertices of $\mathcal{P}_A^\Lambda$ in strongly polynomial oracle time using $O(n^{d^2p^3})$ operations and queries.*

Proof. By Lemma 3.7, each vertex of $\mathcal{P}_A^\Lambda$ equals the A-matrix A^π of some partition in Π_A^p. Therefore, the number of vertices of $\mathcal{P}_A^\Lambda$ is bounded above by $|\Pi_A^p|$, hence, by Lemma 4.1, is $O(n^{d\binom{p}{2}})$. To construct the set of vertices given a rational matrix A, proceed as follows. Use the algorithm of Lemma 4.1 to construct the set Π_A^p of A-generic p-partitions in strongly polynomial time using $O(n^{dp^2})$ arithmetic operations. Test the shapes of the partitions in the list to obtain its subset $\Pi^\Lambda := \{\pi \in \Pi_A^p : |\pi| \in \Lambda\}$ of A-generic Λ-partitions by querying the Λ-oracle on each of the $|\Pi_A^p| = O(n^{d\binom{p}{2}})$ partitions in Π_A^p. Construct the set of matrices $U := \{A^\pi : \pi \in \Pi^\Lambda\}$ with multiple copies identified. This set U is contained in $\mathcal{P}_A^\Lambda$, and by Lemma 3.7 contains the set of vertices of $\mathcal{P}_A^\Lambda$. So $u \in U$ will be a vertex precisely when it is not a convex combination of other elements of U. This could be tested using any linear programming algorithm, but to obtain a strongly polynomial time procedure, we proceed as follows. By Carathéodory's theorem, u will be a vertex if and only if it is not in the convex hull of any *affine basis* of $U \setminus \{u\}$. So, to test if $u \in U$ is a vertex of $\mathcal{P}_A^\Lambda$, compute the affine dimension a of $U \setminus \{u\}$. For each $(a+1)$-subset $\{u_0, \ldots, u_a\}$ of $U \setminus \{u\}$, test if it is an affine basis of $U \setminus \{u\}$, and if it is, compute the unique $\mu_0, \ldots, \mu_a$ satisfying $u = \sum_{i=0}^a \mu_i u_i$ and $\sum_{i=0}^a \mu_i = 1$. Then u is in the convex hull of $\{u_0, \ldots, u_a\}$ if and only if $\mu_0, \ldots, \mu_a \geq 0$. So u is a vertex of $\mathcal{P}_A^\Lambda$ if and only if for each affine basis we get some $\mu_i < 0$. Computing the affine dimension a, testing if an $(a+1)$-subset of $U \setminus \{u\}$ is an affine basis, and computing the μ_i can all be done by Gaussian elimination in strongly polynomial time. Since we have to perform the entire procedure for each of the $|U| \leq |\Pi^\Lambda| = O(n^{d\binom{p}{2}})$ elements $u \in U$, and for each such u the number of affine bases of $U \setminus \{u\}$ is at most $\binom{|U|-1}{dp+1}$, the number of arithmetic operations involved is $O(|U|\binom{|U|-1}{dp+1}) = O(n^{d^2p^3})$, which absorbs the work of constructing Π^Λ and obeys the claimed bound. □

As an immediate corollary of Theorem 4.3, we get the following polynomial bound on the number of facets of any shaped partition polytope and a strongly polynomial oracle time procedure for producing all facets (by which we mean finding, for each facet F, a hyperplane $\{X \in \mathbb{R}^{d\times p} : \langle H, X\rangle = h\}$ supporting $\mathcal{P}_A^\Lambda$ at F).

COROLLARY 4.4. *Let d, p be fixed. For any $A \in \mathbb{R}^{d\times n}$ and nonempty set Λ of p-shapes of n, the number of facets of the shaped partition polytope $\mathcal{P}_A^\Lambda$ is $O(n^{\frac{d^2p^3}{2}})$. Further, there is an algorithm that, given $n \in \mathbb{N}$, $A \in \mathbb{Q}^{d\times n}$, and oracle-presented Λ, produces all facets of $\mathcal{P}_A^\Lambda$ in strongly polynomial oracle time using $O(n^{d^2p^3})$ operations and queries.*

Proof. By the well-known upper bound theorem [18], the number of facets of any k-dimensional polytope with m vertices is $O(m^{\frac{k}{2}})$. Applying this to $\mathcal{P}_A^\Lambda$ with $k \leq dp$ and $m = O(n^{d\binom{p}{2}})$, we get the bound on the number of facets of $\mathcal{P}_A^\Lambda$. To construct the facets, first construct the set V of vertices using the algorithm of Theorem 4.3. Compute the dimension a of $\mathrm{aff}(P) = \mathrm{aff}(V)$ and compute a (possibly empty) set S of $dp - a$ points that, together with V, affinely span $\mathbb{R}^{d\times p}$. For each affinely independent

a-subset T of V, compute the hyperplane $\{X \in \mathbb{R}^{d\times p} : \langle H, X\rangle = h\}$ spanned by $S \cup T$. This hyperplane supports a facet of $\mathcal{P}_A^\Lambda$ if and only if all points in V lie on one of its closed half-spaces. Clearly, all facets of $\mathcal{P}_A^\Lambda$ are obtained that way, in strongly polynomial time and with the number of arithmetic operations and oracle queries bounded as claimed. □

Acknowledgment. Shmuel Onn thanks the Mathematical Sciences Research Institute at Berkeley for its support while part of this research was done.

REFERENCES

[1] N. Alon and S. Onn, *Separable partitions*, Discrete Appl. Math., 91 (1999), pp. 39–51.

[2] F. Aurenhammer and O. Schwarzkopf, *A simple on-line randomized incremental algorithm for computing higher order (Voronoi) diagrams*, in Proceedings of the 7th ACM Symposium on Computational Geometry, 1991, pp. 142–151.

[3] E.R. Barnes, A.J. Hoffman, and U.G. Rothblum, *Optimal partitions having disjoint convex and conic hulls*, Math. Programming, 54 (1992), pp. 69–86.

[4] I. Bárány and S. Onn, *Colourful linear programming and its relatives*, Math. Oper. Res., 22 (1997), pp. 550–567.

[5] A.K. Chakravarty, J.B. Orlin, and U.G. Rothblum, *Consecutive optimizers for a partitioning problem with applications to optimal inventory groupings for joint replenishment*, Oper. Res., 33 (1985), pp. 820–834.

[6] H. Edelsbrunner, P. Valtr, and E. Welzl, *Cutting dense point sets in half*, in Proceedings of the 10th ACM Symposium on Computational Geometry, 1994, pp. 203–210.

[7] P. Erdös, *The Art of Counting*, Joel Spencer, ed., MIT Press, Cambridge, MA, 1973.

[8] B. Gao, F.K. Hwang, W.-C.W. Li, and U.G. Rothblum, *Partition polytopes over* 1-*dimensional points*, 1996, Math. Progr., to appear.

[9] D. Granot and U.G. Rothblum, *The Pareto set of the partition bargaining game*, Games Econom. Behav., 3 (1991), pp. 163–182.

[10] F.K. Hwang and C.L. Mallows, *Enumerating nested and consecutive partitions*, J. Combin. Theory Ser. A, 70 (1995), pp. 323–333.

[11] F.K. Hwang, S. Onn, and U.G. Rothblum, *Representations and characterizations of the vertices of bounded-shape partition polytopes*, Linear Algebra Appl., 278 (1998), pp. 263–284.

[12] F.K. Hwang, S. Onn, and U.G. Rothblum, *Linear Programming over Partitions*, in preparation.

[13] F.K. Hwang and U.G. Rothblum, *Directional-quasi-convexity, asymmetric Schur-convexity and optimality of consecutive partitions*, Math. Oper. Res., 21 (1996), pp. 540–554.

[14] F.K. Hwang and U.G. Rothblum, *Partitions: Clustering and Optimality*, in preparation.

[15] F.K. Hwang, J. Sun, and E.Y. Yao, *Optimal set partitioning*, SIAM J. Algebraic Discrete Math., 6 (1985), pp. 163–170.

[16] P. Kleinschmidt and S. Onn, *Signable posets and partitionable simplicial complexes*, Discrete Comput. Geom., 15 (1996), pp. 443–466.

[17] L. Lovász, *On the number of halving lines*, Ann. Univ. Sci. Budapest. Eötvös Sect. Math., 14 (1970), pp. 107–108.

[18] P. McMullen, *The maximum numbers of faces of a convex polytope*, Mathematika, 17 (1970), pp. 179–184.

[19] S. Onn, *Geometry, complexity, and combinatorics of permutation polytopes*, J. Combin. Theory Ser. A, 64 (1993), pp. 31-49.

[20] A. Schrijver, *Theory of Linear and Integer Programming*, John Wiley, New York, 1986.

[21] G.M. Ziegler, *Lectures in Polytopes*, Graduate Texts in Mathematics, Springer-Verlag, New York, 1995.

SIAM J. Optim.
Vol. 10, No. 1, pp. 82–98

STABILITY AND WELL-POSEDNESS IN LINEAR SEMI-INFINITE PROGRAMMING[*]

M. J. CÁNOVAS[†], M. A. LÓPEZ[‡], J. PARRA[†], AND M. I. TODOROV[§]

Abstract. This paper presents an approach to the stability and the Hadamard well-posedness of the linear semi-infinite programming problem (LSIP). No standard hypothesis is required in relation to the set indexing of the constraints and, consequently, the functional dependence between the linear constraints and their associated indices has no special property. We consider, as parameter space, the set of all LSIP problems whose constraint systems have the same index set, and we define in it an extended metric to measure the size of the perturbations. Throughout the paper the behavior of the optimal value function and of the optimal set mapping are analyzed. Moreover, a certain type of Hadamard well-posedness, which does not require the boundedness of the optimal set, is characterized. The main results provided in the paper allow us to point out that the lower semicontinuity of the feasible set mapping entails high stability of the whole problem, mainly when this property occurs simultaneously with the boundedness of the optimal set. In this case all the stability properties hold, with the only exception being the lower semicontinuity of the optimal set mapping.

Key words. stability, Hadamard well-posedness, semi-infinite programming, feasible set mapping, optimal set mapping, optimal value function

AMS subject classifications. 15A39, 49D39, 49M39, 52A40, 65F99, 90C34

PII. S1052623497319869

1. Introduction.

We consider the linear optimization problem in $\mathbb{R}^n$:

$$\pi : \quad \text{Inf } \{c'x \mid a_t'x \geq b_t\,,\ t \in T\}\,,$$

where c, x, and a_t belong to $\mathbb{R}^n$, $b_t \in \mathbb{R}$, and y' denotes the transpose of $y \in \mathbb{R}^n$. π is alternatively represented by the couple (c,σ), or by $\left(c,(a_t,b_t)_{t\in T}\right)$.

If the *index set* T of the *constraints system,* $\sigma := \{a_t'x \geq b_t\,,\ t \in T\}$, is infinite, we have a *linear semi-infinite programming problem* (LSIP). We shall not assume any structure for T and, consequently, the functions $t \mapsto a_t$ and $t \mapsto b_t$ have no particular property.

The *parameter space*, in our approach, is the set Π of all the problems $\pi = (c,\sigma)$, with $c \neq 0_n$, whose constraint systems have the same index set T. When different problems are considered in Π, they and their associated elements will be distinguished by means of sub- and superscripts. So, if π_1 also belongs to Π, we write $\pi_1 = \left(c^1,\sigma_1\right)$ and $\sigma_1 := \{\left(a_t^1\right)' x \geq b_t^1\,,\ t \in T\}$. Obviously, we can identify Π with $(\mathbb{R}^n \backslash \{0_n\}) \times (\mathbb{R}^n \times \mathbb{R})^T$, where the set of possible systems is itself identified with $(\mathbb{R}^n \times \mathbb{R})^T$.

[*] Received by the editors April 11, 1997; accepted for publication (in revised form) September 10, 1998; published electronically October 20, 1999. This research was partially supported by grants PB95-0687 and SAB 95-0311 from DGES and by grants GV-2219/94 and GV-C-CN-10-067-96 from Generalitat Valenciana.

http://www.siam.org/journals/siopt/10-1/31986.html

[†] Department of Statistics and Applied Mathematics, Miguel Hernández University, 03202 Elche (Alicante), Spain (canovas@umh.es, parra@umh.es).

[‡] Department of Statistics and Operations Research, University of Alicante, 03071 Alicante, Spain (marco.antonio@ua.es).

[§] Bulgarian Academy of Sciences, Institute of Mathematics, 29 Ph. Macedonsky Str., r. 401, 4002 Plovdiv, Bulgaria (todorovm@bgearn.bitnet).

Many LSIP problems have coefficients whose values either are known only approximately or have to be rounded off in the computing process. Therefore, we actually solve a different problem, $\pi_1 = \left(c^1, \sigma_1\right)$, proximal to the original one, $\pi = (c, \sigma)$. An *extended distance* $\delta : \Pi \times \Pi \to [0, +\infty]$ is introduced by means of

$$\delta\left(\pi_1, \pi\right) := \max\left\{ \left\| c^1 - c \right\|_\infty ,\ \sup\nolimits_{t \in T} \left\| \begin{pmatrix} a_t^1 \\ b_t^1 \end{pmatrix} - \begin{pmatrix} a_t \\ b_t \end{pmatrix} \right\|_\infty \right\}.$$

(Π, δ) is a Hausdorff space, whose topology satisfies the first axiom of countability (i.e., convergence is established by means of sequences, since each point has a countable base of neighborhoods), and describes the uniform convergence topology on Π. If T is a compact Hausdorff space and the functions $t \mapsto a_t$ and $t \mapsto b_t$ are continuous, π is said to be *continuous.* We shall denote by Π_o the set of continuous LSIP problems. ((Π_o, δ) is a metric space.)

In this paper, we study the stability properties of π. More precisely, we analyze the lower and upper semicontinuity of the *optimal value function*, ϑ, and the *optimal set mapping*, $\mathcal{F}^*$. The former assigns to each problem π its *optimal value* v (i.e., $\vartheta(\pi) = v$), and the latter assigns to π the (possibly empty) *optimal set,* represented by F^* (i.e., $\mathcal{F}^*(\pi) = F^*$). We prove that the lower semicontinuity of the *feasible set mapping,* $\mathcal{F}$, assigning to π the (possibly empty) *feasible set* F (i.e., $\mathcal{F}(\pi) = F$), and the boundedness of F^* (especially when both hold simultaneously), yield nice stability properties of ϑ and $\mathcal{F}^*$ at π. So, we devote section 3 to presenting different characterizations of the lower semicontinuity of $\mathcal{F}$ at π, which are used throughout the paper. As a counterpart of the important lower semicontinuity property, Lemma 4.1 states that the boundedness of the optimal set, assumed to be nonempty, is equivalent to a certain stability of π: any sufficiently close problem, with nonempty feasible set, also has optimal solutions.

Section 4 contains the main results concerning the optimal value function. Theorem 4.2 deals with the continuity properties of ϑ, whereas in the second part of this section we propose a definition of Hadamard well-posedness, based on the strategy of solving, in an approximated way, the sequence of problems approaching π. Our concept of Hadamard well-posedness, which does not require the uniqueness of the optimal solution, is oriented toward the stability of the optimal value function and can be traced out from Dontchev and Zolezzi [4]. Theorem 4.3 delimits the scope of this new concept.

Section 5 focuses on the stability behavior of the optimal set mapping, $\mathcal{F}^*$. Theorem 5.1 clarifies the role played by the closedness of this mapping. At the end of section 5, Table 5.1 summarizes the theory developed in the paper, emphasizing the importance of the lower semicontinuity of $\mathcal{F}$ at π, and of the boundedness of F^*, in the global stability of π. Section 6 supplies examples showing that every unfixed possibility in Table 5.1 can actually occur.

Some statements in sections 4 and 5 constitute extensions to the general LSIP of different results obtained by Brosowski [2] and Fischer [5] for the continuous LSIP. Moreover, in a forthcoming paper [3], we prove that, under the unicity of the optimal solution, our concept of Hadamard well-posedness is equivalent to other concepts [17] that, at first glance, seem much more restrictive.

2. Preliminaries. The optimal value function ϑ will take values in $[-\infty, +\infty]$ if we define $\vartheta(\pi) = +\infty$ when π is *inconsistent* (i.e., when $F = \emptyset$) and $\vartheta(\pi) = -\infty$ when π is *unbounded* (i.e., when $c'x$ is not bounded from below on F). Hereafter, Π_c represents the *consistent problems* subset ($\pi \in \Pi_c \Leftrightarrow F \neq \emptyset \Leftrightarrow \vartheta(\pi) < +\infty$), and Π_b

denotes the set of *bounded problems* ($\pi \in \Pi_b \Leftrightarrow \vartheta(\pi)$ is finite). In addition, Π_s will be the set of *solvable problems* ($\pi \in \Pi_s \Leftrightarrow F^* \neq \emptyset \Leftrightarrow \vartheta(\pi)$ is attained). Obviously, $\Pi_s \subset \Pi_b \subset \Pi_c$.

At this point we introduce some necessary notation. Given $\emptyset \neq X \subset \mathbb{R}^p$, by $conv(X)$, $cone(X)$, $O^+(X)$, and X^o we denote the *convex hull* of X, the *conical convex hull* of X, the *recession cone* of X (assuming that X is convex), and the *dual cone* of X (i.e., $X^o = \{y \in \mathbb{R}^p \mid y'x \geq 0 \text{ for all } x \in X\}$), respectively. It is assumed that $cone\,(X)$ always contains the zero-vector, and so $cone(\emptyset) = \{0_n\}$. The *Euclidean* and *Chebyshev norms* of $x \in \mathbb{R}^p$ will be $\|x\|$ and $\|x\|_\infty$, respectively, and the *Euclidean distance* from x to X ($\neq \emptyset$) is $d\,(x, X) := \inf\{\|x-y\| : y \in X\}$. The *unit open ball*, in $\mathbb{R}^p$, for the Euclidean norm is represented by B. From the topological side, if X is a subset of any topological space, $\text{int}(X)$, $\text{cl}(X)$, and $\text{bd}(X)$ represent the *interior*, the *closure*, and the *boundary* of X, respectively. Finally, $\lim_r$ should be interpreted as $\lim_{r\to\infty}$.

If $\{X_r\}$ is a sequence of nonempty sets in $\mathbb{R}^p$, $\liminf_r X_r$ ($\limsup_r X_r$) is the set of all the limits (cluster points) of all the possible sequences $\{x^r\}$, $x^r \in X_r$, $r = 1, 2, \ldots$, and it can be characterized as the set of points x such that every neighborhood of x intersects all the sets X_r except a finite number of them (it intersects infinitely many sets X_r). It is said that $\{X_r\}$ converges to X, in the *Painlevé–Kuratowski sense* (see, for instance, [15]) if $X = \liminf_r X_r = \limsup_r X_r$. In this case we write $X = \lim_r X_r$.

Next we recall some well-known continuity concepts for set-valued mappings. If $\mathcal{Y}$ and $\mathcal{Z}$ are two topological spaces and $\mathcal{S} : \mathcal{Y} \rightarrow 2^{\mathcal{Z}}$ is a set-valued mapping, we shall consider the following properties of $\mathcal{S}$.

If both spaces verify the first axiom of countability, we say that $\mathcal{S}$ is *closed* at $y \in \mathcal{Y}$ if for all sequences $\{y^r\} \subset \mathcal{Y}$ and $\{z^r\} \subset \mathcal{Z}$ satisfying $\lim_r y^r = y$, $\lim_r z^r = z$, and $z^r \in \mathcal{S}(y^r)$, one has $z \in \mathcal{S}(y)$.

The mapping $\mathcal{S}$ is *lower semicontinuous* (lsc) at $y \in \mathcal{Y}$ if for each open set $W \subset \mathcal{Z}$ such that $W \cap \mathcal{S}(y) \neq \emptyset$, there exists an open set $U \subset \mathcal{Y}$, containing y, such that $W \cap \mathcal{S}(y^1) \neq \emptyset$ for each $y^1 \in U$.

$\mathcal{S}$ is said to be *upper semicontinuous* (usc) at $y \in \mathcal{Y}$ if for each open set $W \subset \mathcal{Z}$ such that $\mathcal{S}(y) \subset W$, there exists an open neighborhood of y in $\mathcal{Y}$, U, such that $\mathcal{S}(y^1) \subset W$ for every $y^1 \in U$.

Given a consistent system $\sigma := \{a_t'x \geq b_t\,,\ t \in T\}$, with solution set F, we say that $a'x \geq b$ is a *consequence* of σ if it is satisfied at each point of F, i.e., if $a'z \geq b$ for every $z \in F$.

Throughout this paper we shall apply the so-called *nonhomogeneous Farkas lemma* [19], which characterizes the linear inequalities $a'x \geq b$ that are consequences of a consistent system $\sigma := \{a_t'x \geq b_t\,,\ t \in T\}$ as those satisfying

$$\begin{pmatrix} a \\ b \end{pmatrix} \in \text{cl}\left(cone\left(\left\{ \begin{pmatrix} a_t \\ b_t \end{pmatrix}\ ,\ t \in T\ ;\ \begin{pmatrix} 0_n \\ -1 \end{pmatrix} \right\} \right) \right). \tag{2.1}$$

If we introduce the cone, $\mathbb{R}_+^{(T)}$, of all the functions $\lambda : T \to \mathbb{R}_+$ taking positive values only at finitely many points of T, (2.1) is equivalent to the existence of sequences $\{\lambda^r\} \subset \mathbb{R}_+^{(T)}$ and $\{\mu_r\} \subset \mathbb{R}_+$, such that

$$\begin{pmatrix} a \\ b \end{pmatrix} = \lim_r \left\{ \sum_{t\in T} \lambda_t^r \begin{pmatrix} a_t \\ b_t \end{pmatrix} + \mu_r \begin{pmatrix} 0_n \\ -1 \end{pmatrix} \right\},$$

where $\lambda^r = (\lambda_t^r)_{t\in T}$, $r = 1, 2, \ldots$.

3. Feasible set mapping. In [7, sect. 2] it is proved that the mapping $\mathcal{F}$ is always closed at any $\pi \in \Pi_c$. In that paper, and also in [6], different characterizations of the lower semicontinuity of $\mathcal{F}$ at a consistent problem π are provided, most of them based upon different stability concepts taken from the literature ([11], [14], [18], etc.). The following theorem gathers some of these characterizations and adds some new ones, which will be applied below. We recall here the *strong Slater condition* (*SS condition*), which is satisfied by π if there exist a positive scalar ρ and a feasible point $\overline{x}$ satisfying $a_t'\overline{x} \geq b_t + \rho$ for all $t \in T$ ($\overline{x}$ is called an *SS element* of σ). The SS condition is certainly stronger than the well-known *Slater condition*, which only requires the existence of a strict solution, $\overline{x}$, satisfying $a_t'\overline{x} > b_t$ for all $t \in T$ (obviously, if π is continuous, both conditions are equivalent). The set of all the SS elements of σ will be represented by F_{SS}.

THEOREM 3.1. *If* $\pi = (c, \sigma) \in \Pi_c$, *then the following statements are equivalent*:

i. $\mathcal{F}$ *is lsc at* π*;*

ii. $\pi \in \operatorname{int}(\Pi_c)$*;*

iii. $0_{n+1} \notin \operatorname{cl}(conv(\{\binom{a_t}{b_t}, \ t \in T\}))$*;*

iv. π *satisfies the* SS condition*;*

v. *For every sequence* $\{\pi_r\} \subset \Pi$ *converging to* π, *there exists an* r_0 *such that* $\pi_r \in \Pi_c$ *if* $r \geq r_0$, *and* $F = \lim_{r \geq r_0} F_r$*;*

vi. $F = \operatorname{cl}(F_{SS})$.

Proof. The equivalence between the first four statements is established in [7, Thm. 3.1]. Next we prove the equivalence of statements i and v. Let us assume first that statement i holds. Since i $\Leftrightarrow$ ii has already been established, from $\pi \in \operatorname{int}(\Pi_c)$ we conclude the existence of r_0 such that $F_r \neq \emptyset$ if $r \geq r_0$. Then, if $x \in F$ and W is an open neighborhood of x, statement i yields r_1 ($r_1 \geq r_0$) such that $W \cap F_r \neq \emptyset$ for all $r \geq r_1$. In other words, W intersects all the sets F_r, except a finite number of them, which identifies x as a point of $\liminf_{r \geq r_0} F_r$. Moreover, $\limsup_{r \geq r_0} F_r \subset F$ because $\mathcal{F}$ is closed at every π. Since the inclusion $\liminf_{r \geq r_0} F_r \subset \limsup_{r \geq r_0} F_r$ always holds, one concludes that $F = \lim_{r \geq r_0} F_r$.

We proceed by assuming that part v holds and statement i fails. This implies the existence of an open set W such that $F \cap W \neq \emptyset$, whereas for each $r \in \mathbb{N}$ we can find π_r such that $\delta(\pi_r, \pi) < 1/r$ and $F_r \cap W = \emptyset$. Consequently, if $x \in F \cap W$, and whichever r_0 we consider, $x \notin \liminf_{r \geq r_0} F_r$. Thus, $\lim_r \pi_r = \pi$ and $F \neq \liminf_{r \geq r_0} F_r$ for every r_0, contradicting the assumption.

Next we prove i $\Leftrightarrow$ vi. If statement vi holds, since $F \neq \emptyset$ by hypothesis, F_{SS} must be nonempty too, and we apply the equivalence between statements i and iv, already established. Conversely, if statement i is held, given any open set W intersecting F, we can find $\overline{\rho} > 0$ such that $F_1 \cap W \neq \emptyset$ if $\pi_1 := (c, \sigma_1)$ with $\sigma_1 := \{a_t'x \geq b_t + \overline{\rho}, \ t \in T\}$. Since $F_1 = \mathcal{F}(\pi_1) \subset F_{SS}$, we obtain $F_{SS} \cap W \neq \emptyset$. We have just proved that $F \cap W \neq \emptyset$ implies $F_{SS} \cap W \neq \emptyset$, which itself implies $F \subset \operatorname{cl}(F_{SS})$. The opposite inclusion comes from the trivial relation $F_{SS} \subset F$. □

Concerning the upper semicontinuity of $\mathcal{F}$ at $\pi \in \Pi_c$, in the characterization given in [8, Thm. 3.1], the boundedness of F (see [5]) is not required any longer, although this condition is still sufficient [8, Cor. 3.2]. If $n \geq 2$ and $\{a_t, \ t \in T\}$ is bounded and different from $\{0_n\}$, $\mathcal{F}$ will be usc at $\pi \in \Pi_c$ if and only if F is bounded [8, Thm. 3.4]. Finally, in the case $n = 1$, it is remarked in [8, Ex. 3.3] that $\mathcal{F}$ is always usc at every consistent problem.

When we confine ourselves to continuous problems, we shall denote by Π_{oc} the set of consistent continuous LSIP problems in $\mathbb{R}^n$, all of them having constraint systems indexed by a compact Hausdorff space T. It was proved in [7, Thm. 6.2] that the restriction of $\mathcal{F}$ to Π_o, represented by $\mathcal{F}_o$, is lsc at $\pi \in \Pi_{oc}$ if and only if σ satisfies the well-known *Slater condition* or, equivalently, if π belongs to $\operatorname{int}_o(\Pi_{oc})$, the interior set of Π_{oc} in the topology relative to Π_o. Moreover, and since $\{a_t\ ,\ t \in T\}$ is compact when $\pi \in \Pi_o$, it turns out that, for $n \geq 2$, $\mathcal{F}_o$ is usc at $\pi \in \Pi_{oc}$ if and only if either F is bounded or $F = \mathbb{R}^n$.

In section 4 we shall apply the following *uniform metric regularity property.*

LEMMA 3.2. *Given $\pi \in \Pi_c$, assume that $\mathcal{F}$ is lsc at π and that F is bounded. Then, there exists a pair of positive scalars ε and β such that $\delta(\pi_i, \pi) < \varepsilon$, $i = 1, 2$, imply, for every $x^j \in F_j$,*

$$d(x^j, F_i) \leq \beta \left[\sup\nolimits_{t\in T} \left\{b_t^i - \left(a_t^i\right)' x^j\right\}\right]_+, \quad i, j = 1, 2, \quad i \neq j,$$

where $[\alpha]_+ := \max\{0, \alpha\}$.

Proof. The boundedness of F implies that $\mathcal{F}$ is usc at π, and $\widehat{\varepsilon} > 0$ exists such that $F_1 \subset F + B$ if $\delta(\pi_1, \pi) < \widehat{\varepsilon}$. Thus we can find a positive scalar μ such that $\|x^1\| \leq \mu$ for every $x^1 \in F_1$, provided that $\delta(\pi_1, \pi) < \widehat{\varepsilon}$. Moreover, it can be assumed, without loss of generality, that $F_1 \neq \emptyset$ in this $\widehat{\varepsilon}$-neighborhood of π, because of the lower semicontinuity of $\mathcal{F}$ at π.

Now let us consider, in this neighborhood, two problems, π_1 and π_2. Take, for instance, an arbitrary $x^2 \in F_2$ and suppose that $x^2 \notin F_1$ (otherwise the inequality to be proved holds trivially). Suppose that $x^1 \in F_1$ satisfies the $d(x^2, F_1) = \|x^1 - x^2\|$.

Following a reasoning similar to that in [7, Thm. 3.1], we obtain

$$d(x^2, F_1) = \|x^1 - x^2\| \leq \frac{4\mu}{\rho} \sup\nolimits_{t\in T} \left\{b_t^1 - \left(a_t^1\right)' x^2\right\},$$

provided that $\delta(\pi_i, \pi) < \varepsilon := \min\{\widehat{\varepsilon}, \frac{\rho}{2}\left(1 + n^{1/2}\mu\right)^{-1}\}$, $i = 1, 2$, where ρ is the "slack" associated with an arbitrarily chosen SS element, $\overline{x}$, of σ (i.e., $a_t'\overline{x} \geq b_t + \rho$ for all $t \in T$). We finish the proof by taking $\beta = \frac{4\mu}{\rho}$. □

There are, spread out in the literature, many contributions to the stability theory of $\mathcal{F}$ for a class of semi-infinite systems structurally richer than our linear inequality systems. This class is formed by those systems σ whose index set T is a compact set in the Euclidean space, defined as a solution set of finitely many analytic constraints. Moreover, the coefficient functions $a(\cdot)$ and $b(\cdot)$ are assumed to belong to $\mathcal{C}^1(T)$. Obviously, this class of $\mathcal{C}^1$-systems is a subclass of continuous systems.

Assuming that $\mathcal{C}^1(T)$ is equipped with the so-called Whitney topology, it is established in [11] that, under the assumption of the boundedness of F, $\mathcal{F}$ will be topologically stable at π (homeomorphic feasible sets in a neigborhood of π) if and only if the Mangasarian–Fromovitz constraint qualification (MFCQ) is held. The extension of this result for an unbounded F can be found in [10]. In this semi-infinite programming context (with $\mathcal{C}^1$ data), the equivalence between the MFCQ and the metric regularity of the constraints has been established in [9]. Parametric versions of these results are given in [12] and [13], again in the $\mathcal{C}^1$-data context (see also [16]). When one is confined to the context of linear data without any structure for T, the corresponding counterparts of these results were provided in [6] and [7], using ad hoc techniques based exclusively upon the semi-infinite version of the alternative theorems. (The *analytic* approach does not make sense in our context since nothing is known about the functions $a(\cdot)$ and $b(\cdot)$.)

4. Optimal value function and Hadamard well-posedness. Let us consider the *sublevel sets* of the problem π:

$$L(\alpha) := \{x \in F \mid c'x \leq \alpha\} = \{x \in \mathbb{R}^n \mid a_t'x \geq b_t,\ t \in T;\ c'x \leq \alpha\},\ \ \alpha \in \mathbb{R}.$$

$L(\alpha)$ depends on π. So, the sublevel sets of a different problem π_1 will be denoted $L_1(\alpha)$.

Obviously, $O^+(L(\alpha)) = \{y \in \mathbb{R}^n \mid a_t'y \geq 0\ ,\ t \in T\ ;\ c'y \leq 0\} = \{a_t,\ t \in T;\ -c\}^o$, which is independent of α, so that all the nonempty sublevel sets will have the same recession cone.

In the following key lemma, $\operatorname{int}_c(\Pi_s)$ will represent the interior of Π_s in the topology relative to Π_c. Theorem 2.7 in [5] can be obtained as an immediate corollary of this lemma together with Theorem 3.1.

LEMMA 4.1. *$\pi \in \operatorname{int}_c(\Pi_s)$ if and only if F^* is a nonempty bounded set.*

Proof. If F^* is a nonempty bounded set, $O^+(F^*) = \{0_n\} = \{a_t\ ,\ t \in T\ ;\ -c\}^o$ and, consequently, $0_n \in \operatorname{int}(\mathbb{R}^n) = \operatorname{int}(cone\,(\{a_t\ ,\ t \in T\ ;\ -c\}))$. Now, let us note that if $\delta(\pi_1,\pi)$ is small enough one still has $0_n \in \operatorname{int}\left(cone\ \left(\left\{a_t^1\ ,\ t \in T\ ;\ -c^1\right\}\right)\right)$ [7, Lem. 4.2]. Thus, by reversing our previous argument, we observe that every nonempty sublevel set of any consistent problem π_1 in a certain neighborhood of π is bounded and, then, F_1^* is nonempty ($c'x$ attains its minimum in a compact set).

Conversely, if $\pi = (c,\sigma) \in \operatorname{int}_c(\Pi_s)$ and F^* is unbounded, we shall take $u \in O^+(F^*)$, $u \neq 0_n$, and then we shall construct the sequence of problems $\{\pi_r := \left(c - \frac{1}{r}u,\ \sigma\right)\}$.

Obviously, $\lim_r \pi_r = \pi$ and $\{\pi_r\} \subset \Pi_c \backslash \Pi_b$, because, whichever $x^* \in F^*$ we take, one has $x^* + \lambda u \in F^* \subset F = F_r$ for all $\lambda > 0$, and $\lim_{\lambda\to\infty}\left(c - \frac{1}{r}u\right)'(x^* + \lambda u) = v - \frac{1}{r}u'x^* - \lim_{\lambda\to\infty}\frac{\lambda}{r}\|u\|^2 = -\infty$. Hence, the existence of such a sequence $\{\pi_r\}$ contradicts our current hypothesis. □

The continuity properties of the optimal value function ϑ are established in the following theorem.

THEOREM 4.2. *Let $\pi = (c,\sigma) \in \Pi_c$. Then*

i. *$\mathcal{F}$ is lsc at π if and only if ϑ is usc at π.*

ii. *If F^* is a nonempty bounded set, ϑ will be lsc at π. If $\pi \in \Pi_b$, the converse statement holds.*

iii. *If $\mathcal{F}$ is lsc at π and F^* is a nonempty bounded set, then we can find positive scalars, η and k, such that $\delta(\pi_i,\pi) < \eta$, $i = 1,2$, yield the* Lipschitzian inequality

$$|\vartheta(\pi_1) - \vartheta(\pi_2)| \leq k\delta(\pi_1,\pi_2).$$

Proof. i. The "only if" part is a straightforward consequence of [4, Prop. 2, Chap. IX]. In order to prove the converse statement, let us consider that ϑ is usc at π. Let $\mu > v$. Then, there will exist $\eta > 0$ such that $\delta(\pi_1,\pi) < \eta$ implies $v_1 \leq \mu$ and, necessarily, $\pi_1 \in \Pi_c$; i.e., $\pi \in \operatorname{int}(\Pi_c)$ and, so, $\mathcal{F}$ is lsc at π.

ii. Given the scalar $\varepsilon > 0$, we have to prove that $\eta > 0$ exists such that $\delta(\pi_1,\pi) < \eta$ implies $v_1 \geq v - \varepsilon$. If $\rho > 0$ satisfies $F^* \subset \rho B$, we shall take the open set $W := \{x \in \mathbb{R}^n \mid c'x > v - (\varepsilon/2)\} \cap \rho B$. Obviously, $W \supset F^*$.

Let us consider the system

$$\widetilde{\sigma} := \{a_t'x \geq b_t\ ,\ t \in T\ ;\ c'x \leq v\} \tag{4.1}$$

with index set $\widetilde{T} := T \cup \{t_0\}$, where t_0 is the index associated with the last inequality of $\widetilde{\sigma}$ ($t_0 \notin T$). Obviously, its solution set, denoted by $\widetilde{F}$, coincides with F^*, which is a nonempty bounded set by assumption. Consequently, if we represent by $\widetilde{\mathcal{F}}$ the feasible set mapping defined on the parameter space $\widetilde{\Pi}$ of all the LSIP problems with constraint systems having $\widetilde{T}$ as index set, it follows that $\widetilde{\mathcal{F}}$ is usc at $\widetilde{\pi} := (c, \widetilde{\sigma})$. Hence, $\widehat{\eta} > 0$ exists such that, if $\widetilde{\delta}$ denotes the corresponding extended distance in $\widetilde{\Pi}$, $\widetilde{\delta}\left(\widetilde{\pi}_1, \widetilde{\pi}\right) < \widehat{\eta}$ will imply $\widetilde{F}_1 \subset W$ (although $\widetilde{F}_1$ might be empty).

Let us take any problem $\pi_1 = \left(c^1, \sigma_1\right)$ satisfying $\delta\left(\pi_1, \pi\right) < \eta$, with $\eta := \min\{\widehat{\eta}, \varepsilon/(2\rho n^{1/2})\}$. Define the associated problem in $\widetilde{\Pi}$, $\widetilde{\pi}_1 = \left(c^1, \widetilde{\sigma}_1\right)$, where

$$\widetilde{\sigma}_1 := \left\{ \left(a_t^1\right)' x \geq b_t^1 \, , \; t \in T \, ; \; \left(c^1\right)' x \leq v \right\}.$$

(The right-hand side term of the last constraint is fixed at $v = \vartheta\left(\pi\right)$.) It is obvious that $\widetilde{\delta}\left(\widetilde{\pi}_1, \widetilde{\pi}\right) = \delta\left(\pi_1, \pi\right) < \eta \leq \widehat{\eta}$ and, so, $\widetilde{F}_1 = L_1\left(v\right) \subset W$.

Two possibilities can arise. If $\widetilde{F}_1 = \emptyset$, we have $v_1 \geq v > v - \varepsilon$ (possibly, $v_1 = +\infty$). Otherwise (i.e., when $\widetilde{F}_1 \neq \emptyset$), if we take an arbitrary $x^* \in F_1^* \subset \widetilde{F}_1$, it can be written

$$v_1 = \left(c^1\right)' x^* = c'x^* + \left(c^1 - c\right)' x^* > v - \frac{\varepsilon}{2} - \left\|c^1 - c\right\|_\infty \left\|x^*\right\| n^{1/2} > v - \varepsilon.$$

Assume now that ϑ is lsc at $\pi \in \Pi_b$, and let us show that the level set $L(\mu)$, with $\mu > v$, is bounded, in which case π will be solvable and F^* bounded. Otherwise, we can take $u \in O^+(L(\mu))$, $u \neq 0_n$, and then construct the sequence $\{\pi_r := (c - \frac{1}{r}u, \sigma)\}$. Obviously, $\lim_r \pi_r = \pi$ and, reasoning as in Lemma 4.1, we prove that $\{\pi_r\} \subset \Pi_c \setminus \Pi_b$, which contradicts our current hypothesis.

iii. By Theorem 3.1 and Lemma 4.1 there will exist $\widehat{\eta} > 0$ such that $\delta\left(\pi_1, \pi\right) < \widehat{\eta}$ implies $\pi_1 \in \Pi_s$. Given $\varepsilon > 0$, the upper semicontinuity of ϑ at π guarantees that, if $\widehat{\eta}$ is small enough, one also has $v_1 \leq v + \varepsilon$, which is equivalent in this case to $L_1\left(v + \varepsilon\right) \neq \emptyset$, provided that $\delta\left(\pi_1, \pi\right) < \widehat{\eta}$.

If we consider, instead of the system introduced in (4.1), the system

$$\widetilde{\sigma} := \left\{a_t'x \geq b_t \, , \; t \in T \, ; \; c'x \leq v + \varepsilon\right\},$$

we observe that $\widetilde{F} = L\left(v + \varepsilon\right)$ is bounded (all the nonempty sublevel sets are bounded because $F^* = L\left(v\right)$ enjoys this property) and $\widetilde{\mathcal{F}}$ will again be usc at $\widetilde{\pi}$. Taking $\widehat{\eta}$ sufficiently small and $\pi_1 = \left(c^1, \sigma_1\right)$ satisfying $\delta\left(\pi_1, \pi\right) < \widehat{\eta}$, we have

$$L_1\left(v + \varepsilon\right) \subset L\left(v + \varepsilon\right) + B, \tag{4.2}$$

since $L_1\left(v + \varepsilon\right)$ is the feasible set of $\widetilde{\pi}_1 = \left(c^1, \widetilde{\sigma}_1\right)$, where

$$\widetilde{\sigma}_1 := \left\{ \left(a_t^1\right)' x \geq b_t^1 \, , \; t \in T \, ; \; \left(c^1\right)' x \leq v + \varepsilon \right\}$$

(note that $\widetilde{\delta}\left(\widetilde{\pi}_1, \widetilde{\pi}\right) = \delta\left(\pi_1, \pi\right)$).

Statement (4.2) means that $\mu > 0$ can be found such that $\|x\| \leq \mu$ for all $x \in L_1\left(v + \varepsilon\right)$ and for every π_1 in the $\widehat{\eta}$-neighborhood of π.

Applying Lemma 3.2 to $\widetilde{\pi} = (c, \widetilde{\sigma})$, we conclude the existence of $\eta > 0$ (we shall take $\eta < \min\left\{1, \widehat{\eta}\right\}$) and $\beta > 0$ such that, if $\pi_i = \left(c^i, \sigma_i\right)$, $i = 1, 2$, are contained in the η-neighborhood of π and, since $L_i\left(v + \varepsilon\right)$, $i = 1, 2$, is the feasible set of $\widetilde{\pi}_i = \left(c^i, \widetilde{\sigma}_i\right)$,

$\widetilde{\sigma}_i := \left\{ \left(a_t^i\right)' x \geq b_t^i\,,\ t \in T\,;\ \left(c^i\right)' x \leq v + \varepsilon \right\}$, one has for $x^2 \in L_2(v+\varepsilon)$

$$\begin{aligned} d\left(x^2, L_1\left(v+\varepsilon\right)\right) &\leq \beta \max \left[\sup\nolimits_{t\in T} \left\{ b_t^1 - \left(a_t^1\right)' x^2 \right\}\ ,\ \left(c^1\right)' x^2 - v - \varepsilon\ ,\ 0\right] \\ &= \beta \max \Big[\sup\nolimits_{t\in T} \left\{ \left[b_t^2 - \left(a_t^2\right)' x^2\right] + \left[\left(b_t^1 - b_t^2\right) - \left(a_t^1 - a_t^2\right)' x^2\right] \right\}\ , \\ &\qquad\qquad \left(c^2\right)' x^2 - v - \varepsilon + \left(c^1 - c^2\right)' x^2\ ,\ 0\Big] \\ &\leq \beta \max \left[\sup\nolimits_{t\in T} \left\{ \left(b_t^1 - b_t^2\right) - \left(a_t^1 - a_t^2\right)' x^2 \right\}\ ,\ \left(c^1 - c^2\right)' x^2\ ,\ 0\right] \\ &\qquad \leq \beta \left(1 + \mu\, n^{1/2}\right) \delta\left(\pi_1, \pi_2\right) = \beta_0 \delta\left(\pi_1, \pi_2\right), \end{aligned}$$

where $\beta_0 := \beta\left(1 + \mu\, n^{1/2}\right)$. Now, if $x^2 \in F_2^* \subset L_2\left(v+\varepsilon\right)$, and taking $x^1 \in L_1\left(v+\varepsilon\right)$ such that $\left\|x^1 - x^2\right\| = d\left(x^2, L_1\left(v+\varepsilon\right)\right)$, it follows that

$$\begin{aligned} v_1 - v_2 = v_1 - \left(c^2\right)' x^2 &\leq \left(c^1\right)' x^1 - \left(c^2\right)' x^2 \leq \left\|c^1 - c^2\right\| \left\|x^2\right\| + \left\|c^1\right\| \left\|x^1 - x^2\right\| \\ &\leq \mu\, n^{1/2} \delta\left(\pi_1, \pi_2\right) + n^{1/2} \left(\left\|c\right\|_\infty + \eta\right) \beta_0 \delta\left(\pi_1, \pi_2\right) \leq k\delta\left(\pi_1, \pi_2\right), \end{aligned}$$

provided that $k := n^{1/2}\left[\mu + \beta_0\left(\left\|c\right\|_\infty + 1\right)\right]$.

Repeating the procedure for $v_2 - v_1$, one obtains $\left|v_1 - v_2\right| \leq k\delta\left(\pi_1, \pi_2\right)$. □

In LSIP, existence and continuous dependence of the optimal solutions from problem's data might be established as follows.

Given $\{\pi_r = (c^r, \sigma_r)\} \subset \Pi_b$ such that $\lim_r \pi_r = \pi$, the sequence $\{x^r\}$ is said to be an *asymptotically minimizing sequence* (a.m.s.) for π *associated with* $\{\pi_r\}$ if $x^r \in F_r$ for all r, and

$$\lim\nolimits_r \left\{ \left(c^r\right)' x^r - v_r \right\} = 0;$$

i.e., as r increases, x^r approximately solves the approximating problem π_r.

The problem $\pi \in \Pi_s$ will be *Hadamard well posed* (Hwp) if for each $x^* \in F^*$ and for each possible sequence $\{\pi_r\} \subset \Pi_b$ converging to π, there exists at least an associated a.m.s. converging to x^*.

THEOREM 4.3. *Given* $\pi = (c, \sigma) \in \Pi_s$, *the following statements hold*:

i. *If* π *is Hwp, then the restriction of* ϑ *to* Π_b, *denoted by* ϑ_b, *is continuous at* π. *If* $\mathcal{F}$ *is lsc at* π, *the converse statement is also true.*

ii. *Provided that* F^* *is bounded,* π *is Hwp if and only if either* $\mathcal{F}$ *is lsc at* π *or* F *is a singleton.*

iii. *When* F^* *is unbounded and* π *is Hwp,* $\mathcal{F}$ *has to be lsc at* π.

Proof. i. First, we assume that π is Hwp, and take $\{\pi_r\} \subset \Pi_b$ converging to π. We will see that $\lim_r v_r = v$.

The definition of Hadamard well-posedness states that, given $x^* \in F^*$, there will exist a sequence $\{x^r\}$ tending to x^*, such that $x^r \in F_r$ and $\lim_r \left\{ \left(c^r\right)' x^r - v_r \right\} = 0$. Since $\lim_r \left(c^r\right)' x^r = c' x^* = v$, we obtain $\lim_r v_r = v$.

In order to prove the converse, we start from the continuity of ϑ_b at π and from the lower semicontinuity of $\mathcal{F}$ at π. If $\{\pi_r\} \subset \Pi_b$ converges to π and $x^* \in F^* \subset F$, the lower semicontinuity of $\mathcal{F}$ at π implies $x^* \in \liminf_r F_r$ (condition v in Theorem 3.1). In other words, there must exist a sequence $\{x^r\}$ converging to x^* and such that $x^r \in F_r$. Then $\{x^r\}$ turns out to be an a.m.s. for π associated with $\{\pi_r\}$, since $\lim_r \left\{ \left(c^r\right)' x^r - v_r \right\} = 0$.

ii. Assume that π is Hwp, that F^* is bounded, and that, at the same time, F is not a singleton and $\mathcal{F}$ fails to be lsc at π.

Pick an optimal point $x^* \in F^*$ and an arbitrary $y \in F \backslash \{x^*\}$. Define $u := y - x^*$ and, associated with each $r \in \mathbb{N}$, take a positive scalar k_r satisfying

$$\left\| \frac{1}{k_r} u \right\|_\infty < \frac{1}{r} \qquad \text{and} \qquad \left| \frac{1}{k_r} u' y \right| < \frac{1}{r}.$$

According to condition iii in Theorem 3.1, the unfulfillment of the lower semicontinuity of $\mathcal{F}$ at π gives rise to the existence of a sequence $\{\lambda^p\} \subset \mathbb{R}_+^{(T)}$, satisfying $\sum_{t \in T} \lambda_t^p = 1$, $p = 1, 2, \ldots$, and

$$0_{n+1} = \lim_p \sum_{t \in T} \lambda_t^p \begin{pmatrix} a_t \\ b_t \end{pmatrix}. \tag{4.3}$$

Let us introduce, for each $r \in \mathbb{N}$, the problem $\pi_r = (c, \sigma_r)$ with

$$\sigma_r := \left\{ \left(a_t + \frac{1}{k_r} u \right)' x \geq b_t + \frac{1}{k_r} u' y, \ t \in T \right\}.$$

Obviously, $\delta(\pi_r, \pi) < \frac{1}{r}$ and, so, $\lim_r \pi_r = \pi$. Moreover, $y \in F_r$ for every r, and $u'x \geq u'y$ is a consequence of each σ_r, since (4.3) implies

$$\lim_p \sum_{t \in T} \lambda_t^p \begin{pmatrix} a_t + \frac{1}{k_r} u \\ b_t + \frac{1}{k_r} u' y \end{pmatrix} = \frac{1}{k_r} \begin{pmatrix} u \\ u' y \end{pmatrix}.$$

According to Lemma 4.1, the boundedness of F^* entails that $\{\pi_r\}_{r \geq m} \subset \Pi_s$ for a certain m.

We have realized that $u'x \geq u'y$ for every $x \in F_r$, but $u'(x^* - y) = -\|u\|^2$ and, so, $u'x^* < u'y$. This implies that, for this optimal point x^* and for this particular sequence of bounded problems converging to π, there is no associated a.m.s. $\{x^r\}_{r \geq m}$ converging to x^*, and the Hadamard well-posedness fails.

Let us proceed with the proof of the converse. First, we assume that F^* is bounded and $\mathcal{F}$ is lsc at π. By Theorem 4.2, parts i and ii, we conclude that ϑ is continuous at π and then apply the converse statement in part i. If, alternatively, $F = F^* = \{x^*\}$, our first preliminary conclusion is that $\mathcal{F}$ is usc at π, and we shall check that the condition for the Hadamard well-posedness of π is fulfilled in this case.

Let us consider an arbitrary sequence $\{\pi_r\} \subset \Pi_b$ converging to π. Lemma 4.1 applies again, yielding m such that $\pi_r \in \Pi_s$ if $r \geq m$. Take $x^r \in F_r^*$ for $r \geq m$ and $x^r \in F_r$ if $r < m$. Then, $\{x^r\}$ is obviously an a.m.s. for π associated with $\{\pi_r\}$.

The upper semicontinuity of $\mathcal{F}$ at π implies that, given any open set W containing $F = \{x^*\}$, there will exist an integer r_0 such that $F_r \subset W$ if $r \geq r_0$. In other words, $x^r \in W$ when $r \geq r_0$, and this means $\lim_r x^r = x^*$.

iii. Take $x^* \in F^*$ and $u \in O^+(F^*)$ with $\|u\|_\infty = 1$. Then define $\mu_r = \frac{1}{u'x^* + r}$, with r sufficiently large to guarantee the positiveness of the denominator, and take $c^r := c - \mu_r u$ and $y^r := x^* + ru$. Obviously, $y^r \in F^*$ and $(c^r)' y^r = v - 1$.

Now let us define the systems

$$\sigma_r := \left\{ \left(a_t + \frac{1}{k_r} c^r \right)' x \geq b_t + \frac{v - 1}{k_r}, \ t \in T \right\}, \quad r = 1, 2, \ldots,$$

where the constants k_r are chosen in such a way that

$$\left\| \frac{c^r}{k_r} \right\|_\infty < \frac{1}{r} \quad \text{and} \quad \left| \frac{v-1}{k_r} \right| < \frac{1}{r}.$$

Finally, we shall introduce the associated problems $\pi_r := (c^r, \sigma_r)$, which obviously verify $\lim_r \pi_r = \pi$ and $\pi_r \in \Pi_c$ (because $y^r \in F_r$).

If $\mathcal{F}$ is not lsc at π, condition iii in Theorem 3.1 will fail, and a sequence $\{\lambda^p\} \subset \mathbb{R}_+^{(T)}$ exists verifying $\sum_{t \in T} \lambda_t^p = 1$, $p = 1, 2, \ldots$, and (4.3). This implies, for each $r \in \mathbb{N}$,

$$\lim_p \sum_{t \in T} \lambda_t^p \begin{pmatrix} a_t + \frac{1}{k_r} c^r \\ b_t + \frac{v-1}{k_r} \end{pmatrix} = \frac{1}{k_r} \begin{pmatrix} c^r \\ v-1 \end{pmatrix},$$

and the nonhomogeneous Farkas lemma allows us to conclude that $(c^r)' x \geq v - 1$ is a consequence of σ_r, which in fact entails $y^r \in F_r^*$ and $v_r = v - 1$, contradicting part i. □

COROLLARY 4.4. *Let π be a Hwp problem. If x^* is the limit of a certain a.m.s, then x^* will be optimal for π (i.e., $x^* \in F^*$).*

Proof. There must exist a sequence $\{\pi_r\} \subset \Pi_b$ converging to π, and an associated sequence $\{x^r\}$, $x^r \in \mathcal{F}(\pi_r)$ for every $r \in \mathbb{N}$, such that

$$\lim_r \left\{ (c^r)' x^r - v_r \right\} = 0 \quad \text{and} \quad \lim_r x^r = x^*.$$

Statement i in Theorem 4.3 establishes the continuity of ϑ_b at π, entailing $\lim_r v_r = v$. Thus,

$$0 = \lim_r \left\{ (c^r)' x^r - v_r \right\} = c' x^* - v$$

at the same time that x^* is feasible for π since, for every $t \in T$,

$$0 \leq \lim_r \left\{ (a_t^r)' x^r - b_t^r \right\} = a_t' x^* - b_t$$

(convergence in (Π, δ) yields $\lim_r c^r = c$, $\lim_r a_t^r = a_t$, and $\lim_r b_t^r = b_t$ for all $t \in T$). □

The only antecedents of the results presented in this section come from the field of continuous LSIP, and they can be traced out from [2] and the references therein. Most of the statements in [2, sects. 2 and 3] can be obtained as corollaries of Theorem 4.2, emphasizing the fact that our results subsume all the previous contributions to the continuous problem. Additionally, the Lipschitzian condition given in [2, Thm. 3.5] is a trivial consequence of the inequality established in part iii of Theorem 4.2.

5. Optimal set mapping.

The only theorem in this section concerns the stability behavior of $\mathcal{F}^*$.

THEOREM 5.1. *Given $\pi \in \Pi_s$, the following propositions hold*:

i. *$\mathcal{F}^*$ is closed at π if and only if either $\mathcal{F}$ is lsc at π or $F = F^*$.*

ii. *If $\mathcal{F}^*$ is usc at π, then $\mathcal{F}^*$ is closed at π. The converse statement holds if F^* is bounded.*

iii. *$\mathcal{F}^*$ is lsc at π if and only if $\mathcal{F}$ is lsc at π and F^* is a singleton.*

Proof. i. Suppose that $\mathcal{F}^*$ is closed at π and that, simultaneously, $F \neq F^*$ and $\mathcal{F}$ is not lsc at π.

Let $y \in F \backslash F^*$. Then $c'y = v + \alpha$ for a certain $\alpha > 0$, and we shall consider a sequence of problems $\{\pi_r = (c, \sigma_r)\}$, where

$$\sigma_r := \left\{ \left(a_t + r^{-1} c\right)' x \geq b_t + r^{-1} (v + \alpha), \ t \in T \right\}, \qquad r = 1, 2, \ldots.$$

It follows that $\lim_r \pi_r = \pi$, that $y \in F_r$ for all r, and that $c'x \geq c'y$ is a consequence of σ_r, again for every r (we should apply the technique used in the proof of proposition ii in Theorem 4.3). This fact actually implies $y \in F_r^*$, $r = 1, 2, \ldots$, and the closedness of $\mathcal{F}^*$ at π gives rise to the contradiction $y \in F^*$.

We continue with the proof of the converse statement. If $F = F^*$, we take sequences $\{\pi_r\}$ and $\{x^r\}$, converging to π and $\overline{x}$, respectively, and also verifying $x^r \in F_r^*$. Since $F_r^* \subset F_r$ and $\mathcal{F}$ is always closed at π, one attains $\overline{x} \in F = F^*$.

Alternatively, if $\mathcal{F}$ is lsc at π and we have $\lim_r \pi_r = \pi$, $\lim_r x^r = \overline{x}$, and $x^r \in F_r^*$, $r = 1, 2, \ldots$, we shall prove that $c'\overline{x} \leq c'x^0$ for any possible SS element of σ, x^0. First, we prove that $x^0 \in F_r$ if $r \geq r_0$ for a certain r_0. Actually, if $\rho > 0$ satisfies $a_t' x^0 \geq b_t + \rho$ for all $t \in T$, and $\delta(\pi_0, \pi) < \frac{\rho}{2} \min\{1, \ n^{-1/2} \left\|x^0\right\|^{-1}\}$ (writing $\left\|x^0\right\|^{-1} = +\infty$ in the case $x^0 = 0_n$), the Cauchy–Schwarz inequality leads us to

$$\left(a_t^0\right)' x^0 \geq a_t' x^0 - \left\|x^0\right\| \left\|a_t^0 - a_t\right\| \geq a_t' x^0 - \frac{\rho}{2} \geq b_t + \frac{\rho}{2} \geq b_t^0.$$

Once we have established $x^0 \in F_r$, if r is sufficiently large, we write $(c^r)' x^r \leq (c^r)' x^0$ and, taking limits for $r \to \infty$, $c'\overline{x} \leq c'x^0$ results.

Since F is, in this case, the closure of the set of all the SS elements of σ (condition vi in Theorem 3.1), one concludes that $c'\overline{x} \leq c'y$ for every feasible point $y \in F$, i.e., $\overline{x} \in F^*$.

ii. Since (Π, δ) behaves locally as the metric space (Π, d) with $d(\pi_1, \pi) = \min\{1, \delta(\pi_1,\pi)\}$, we can apply any property of set-valued mappings between metric spaces (see, for instance, [1]). In particular the upper semicontinuity of $\mathcal{F}^*$ at π and the closedness of the set F^* imply that $\mathcal{F}^*$ is a closed mapping at π.

In order to prove the converse statement, we assume that F^* is bounded. If $F = F^*$, we have that $\mathcal{F}$ is usc at π, entailing the upper semicontinuity of $\mathcal{F}^*$ at the same problem π. When $\mathcal{F}$ is lsc at π, we use the following reasoning.

Let W be an open set containing F^*, the last one being interpreted as the solution set of the system $\widetilde{\sigma}$ introduced in (4.1). The boundedness of $\widetilde{F} \equiv F^*$ implies the upper semicontinuity of $\widetilde{\mathcal{F}}$ at $\widetilde{\pi} := (c, \widetilde{\sigma})$. In other words, $\eta_1 > 0$ exists such that $\widetilde{\delta}(\widetilde{\pi}_1, \widetilde{\pi}) \leq \eta_1$ guarantees $\widetilde{F}_1 \subset W$. In particular, if we consider $\widetilde{\pi}_1 := (c, \widetilde{\sigma}_1)$, with

$$\widetilde{\sigma}_1 := \{a_t' x \geq b_t \ , \quad t \in T \ ; \quad c'x \leq v + \eta_1\},$$

we deduce the inclusion $\widetilde{F}_1 = L(v + \eta_1) \subset W$.

Let $\overline{x}$ be an SS element of σ (remember that $\mathcal{F}$ is lsc at π). If $c'\overline{x} < v + \eta_1$, it is evident that $\overline{x}$ is an SS element of $\widetilde{\sigma}_1$ too. Otherwise, we pick $\widetilde{x} \in \widetilde{F}_1$ satisfying $c'\widetilde{x} < v + \eta_1$. Then, if λ is sufficiently small, $\lambda\overline{x} + (1 - \lambda)\widetilde{x}$ will be an SS element of $\widetilde{\sigma}$. In any case, we conclude that $\widetilde{\mathcal{F}}$ is lsc at $\widetilde{\pi}_1$, implying the existence of $\eta_2 > 0$ such that $\widetilde{\delta}(\widetilde{\pi}_2, \widetilde{\pi}_1) \leq \eta_2$ leads us to $\widetilde{F}_2 \neq \emptyset$.

Moreover, the boundedness of $\widetilde{F}_1 = L(v + \eta_1)$ implies that $\widetilde{\mathcal{F}}$ is also usc at $\widetilde{\pi}_1$, and for a certain $\eta_3 > 0$, $\widetilde{\delta}(\widetilde{\pi}_2, \widetilde{\pi}_1) \leq \eta_3$ ensures $\widetilde{F}_2 \subset W$.

Now, take a problem π_2 such that $\delta(\pi_2, \pi) < \eta := \min\{\eta_2, \eta_3\}$, and let us associate with it the problem $\widetilde{\pi}_2 := \left(c^2, \widetilde{\sigma}_2\right)$ in $\widetilde{\Pi}$, with

$$\widetilde{\sigma}_2 := \left\{ \left(a_t^2\right)' x \geq b_t^2 \ , \quad t \in T \ ; \quad \left(c^2\right)' x \leq v + \eta_1 \right\}.$$

TABLE 5.1
Stability and Hadamard well-posedness of the LSIP problem.

F^* nonempty		$\mathcal{F}$ lsc at π	$\mathcal{F}$ non-lsc at π
F^* is a singleton	$F = F^*$	I,II,III,IV, Hwp	$\overline{\mathrm{I}}$,II,III,$\overline{\mathrm{IV}}$,Hwp
	$F \neq F^*$		$\overline{\mathrm{I}},\overline{\mathrm{II}}$,III,$\overline{\mathrm{IV}},\overline{\mathrm{Hwp}}$
F^* is bounded, not a singleton	$F = F^*$	$\overline{\mathrm{I}}$,II,III,IV, Hwp	$\overline{\mathrm{I}}$,II,III,$\overline{\mathrm{IV}},\overline{\mathrm{Hwp}}$
	$F \neq F^*$		$\overline{\mathrm{I}}$, $\overline{\mathrm{II}}$, III, $\overline{\mathrm{IV}}$, $\overline{\mathrm{Hwp}}$
F^* is unbounded	$F = F^*$	Cell A: $\overline{\mathrm{I}}$, $\overline{\mathrm{III}_c}$,IV	Cell C: $\overline{\mathrm{I}},\overline{\mathrm{III}_b}$, $\overline{\mathrm{IV}},\overline{\mathrm{Hwp}}$
	$F \neq F^*$	Cell B: $\overline{\mathrm{I}},\overline{\mathrm{III}_c}$,IV	$\overline{\mathrm{I}},\overline{\mathrm{II}},\overline{\mathrm{III}_b},\overline{\mathrm{IV}},\overline{\mathrm{Hwp}}$

Obviously, $\widetilde{\delta}\left(\widetilde{\pi}_2, \widetilde{\pi}_1\right) < \eta$ and, consequently, $\emptyset \neq \widetilde{F}_2 = L_2\left(v + \eta_1\right) \subset W$. Thus, $F_2^* \subset L_2\left(v + \eta_1\right) \subset W$ and the upper semicontinuity of $\mathcal{F}^*$ at π follows.

iii. Let us suppose first that $F^* = \{x^*\}$ and that $\mathcal{F}$ is lsc at π. Then parts i and ii apply, and we conclude that $\mathcal{F}^*$ is usc at π.

Since $\mathcal{F}$ is lsc at π, there will exist $\eta_1 > 0$ such that $\delta\left(\pi_1, \pi\right) < \eta_1$ implies $\pi_1 \in \Pi_c$. Lemma 4.1 allows us to write $\pi_1 \in \Pi_s$ if η_1 is small enough.

Now take an open set W containing x^*. The upper semicontinuity of $\mathcal{F}^*$ at π gives rise to the existence of $\eta_2 > 0$ such that $\delta\left(\pi_1, \pi\right) < \eta_2$ implies $F_1^* \subset W$. If $\eta := \min\left\{\eta_1, \eta_2\right\}$, one gets $\emptyset \neq F_1^* \subset W$, when $\delta\left(\pi_1, \pi\right) < \eta$, so that $F_1^* \cap W \neq \emptyset$ and $\mathcal{F}^*$ is certainly lsc at π.

Next we shall prove that the lower semicontinuity of $\mathcal{F}^*$ at π implies that π has a unique optimal solution. If this is not the case, we pick two different points in F^*, x^*, and y^* and define $u := y^* - x^*$. We shall introduce the sequence of problems $\pi_r := \left(c^r, \sigma\right)$, $r = 1, 2, \ldots$, with $c^r := c - \frac{1}{r}u$. Obviously, $\lim_r \pi_r = \pi$, and a contradiction will be attained.

Since $u'\left(y^* - x^*\right) > 0$, an open neighborhood of x^*, W, can be found such that $u'\left(y^* - x\right) > 0$ for every $x \in W$. Let us take an arbitrary $x \in W \cap F$, and notice that $\left(c^r\right)'\left(y^* - x\right) = c'\left(y^* - x\right) - \frac{1}{r}u'\left(y^* - x\right) < 0$. Hence $x \notin F_r^*$, and this contradicts the lower semicontinuity of $\mathcal{F}^*$ at π.

The last step in the proof will establish that the lower semicontinuity of $\mathcal{F}^*$ at π implies that this property also holds for $\mathcal{F}$. Actually, we shall see that $\pi \in \operatorname{int}\left(\Pi_c\right)$. In fact, if W is an open set such that $F^* \cap W$ is nonempty, there will exist $\eta > 0$ such that $\delta\left(\pi_1, \pi\right) < \eta$ yields $F_1^* \cap W \neq \emptyset$, and $F_1 \neq \emptyset$ in this neighborhood of π. □

In [5, Thms. 3.3 and 4.2], the continuity properties of the optimal set mapping at a continuous solvable problem π are analyzed. The optimal set mapping considered there is the restriction, $\mathcal{F}_{os}^*$, of $\mathcal{F}^*$ to the subset of continuous solvable problems, Π_{os}. So, the characterization of the lower semicontinuity of $\mathcal{F}_{os}^*$ at $\pi \in \Pi_{os}$ given in [5, Thm. 4.2] requires the existence of an extreme point of F to guarantee the existence of solvable problems in a neighborhood of π.

Table 5.1 summarizes all the results presented in the previous sections. The following symbols are used: I $\Leftrightarrow$ $\mathcal{F}^*$ is lsc at π; II $\Leftrightarrow$ $\mathcal{F}^*$ is usc at π; III $\Leftrightarrow$ ϑ is lsc at π; IV $\Leftrightarrow$ ϑ is usc at π; III_b $\Leftrightarrow$ ϑ_b is lsc at π; III_c $\Leftrightarrow$ ϑ_c is lsc at π ($\vartheta_c \equiv \vartheta\mid_{\Pi_c}$); $\overline{\mathrm{I}}$ means that I does not hold (etc.).

6. Examples. By means of the following series of examples it is shown that in the cells of Table 5.1 associated with the cases "F^* unbounded," every possible combination for the nonfixed properties can occur, showing that there is no additional underlying connection between them. All the examples are LSIP problems in $\mathbb{R}^2$, except Examples 6.3 and 6.4, which are posed in $\mathbb{R}^3$.

Cell A: F^* unbounded, $F = F^*$, and $\mathcal{F}$ lsc at π

Example 6.1. II and Hwp.

$$\begin{aligned} \pi : \text{Inf} \;\; & x_1 \\ \text{s.t.} \;\; & tx_1 + x_2 \geq -1 \;,\; t \in \mathbb{Z}, \\ & sx_2 \geq -1 \;,\; s \in \mathbb{N}. \end{aligned}$$

$F = F^* = \{0\} \times \mathbb{R}_+$ and $v = 0$. Moreover, 0_2 is an SS element and $\mathcal{F}$ is lsc at π. If $\delta(\pi_1, \pi) < 1$, we have $F_1^* \subset F_1 = F = F^*$ and $\mathcal{F}^*$ is trivially usc at π. In order to prove that π is Hwp, we have to establish the lower semicontinuity of ϑ_b at π, since this function is already usc at π as a consequence of the lower semicontinuity of $\mathcal{F}$ at π. In fact, $\pi_1 \in \Pi_b$ and $\delta(\pi_1, \pi) < 1$ implies that v_1 is attained in the only extreme point of F_1, namely, 0_2. In other words, $v_1 = 0$ and this entails the required continuity.

Example 6.2. II and $\overline{\text{Hwp}}$.

$$\begin{aligned} \pi : \text{Inf} \;\; & x_1 \\ \text{s.t.} \;\; & tx_1 + 0x_2 \geq -1 \;,\; t \in \mathbb{Z}. \end{aligned}$$

$F = F^* = \{0\} \times \mathbb{R}$ and $v = 0$. Since 0_2 is an SS element, $\mathcal{F}$ is lsc at π. If we define, for $r = 1, 2, \ldots$, the problem

$$\begin{aligned} \pi_r : \text{Inf} \;\; & \left(x_1 + \tfrac{1}{r}x_2\right) \\ \text{s.t.} \;\; & tx_1 + \tfrac{1}{r}x_2 \geq -1 \;,\; t \in \mathbb{Z}, \end{aligned}$$

whose feasible set is $F_r = \{0\} \times [-r, +\infty[$, we observe that $\lim_r \pi_r = \pi$ and $v_r = -1$. Thus ϑ_b fails to be lsc at π.

Example 6.3. $\overline{\text{II}}$ and Hwp.

$$\begin{aligned} \pi : \text{Inf} \;\; & x_1 \\ \text{s.t.} \;\; & tx_1 + x_2 + x_3 \geq -1 \;,\; t \in \mathbb{Z}, \\ & x_1 + sx_2 + x_3 \geq -1 \;,\; s \in \mathbb{N}, \\ & x_1 + x_2 + ux_3 \geq -1 \;,\; u \in \mathbb{N}, \\ & -x_2 + x_3 \geq -1. \end{aligned}$$

0_3 is an SS element and, so, $\mathcal{F}$ is lsc at π. It can be seen that $x_1 \geq 0$, $-x_1 \geq 0$, $x_2 \geq 0$, and $x_3 \geq 0$ are consequent relations of the constraint system σ. To this end, we divide the first (second, third) block of constraints by t (s, u) and take limits for $t \to \pm\infty$ $(s \to +\infty,\, u \to +\infty)$. Conversely, the infinitely many constraints in the first three blocks are themselves consequences of $x_1 = 0$, $x_2 \geq 0$, and $x_3 \geq 0$. Consequently, $F = F^* = \left\{x \in \mathbb{R}^3 \mid x_1 = 0,\; x_2 \geq 0,\; x_3 \geq 0,\; x_3 - x_2 \geq -1\right\}$.

If π_1 is a problem such that $\delta(\pi_1, \pi) < \varepsilon < 1$, we can write it as follows:

$$\begin{aligned} \pi_1 : \;\; \text{Inf} \;\; & \{(1 + \varepsilon_1)\, x_1 + \varepsilon_2 x_2 + \varepsilon_3 x_3\} \\ \text{s.t.} \;\; & (t + \varepsilon_1^t)\, x_1 + (1 + \varepsilon_2^t)\, x_2 + (1 + \varepsilon_3^t)\, x_3 \geq -1 + \varepsilon_4^t \;, \quad t \in \mathbb{Z}, \\ & (1 + \varepsilon_1^s)\, x_1 + (s + \varepsilon_2^s)\, x_2 + (1 + \varepsilon_3^s)\, x_3 \geq -1 + \varepsilon_4^s \;, \quad s \in \mathbb{N}, \\ & (1 + \varepsilon_1^u)\, x_1 + (1 + \varepsilon_2^u)\, x_2 + (u + \varepsilon_3^u)\, x_3 \geq -1 + \varepsilon_4^u \;, \quad u \in \mathbb{N}, \\ & \varepsilon_1^w x_1 + (-1 + \varepsilon_2^w)\, x_2 + (1 + \varepsilon_3^w)\, x_3 \geq -1 + \varepsilon_4^w. \end{aligned}$$

It is also obvious that $0_3 \in F_1$ and that the first three blocks of constraints are still equivalent to the finite system $\{x_1 = 0,\ x_2 \geq 0,\ x_3 \geq 0\}$; i.e.,

$$F_1 = \{x \in \mathbb{R}^3 \mid x_1 = 0,\ x_2 \geq 0,\ x_3 \geq 0 \text{ and } (-1 + \varepsilon_2^w)\, x_2 + (1 + \varepsilon_3^w)\, x_3 \geq -1 + \varepsilon_4^w\}.$$

The first part of our argument consists in showing that $\mathcal{F}^*$ fails to be usc at π. Actually, if one introduces the approximating sequence (ap.s.) of problems π_r, $r = 1, 2, \ldots$, which differ from π only in that the last constraint has been replaced by $-x_2 + \left(1 + \frac{1}{r}\right) x_3 \geq -1$, respectively, it becomes evident that the open set $W = \left\{x \in \mathbb{R}^3 \mid -x_2 + x_3 > -2\right\}$ contains F^*, but $x^r = (0, r+2, r)' \in F_r \setminus W = F_r^* \setminus W$. So, $F_r^* \nsubseteq W$ for every r and $\mathcal{F}^*$ is not usc at π.

The second part is devoted to establishing the lower semicontinuity of ϑ_b at π. Since F_1 is a polyhedral set, if $\pi_1 \in \Pi_b$, its optimal value will be attained in any of its two extreme points, namely,

$$0_3 \quad \text{and} \quad \left(0, \frac{-1 + \varepsilon_4^w}{-1 + \varepsilon_2^w}, 0\right)' ; \quad \text{i.e., } v_1 = \min\left\{0, \frac{\varepsilon_2\,(\varepsilon_4^w - 1)}{\varepsilon_2^w - 1}\right\}.$$

Accordingly, we shall write

$$v_1 \geq \frac{-\left|\varepsilon_2\right|(\varepsilon_4^w - 1)}{\varepsilon_2^w - 1} > \frac{-\varepsilon\,(1 + \varepsilon)}{1 - \varepsilon}$$

and, since $\lim_{\varepsilon \to 0} \frac{\varepsilon(1+\varepsilon)}{1-\varepsilon} = 0$, we conclude the intended property.

Example 6.4. $\overline{\text{II}}$ and $\overline{\text{Hwp}}$.

$$\begin{array}{rll} \pi: & \text{Inf } x_1 & \\ \text{s.t.} & tx_1 + x_2 + x_3 \geq -1\ , & t \in \mathbb{Z}, \\ & x_1 + sx_2 + x_3 \geq -1\ , & s \in \mathbb{N}, \\ & x_1 + x_2 + ux_3 \geq -1\ , & u \in \mathbb{N}, \\ & -x_2 \geq -1. & \end{array}$$

0_3 is an SS element and $F = F^* = \left\{x \in \mathbb{R}^3 \mid x_1 = 0, x_3 \geq 0 \text{ and } x_2 \in [0,1]\right\}$.

Now, let us introduce the ap.s. $\{\pi_r\}$ with π_r differing from π only in the last constraint, which is replaced by $-x_2 + \frac{1}{r}x_3 \geq -1$. Consider the open set $W = \{x \in \mathbb{R}^3 \mid -x_2 > -2\}$, the points $x^r = (0, 2, r)'$, $r = 1, 2, \ldots$, and observe that $W \supset F^*$ but $x^r \in F_r \setminus W = F_r^* \setminus W$, $r = 1, 2, \ldots$; i.e., $\mathcal{F}^*$ is not usc at π.

Next we prove that π is not Hwp. Now we take into account the ap.s. $\{\widetilde{\pi}_r\}$, such that $\widetilde{\pi}_r$ is obtained from changing the objective function of π by $x_1 - \frac{1}{r}x_3$ and substituting the last constraint of π by $-x_2 - \frac{1}{r}x_3 \geq -1$. We get $\widetilde{F}_r = conv\{0_3, (0,1,0)', (0,0,r)'\}$ and, consequently, $\widetilde{v}_r = -1$, $r = 1, 2, \ldots$; i.e., ϑ_b is not lsc at π.

Cell B: F^* unbounded, $F \neq F^*$, and $\mathcal{F}$ lsc at π.

Example 6.5. II and Hwp.

$$\begin{array}{rll} \pi: & \text{Inf } x_1 & \\ \text{s.t.} & tx_1 + x_2 \geq -1\ , & t \in \mathbb{N}, \\ & x_1 + sx_2 \geq -1\ , & s \in \mathbb{N}. \end{array}$$

0_2 is an SS element of π, entailing the lower semicontinuity of $\mathcal{F}$ at π. The constraints system is obviously equivalent to $x_1 \geq 0$ and $x_2 \geq 0$, so $F = \{x \in \mathbb{R}^2 \mid x_1 \geq 0, x_2 \geq 0\}$ and $F^* = \{0\} \times \mathbb{R}_+$. In addition, if $\delta\,(\pi_1, \pi) < 1$, we shall find $F_1 = F$ and, since the objective function of π_1 is $\left(c^1\right)' x = (1 + \varepsilon_1)\, x_1 + \varepsilon_2 x_2$, one gets $\left(c^1\right)' \binom{1}{0} = 1 + \varepsilon_1 > 0 = \left(c^1\right)' 0_2$. Hence, the point $(1,0)'$ is not optimal for π_1, which implies that in the nontrivial case, namely, $\pi_1 \in \Pi_b$, we have $\emptyset \neq F_1^* \subset F^*$; i.e., $\mathcal{F}^*$ is trivially usc at π. Simultaneously, $v_1 = 0 = v$ and ϑ_b is obviously lsc at π.

EXAMPLE 6.6. II and $\overline{\text{Hwp}}$.

$$\begin{aligned} \pi: \ & \text{Inf } x_1 \\ \text{s.t.} \ & tx_1 + x_2 \geq -1 \ , \quad t \in \mathbb{N}, \\ & x_1 + sx_2 \geq -1 \ , \quad s \in \mathbb{N}, \\ & -x_1 \geq -1. \end{aligned}$$

$\mathcal{F}$ is lsc at π because 0_2 is, once more, an SS element. It can be easily verified that $F = [0,1] \times \mathbb{R}_+$ and $F^* = \{0\} \times \mathbb{R}_+$. If $\delta(\pi_1, \pi) < 1$ we have $c^1 = (1+\varepsilon_1, \varepsilon_2)'$ and $F_1 = \left\{x \in \mathbb{R}^2 \mid x_1 \geq 0,\ x_2 \geq 0 \text{ and } \left(-1+\varepsilon_1^\omega\right) x_1 + \varepsilon_2^\omega x_2 \geq -1 + \varepsilon_3^\omega\right\}$. We shall distinguish two cases:

i. $\varepsilon_2^\omega < 0$. Then

$$F_1 = conv \left\{ 0_2,\ \left(\frac{1-\varepsilon_3^\omega}{1-\varepsilon_1^\omega}, 0\right)',\ \left(0, \frac{1-\varepsilon_3^\omega}{-\varepsilon_2^\omega}\right)' \right\}.$$

ii. $\varepsilon_2^\omega \geq 0$. Now F_1 is unbounded, with two extreme points,

$$0_2 \quad \text{and} \quad \left(\frac{1-\varepsilon_3^\omega}{1-\varepsilon_1^\omega}, 0\right)'.$$

In any case, if $\pi_1 \in \Pi_s$ (or, equivalently, $\pi_1 \in \Pi_b$ since π_1 is equivalent to an ordinary linear programming problem), the optimal value is attained at some extreme point. Notice that

$$\left(\frac{1-\varepsilon_3^\omega}{1-\varepsilon_1^\omega}, 0\right)'$$

will never be optimal, because

$$\left(c^1\right)' 0_2 = 0 < \left(1+\varepsilon_1\right) \frac{1-\varepsilon_3^\omega}{1-\varepsilon_1^\omega}$$

(remember that all the epsilons have absolute values smaller than 1). Consequently, v_1 will be attained at points with the first coordinate equal to zero; i.e., $F_1^* \subset F^*$ and $\mathcal{F}^*$ turns out to be usc at π.

Let us proceed, providing an ap.s. of problems for π, $\{\pi_r\}$, such that $v_r = -1$, $r = 1, 2, \ldots$, and, accordingly, ϑ_b will not be lsc at π. The problem π_r is derived from π, replacing the objective function by $\left(c^r\right)' x = x_1 - \frac{1}{r} x_2$ and the last constraint by $-x_1 - \frac{1}{r} x_2 \geq -1$. Since $F_r = conv\{0_2, (1,0)', (0,r)'\}$, $v_r = \left(c^r\right)' (0,r)' = -1$ results.

EXAMPLE 6.7. $\overline{\text{II}}$ and Hwp.

$$\begin{aligned} \pi: \ & \text{Inf } x_1 \\ \text{s.t.} \ & x_1 + 0x_2 \geq 0. \end{aligned}$$

$x^0 = (1,0)'$ is an SS element, $F = \mathbb{R}_+ \times \mathbb{R}$, and $F^* = \{0\} \times \mathbb{R}$.

Consider the approximating problem $\pi_r := \text{Inf}\left\{x_1 + \frac{1}{r} x_2 \mid x_1 + \frac{1}{r} x_2 \geq 0\right\}$, for which $\delta(\pi_r, \pi) = \frac{1}{r}$. Taking the open set $W := \left\{x \in \mathbb{R}^2 \mid x_1 > -1\right\}$ and the points $x^r := (-1, r)'$, one has $W \supset F^*$ but $x^r \in F_r^* \backslash W$, and the upper semicontinuity of $\mathcal{F}^*$ does not hold at π.

In the following step, the Hadamard well-posedness of π is shown. If π_1 is any problem obtained by perturbation of π, and $\delta(\pi_1, \pi) < \varepsilon < 1$, we can write

$$\pi_1 := \text{Inf } \left\{ \left(1+\varepsilon_1\right) x_1 + \varepsilon_2 x_2 \mid \left(1+\varepsilon_1^1\right) x_1 + \varepsilon_2^1 x_2 \geq \varepsilon_3^1 \right\},$$

with all the parameters having values in $]-1,1[$. Then, $\pi_1 \in \Pi_b$ if and only if

$$\frac{1+\varepsilon_1}{1+\varepsilon_1^1} = \frac{\varepsilon_2}{\varepsilon_2^1},$$

in which case

$$v_1 = \varepsilon_3^1 \frac{1+\varepsilon_1}{1+\varepsilon_1^1} \quad \text{and} \quad v_1 \geq v - \frac{\varepsilon\,(1+\varepsilon)}{1-\varepsilon}.$$

Since $\lim_{\varepsilon \to 0} \frac{\varepsilon(1+\varepsilon)}{1-\varepsilon} = 0$, ϑ_b comes to be lsc at π.

EXAMPLE 6.8. $\overline{\text{II}}$ and $\overline{\text{Hwp}}$.

$$\begin{aligned} \pi:\ & \text{Inf } x_1 \\ \text{s.t. } & x_1 + 0x_2 \geq 0, \\ & -x_1 + 0x_2 \geq -1. \end{aligned}$$

$x^0 = (\frac{1}{2}, 0)'$ is an SS element, $F = [0,1] \times \mathbb{R}$, and $F^* = \{0\} \times \mathbb{R}$. On this occasion, $\pi_r := \text{Inf}\left\{x_1 + \frac{1}{r}x_2 \mid x_1 + \frac{1}{r}x_2 \geq 0, -x_1 \geq -1\right\}$, and the argument uses exactly the same terms as in the previous example to conclude that $\mathcal{F}^*$ is not usc at π.

In order to check that π is not Hwp, take $\widetilde{\pi}_r := \text{Inf}\ \{x_1 - \frac{1}{r}x_2 \mid x_1 - \frac{1}{2r}x_2 \geq 0,\ -x_1 \geq -1\}$. Note that $\delta\,(\widetilde{\pi}_r, \pi) = \frac{1}{r}$ and $\{\widetilde{\pi}_r\}$ is an ap.s. for π. Moreover, $\widetilde{F}_r^* = \left\{(1, 2r)'\right\}$ and $\widetilde{v}_r = -1$, precluding the lower semicontinuity of ϑ_b at π.

Cell C: F^* unbounded, $F = F^*$, and $\mathcal{F}$ non-lsc at π.

EXAMPLE 6.9. II.

$$\begin{aligned} \pi:\ & \text{Inf } x_1 \\ \text{s.t. } & tx_1 + 0x_2 \geq -1\,, \quad t \in \mathbb{Z}, \\ & x_1 + 0x_2 \geq 0, \\ & -x_1 + 0x_2 \geq 0. \end{aligned}$$

$F = F^* = \{0\} \times \mathbb{R}$. There is no SS element and, so, $\mathcal{F}$ is not lsc at π. If $\delta\,(\pi_1, \pi)$ is finite, $F_1^* \subset F_1 \subset F = F^*$, implying II.

EXAMPLE 6.10. $\overline{\text{II}}$.

$$\begin{aligned} \pi:\ & \text{Inf } x_1 \\ \text{s.t. } & x_1 + 0x_2 \geq 0, \\ & -x_1 + 0x_2 \geq 0. \end{aligned}$$

$F = F^* = \{0\} \times \mathbb{R}$, and we have no SS element. Defining $\pi_r := \text{Inf}\{x_1 + \frac{1}{r}x_2 \mid x_1 + \frac{1}{r}x_2 \geq 0, -x_1 - \frac{1}{r}x_2 \geq 0\}$, one has $\delta\,(\pi_r, \pi) = \frac{1}{r}$, $x^r := (-1, r)' \in F_r^*$, but $x^r \notin W := \{x \in \mathbb{R}^2 \mid x_1 > -1\} \supset F^*$. This yields $\overline{\text{II}}$.

Acknowledgments. The authors wish to thank M. A. Goberna for his many useful suggestions and the anonymous referees for their valuable critical comments.

REFERENCES

[1] B. BROSOWSKI, *Parametric Semi-Infinite Optimization*, Verlag Peter Lang, Frankfurt-Am-Main, Germany, 1982.

[2] B. BROSOWSKI, *Parametric semi-infinite linear programming* I. *Continuity of the feasible set and of the optimal value*, Math. Programming Stud., 21 (1984), pp. 18–42.

[3] M. J. CÁNOVAS, M. A. LÓPEZ, J. PARRA, AND M. I. TODOROV, *Solving Strategies and Well-Posedness in Linear Semi-Infinite Programming*, Dept. of Statistics and Operations Research Working Paper Series, University of Alicante, Alicante, Spain, 1998.

[4] A. L. DONTCHEV AND T. ZOLEZZI, *Well-Posed Optimization Problems*, Springer-Verlag, Berlin, 1993.

[5] T. Fischer, *Contributions to semi-infinite linear optimization*, in Approximation and Optimization in Mathematical Physics, B. Brosowski and E. Martensen, eds., Verlag Peter Lang, Frankfurt-Am-Main, Germany, 1983, pp. 175–199.

[6] M. A. Goberna and M. A. López, *Topological stability of linear semi-infinite inequality systems*, J. Optim. Theory Appl., 89 (1996), pp. 227–236.

[7] M. A. Goberna, M. A. López, and M. I. Todorov, *Stability theory for linear inequality systems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 730–743.

[8] M. A. Goberna, M. A. López, and M. I. Todorov, *Stability theory for linear inequality systems* II: *Upper semicontinuity of the solution set mapping*, SIAM J. Optim., 7 (1997), pp. 1138–1151.

[9] R. Henrion and D. Klatte, *Metric regularity of the feasible set mapping in semi-infinite optimization*, Appl. Math. Optim., 30 (1994), pp. 103–109.

[10] M. A. Jiménez and J. J. Rückmann, *On equivalent stability properties in semi-infinite optimization*, Z. Oper. Res., 41 (1995), pp. 175–190.

[11] H. Th. Jongen, F. Twilt, and G. W. Weber, *Semi-infinite optimization: Structure and stability of the feasible set*, J. Optim. Theory Appl., 72 (1992), pp. 529–552.

[12] H. Th. Jongen, J.-J. Rückmann, and G.-W. Weber, *One-parametric semi-infinite optimization: On the stability of the feasible set*, SIAM J. Optim., 4 (1994), pp. 637–48.

[13] D. Klatte, *Stable local minimizers in semi-infinite optimization: Regularity and second-order conditions*, J. Comput. Appl. Math., 56 (1994), pp. 137–57.

[14] S. M. Robinson, *Stability theory for systems of inequalities. Part* I: *Linear systems*, SIAM J. Numer. Anal., 12 (1975), pp. 754–769.

[15] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer-Verlag, Berlín, 1998.

[16] J. J. Rückmann, *Topological stability of feasible sets in semi-infinite optimization: A tutorial*, Institut für Geometrie und Praktische Mathematik, RWTH Aachen, Germany, 1995.

[17] M. I. Todorov, *Generic existence and uniqueness of the solution set to linear semi-infinite optimization problems*, Numer. Funct. Anal. Optim., 8 (1985/1986), pp. 27–39.

[18] H. Tuy, *Stability property of a system of inequalities*, Math. Oper. Statist. Ser. Opt., 8 (1977), pp. 27–39.

[19] Y. J. Zhu, *Generalizations of some fundamental theorems on linear inequalities*, Acta Math. Sinica, 16 (1966), pp. 25–40.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 99–120

RATES OF CONVERGENCE FOR A CLASS OF GLOBAL STOCHASTIC OPTIMIZATION ALGORITHMS[*]

G. YIN[†]

Abstract. Inspired and motivated by the recent advances in simulated annealing algorithms, this paper analyzes the convergence rates of a class of recursive algorithms for global optimization via Monte Carlo methods. By using perturbed Liapunov function methods, stability results of the algorithms are established. Then the rates of convergence are ascertained by examining the asymptotic properties of suitably scaled estimation error sequences.

Key words. global optimization, Monte Carlo methods, simulated annealing, rate of convergence, weak convergence

AMS subject classifications. 60F05, 62L20, 65K10, 93E23

PII. S1052623497319225

1. Introduction.

An important task in control, optimization, and related fields is to locate the global minimum of $f(\cdot) : \mathbb{R}^r \mapsto [0, \infty)$, a smooth function, which has multiple local minima. The following situation is of interest: we cannot calculate the gradient of $f(\cdot)$ explicitly and only noise-corrupted gradient estimates or measurements, "$\nabla f(x)$+noise," are available. Consequently, standard deterministic algorithms are not able to produce desirable results. One needs to rely on stochastic approximation algorithms. Nevertheless, a stochastic approximation algorithm of the form

$$X_{n+1} = X_n - a_n(\nabla f(X_n) + \xi_n) \tag{1.1}$$

may yield convergence to a local minimum. Let S_l denote the collection of all the minima of $f(x)$. Under broad conditions (see, for example, Kushner and Clark [12], Benveniste, Métivier, and Priouret [1], or the more up-to-date treatment of Kushner and Yin [17]), $X_n \to S_l$ with probability 1 (w.p.1). But very often the iterates will be trapped at a local minimum and will miss the global one.

In this work we examine the algorithms

$$X_{n+1} = X_n - \frac{A}{n^\gamma}(\nabla f(X_n) + \xi_n) + \frac{B}{n^{\gamma/2}\sqrt{\ln[n^{1-\gamma} + A_0]}} W_n \tag{1.2}$$

for $0 < \gamma < 1$, and

$$X_{n+1} = X_n - \frac{A}{n}(\nabla f(X_n) + \xi_n) + \frac{B}{\sqrt{n \ln \ln(n + A_0)}} W_n, \tag{1.3}$$

where A, A_0, and B are some positive constants. These algorithms are Monte Carlo versions of the "simulated annealing" procedure. Notice that there are two noise sequences, of which $\{\xi_n\}$ is a sequence of measurement noise and $\{W_n\}$ is a sequence

[*] Received by the editors March 31, 1997; accepted for publication (in revised form) August 18, 1998; published electronically October 20, 1999. This research was supported in part by National Science Foundation grant DMS-9529738 and by Wayne State University.

http://www.siam.org/journals/siopt/10-1/31922.html

[†] Department of Mathematics, Wayne State University, Detroit, MI 48202 (gyin@math.wayne.edu).

of added random perturbations. Following the basic premise of the annealing scheme, the purpose of $\{W_n\}$ is to give the iterates enough excitation so that they will not be trapped at one of the local minima. Corresponding to the two noise processes there are two sequences of step sizes as well: the sequence $a_n = A/n^\gamma$ (respectively, $a_n = A/n$) and the sequence $b_n = B/(n^{\gamma/2}\sqrt{\ln(n^{1-\gamma}+A_0)})$ (respectively, $b_n = B/(\sqrt{n \ln\ln(n+A_0)})$).

Our work is inspired by the algorithms developed recently by Geman and Hwang [3], Kushner [11], and Gelfand and Mitter [6]. The focus of this paper is the rates of convergence of global stochastic optimization algorithms.

To overcome the difficulties noted above with regard to stochastic approximation algorithms, much effort has been made to search for suitable procedures for global optimization. In the 1980s, one such global optimization method, simulated annealing, started attracting the attention of researchers and practitioners. In [9], Kirkpatrick, Gelatt, and Vecchi proposed a solution method by running the Metropolis algorithm [19] while gradually lowering the temperature.

To some extent, these approaches can be thought of as dynamical systems perturbed by a small noise (see Freidlin and Wentzell [5]). Due to the nature of the algorithms, generally we cannot expect w.p.1 convergence results. In [6], the convergence, in probability, of (1.3) was proved via the use of properties of diffusion processes (see Chiang, Hwang, and Sheu [2]). It provides sufficient conditions guaranteeing the convergence of the algorithms and relates the discrete iterates to the stochastic differential equation

$$(1.4) \qquad dx(t) = -\nabla f(x(t))dt + \frac{C}{\sqrt{\ln t}} d\widehat{w}(t), \quad x(t^0) = x_0 \quad \text{for some } t^0 > 1,$$

where $\widehat{w}(\cdot)$ is a Brownian motion and $C = B/\sqrt{A}$ is sufficiently large. Note that we use $t^0 > 1$ as the initial time due to the presence of $1/\sqrt{\ln t}$. The requirement $B/\sqrt{A} > C_0$ (with C_0 sufficiently large) comes from the work [2], where the critical constant C_0 is given.

It will be shown in this work that the rates of convergence mainly depend on the step size of the perturbation terms and are determined by the random perturbation, not the observation noise. By a suitable scaling, we show that the interpolated processes of the normalized estimation errors converge to a diffusion process. The limit stochastic differential equations are the same for normalized sequences resulting from both (1.2) and (1.3). The stationary covariance of the diffusions for these cases are the same. The scaling factors for the two algorithms are different, however.

The rest of the paper is arranged as follows. We present the main assumptions and conditions in section 2. Section 3 is devoted to bounds and moment estimates of a suitably scaled sequence of the estimation errors. We first prove an auxiliary estimate, and then under appropriate conditions, we demonstrate that for algorithms (1.2) and (1.3), the corresponding scalings are $\sqrt{\ln(n^{1-\gamma}+A_0)}$ and $\sqrt{\ln\ln(n+A_0)}$, respectively, which indicate that (1.2) performs better than (1.3). In section 4, we obtain a local limit result by showing that the interpolated sequences converge to stochastic differential equations. Denoting the global minimizer of $f(\cdot)$ by x^*, it follows that for algorithm (1.2), $\sqrt{\ln(n^{1-\gamma}+A_0)}(X_n - x^*)$, and for algorithm (1.3), $\sqrt{\ln\ln(n+A_0)}(X_n - x^*)$, are asymptotically normal. These scaling factors and the corresponding asymptotic covariance matrices give us the desired convergence-rate results. Finally, we make a number of further remarks in section 5 and close the paper with an appendix containing the proofs of two lemmas.

2. Formulation and conditions. Since the main theme of this work is on the rates of convergence, throughout the paper we will assume the convergence of algorithms (1.2) and (1.3). To carry out the analysis, we make the following assumptions:

A1. $f : \mathbb{R}^r \mapsto [0, \infty)$ is a three times continuously differentiable function such that $\min f(x) = 0$; the set $\mathsf{M} = \{x \in \mathbb{R}^r; \nabla f(x) = 0\}$ consists of finitely many isolated points; and there is an x^* that is the global minimum of $f(\cdot)$. Without loss of generality, assume $x^* = 0$ henceforth.

A2. For both algorithms (1.2) and (1.3), $X_n \xrightarrow{n} 0$ in probability.

A3. The noise sequences satisfy:

a. $\{W_n\}$ is a sequence of independent and identically distributed (i.i.d.) ($\mathbb{R}^r$-valued) random variables with $EW_n = 0$ and $EW_n W_n' = I$, where z' denotes the transpose of z for any $z \in \mathbb{R}^{r\times l}$ for some $l \geq 1$. $\{W_n\}$ is independent of $\{\xi_n\}$ and the initial estimate X_1.

b. $E\xi_n = 0$ for each n, and $\sup_n E|\xi_n|^2 < \infty$.

c. There exists a sequence $\{\rho(n)\}$ of nonnegative real numbers such that for $j \geq n$, $E^{\frac{1}{2}}|E_n\xi_j - E\xi_j|^2 \leq \rho(j-n)$ and $\sum_{k=0}^{\infty} \rho(k) < \infty$, where E_n denotes the conditioning on $\mathcal{F}_n$, the σ-algebra generated by $\{X_1, \xi_j, W_j;\ j < n\}$.

A4. There is a twice continuously differentiable function $V(\cdot) : \mathbb{R}^r \mapsto [0, \infty)$ such that

a. $V(x) \to \infty$ as $|x| \to \infty$;

b. there is a $\lambda > 0$ such that $V_x'(x)\nabla f(x) \geq \lambda V(x)$ for all $x \notin \mathsf{M}$, where $V_x(\cdot)$ denotes the first-order partial derivative of $V(\cdot)$ with respect to x and, similarly, $V_{xx}(\cdot)$ denotes the second-order derivative;

c. for each $0 \leq s \leq 1$, $E_n|V_x(x + s\delta(x,n))\nabla f(x + s\delta(x,n))| \leq K(1 + V(x))$, where $\delta(x,n) = -a_n(\nabla f(x) + \xi_n) + b_n W_n$, $V_{xx}(\cdot)$ is bounded, and $|\nabla f(x)|^2 \leq K(1 + V(x))$.

Remarks. We make the following comments concerning the conditions:

1. The assumption that $f(\cdot)$ is a nonnegative real-valued function is not a restriction. The condition $\min f(x) = 0$ is purely for convenience. We can always translate the axis; i.e., if $\min f(x) = c_0 \neq 0$, we can define $\widehat{f}(x) = f(x) - c_0$. Then $\widehat{f}(\cdot)$ is nonnegative, with $\min \widehat{f}(x) = 0$.

2. The function $V(\cdot)$ in A4 is simply a Liapunov function; we assume its existence together with some conditions. Such a function is frequently used in analyzing stochastic recursive algorithms. As an alternative, $f(\cdot)$ itself may be considered as a Liapunov function. In this case, additional conditions on $f(\cdot)$, similar to (A1) and (A2) in [6], are needed. These conditions were originally used in [2].

3. In part a of A3, it is assumed that $\{W_n\}$ is an i.i.d. sequence. We can deal with a more complex and correlated sequence. Since the algorithm is a Monte Carlo–based approach, the sequence $\{W_n\}$ is at our disposal (we can choose it in accordance with our needs). Thus the i.i.d. assumption appears to be sufficient.

4. In part b of A3, we require only that the sequence have zero mean and that the second moment of the noise be finite. In view of A3, part c, for each $m \geq 1$,

$$E\left|\sum_{j=m}^{n+m} E_m\xi_j\right| \leq \sum_{j=m}^{n+m} E^{\frac{1}{2}}\left|E_m\xi_j - E\xi_j\right|^2 \leq \sum_{j=m}^{\infty} \rho(j-m) < \infty. \tag{2.1}$$

Thus, as $n \to \infty$,

$$E\left|\frac{1}{n}\sum_{j=m}^{n+m} E_m\xi_j\right| = O\left(\frac{1}{n}\right) \to 0,$$
$$\frac{1}{n}\sum_{j=m}^{n+m} E_m\xi_j \to 0 \text{ in probability,}$$

and the sequence verifies a law of large numbers type of condition in the sense of convergence in probability. Such an averaging condition is satisfied by a large class of stochastic processes, including i.i.d. zero mean sequences, martingale difference sequences, certain autoregressive moving average (ARMA) processes, mixing processes, and functions of mixing processes. For example, if $\{\xi_n\}$ is a stationary φ-mixing sequence, then it is strongly ergodic. The averaging condition is satisfied. This kind of condition and similar ones have been used extensively in Kushner [13] (see also [17] and the references therein).

5. As for part c of A3, if the noise process is stationary φ-mixing with mixing rate $\tilde{\phi}(n)$ such that $\sum_i \tilde{\phi}(i) < \infty$, using the well-known mixing inequality (see Ethier and Kurtz [4, p. 347] and [13, Lemma 4, p. 82]), this condition is easily verified.

3. Error bounds.

The purpose of this section is to establish error bounds of scaled sequences of estimation errors for algorithms (1.2) and (1.3). To do so, we first prove an auxiliary result for the bounds on $\{EV(X_n)\}$ and then modify the argument to establish the desired results.

3.1. Bounds on $EV(X_n)$.

THEOREM 3.1. *Suppose* A1–A4 *are satisfied. Then* $EV(X_n) = O(1)$ *for* $\{X_n\}$ *defined by* (1.2) *and* (1.3).

Proof. We will only work out the details for (1.2) because the argument for (1.3) is essentially the same. We use a perturbed Liapunov function method to prove the assertion. The idea of the perturbed Liapunov function method is to add a small perturbation to the Liapunov function $V(\cdot)$ to cancel the unwanted terms in the process of averaging. The main techniques were developed by Kushner; for many applications in approximation of various random processes, see [13] and the references therein.

By virtue of A3, $E_nW_n = 0$. Using A4,

$$\begin{aligned}
&E_nV(X_{n+1}) - V(X_n) \\
&\quad = -\frac{A}{n^\gamma}V_x'(X_n)\nabla f(X_n) - \frac{A}{n^\gamma}V_x'(X_n)E_n\xi_n \\
&\quad\quad + \frac{A^2}{2n^{2\gamma}}E_n(\nabla f(X_n) + \xi_n)'\int_0^1 V_{xx}(X_n + s(X_{n+1} - X_n))ds(\nabla f(X_n) + \xi_n) \\
&\quad\quad + \frac{B^2}{2n^\gamma \ln(n^{1-\gamma} + A_0)}E_nW_n'\int_0^1 V_{xx}(X_n + s(X_{n+1} - X_n))dsW_n.
\end{aligned} \tag{3.1}$$

Define

$$V_1(x,n) = -\sum_{j=n}^{\infty}\frac{A}{j^\gamma}E_nV_x'(x)\xi_j. \tag{3.2}$$

By virtue of (2.1) and A4,

$$E|V_1(x,n)| \leq \frac{K}{n^\gamma}(1+V(x))E\left|\sum_{j=n}^{\infty} E_n\xi_j\right| \leq \frac{K}{n^\gamma}(1+V(x)). \tag{3.3}$$

(Here and hereafter, K denotes a generic positive constant. Its value may be different for different uses. The expressions $K+K=K$ and $KK=K$ are understood in an appropriate sense.) Moreover,

$$\begin{aligned} &E_nV_1(X_{n+1},n+1) - V_1(X_n,n) = \frac{A}{n^\gamma}V'_x(X_n)E_n\xi_n \\ &+\frac{1}{2}\sum_{j=n+1}^{\infty}\frac{A^2}{j^\gamma n^\gamma}E_n\left(\nabla f(X_n)+\xi_n\right)'\int_0^1 V_{xx}(X_n+s(X_{n+1}-X_n))ds\xi_j. \end{aligned} \tag{3.4}$$

Define

$$\widehat{V}(n) = V(x) + V_1(x,n).$$

In view of (3.1) and (3.4),

$$E_n\widehat{V}(n+1) - \widehat{V}(n) = -\frac{A}{n^\gamma}V'_x(X_n)\nabla f(X_n) + \frac{\mu_n}{2},$$

where

$$\begin{aligned} \mu_n = &\frac{A^2}{n^{2\gamma}}E_n(\nabla f(X_n)+\xi_n)'\int_0^1 V_{xx}(X_n+s(X_{n+1}-X_n))ds(\nabla f(X_n)+\xi_n) \\ &+\frac{B^2}{n^\gamma\ln(n^{1-\gamma}+A_0)}E_nW'_n\int_0^1 V_{xx}(X_n+s(X_{n+1}-X_n))dsW_n \\ &+\sum_{j=n+1}^{\infty}\frac{A^2}{j^\gamma n^\gamma}E_n\left(\nabla f(X_n)+\xi_n\right)'\int_0^1 V_{xx}(X_n+s(X_{n+1}-X_n))ds\xi_j. \end{aligned} \tag{3.5}$$

Note that

$$\begin{aligned} E\left|\sum_{j=n+1}^{\infty}E_n\xi'_n\xi_j\right| &\leq \sum_{j=n+1}^{\infty}E|E_n\xi'_n\xi_j| = \sum_{j=n+1}^{\infty}E|E_n\xi'_n(E_{n+1}\xi_j)| \\ &\leq \sum_{j=n+1}^{\infty}E^{\frac{1}{2}}|\xi_n|^2E^{\frac{1}{2}}|E_{n+1}\xi_j - E\xi_j|^2 \\ &\leq \left(\sup_n E^{\frac{1}{2}}|\xi_n|^2\right)\sum_{j=n+1}^{\infty}\rho(j-(n+1)) \leq K. \end{aligned} \tag{3.6}$$

Owing to A3 and A4, there is a sequence of nonnegative real-valued random variables $\{\zeta_n\}$ satisfying $\sup_n E\zeta_n < \infty$, and

$$\begin{aligned} &E\left|\sum_{j=n+1}^{\infty}\frac{A^2}{j^\gamma n^\gamma}E_n\left(\nabla f(X_n)+\xi_n\right)'\int_0^1 V_{xx}(X_n+s(X_{n+1}-X_n))ds\xi_j\right| \\ &= O\left(\frac{1}{n^{2\gamma}}\right)(1+EV(X_n)+E\zeta_n) \\ &= O\left(\frac{1}{n^{2\gamma}}\right)(1+EV(X_n)). \end{aligned} \tag{3.7}$$

Moreover, in view of A1 and A4, there is a constant vector $h \in \mathbb{R}^r$ such that

$$\begin{aligned}-V_x'(X_n)\nabla f(X_n) &= -V_x'(X_n)\nabla f(X_n) I_{\{X_n \notin \mathsf{M}\}} - V_x'(X_n)\nabla f(X_n) I_{\{X_n \in \mathsf{M}\}} \\ &\le -\lambda V(X_n) I_{\{X_n \notin \mathsf{M}\}} - V_x'\left(X_n + \frac{A}{n^\gamma}h\right)\nabla f\left(X_n + \frac{A}{n^\gamma}h\right) I_{\{X_n \in \mathsf{M}\}} \\ &\qquad + V_x'\left(X_n + \frac{A}{n^\gamma}\right)\nabla f\left(X_n + \frac{A}{n^\gamma}h\right) I_{\{X_n \in \mathsf{M}\}}.\end{aligned}$$

Note that

$$\begin{aligned}&-V_x'\left(X_n + \frac{A}{n^\gamma}\right)\nabla f\left(X_n + \frac{A}{n^\gamma}\right) I_{\{X_n \in \mathsf{M}\}} \\ &\quad \le -\lambda V\left(X_n + \frac{A}{n^\gamma}h\right) I_{\{X_n \in \mathsf{M}\}} \\ &\quad \le -\lambda V(X_n) I_{\{X_n \in \mathsf{M}\}} + \left|\frac{A}{n^\gamma}h \int_0^1 V_x'(X_n + s(A/n^\gamma)h) ds I_{\{X_n \in \mathsf{M}\}}\right| \\ &\quad \le -\lambda V(X_n) I_{\{X_n \in \mathsf{M}\}} + O\left(\frac{1}{n^\gamma}\right)(1 + V(X_n)),\end{aligned}$$

and

$$\begin{aligned}&E_n\left|V_x'\left(X_n + \frac{A}{n^\gamma}h\right)\nabla f\left(X_n + \frac{A}{n^\gamma}h\right) I_{\{X_n \in \mathsf{M}\}}\right| \\ &\quad = E_n\left|\frac{A}{n^\gamma}h\left(\int_0^1 V_x'(X_n + s(A/n^\gamma)h)\nabla f(X_n + (A/n^\gamma)h) ds\right)_x I_{\{X_n \in \mathsf{M}\}}\right| \\ &\quad \le \frac{K}{n^\gamma}(1 + V(X_n)).\end{aligned}$$

It follows that

$$\begin{aligned}-E_n V_x(X_n)\nabla f(X_n) &\le -\lambda V(X_n) I_{\{X_n \notin \mathsf{M}\}} - \lambda V(X_n) I_{\{X_n \in \mathsf{M}\}} \\ &\qquad + O\left(\frac{1}{n^\gamma}\right)(1 + V(X_n)).\end{aligned}$$

Taking the expectation in (3.5) and using the estimates above, it is easy to verify that the first and second terms in $E\mu_n$ are bounded above by $Kn^{-2\gamma}(1 + EV(X_n))$ and $K(n^\gamma \ln(n^{1-\gamma} + A_0))^{-1}$, respectively. Putting the estimates above together, we arrive at

$$\begin{aligned}&E\widehat{V}(n+1) \\ &\quad \le \left(1 - \frac{A\lambda}{n^\gamma}\right) EV(X_n) + \left(O\left(\frac{1}{n^\gamma \ln(n^{1-\gamma} + A_0)}\right) + O\left(\frac{1}{n^{2\gamma}}\right)\right)(1 + EV(X_n)) \\ &\quad \le \left(1 - \frac{A\lambda}{n^\gamma}\right) E\widehat{V}(n) + \left(O\left(\frac{1}{n^\gamma \ln(n^{1-\gamma} + A_0)}\right) + O\left(\frac{1}{n^{2\gamma}}\right)\right)(1 + E\widehat{V}(n)).\end{aligned} \tag{3.8}$$

The second inequality in (3.8) follows from the bound on $V_1(\cdot)$. Iterating on (3.8) yields

$$E\widehat{V}(n+1) \le K_n + K\sum_{j=1}^n \left(\frac{1}{j^\gamma \ln(j^{1-\gamma} + A_0)} + \frac{1}{j^{2\gamma}}\right) B_{nj} E\widehat{V}(j), \tag{3.9}$$

where

$$B_{nj} = \begin{cases} \displaystyle\prod_{l=j+1}^{n} \left(1 - \frac{A\lambda}{l^{\gamma}}\right), & j < n, \\ 1, & j = n, \end{cases}$$

and

$$K_n = B_{n,0} E\widehat{V}(1) + K \sum_{j=1}^{n} \frac{1}{j^{\gamma}} \left(\frac{1}{j^{\gamma}} + \frac{1}{\ln(j^{1-\gamma} + A_0)} \right) B_{nj} < \infty.$$

An application of Gronwall's inequality implies

$$E\widehat{V}(n+1) \le K_n \exp \left(K \sum_{j=1}^{n} \frac{1}{j^{\gamma}} \left(\frac{1}{j^{\gamma}} + \frac{1}{\ln(j^{1-\gamma} + A_0)} \right) B_{nj} \right) < \infty.$$

Owing to (3.3), we also have $EV(X_{n+1}) \le K$. The desired estimate for $\{EV(X_n)\}$ then follows. □

3.2. Normalized sequences. Choosing appropriate scaling factors is crucial to the study of the convergence speed. In this section, we derive the error bounds of suitably scaled sequences, while the next section is on local limit results.

Define

$$v_n = \begin{cases} \sqrt{\ln(n^{1-\gamma} + A_0)} X_n & \text{for } 0 < \gamma < 1, \\ \sqrt{\ln\ln(n + A_0)} X_n & \text{for } \gamma = 1. \end{cases} \tag{3.10}$$

To derive the desired tightness, use the Liapunov function $V(\cdot)$ again. We first obtain the following result.

THEOREM 3.2. *Under the conditions of Theorem* 3.1,

$$\sup_n [\ln((n+1)^{1-\gamma} + A_0)] EV(X_{n+1}) = O(1)$$

with $0 < \gamma < 1$ *for algorithm* (1.2), *and*

$$\sup_n [\ln\ln((n+1) + A_0)] EV(X_{n+1}) = O(1)$$

for algorithm (1.3).

Proof. The proof is very similar to the previous result, and we point out only the differences. Again, we examine algorithm (1.2) only. It is easily seen that

$$\sup_n \frac{\ln((n+1)^{1-\gamma} + A_0)}{\ln(n^{1-\gamma} + A_0)} \le K.$$

Now (3.9) is replaced by

$$\begin{aligned} &[\ln((n+1)^{1-\gamma} + A_0)] E\widehat{V}(n+1) \\ &\quad \le \frac{\ln((n+1)^{1-\gamma} + A_0)}{\ln(n^{1-\gamma} + A_0)} \ln(n^{1-\gamma} + A_0) \\ &\quad \times \left(K_n + K \sum_{j=1}^{n} \left(\frac{1}{j^{\gamma} \ln(j^{1-\gamma} + A_0)} + \frac{1}{j^{2\gamma}} \right) B_{nj} E\widehat{V}(j) \right) \\ &\quad \le K \ln(n^{1-\gamma} + A_0) \left(K_n + K \sum_{j=1}^{n} \left(\frac{1}{j^{\gamma} \ln(j^{1-\gamma} + A_0)} + \frac{1}{j^{2\gamma}} \right) B_{nj} E\widehat{V}(j) \right). \end{aligned} \tag{3.11}$$

It thus suffices to show that the third line of (3.11) is bounded. To proceed, we state a lemma. Its proof is in the appendix.

LEMMA 3.3. *The following estimates hold:*

$$
\begin{aligned}
&\sum_{j=k}^{m} \frac{1}{j^{\gamma}} B_{nj} < \infty \qquad \text{for } 1 \le k \le m \le n,\\
&\ln(n^{1-\gamma}+A_0) \sum_{j=1}^{n} \frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)} B_{nj} < \infty,\\
&\ln(n^{1-\gamma}+A_0) \sum_{j=1}^{n} \frac{1}{j^{2\gamma}} B_{nj} < \infty.
\end{aligned}
$$

In view of the definition of K_n and the lemma above,

$$
\begin{aligned}
\ln(n^{1-\gamma}+A_0)K_n &= \ln(n^{1-\gamma}+A_0)B_{n,0}E\widehat{V}(1)\\
&\quad + \ln(n^{1-\gamma}+A_0)K\sum_{j=1}^{n}\frac{1}{j^{\gamma}}\left(\frac{1}{j^{\gamma}}+\frac{1}{\ln(j^{1-\gamma}+A_0)}\right)B_{nj} \le K < \infty.
\end{aligned}
\tag{3.12}
$$

Define

$$
\widetilde{V}(n) = \ln(n^{1-\gamma}+A_0)\widehat{V}(n).
$$

Then

$$
\begin{aligned}
&\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\left(\frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)}+\frac{1}{j^{2\gamma}}\right)B_{nj}E\widehat{V}(j)\\
&\quad = \ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\left(\frac{1}{j^{\gamma}[\ln(j^{1-\gamma}+A_0)]^2}+\frac{1}{j^{2\gamma}\ln(j^{1-\gamma}+A_0)}\right)B_{nj}E\widetilde{V}(j).
\end{aligned}
\tag{3.13}
$$

Using (3.11)–(3.13), Lemma 3.3, and the familiar Gronwall's inequality, we arrive at

$$
\begin{aligned}
&E\widetilde{V}(n+1)\\
&\le K\exp\left(\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\left(\frac{1}{j^{\gamma}[\ln(j^{1-\gamma}+A_0)]^2}+\frac{1}{j^{2\gamma}\ln(j^{1-\gamma}+A_0)}\right)B_{nj}\right) < \infty.
\end{aligned}
$$

This, together with the bound on $V_1(\cdot)$, then yields the desired estimate. □

COROLLARY 3.4. *If the Liapunov function $V(\cdot)$ is locally quadratic, i.e., $V(x) = x'Qx + o(|x|^2)$, where Q is a symmetric positive definite matrix, then $\{v_n\}$ defined for algorithm* (1.2) (*respectively, algorithm* (1.3)) *is tight.*

Proof. The proof is very similar to the verification of the last part of Theorem 1 in [15], which mainly uses the local quadratic structure of the Liapunov function. We thus omit the details. □

4. Diffusion limits. This section deals with a local limit theorem for the interpolated sequence of v_n. The main idea is to linearize the difference equation around the global minimum 0 and to carry out the limit process for the scaled sequences. As before, detailed derivation is given for the sequence defined by (1.2) only.

Note that

$$\nabla f(x) = Hx + \frac{1}{2}x'\left(\int_0^1 f_{xxx}(sx)ds\right)x,$$

where $H = f_{xx}(0)$ is the Hessian of $f(\cdot)$ at 0. Linearizing (1.2) yields that

$$X_{n+1} = X_n - \frac{A}{n^\gamma}(HX_n + \xi_n) - \frac{A}{2n^\gamma}X_n' \int_0^1 f_{xxx}(sX_n)dsX_n + \frac{B}{n^{\frac{\gamma}{2}}\sqrt{\ln(n^{1-\gamma}+A_0)}}W_n.$$

For future use, note that

$$\left(\frac{\ln((n+1)^{1-\gamma}+A_0)}{\ln(n^{1-\gamma}+A_0)}\right)^{1/2} = 1 + O\left(\frac{1}{n^{2-2\gamma}\ln(n^{1-\gamma}+A_0)}\right). \tag{4.1}$$

By virtue of the definition of v_n (see (3.10)) and using (4.1), we obtain

$$\begin{aligned} v_{n+1} &= v_n - \frac{AH}{n^\gamma}v_n - \frac{A}{n^\gamma}\widetilde{\xi}_n + \frac{B}{n^{\gamma/2}}W_n + g_n + \widetilde{g}_n + \widehat{g}_n, \text{ where} \\ g_n &= O\left(\frac{1}{n^{2-\gamma}\ln(n^{1-\gamma}+A_0)}\right)v_n + O\left(\frac{\left|\int_0^1 f_{xxx}(sX_n)ds\right|}{n^\gamma\sqrt{\ln(n^{1-\gamma}+A_0)}}|v_n|^2\right), \\ \widetilde{g}_n &= O\left(\frac{1}{n^{2\gamma}\ln(n^{1-\gamma}+A_0)}\right)\widetilde{\xi}_n \text{ with} \\ \widetilde{\xi}_n &= \sqrt{\ln(n^{1-\gamma}+A_0)}\xi_n \text{ and} \\ \widehat{g}_n &= O\left(\frac{1}{n^{2-\frac{\gamma}{2}}[\ln(n^{1-\gamma}+A_0)]^{\frac{3}{2}}}\right)W_n. \end{aligned} \tag{4.2}$$

Define

$$t_n = \sum_{i=1}^n \frac{A}{i^\gamma}$$

and $m(t) = \max\{n;\ t_n \le t\}$.

To proceed, we define the piecewise constant interpolation of v_n by

$$\begin{aligned} v^0(t) &= v_n \quad \text{for } t \in [t_n, t_{n+1}), \\ v^n(t) &= v^0(t + t_n). \end{aligned}$$

THEOREM 4.1. *Suppose the conditions of Corollary* 3.4 *are satisfied. Then* $\{v^n(\cdot)\}$ *is tight in* $D^r[0,\infty)$. *Any weakly convergent subsequence of* $\{v^n(\cdot)\}$ *has a limit satisfying the following stochastic differential equation:*

$$dv = -Hvdt + Cdw, \tag{4.3}$$

where $C = B/\sqrt{A}$ *and* $w(\cdot)$ *is an* r*-dimensional standard Brownian motion.*

Remark. In view of A3, part c, we can derive an appropriate moment bound for the scaled sequence of measurement noise. Very often, we also have the situation in which

$$\sum_{i=n}^{m(t_n+t)} \frac{A}{i^{\gamma/2}}\xi_i \text{ converges weakly to } \widetilde{w}(t),$$

a Brownian motion process. Moreover, we note that since (4.3) is linear, it has a unique solution for each initial condition.

Proof. The proof uses the methods of direct averaging [13]. First note that $v^n(0)$ is tight owing to Corollary 3.4. To proceed, we divide the proof into several steps.

Step 1. Use an N-truncation device: since it is not known a priori that $\{v_n\}$ is bounded, we first use a truncation device (see [13, p. 43]). The idea is to apply the truncations to X_n and v_n, but not to the noise processes. For each $0 < N < \infty$, consider the discrete system

$$v_{n+1}^N = v_n^N - \frac{AH}{n^\gamma} v_n^N - \frac{A}{n^\gamma}\widetilde{\xi}_n + \frac{B}{n^{\gamma/2}} + g_n^N + \widetilde{g}_n + \widehat{g}_n, \tag{4.4}$$

where

$$g_n^N = O\left(\frac{1}{n^{2-\gamma}\ln(n^{1-\gamma}+A_0)}\right) v_n^N + O\left(\frac{1}{n^\gamma\sqrt{\ln(n^{1-\gamma}+A_0)}}|v_n^N|^2\right) q_N(v_n^N),$$

$$q_N(v) = \begin{cases} 1, & v \in S_N, \\ 0, & v \in \mathbb{R}^r - S_{N+1}, \\ \text{smooth}, & \text{otherwise}, \end{cases}$$

and S_N denotes the sphere with radius N, i.e., $S_N = \{v;\ |v| \le N\}$. In the above, we have noticed that $\{X_n^N\}$ is bounded; so is $\int_0^1 f_{xxx}(sX_n^N)ds$. Define $v^{n,N}(\cdot)$ to be the corresponding piecewise constant interpolation of v_n^N. Then $v^{n,N}(\cdot)$ is the N-truncation of $v^n(\cdot)$, i.e., $v^{n,N}(t) = v^n(t)$, up until the first exit from S_N, and

$$\lim_{\kappa\to\infty} \limsup_{n\to\infty} P\left\{\sup_{t\le T} |v^{n,N}(t)| \ge \kappa\right\} = 0 \quad \text{for each } T < \infty,\ N < \infty.$$

The function $q_N(\cdot)$ is smooth and is termed a truncation function (see [13, p. 43]). As a result,

$$\begin{aligned} & v^{n,N}(t+s) - v^{n,N}(t) \\ & = -AH \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{v_i^N}{i^\gamma} + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A^{1/2}C}{i^{\gamma/2}} W_i \\ & \quad - \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma}\widetilde{\xi}_i + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} g_i^N + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \widetilde{g}_i + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \widehat{g}_i. \end{aligned} \tag{4.5}$$

Step 2. Derive the tightness of the process $\{v^{n,N}(\cdot)\}$. We apply Kurtz's tightness criterion (see [13, p. 47]) in what follows. For each $T < \infty$, and each $t \le T$, $N < \infty$, $\delta > 0$, use $E^{\mathcal{F}_t}$ and E_k to denote the conditioning on the σ-algebra generated by $\sigma\{v^{n,N}(\tau),\ \tau \le t\}$ and $\sigma\{X_1^N, \xi_j, W_j,\ j < k\}$, respectively. Using the piecewise constant interpolation, we have for some $K > 0$,

$$E^{\mathcal{F}_t}\left|v^{n,N}(t+\delta) - v^{n,N}(t)\right|^2 \le K\sum_{i=1}^{6} \Gamma_i^n, \tag{4.6}$$

where

$$\begin{aligned}
\Gamma_1^n &= E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{AH}{i^\gamma} v_i^N\right|^2,\\
\Gamma_2^n &= E^{\mathcal{F}_t}\left|A\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{\widetilde{\xi}_i}{i^\gamma}\right|^2,\\
\Gamma_3^n &= E^{\mathcal{F}_t}\left|B\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{W_i}{i^{\gamma/2}}\right|^2,\\
\Gamma_4^n &= E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} g_i^N\right|^2,\\
\Gamma_5^n &= E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \widetilde{g}_i\right|^2,\\
\Gamma_6^n &= E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \widehat{g}_i\right|^2.
\end{aligned}$$

By virtue of the N-truncation and hence the boundedness of v_i^N,

$$\Gamma_1^n \le K\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{1}{i^\gamma}\right|^2 \le \widetilde{\Gamma}_1^n(\delta),$$

where we wish to prove that $\widetilde{\Gamma}_1^n(\delta)$ is some upper bound that converges to 0 in expectation. Owing to the definition of $m(\cdot)$,

$$\begin{aligned}
\sum_{i=1}^{m(t_n+t+\delta)} \frac{A}{i^\gamma} &\le t_n+t+\delta\\
&\le \sum_{i=1}^{m(t_n+t)-1} \frac{A}{i^\gamma} + \frac{A}{(m(t_n+t))^\gamma} + \frac{A}{(m(t_n+t)+1)^\gamma} + \delta.
\end{aligned}$$

As $n \to \infty$, $(m(t_n+t))^\gamma \to \infty$. Hence

$$\begin{aligned}
&\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)} \frac{A}{i^\gamma} \le \delta + \frac{A}{(m(t_n+t))^\gamma} + \frac{A}{(m(t_n+t)+1)^\gamma} \text{ and}\\
&\limsup_{n\to\infty} \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma} \le \limsup_{n\to\infty} \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)} \frac{A}{i^\gamma} \le \delta.
\end{aligned} \tag{4.7}$$

In view of (4.7),

$$\lim_{\delta\to 0}\limsup_{n\to\infty} E\widetilde{\Gamma}_1^n(\delta) = 0.$$

Likewise,

$$\Gamma_4^n \le \widetilde{\Gamma}_4^n(\delta) \quad \text{ such that } \lim_{\delta\to 0}\limsup_{n\to\infty} E\widetilde{\Gamma}_4^n(\delta) = 0.$$

Owing to the fact that $\{W_n\}$ is a sequence of i.i.d. random variables with zero mean and $EW_nW_n' = I$,

$$\begin{aligned}\Gamma_3^n &= \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{B}{i^\gamma} E^{\mathcal{F}_t}|W_i|^2 \\ &= \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{B}{i^\gamma} \operatorname{tr}(EW_iW_i') \\ &= Br \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{1}{i^\gamma} \le \widetilde{\Gamma}_3^n(\delta),\end{aligned}$$

so (in view of (4.7)), we also have

$$\lim_{\delta\to 0} \limsup_{n\to\infty} E\widetilde{\Gamma}_3^n(\delta) = 0.$$

Similarly,

$$\Gamma_6^n \le \widetilde{\Gamma}_6^n(\delta) \quad \text{and} \quad \lim_{\delta\to 0} \limsup_{n\to\infty} E\widetilde{\Gamma}_6^n(\delta) = 0.$$

Proceeding to the term Γ_2^n,

$$\begin{aligned}&E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma}\widetilde{\xi}_i\right|^2 \\ &= \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \sum_{j=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma}\frac{A}{j^\gamma}\sqrt{\ln(i^{1-\gamma}+A_0)}\sqrt{\ln(j^{1-\gamma}+A_0)}E^{\mathcal{F}_t}\xi_i'\xi_j \\ &\le K \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma} \sum_{j\ge i} \frac{\sqrt{\ln(i^{1-\gamma}+A_0)}}{j^{\frac{\gamma}{2}}} \frac{\sqrt{\ln(j^{1-\gamma}+A_0)}}{j^{\frac{\gamma}{2}}} \left|E^{\mathcal{F}_t}\xi_i'\xi_j\right|.\end{aligned}$$

Notice that for $j \ge i$,

$$\frac{\sqrt{\ln(i^{1-\gamma}+A_0)}}{j^{\frac{\gamma}{2}}} \le 1 \quad \text{and} \quad \frac{\sqrt{\ln(j^{1-\gamma}+A_0)}}{j^{\frac{\gamma}{2}}} \le 1.$$

Furthermore, for $j \ge i$,

$$\left|E^{\mathcal{F}_t}\xi_i'\xi_j\right| \left|E^{\mathcal{F}_t}\xi_i'(E_i\xi_j - E\xi_j)\right| \le \left(E^{\mathcal{F}_t}|\xi_i|^2\right)^{1/2} \left(E^{\mathcal{F}_t}\left|E_i\xi_j - E\xi_j\right|^2\right)^{1/2}.$$

In view of A3, part c, and owing to the estimates above,

$$\begin{aligned}E^{\mathcal{F}_t}\left|\sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma}\widetilde{\xi}_i\right|^2 &\le K \sum_{i=m(t_n+t)}^{m(t_n+t+\delta)-1} \frac{A}{i^\gamma} \left(E^{\mathcal{F}_t}|\xi_i|^2\right)^{1/2} \sum_{j\ge i} \left(E^{\mathcal{F}_t}|E_i\xi_j - \xi_j|^2\right)^{1/2} \\ &\le E^{\mathcal{F}_t}\widetilde{\Gamma}_2^n(\delta).\end{aligned}$$

Taking expectation and using the Cauchy–Schwarz inequality,

$$\lim_{\delta\to 0} \limsup_{n\to\infty} E\widetilde{\Gamma}_2^n(\delta) \le K \lim_{\delta\to 0} \limsup_{n\to\infty} \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma} \sum_{j\ge i} \rho(j-i) = 0.$$

Likewise,

$$\Gamma_5^n \le E^{\mathcal{F}_t}\widetilde{\Gamma}_5^n(\delta) \quad \text{such that } \lim_{\delta\to 0}\limsup_{n\to\infty} E\widetilde{\Gamma}_5^n(\delta) = 0.$$

Putting things together and using $\widetilde{\Gamma}^n(\delta) = K\sum_{i=1}^6 \widetilde{\Gamma}_i^n(\delta)$, we arrive at

$$E^{\mathcal{F}_t}\left|v^{n,N}(t+\delta) - v^{n,N}(t)\right|^2 \le E^{\mathcal{F}_t}\widetilde{\Gamma}^n(\delta)$$

such that

$$\lim_{\delta\to 0}\limsup_{n\to\infty} E\widetilde{\Gamma}^n(\delta) = 0.$$

It follows from Theorem 3.3 of [13] that $\{v^{n,N}(\cdot)\}$ is tight in $D^r[0,\infty)$.

Step 3. Characterize the limit process. We utilize the direct averaging techniques here (see [13, Chapter 5]). Since $\{v^{n,N}(\cdot)\}$ is tight, by Prohorov's theorem, we may extract a convergent subsequence. We do so and still denote the subsequence by $\{v^{n,N}(\cdot)\}$ for notational simplicity. Furthermore, denote the limit by $v^N(\cdot)$. Using the Skorokhod representation (see [4] and [13]), without changing notation, suppose that $v^{n,N}(\cdot)$ converges to $v^N(\cdot)$ w.p.1; the convergence is uniform on any bounded time interval.

We prove that $v^N(\cdot)$ is a solution of the truncated equation

$$dv^N(t) = -Hv^N(t)dt + Cdw(t). \tag{4.8}$$

To this end, define

$$M^N(t) = v^N(t) - v^N(0) + \int_0^t Hv^N(u)du - C\int_0^t dw(u).$$

We claim that $M^N(\cdot)$ is a martingale.

To verify the martingale property, we need only prove that for any bounded and continuous function $h(\cdot)$, arbitrary t and s, any integer ν, and $s_j < t < t+s$, for $j \le \nu$,

$$\begin{aligned}&Eh(v^N(s_j), j\le\nu)(M^N(t+s) - M^N(t))\\ &\quad = Eh(v^N(s_j), j\le\nu)\bigg(v^N(t+s) - v^N(t) + \int_t^{t+s} Hv^N(u)du - \int_t^{t+s} Cdw(u)\bigg) = 0.\end{aligned}$$

To verify this equation, we start with the prelimit process $v^{n,N}(\cdot)$. Notice that

$$\begin{aligned}&\lim_{n\to\infty} Eh(v^{n,N}(s_j), j\le\nu)\left(v^{n,N}(t+s) - v^{n,N}(t)\right)\\ &= \lim_{n\to\infty} Eh(v^{n,N}(s_j), j\le\nu)\left(-\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{A}{i^\gamma}Hv_i^N + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{A^{1/2}C}{i^{\gamma/2}}W_i\right.\\ &\qquad\left. - \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{A}{i^\gamma}\widetilde{\xi}_i + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} g_i^N + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\widetilde{g}_i + \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\widehat{g}_i\right).\end{aligned} \tag{4.9}$$

We detail the estimates below for each of the terms in (4.9). The basic idea is that the observation noise $\{\xi_n\}$ is averaged out and a scaled sequence of the added noise $\{W_n\}$ results in a Brownian motion limit.

We have that

$$E \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{1}{i^\gamma \sqrt{\ln(i^{1-\gamma}+A_0)}} |v_i^N|^2 q_N(v_i^N) \leq \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{1}{i^\gamma [\ln(i^{1-\gamma}+A_0)]^{\frac{1}{2}}} \xrightarrow{n} 0 \quad \text{uniformly in } t,$$

and

$$E \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{1}{i^{2-\gamma} \ln(i^{1-\gamma}+A_0)} |v_i^N| \xrightarrow{n} 0 \quad \text{uniformly in } t.$$

Thus,

$$\lim_{n\to\infty} E \left| \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} g_i^N \right| = 0 \quad \text{uniformly in } t. \tag{4.10}$$

By virtue of a summation by parts,

$$\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma} \widetilde{\xi}_i = J_n^1 + J_n^2,$$

where

$$\begin{aligned} J_n^1 &= \varepsilon_n^1 \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^{\gamma/2}} \xi_i, \\ J_n^2 &= \sum_{i=m(t_n+t)}^{m(t_n+t+s)-2} \varepsilon_i^2 \sum_{j=m(t_n+t)}^{i} \frac{A}{j^{\gamma/2}} \xi_j, \\ \varepsilon_n^1 &= \frac{\sqrt{\ln([m(t_n+t+s)-1]^{1-\gamma}+A_0)}}{(m(t_n+t+s)-1)^{\gamma/2}}, \\ \varepsilon_n^2 &= \frac{\sqrt{\ln(n^{1-\gamma}+A_0)}}{n^{\gamma/2}} - \frac{\sqrt{\ln((n+1)^{1-\gamma}+A_0)}}{(n+1)^{\gamma/2}}. \end{aligned}$$

As in the development of Step 2,

$$\sum_{j\geq i} |E\xi_i\xi_j| \leq \sum_{j\geq i} E^{\frac{1}{2}} |\xi_i|^2 E^{\frac{1}{2}} |E_i\xi_j - E\xi_j|^2,$$

so

$$\begin{aligned} E|J_n^1| &\leq \varepsilon_n^1 E^{1/2} \left| \sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^{\gamma/2}} \xi_i \right|^2 \\ &\leq K\varepsilon_n^1 \left(\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^{\gamma/2}} \sum_{j\geq i} \frac{A}{j^{\gamma/2}} |E\xi_i' \xi_j| \right)^{1/2} \\ &\leq K\varepsilon_n^1 \left(\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma} \sum_{j\geq i} \rho(j-i) \right)^{1/2} \\ &\to 0 \quad \text{as } n\to\infty \text{ uniformly in } t. \end{aligned}$$

Likewise, since

$$\sup_{i\le m(t_n+t+s)-1} E\left|\sum_{j=m(t_n+t)}^{i}\frac{A}{j^{\gamma/2}}\xi_j\right| \le \sup_{i\le m(t_n+t+s)-1} E^{\frac{1}{2}}\left|\sum_{j=m(t_n+t)}^{i}\frac{A}{j^{\gamma/2}}\xi_j\right|^2 \le K,$$

we have

$$\begin{aligned}E|J_n^2| &\le \sum_{i=m(t_n+t)}^{m(t_n+t+s)-2}\varepsilon_i^2 \sup_{i\le m(t_n+t+s)-1} E\left|\sum_{j=m(t_n+t)}^{i}\frac{A}{j^{\gamma/2}}\xi_j\right|\\ &\xrightarrow{n}0 \quad \text{uniformly in } t.\end{aligned}$$

Thus

$$\lim_{n\to\infty} E\left|\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{A}{i^\gamma}\widetilde{\xi}_i\right| = 0, \text{ and } \lim_{n\to\infty} E\left|\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\widetilde{g}_i\right| = 0, \quad \text{uniformly in } t. \qquad \square$$

To proceed, we state a variant of Donsker's invariance theorem. A brief sketch of the proof is in the appendix.

LEMMA 4.2. *Define*

$$w^n(t) = \sum_{i=n}^{m(t_n+t)-1}\frac{A^{1/2}}{i^{\gamma/2}}W_i.$$

Then $w^n(\cdot)$ converges weakly to $w(\cdot)$, a standard Brownian motion process.

By virtue of Lemma 4.2 above,

$$\lim_{n\to\infty} E\left|\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\widehat{g}_i\right| = 0 \quad \text{uniformly in } t.$$

Now, (4.5) can be rewritten as

$$v^{n,N}(t+s) - v^{n,N}(t) = -\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{AH}{i^\gamma}v_i^N + C\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1}\frac{A^{1/2}}{i^{\gamma/2}}W_i + o(1),$$

where $o(1)\xrightarrow{n}0$, in probability, uniformly in t.

Let Δ_n be chosen so that $\Delta_n > 0$, $\Delta_n \to 0$ as $n\to\infty$, and

$$\limsup_{n\to\infty}\frac{\{j^{-\gamma};\ j\ge n\}}{\Delta_n} = 0.$$

For each l,

$$\lim_{n\to\infty}\frac{1}{\Delta_n}\sum_{i=m(t_n+l\Delta_n)}^{m(t_n+l\Delta_n+\Delta_n)-1}\frac{A}{i^\gamma} = 1.$$

This implies that

$$\frac{1}{\Delta_n}(t_{m(t_n+l\Delta_n+\Delta_n)} - t_{m(t_n+l\Delta_n)})\xrightarrow{n}1.$$

Partitioning $[0, t+s]$ into subintervals, we have

$$\begin{aligned}
&\lim_{n\to\infty} Eh(v^{n,N}(s_j), j \le \nu)\left(-\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma} H v_i^N\right)\\
&\quad= \lim_{n\to\infty} Eh(v^{n,N}(s_j), j \le \nu)\left(-\sum_{l\in I_n} \Delta_n \frac{1}{\Delta_n} \sum_{i=m(t_n+l\Delta_n)}^{m(t_n+l\Delta_n+\Delta_n)-1} \frac{A}{i^\gamma} H v_i^N\right)\\
&\quad= \lim_{n\to\infty} Eh(v^{n,N}(s_j), j \le \nu)\left(-\sum_{l\in I_n} \Delta_n \frac{1}{\Delta_n} \sum_{i=m(t_n+l\Delta_n)}^{m(t_n+l\Delta_n+\Delta_n)-1} \frac{A}{i^\gamma} H v_{m(t_n+l\Delta_n)}^N\right),
\end{aligned} \tag{4.11}$$

where $I_n = \{l \in Z_+;\ \ t \le l\Delta_n < t+s\}$. Letting $t_{m(t_n+l\Delta_n)} \to u$, the choice of Δ_n leads to

$$\begin{aligned}
&\lim_{n\to\infty} Eh(v^{n,N}(s_j), j \le \nu)\left(-\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A}{i^\gamma} H v_i^N\right)\\
&\qquad= Eh(v^N(s_j), j \le \nu)\left(-\int_t^{t+s} H v^N(u)du\right)
\end{aligned} \tag{4.12}$$

by the weak convergence of $v^{n,N}(\cdot)$ and the Skorokhod representation. In addition, by Lemma 4.2,

$$\sum_{i=m(t_n+t)}^{m(t_n+t+s)-1} \frac{A^{1/2}}{i^{\gamma/2}} W_i = \sum_{l\in I_n} \sum_{i=m(t_n+l\Delta_n)}^{m(t_n+l\Delta_n+\Delta_n)-1} \frac{A^{1/2}}{i^{\gamma/2}} W_i \Rightarrow \int_t^{t+s} dw(u). \tag{4.13}$$

Combining the estimates obtained so far, $(v^{n,N}(\cdot), w^n(\cdot))$ converges weakly to $(v^N(\cdot), w(\cdot))$ such that

$$v^N(t) = v^N(0) - H\int_0^t v^N(s)ds + C\int_0^t dw(s).$$

Step 4. Let the truncation level grow. The argument here is similar to that of [13, Corollary to Theorem 3.2]. Let $P_{v(0)}(\cdot)$ and $P^N(\cdot)$ be the measures induced by $v(\cdot)$ and $v^N(\cdot)$, respectively. The measure $P_{v(0)}(\cdot)$ is unique by virtue of A4. For each $T < \infty$, $P_{v(0)}(\cdot)$ agrees with $P^N(\cdot)$ on all Borel subsets of the set of paths in $D^r[0,\infty)$ whose values are in S_N for $t \le T$. Notice that

$$P_{v(0)}\left(\sup_{t\le T} |v(t)| \le N\right) \xrightarrow{N} 1.$$

This, together with the weak convergence of $v^{n,N}(\cdot)$ to $v^N(\cdot)$, yields $v^n(\cdot) \Rightarrow v(\cdot)$. The proof of the theorem is completed. □

Remark. The proof above concentrates on algorithm (1.2). For algorithm (1.3) the argument is very similar; the main difference is the scaling. In this case, define

$$t_n = \sum_{i=1}^n \frac{A}{i} \quad \text{and} \quad m(t) = \max\{n;\ t_n \le t\}.$$

The motivation for using such interpolation intervals comes from ODE (ordinary differential equation) methods for stochastic approximation algorithms.

5. Further discussions.

This section consists of three parts. The first subsection discusses a number of issues related to the asymptotic results obtained thus far. Section 5.2 concentrates on the case in which $\nabla f(x)$+noise cannot be obtained and one can only use gradient estimates instead. Finally, we conclude with a few more remarks.

5.1. A few observations.

In view of the limit results obtained, the following points are worth noticing.

A. Since $H = f_{xx}(0)$, $-H$ is a stable matrix. The stationary covariance Σ of the diffusion in (4.3) can be obtained by solving the Liapunov equation $\Sigma H + H'\Sigma = -C^2 I$.

B. A direct consequence of Theorem 4.1 and the observation above is that for both algorithms (1.2) and (1.3), $\sqrt{\ln(n^{1-\gamma}+A_0)}X_n$ and $\sqrt{\ln\ln(n+A_0)}X_n$ are asymptotically normal with zero mean and covariance Σ.

C. The rates of convergence of the global stochastic approximation algorithms depend mainly on the step size of the added random perturbations. Since the step size a_n is much smaller than b_n, the observation noise is averaged out in the limit.

D. A consequence of statement C above is that the step size in (1.2) is preferable. From a computation point of view, one always wants to choose a larger step size whenever possible to force the iterates to approach the desired target value faster. This is another reason that $a_n = O(1/n^\gamma)$ for $0 < \gamma < 1$ is preferred over $a_n = O(1/n)$. For more discussions on the step size sequences, see [17, Chapters 10 and 11], and also [18] for related issues.

E. In the limit stochastic differential equation (4.3) for both algorithms (1.2) and (1.3), the drift is exactly the same. This reveals another feature of the global stochastic optimization algorithms. For classical stochastic approximation algorithms, e.g., (1.2) and (1.3) without the added random perturbations, the scalings are $n^{\gamma/2}$ and $\sqrt{n}$, respectively. The corresponding drifts of the limit stochastic differential equations are $\widetilde{H} = H$ for $0 < \gamma < 1$ and $\widetilde{H} = H - I/2$ for $\gamma = 1$, respectively.

5.2. Kiefer–Wolfowitz-type algorithms.

Whenever possible, one tries to observe the gradient $\nabla f(x)$ directly without recourse to the finite difference method. However, the situation in which one can obtain only noisy observations of the function values $f(\cdot)$ (not that of $\nabla f(\cdot)$) is of practical concern. This subsection is devoted to Kiefer–Wolfowitz (KW)-type algorithms. We show that the framework developed in this paper can be adopted to treat KW algorithms with added random perturbations. It is conceivable that such an approach can also be used for variants of stochastic optimization algorithms such as random directions methods and stochastic optimization algorithms in conjunction with the so-called infinitesimal perturbation analysis methods.

For some $A > 0$, $A_0 > 0$, and $B > 0$, define the step size sequences and the finite difference intervals as

$$a_n = \begin{cases} \dfrac{A}{n^\gamma}, & \text{when } 0 < \gamma < 1, \\ \dfrac{A}{n}, & \text{when } \gamma = 1, \end{cases} \qquad b_n = \begin{cases} \dfrac{B}{n^{\frac{\gamma}{2}}\sqrt{\ln(n^{1-\gamma}+A_0)}}, & \text{when } 0 < \gamma < 1, \\ \dfrac{B}{\sqrt{n\ln\ln(n+A_0)}}, & \text{when } \gamma = 1, \end{cases}$$

and

$$c_n = \frac{c_0}{n^\alpha} \quad \text{for some } c_0 > 0.$$

Consider KW-type algorithms (with centered finite difference) of the form

$$X_{n+1} = X_n - a_n \left(\frac{Y_n^+ - Y_n^-}{2c_n} \right) + b_n W_n, \tag{5.1}$$

where $\{W_n\}$ is a sequence of added random perturbations as in (1.2) and (1.3),

$$Y_n^{\pm} = (Y_{n,1}^{\pm}, \dots, Y_{n,r}^{\pm})' \in \mathbb{R}^r,$$

and $Y_{n,i}^{\pm}$, for $i = 1, \dots, r$, are the observations at time n and parameter values $X_n \pm e_i c_n$, with e_i being the ith standard unit vector of $\mathbb{R}^r$.

To proceed, define the bias and noise by

$$\begin{aligned} &\beta_n(x) = (\beta_{n,1}(x), \dots, \beta_{n,r}(x))' \in \mathbb{R}^r, \qquad \text{where} \\ &\beta_{n,i}(x) = \frac{\partial f(x)}{\partial x_i} - \frac{f(x + c_n e_i) - f(x - c_n e_i)}{2c_n}, \qquad i = 1, \dots, r, \end{aligned}$$

and

$$\begin{aligned} &\eta_n = (\eta_{n,1}, \dots, \eta_{n,r})' \in \mathbb{R}^r, \qquad \text{where} \\ &\eta_{n,i} = [f(X_n + c_n e_i) - Y_n^+] - [f(X_n - c_n e_i) - Y_n^-], \qquad i = 1, \dots, r. \end{aligned}$$

Then algorithm (5.1) above can be rewritten as

$$X_{n+1} = X_n - a_n \nabla f(X_n) + a_n \frac{\eta_n}{2c_n} + a_n \beta_n(X_n) + b_n W_n. \tag{5.2}$$

The main differences between (1.2) and (5.2) (respectively, (1.3) and (5.2)) are the addition of the bias term and the appearance of the step size c_n. With the term $b_n W_n$ dropped from (5.2) above, it reduces to a standard form of the KW algorithm. It is well known that with the selection of a_n and c_n above, the convergence rate of the KW algorithm is of the order $O(n^{-\tilde{\rho}})$, where the best value of $\tilde{\rho}$ is given by $\tilde{\rho} = 2\alpha$ and $\tilde{\rho} + \alpha = \gamma/2$ (see [12] and [17, Chapter 10]). Thus, for $0 < \gamma \le 1$, choose $\alpha = \gamma/6$. With the addition of the added perturbation, the convergence takes place at a much slower rate. In fact, expanding the bias term and noticing that the second-order term does not show up because of the central finite difference, we have (see [17, p. 292])

$$\beta_n(X_n) = -\frac{f_{xxx}(X_n) c_n^2}{3!} + o(c_n^2).$$

A detailed calculation reveals that we still have (3.1) with the addition of terms involving $a_n \beta_n(X_n)$ in the first and the second lines and with ξ_n replaced by $-\eta_n/(2c_n)$. Now redefine

$$V_1(x, n) = -\sum_{j=n}^{\infty} \frac{A}{2c_0} \frac{1}{j^{5\gamma/6}} E_n V_x'(x) \eta_j.$$

As in the previous derivation,

$$E|V_1(x, n)| \le \frac{K}{n^{5\gamma/6}} (1 + EV(x)).$$

Defining

$$\widehat{V}(n) = V(x) + V_1(x, n)$$

and proceeding as before, we derive an expression like (3.8), with $O(1/n^{2\gamma})$ replaced by $O(1/n^{5\gamma/3})$. Using the same ideas as in the developments of sections 3 and 4 with slight changes, we can derive the following result.

THEOREM 5.1. *Suppose $c_n = c_0/n^{\gamma/6}$ for some $c_0 > 0$, and for each x,*

$$|f_{xxx}(x)| \le K(1 + V^{1/2}(x)).$$

Then, Theorems 3.2 *and* 4.1 *hold with ξ_n replaced by $-\eta_n/2c_n$.*

5.3. Concluding remarks. In this paper, we developed rates of convergence results for two stochastic global optimization algorithms. Asymptotic properties of the algorithms were obtained via weak convergence methods. Our results indicate that algorithm (1.2) has a faster rate of convergence than (1.3). The limit diffusion for the estimation error depends mainly on the random perturbation term.

Related work on globally convergent stochastic approximation can be found in Yakowitz [21]. The study of the global optimization algorithms is often closely related to the properties of the diffusion processes. It will be interesting to see the connection between such algorithms and singularly perturbed diffusions (see, for instance, Khasminskii and Yin [10] and Yin and Zhang [22, Chapters 8–10] for related control problems of singularly perturbed Markovian systems). To relax the conditions on the growth rate of the function or to incorporate with various constraints, one may wish to consider projection algorithms. Suppose that the global minimum is interior to the projection region; then one can use essentially the same kind of analysis as in this work to analyze the rates of convergence of the corresponding algorithms. It is an interesting problem to treat functions with multiple global minima. One of the possible approaches is Kaniovskii [8]. Moreover, various perturbation and random directions methods for estimating the gradient may be incorporated in the algorithms; see, for example, Ho and Cao [7], Kushner and Yin [17], and Spall [20], among others.

When simulations are used to obtain the gradient estimates of $f(\cdot)$, one may wish to consider the corresponding budget-dependent convergence rates (see L'Ecuyer and Yin [18]). If large-dimensional problems are encountered, one may wish to use parallel processing methods with multiprocessors to implement the computation task (see Kushner and Yin [16] for discussion of related matters). Effort may also be directed to finding fine tuning procedures for the step size sequences $\{a_n\}$ and $\{b_n\}$ (see related discussions for stochastic approximation algorithms in [17, Chapter 11]).

6. Appendix.

6.1. Proof of Lemma 3.3. We verify only the first two inequalities, since the third one is an easy consequence of the second one. Note that for $m \ge k \ge 1$,

$$\sum_{j=k}^{m} \frac{1}{j^\gamma} B_{nj} = \frac{1}{A\lambda} \sum_{j=k}^{m} (B_{nj} - B_{n,j-1}) = \frac{1}{A\lambda}(B_{nm} - B_{n,k-1}), \tag{6.1}$$

and hence the first inequality holds.

To prove the second inequality, use a partial summation together with (6.1),

$$\begin{aligned}
&\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)}B_{nj}\\
&\quad=\sum_{j=1}^{n}\frac{1}{j^{\gamma}}B_{nj}+\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n-1}\left[\frac{1}{\ln(j^{1-\gamma}+A_0)}-\frac{1}{\ln((j+1)^{1-\gamma}+A_0)}\right]\sum_{k=1}^{j}\frac{1}{k^{\gamma}}B_{nk}\\
&\quad\le K+\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n-1}\frac{\ln((j+1)^{1-\gamma}+A_0)-\ln(j^{1-\gamma}+A_0)}{[\ln(j^{1-\gamma}+A_0)]^2}B_{nj}.
\end{aligned}$$

(6.2)

Note that

$$\begin{aligned}
&\ln[(j+1)^{1-\gamma}+A_0]-\ln(j^{1-\gamma}+A_0)\\
&\quad=(1-\gamma)\ln\left(1+\frac{1}{j}\right)+\ln\left(1+\frac{A_0}{(j+1)^{1-\gamma}}\right)-\ln\left(1+\frac{A_0}{j^{1-\gamma}}\right)\\
&\quad\le\ln\left(1+\frac{1}{j}\right)<\frac{1}{j}.
\end{aligned}$$

It then readily follows that

$$\begin{aligned}
&\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n-1}\frac{\ln((j+1)^{1-\gamma}+A_0)-\ln(j^{1-\gamma}+A_0)}{[\ln(j^{1-\gamma}+A_0)]^2}B_{nj}\\
&\quad\le\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\frac{1}{j[\ln(j^{1-\gamma}+A_0)]^2}B_{nj}.
\end{aligned}$$

Thus, for some $0<\widetilde{\kappa}_0<1$, (6.2) becomes

$$\begin{aligned}
&\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)}B_{nj}\\
&\quad\le K+\widetilde{\kappa}_0\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)}B_{nj}.
\end{aligned}$$

This yields that

$$(1-\widetilde{\kappa}_0)\ln(n^{1-\gamma}+A_0)\sum_{j=1}^{n}\frac{1}{j^{\gamma}\ln(j^{1-\gamma}+A_0)}B_{nj}\le K<\infty.$$

The desired inequality thus follows. □

6.2. A sketch of the proof of Lemma 4.2. Only a rough sketch is given here; the lemma can be proved as in [14, Part two of Theorem 2]; see also related results in [4, 17]. Since $\{W_n\}$ is a sequence of i.i.d. random variables, it is readily verified that

$$\begin{aligned}
E|w^n(t+s)-w^n(t)|^2&=\sum_{i=m(t_n+t)}^{m(t_n+t+s)}\sum_{j=m(t_n+t)}^{m(t_n+t+s)}E\frac{A^{1/2}}{i^{\gamma/2}}\frac{A^{1/2}}{j^{\gamma/2}}W_i'W_j\\
&=\sum_{i=m(t_n+t)}^{m(t_n+t+s)}\frac{A}{i^{\gamma}}EW_i'W_i\le Ks,
\end{aligned}$$

and similarly,

$$E|w^n(t+s)-w^n(t)|^4 \leq K \sum_{i=m(t_n+t)}^{m(t_n+t+s)} \sum_{j=m(t_n+t)}^{m(t_n+t+s)} \frac{A}{i^\gamma}\frac{A}{j^\gamma} \leq Ks^2.$$

Thus $\{w^n(\cdot)\}$ is tight and all limits have continuous paths w.p.1.

Define

$$\Sigma^n(t) = \sum_{j=n}^{m(t_n+t)} \frac{A}{j^\gamma} E_j W_j W_j'.$$

Then $\Sigma^n(t) \to tI$ as $n \to \infty$. Moreover, for each $T > 0$,

$$\lim_{s\to 0} E \sup_{t\leq T} |w^n(t+s)-w^n(t)|^2 = 0,$$
$$\lim_{s\to 0} E \sup_{t\leq T} |\Sigma^n(t+s)-\Sigma^n(t)| = 0.$$

Furthermore, $w^n(t)w^{n,\prime}(t)-\Sigma^n(t)$ is a martingale. It then follows that $w^n(\cdot) \Rightarrow w(\cdot)$, a standard Brownian motion process (see, for example, Ethier and Kurtz [4, Chapter 7]). □

Acknowledgments. I thank Saul Gelfand for his comments on an early draft of the manuscript; my thanks also go to the referees and the editor, Alan King. Their remarks, comments, and suggestions led to much improvement in the presentation.

REFERENCES

[1] A. Benveniste, M. Métivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximation*, Springer-Verlag, Berlin, 1990.

[2] T.-S. Chiang, C.-R. Hwang, and S.-J. Sheu, *Diffusion for global optimization in* $\mathbb{R}^n$, SIAM J. Control Optim., 25 (1987), pp. 737–753.

[3] S. Geman and C.-R. Hwang, *Diffusions for global optimization*, SIAM J. Control Optim., 24 (1986), pp. 1031–1043.

[4] S. N. Ethier and T. G. Kurtz, *Markov Processes, Characterization and Convergence*, Wiley, New York, 1986.

[5] M. I. Freidlin and A. D. Wentzell, *Random Perturbations of Dynamical Systems*, Springer-Verlag, Berlin, 1984.

[6] S. B. Gelfand and S. K. Mitter, *Recursive stochastic algorithms for global optimization in* $\mathbb{R}^d$, SIAM J. Control Optim., 29 (1991), pp. 999–1018.

[7] Y. C. Ho and X. R. Cao, *Perturbation Analysis of Discrete Event Dynamic Systems*, Kluwer, Boston, MA, 1991.

[8] Y. M. Kaniovskii, *On the limit distribution of processes of stochastic approximation type when the regression function has several roots*, Soviet Math. Dokl., 38 (1989), pp. 210–211.

[9] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.

[10] R. Z. Khasminskii and G. Yin, *On transition densities of singularly perturbed diffusions with fast and slow components*, SIAM J. Appl. Math., 56 (1996), pp. 1794–1819.

[11] H. J. Kushner, *Asymptotic global behavior for stochastic approximation and diffusions with slowly decreasing noise effects: Global minimization via Monte Carlo*, SIAM J. Appl. Math., 47 (1987), pp. 169–185.

[12] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Springer-Verlag, New York, 1978.

[13] H. J. Kushner, *Approximation and Weak Convergence Methods for Random Processes, with Applications to Stochastic Systems Theory*, MIT Press, Cambridge, MA, 1984.

[14] H. J. Kushner and H. Huang, *Rates of convergence for stochastic approximation type algorithms*, SIAM J. Control Optim., 17 (1979), pp. 607–617.

[15] H. J. Kushner and H. Huang, *Asymptotic properties of stochastic approximations with constant coefficients*, SIAM J. Control Optim., 19 (1981), pp. 87–105.

[16] H. J. Kushner and G. Yin, *Asymptotic properties of distributed and communicating stochastic approximation algorithms*, SIAM J. Control Optim., 25 (1987), pp. 1266–1290.

[17] H. J. Kushner and G. Yin, *Stochastic Approximation Algorithms and Applications*, Springer-Verlag, New York, 1997.

[18] P. L'Ecuyer and G. Yin, *Budget-dependent convergence rate of stochastic approximation*, SIAM J. Optim., 8 (1998), pp. 217–247.

[19] M. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, *Equations of state calculations by fast computing machines*, J. Chem. Phys., 21 (1953), pp. 1087–1091.

[20] J. C. Spall, *Multivariate stochastic approximation using a simultaneous perturbation gradient approximation*, IEEE Trans. Automat. Control, AC-37 (1992), pp. 331–341.

[21] S. Yakowitz, *A globally convergent stochastic approximation*, SIAM J. Control Optim., 31 (1993), pp. 30–40.

[22] G. Yin and Q. Zhang, *Continuous-Time Markov Chains and Applications: A Singular Perturbation Approach*, Springer-Verlag, New York, 1998.

SIAM J. Optim.
Vol. 10, No. 1, pp. 121–131

MULTICOORDINATION METHODS FOR SOLVING CONVEX BLOCK-ANGULAR PROGRAMS[*]

R. R. MEYER[†] AND G. ZAKERI[‡]

Abstract. Several decomposition methods are considered for solving block-angular programs (BAPs) in parallel. We present a computational comparison of synchronous multicoordination methods. The most efficient of these approaches is shown to involve an intermediate number of blocks in the coordination phase.

Key words. large-scale optimization, decomposition, coordination, parallel computing

AMS subject classifications. 90C06, 90C35, 49M27, 65Y05

PII. S1052623496313635

1. Introduction.

We are concerned with parallel solution methods for mathematical programs of the following block-angular form:

$$
\boxed{
\begin{array}{llllll}
\text{BAP} & \underset{x}{\text{minimize}} & c(x) & & & \\
& \text{subject to} & A_{[1]}x_{[1]} & & = & b_{[1]}, \\
& & A_{[2]}x_{[2]} & & = & b_{[2]}, \\
& & \ddots & & & \vdots \\
& & & A_{[K]}x_{[K]} & = & b_{[K]}, \\
& & D(x) & & \leq & d, \\
& & \mathbf{0} \leq x \leq u. & & &
\end{array}
}
$$

We assume that the functions c and D are convex and at least once continuously differentiable. The $x_{[k]}$'s are blocks of variables and the $A_{[k]}$'s are constraint matrices, which in the case of multicommodity network flow problems will be node-arc incidence matrices. We assume the upper bounds $u_{[k]}$ are finite. In the constraints, the blocks $x_{[k]}$ are only coupled together through the J coupling constraints $D(x) \leq d$.

In general, P processors will be available to solve BAPs with K blocks (we assume here that $P \geq K$; see [15] for an alternative approach when $P < K$). Decomposition methods (e.g., Dantzig–Wolfe [3], Benders [16], and Schultz–Meyer [12, 4] methods) work by first removing the coupling constraints and replacing $c(x)$ with a block-separable approximation (if $c(x)$ is not block separable). This process makes the altered BAP block separable. Decomposition methods then continue by iteratively

- solving a subproblem for each block k of BAP (this can be done in parallel),
- incorporating the solution of these subproblems in a coordinator problem to determine the next iterate.

[*] Received by the editors December 12, 1996; accepted for publication (in revised form) July 22, 1998; published electronically October 20, 1999. This research was partially funded by National Science Foundation grants CDA-9024618 and CCR-9306807, Air Force Office of Scientific Research grant F 49620-94-1-0036, and the New Zealand Foundation for Research Science and Technology grant UOA-403.

http://www.siam.org/journals/siopt/10-1/31363.html

[†] Center for Parallel Optimization, Computer Sciences Department, University of Wisconsin–Madison, 1210 West Dayton Street, Madison, WI 53706 (rrm@cs.wisc.edu).

[‡] MCS Division, Argonne National Laboratories, 9700 S. Cass Avenue, Argonne, IL 60439-4844 (zakeri@ mcs.anl.gov).

The solution of each subproblem is referred to below by a "search direction" for the corresponding block. The main aim of this paper is to examine the achievement of further parallelism by simultaneously solving K *coordinator* problems instead of just one, an approach we term *multicoordination.*

Previously, in [4], we presented synchronous multicoordination schemes for the solution of BAP using barrier decomposition. Here we remind the reader of those methods and discuss implementation issues. We will present new computational results on two classes of linear multicommodity network flow problems.

2. Review of synchronous multicoordination methods. In [11, 12], Schultz and Meyer developed specialized barrier methods for the solution of BAP. Let the barrier problem (denoted BP) be defined as

$$\boxed{\begin{array}{lrl} \text{BP} & \underset{\mathrm{x}}{\text{minimize}} & c(\mathrm{x}) + \tau\rho(d - D(\mathrm{x})) \\ & \text{subject to} & A_{[k]}\mathrm{x}_{[k]} = b_{[k]}, \quad k = 1, \dots, K, \\ & & \mathbf{0} \le \mathrm{x} \le u, \end{array}}$$

where τ is a penalty parameter ($\tau > 0$) and ρ is a differentiable convex barrier function (see [5, 11]). The barrier function methods solve a sequence of BPs with $\tau^i \downarrow 0$, in order to generate a good approximation of the optimal objective of BAP. We may not, however, commence solving a sequence of BPs unless an interior point (relative to coupling constraints) is available. To obtain an initial interior feasible point for BP we (approximately) solve a sequence of shifted barrier problems of the form

$$\boxed{\begin{array}{lrl} \text{SBP} & \underset{\mathrm{x}}{\text{minimize}} & c(\mathrm{x}) + \tau^i\rho(\theta^i - D(\mathrm{x})) \\ & \text{subject to} & A_{[k]}\mathrm{x}_{[k]} = b_{[k]}, \quad k = 1, \dots, K, \\ & & \mathbf{0} \le \mathrm{x} \le u, \end{array}}$$

where θ^i are the shifted barriers. In order to outline the construction of θ^i, let x^0 be the solution of BAP without the coupling constraints (we refer to this as the "relaxed" problem). The value of θ^0 is determined by

$$\theta_j^0 = \begin{cases} d_j & \text{if} \quad D_j(\mathrm{x}^0) < d_j, \\ D_j(\mathrm{x}^0) + \Theta & \text{if} \quad D_j(\mathrm{x}^0) \ge d_j, \end{cases} \tag{1}$$

where $\Theta > 0$ is a constant. In later iterations we vary θ according to

$$\theta_j^i = \begin{cases} d_j & \text{if} \quad D_j(\mathrm{x}^i) < d_j, \\ \lambda_\theta D_j(\mathrm{x}^i) + (1 - \lambda_\theta)\theta_j^{i-1} & \text{if} \quad D_j(\mathrm{x}^i) \ge d_j, \end{cases} \tag{2}$$

where $0 < \lambda_\theta < 1$ is a constant.

We (approximately) solve each problem of types BP and SBP using an iterative fork-join scheme. Here we refer to each iteration as an inner iteration and use x^t to denote the iterate (using x^i for the outer iterates). Here and in [4] we define $f(\tau, \theta, \mathrm{x}) = c(\mathrm{x}) + \tau\rho(\theta - D(\mathrm{x}))$. We start the iterative process at outer iteration i with $x^0 = \mathrm{x}^i$; at inner iteration t, assuming a base point x^t, we first construct $\overline{\mathrm{R}}(x^t)$, a type of trust region around x^t, and then find search directions by solving the linear problem

$$\boxed{\begin{array}{lrl} \text{SLP} & \underset{y}{\text{minimize}} & \nabla f(x^t)(y - x^t) \\ & \text{subject to} & Ay = b, \\ & & \mathbf{0} \le y \le \overline{\mathrm{R}}(x^t). \end{array}}$$

By the convergence theory developed in [4] and [14], only one inner iteration is required per outer iteration; however, additional iterations (which are also theoretically valid) have proved efficient in the implementation. Let $\mathrm{B} := \{\, x \mid Ax = b \text{ and } \mathbf{0} \le x \le u \,\}$. We require $\overline{\mathrm{R}}(x^t)$ to be a continuous function of x^t satisfying $\mathbf{0} \le x \le \overline{\mathrm{R}}(x^t) \le u$ for any $x \in \mathrm{B}$. Define a decoupled resource allocation for B by

$$\mathrm{R}(x^t) := \left\{\, z \in R^n \mid \mathbf{0} \le z \le \overline{\mathrm{R}}(x^t) \,\right\}.$$

$\overline{\mathrm{R}}(x^t)$ is constructed so that the following hold:

- For any bounded sequence $\{x^t\} \subseteq \mathrm{B}$, the set $\bigcup_{t=0}^{\infty} \mathrm{R}(x^t)$ is bounded. This condition is referred to as the boundedness of resource allocation.
- For any $z \in \mathrm{B}$ and any bounded sequence $\{x^t\} \subseteq \mathrm{B}$ with

  $$\alpha^t := \max\left\{\, \alpha \mid 0 \le \alpha \le 1 \text{ and } x^t + \alpha(z - x^t) \in \mathrm{R}(x^t) \,\right\},$$

  we have $\liminf_{t\to\infty} \alpha^t > 0$. This condition ensures that we can always take a step in any feasible search direction.

For more details on the decoupled resource allocation, see [11].

Problem SLP may be decomposed into K independent subproblems of the form

$$\begin{array}{lll} \mathrm{LP}_k & \underset{y_{[k]}}{\text{minimize}} & \nabla f(x^t)_{[k]}\big(y_{[k]} - x^t_{[k]}\big) \\ & \text{subject to} & A_{[k]} y_{[k]} = b_{[k]}, \\ & & \mathbf{0} \le y_{[k]} \le \overline{\mathrm{R}}(x^t)_{[k]}, \end{array}$$

one for each block k. We solve LP_k on processor k and denote the solutions by $y^t_{[k]}$.

Once the search directions $y^t_{[k]}$ are determined we need to assign stepsizes to be taken in each direction. This was originally done by Schultz and Meyer [11, 12] using a complex coordinator (a single coordination problem involving all blocks). Instead, we perform this step using multiple simpler coordinators, hence taking advantage of parallelism. The single-variable and group multicoordination methods are discussed in [4]. We briefly review them now.

2.1. Single-variable multicoordination. Here, once processor k has obtained the search direction from LP_k, it solves the single-variable coordinator problem,

$$\begin{array}{lll} \mathrm{SVMC}_k & \underset{w_k \in \Re}{\text{minimize}} & f\big(x^t_{[1]}, \ldots, x^t_{[k-1]}, x^t_{[k]} + (y^t_{[k]} - x^t_{[k]})w_k, x^t_{[k+1]}, \ldots, x^t_{[K]}\big) \\ & \text{subject to} & \mathbf{0} \le x^t_{[k]} + (y^t_{[k]} - x^t_{[k]})w_k \le u_{[k]}, \end{array}$$

to obtain w^*_k. For each k, define

$$x^{t+\frac{1}{2}}_{[k]} = x^t_{[k]} + (y^t_{[k]} - x^t_{[k]})w^*_k.$$

We then find the coordinator with the least objective at time t (we will refer to the index of that coordinator as $c(t)$). This amounts to a simple pass through the objective values of the coordinators. Now the new iterate is determined by

$$x^{t+1}_{[k]} = \begin{cases} x^{t+\frac{1}{2}}_{[k]} & \text{if} \quad k = c(t), \\ x^t_{[k]} & \text{otherwise.} \end{cases}$$

The above coordination scheme is simple and fast and highly parallelizeable. However, at each iteration only one block of variables is updated.

2.2. Group multicoordination. In group multicoordination, each processor is responsible for several blocks of variables rather than just one. Suppose $\{p\} \subset \Gamma_p \subset \{1, \ldots, K\}$ is the set of blocks involved in coordinator p. The group coordinator problem for processor p is then given by

$$\boxed{\begin{array}{lll} \mathrm{GMC}_p \qquad & \underset{w \in \Re^K}{\text{minimize}} & f\big(x^t_{[1]} + (y^t_{[1]} - x^t_{[1]})w_1, \ldots, x^t_{[K]} + (y^t_{[K]} - x^t_{[K]})w_K\big) \\ & \text{subject to} & \mathbf{0} \le x^t_{[k]} + (y^t_{[k]} - x^t_{[k]})w_k \le u_{[k]}, \quad k \in \Gamma_p, \\ & & w_j = 0, \quad j \notin \Gamma_p. \end{array}}$$

Again we look for the coordinator with the least objective (coordinator $c(t)$). The new iterate is obtained by updates in the blocks that are members of $\Gamma_{c(t)}$. The computational experience below indicates that group multicoordination is more efficient than both single-variable multicoordination and full coordination (using all search directions, as in Schultz–Meyer). Note that only one processor will be busy during full coordination. The convergence proofs for the multicoordination methods given above (as well as a third method called block-plus-group multicoordination) can be found in [4].

3. Computational results. In this section we present computational results from the implementation of single-variable and group multicoordination schemes. We implemented these methods on two sets of linear multicommodity network flow problems. The first test set consists of the well-known patient distribution system (PDS) problems that arise from a logistic application. The second test set consists of MNETGEN problems, which are randomly generated via the multicommodity version of the network flow problem generator NETGEN. We start by discussing the implementation issues. We then present our computational results for each of the above-mentioned schemes and compare them to those of De Leone et al. [4], Pinar and Zenios [10], McBride and Mamer [8], Schultz and Meyer [12], and Grigoriadis and Khachiyan [6].

3.1. Parallel implementation. Our algorithm follows the basic three-phase method of Schultz and Meyer. Figure 1 presents a sketch of that method. Although we use the same three-phase structure, we generate approximate solutions of the shifted barrier problem using single-variable or group multicoordination schemes. The algorithm for approximately solving SBP using single-variable multicoordination is presented in Figure 2, and the group multicoordination scheme is presented in Figure 3.

We implemented our code on Thinking Machines Corporation's Connection Machine CM-5. This machine contains 64 processors (Sun SPARCstation 10s) in 2 partitions of size 32 each. The machine runs the CMOST 7.3 operating system, and we used the CMMD message-passing library for interprocessor communication.

As evident from the algorithms, K processors are used in both steps 1 and 3 of both schemes. This means that we used K processors, one for each subproblem (step 1). We also used K processors, one to solve each coordinator problem, be it a single-variable coordinator or a group coordinator. In the single-variable coordination scheme, very little interprocessor communication is necessary because each processor uses only the search direction it produced to generate the SMVC problem. In fact, the only interprocessor communication in this algorithm takes place when we determine the candidate with the least coordinator objective (see step 4 of Figure 2). This is done using the CMMD library function `reduce`, which very efficiently searches all processors for a minimum such value.

Assume that the parameters

$$\Theta > 0, \quad \lambda_\theta \in (0,1), \quad \tau^0 > 0, \quad \tau_{\text{inf}} > 0, \quad \lambda_\tau \in (0,1)$$

are given. Also define

$$\mathrm{B} = \{\, \mathrm{x} \mid A\mathrm{x} = b, \quad 0 \le \mathrm{x} \le u \,\}, \quad \mathrm{C} = \{\, \mathrm{x} \mid D(\mathrm{x}) \le d \,\}, \text{ and}$$

$$\mathrm{C}^\circ = \{\, \mathrm{x} \mid D(\mathrm{x}) < d \,\}.$$

```
Relaxed Phase
$i = 0$.
Compute $\mathrm{x}^0$ as the solution of the “relaxed” problem.
If we determine that $\mathrm{B} = \emptyset$ then quit.
Set $\theta^0$ as in (1).
If $\mathrm{x}^0 \in \mathrm{C}^\circ$
        then terminate and declare $\mathrm{x}^0$ optimal,
        else go to the feasibility phase.

Feasibility Phase ($\tau^i = \tau^0$)
Generate $\mathrm{x}^{i+1}$ as an approximate solution of the $i$th SBP.
If $\mathrm{x}^{i+1} \in \mathrm{B} \cap \mathrm{C}^\circ$
        then go to the refine phase with $i = i + 1$,
        else repeat the feasibility phase with $i = i + 1$.

Refine Phase ($\theta^i = d$)
$\tau^i = \max(\tau_{\text{inf}}, \lambda_\tau \tau^{i-1})$.
Generate $\mathrm{x}^{i+1}$ as an approximate solution of BP with penalty parameter $\tau^i$.
If $\mathrm{x}^{i+1}$ meets suitable termination criteria then quit, else repeat the refine phase
with $i = i + 1$.
```

FIG. 1. *The Schultz–Meyer three-phase method.*

To solve SBP approximately using *single-variable* multicoordination, at iteration t:

1. Solve the K linear subproblems LP_k to obtain optimal solutions $y^t_{[k]}$.
2. Define

$$(y^t_k)_{[i]} = \begin{cases} y^t_{[k]} & \text{if } i = k, \\ x^t_{[k]} & \text{otherwise.} \end{cases}$$

3. Solve the K single-variable coordinator problems SVMC_k,

$$\begin{aligned} \underset{w_k \in \Re}{\text{minimize}} \quad & f(x^t + (y^t_k - x^t)w_k) \\ \text{subject to} \quad & \mathbf{0} \le x^t + (y^t_k - x^t)w_k \le u, \end{aligned}$$

   to obtain optimal solutions w^*_k. Set $x^{t,k} = x^t + (y^t_k - x^t)w^*_k$.
4. Choose $x^{t+1} = x^{t,c(t)}$, where $c(t)$ is the index of the block that produces the least objective for the barrier problem.

FIG. 2. *Inner iteration for single-variable multicoordination.*

To solve SBP approximately using *group* multicoordination, at iteration t:

1. Solve the K linear subproblems LP_k to obtain optimal solutions $y^t_{[k]}$.
2. Let Γ_p be the group of blocks assigned to processor p. Define Y^t_p to be the matrix of search directions for blocks in Γ_p. Specifically, Y^t_p is a diagonal matrix with K blocks and the kth block of Y^t_p is given by $y^t_{[k]} - x^t_{[k]}$ provided $k \in \Gamma_p$ and $\mathbf{0}$ otherwise.
3. Solve the K group coordinator problems GMC_p,

$$\begin{array}{rl} \underset{w \in \Re^K}{\text{minimize}} & f(x^t + Y^t_p w) \\ \text{subject to} & \mathbf{0} \le x^t + Y^t_p w \le u, \end{array}$$

   to obtain optimal solutions w^*_p. Set $x^{t,p} = x^t + Y^t_p w^*_p$.
4. Choose $x^{t+1} = x^{t,c(t)}$, where $c(t)$ is the index of the group that produces the least objective for the barrier problem.

FIG. 3. *Inner iteration for group multicoordination.*

In the group multicoordination scheme, we experimented with various sizes for Γ_p (reported in section 3.4.2 below). We chose the group sizes to be odd here for simplicity. Given a group size s, we defined the set Γ_p for each processor p to be

$$\Gamma_p := \left\{ k = p \pm i \quad | \quad 0 \le i \le \frac{s-1}{2}, i \in Z, 1 \le k \le P \right\}.$$

The group multicoordination scheme calls for more communication. Each processor needs the search directions for the blocks in its group. We could have used the `send-and-receive` CMMD library function in such a way that each processor would get only the information regarding the group assigned to it. However, we found the `concat` CMMD library function more flexible and efficient. `concat` is a global communication function that operates by informing all processors of the involved data (in this case, the search directions). Although the total length of the messages communicated under `concat` is longer than the length of messages communicated by `send-and-receive`, we found `concat` much faster. We attribute this property to the fact that `concat` uses specialized broadcast hardware, whereas `send-and-receive` does only point-to-point communication. (For further details, see [1].)

3.2. Parameter values. The algorithm terminates if the maximum subproblem objective absolute value is less than the parameter *spobjtol* or if the number of iterations reaches 100 for the PDS problems or 300 for the MNETGEN problems. The code achieved at least six digits of accuracy in the optimal objective (compared with the previous results obtained by Schultz and Meyer). We scaled the cost coefficients so that $\|c\|_\infty = 1$. The remaining parameter values were as follows:

- $\lambda_\theta = 0.95$ is the parameter used in (2) to produce a sequence of shifted barriers that converge to the original barrier.
- $\tau^0 = 100$ is the initial value of the penalty parameter.
- $\tau_{\inf} = 10^{-6}$ is the minimum possible value for the penalty parameter.
- $\lambda_\tau \in (0.25, 0.4)$ is the factor by which we reduced the penalty parameter. (For the larger problems we used a greater value of λ_τ, which corresponds to a more gradual decrease of the penalty parameter.)

Table 1
Sizes of the PDS problems tested.

Problem	Max node	Max arc	Coupling	Total constr	Total var
pds.1	126	339	87	1,473	3,729
pds.2	252	685	181	2,953	7,535
pds.3	390	1117	303	4,593	12,287
pds.5	686	2,149	553	8,099	23,639
pds.10	1,399	4,433	1,169	16,558	48,763
pds.20	2,857	10,116	2,447	33,874	105,728
pds.30	4,223	15,126	3,491	49,944	154,998
pds.40	5,652	20,698	4,672	66,844	212,859

- spobjtol $= 10^{-6}$ is the bound on the subproblems' optimality gap.

3.3. The PDS problems. We chose the PDS problems because these are real-world problems and quite a few recently developed methods (including Schultz–Meyer) have used them as benchmarks. The PDS model is a logistics model designed to help make decisions about patient evacuation. "pds.n" denotes the problem that models a scenario lasting n days. Table 1 lists the sizes of the problems we considered. Note that the size of pds.n is essentially a linear function of n. PDS problems are linear multicommodity network flow problems with 11 commodities. The columns labeled "Max node" and "Max arc" present the maximum number of nodes and arcs for any commodity. The last two columns in the table present the size of the problem when considered as an LP. The column labeled "Total constr" contains the total number of node constraints plus the number of coupling constraints. The column labeled "Total var" contains the total number of variables. The column labeled "Coupling" contains the number of coupling constraints in each problem.

The block constraint matrices for these BAPs are node-arc incidence matrices. We take advantage of this fact in our code and use a very efficient network flow solver, NSM [13], to solve the subproblems. NSM uses the network simplex method. Since the upper bounds on the subproblems are changed from each iteration to the next (we adjust the decoupled resource allocation) we cannot use "hot starting." That is, we need to start with an all-artificial basis at every iteration.

We used the optimization package MINOS [9] in the form of a subroutine (MINOS 5.4) in order to solve the coordinator problems for both single-variable and group coordination. For such problems, MINOS uses a reduced-gradient algorithm in conjunction with a quasi-Newton algorithm. MINOS requires any domain constraint for the objective function to be specified explicitly. Therefore, we had to put in linear constraints that imposed lower bounds on the arguments of each log term involved in the barrier function. This procedure amounted to imposing upper and lower bounds on w when using single-variable multicoordination. In the group multicoordination case, however, we required J linear constraints for each coordinator problem (recall that J is the number of coupling constraints), which is quite significant for large problems. We used the default parameters except for the feasibility and optimality tolerances, which were set to 10^{-11} in the single-variable case and 10^{-6} in the group case.

3.4. Analysis of the results.

3.4.1. Single-variable multicoordination. In Table 2 we present the solution results for the PDS problems we tested using single-variable multicoordination. The

Table 2
Solution using single-variable multicoordination.

Problem	Feas	Opt	Inner	Rlx time	Sub time	Coor time	Total	Comm
pds.1	5	15	2	0.02	0.63	0.77	2.4	*
pds.2	5	15	2	0.04	1.49	0.90	3.7	*
pds.3	5	18	2	0.07	3.15	1.68	7.1	*
pds.5	5	23	2	0.23	9.14	3.19	16.9	*
pds.10	5	23	2	0.65	28.61	6.13	45.1	20
pds.20	5	23	2	4.60	191.91	12.56	252.8	17
pds.30	5	20	4	13.09	754.36	33.65	901.7	11
pds.40	5	20	4	36.57	1404.14	45.75	1806.4	17

first column, labeled “Feas,” contains the number of outer iterations for the feasibility phase (there were two inner iterations per outer iteration). During the feasibility phase, from one outer iteration to the next, the variable θ was changed as in (2). The “Opt” column contains the number of additional outer iterations to optimality. The “Inner” column contains the number of inner iterations per outer iteration in the optimality phase. “Rlx time” is the time it took to solve the most time-consuming subproblem in the relaxed phase, and “Sub time” contains the sum of solution times for the subproblems that took longest to be solved in each iteration. In column “Coor time” we summed the times for the coordinators that took longest to solve. The column labeled “Total” contains the total time taken to solve the corresponding PDS problem, and “Comm” is the percentage of time spent on communication. The communication overhead is calculated by subtracting the time it took to solve the relaxed problem and the subproblem and coordination time from the total time and then dividing this by the total time. Asterisks indicate that the times were too small to extract a meaningful communications percentage.

3.4.2. Group multicoordination. Table 3 contains information about the solution of large PDS problems using the group multicoordination scheme. We also tried the group method on small PDS problems, but the results were not competitive with those found in the single-variable implementation. In Table 3 we include a new column labeled “Group size.” This column contains the group size we chose for each problem for the optimality phase.

In the group multicoordination, the feasibility phase was implemented using single-variable coordinators. As shown in Table 2, a feasible point was obtained quite efficiently using single-variable coordinators.

As the size of the coordinators increases, the time required to solve each coordinator problem increases significantly (recall that the coordinator problems are nonlinear). Larger coordinator problems incorporate more search directions in a coordination step and hence require a smaller number of iterations to converge. This is shown in the first column of Table 3. The question is whether a gain can be made from this trade-off.

Table 3 shows that for problems smaller than pds.30, group multicoordination is less efficient than single-variable coordination. However, for the larger PDS problems (e.g., pds.30 and pds.40) we can save time (26% to 27% speed-up) if the appropriate group size is chosen. The best size in these cases is 5, smaller than the full size coordinator of 11 used in Schultz–Meyer. Figure 4 plots the solution times for the large PDS problems using both single-variable and group multicoordination.

TABLE 3
Comparison of single-variable and group multicoordination.

Problem	Opt	Sub time	Coor time	Total	Group size	Comm
pds.10	23	29	6	45	1	20
pds.10	15	20	43	102	3	37
pds.20	23	192	13	253	1	17
pds.20	15	141	135	355	3	21
pds.30	20	754	34	902	1	11
pds.30	10	447	182	800	3	20
pds.30	8	308	191	654	5	22
pds.30	6	484	647	1273	7	11
pds.30	5	569	647	1359	11	12
pds.40	20	1404	46	1806	1	17
pds.40	10	1124	189	1704	3	21
pds.40	8	897	199	1434	5	21
pds.40	6	1009	666	1980	7	14
pds.40	5	1254	926	2370	11	7

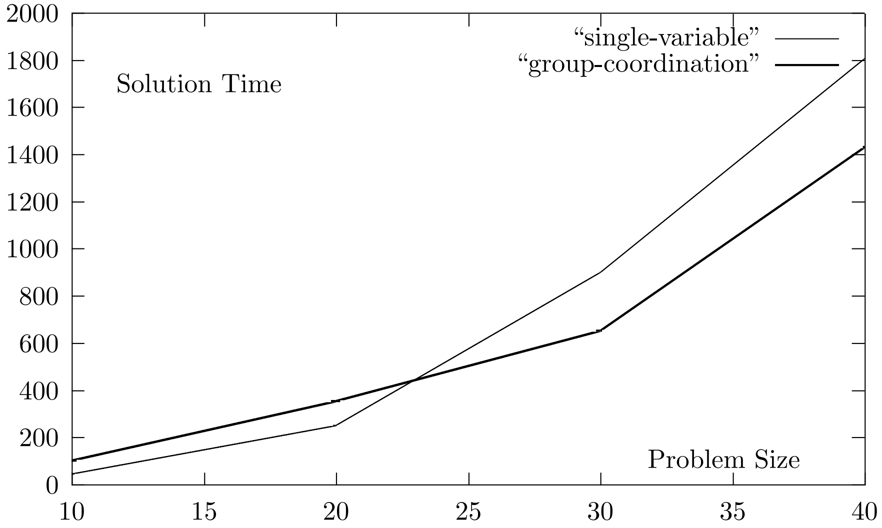


FIG. 4. *Single-variable multicoordination vs. group multicoordination.*

3.4.3. Comparison. In Table 4 we compare the best of our results (column "MC") with other reported results on the same set of problems. The first column gives the timing results obtained by Schultz and Meyer. They implemented their method on the Sequent Symmetry machine, for which each processor is 5–10 times slower than the nodes of the CM-5. Results obtained by Zenios and Pinar (column "ZP") were implemented on a Cray Y-MP with eight processors using the vector units; hence, the processors are 2–4 times faster than the nodes of the CM-5. Grigoriadis and Khachiyan [6] implemented their algorithm on an IBM RS/6000-550 workstation, which is 3 times faster than a CM-5 node. McBride and Mamer [8] implemented their algorithm on an HP 730 workstation, which is also 3 times as fast as a node of the CM-5.

Table 4
Time comparison with other solution methods.

Problem	SM	ZP	GK	MM	MC
pds.10	1711	408	123	40	45
pds.20	7920	1947	372	427	253
pds.30	19380	7504	756	838	654
pds.40	37620	–	1448	4517	1434

3.4.4. The MNETGEN problems. Another set of problems we considered were those produced by MNETGEN [2], a multicommodity network flow generator derived from NETGEN [7]. We discovered that the problems produced by this generator contain some coupling constraints that hold as equations for any feasible point. Therefore, there is no interior for the coupling constraints. Hence we perturbed the right-hand side of the coupling constraints by 1 (this perturbation is 0.06% to 0.25% of the original right-hand side). We anticipated difficulties with these problems because of their random nature and lack of interior. We calculated the optimal values to four digits of accuracy. The largest problem we solved in this test set was the 200.8 problem with 8 blocks, 200 nodes per block, and 449 arcs per block as well as 277 coupling constraints. The most efficient method used group multicoordination with an intermediate group size of 7, which required about 140 seconds. The overall size of this problem is comparable to the smaller PDS problems, so the performance of the method is not as good on this randomly generated set as it is on the real-world PDS problems.

4. Conclusion.

We have developed and tested several multicoordination schemes for the solution of convex BAPs. These multicoordination schemes are highly parallelizeable. We presented numerical results that show the efficiency of the synchronous single-variable and group multicoordination schemes. The results demonstrate significant improvement over the Schultz–Meyer predecessor and are at least comparable with the best of other solution methods.

Acknowledgments. The authors would like thank Michael Saunders and the referee for their helpful comments.

REFERENCES

[1] *CM5 Technical Summary*, Tech. Report, Thinking Machines Corporation, 1991.

[2] A. Ali and J. Kennington, *MNETGEN Program Documentation*, Tech. Report IEOR 77003, Department of Industrial Engineering and Operations Research, Southern Methodist University, Dallas, TX, 1977.

[3] G. Dantzig and P. Wolfe, *Decomposition principle for linear programs*, Oper. Res., 8 (1960), pp. 101–111.

[4] R. De Leone, R. Meyer, S. Kontogiorgis, A. Zakarian, and G. Zakeri, *Coordination in coarse-grained decomposition*, SIAM J. Optim., 4 (1994), pp. 777–793.

[5] A. Fiacco and G. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley, New York, 1968.

[6] M. Grigoriadis and L. Khachiyan, *An Exponential-Function Reduction Method for Block-Angular Convex Programs*, Tech. Report 211, Laboratory for Computer Sciences Research, Computer Sciences Department, Rutgers University, New Brunswick, NJ, May 1993.

[7] D. Klingman, A. Napier, and J. Stutz, *NETGEN—A program for generation of large-scale (un)capacitated assignment, transportation and minimum cost network problems*, Management Sci., 20 (1974), pp. 814–822.

[8] R. D. McBride and J. W. Mamer, *Solving multicommodity flow problems with a primal embedded network simplex algorithm*, INFORMS J. Comput., 9 (1997), pp. 154–163.

[9] B. Murtagh and M. Saunders, *MINOS* 5.4 *Release Notes*, *Appendix to MINOS* 5.1 *User's Guide*, Tech. Report, Stanford University, Stanford, CA, 1992.

[10] M. Pinar and S. Zenios, *Parallel decomposition of multicommodity network flows using a linear-quadratic penalty algorithm*, ORSA J. Comput., 4 (1992), pp. 235–249.

[11] G. Schultz, *Barrier Decomposition for the Parallel Optimization of Block-Angular Programs*, Ph.D. thesis, University of Wisconsin–Madison, Madison, WI, 1991.

[12] G. Schultz and R. Meyer, *An interior point method for block angular optimization*, SIAM J. Optim., 1 (1991), pp. 583–602.

[13] A. Zakarian, *private communication,* 1994.

[14] G. Zakeri, *Multi-Coordination Methods for Parallel Solution of Block-Angular Programs*, Tech. Report 95-08, Computer Sciences Department, University of Wisconsin–Madison, Madison, WI, 1995.

[15] G. Zakeri and R. R. Meyer, *Block Cyclic Multicoordination Schemes for Block-Angular Programs*, in preparation.

[16] G. Zakeri, A. B. Philpott, and D. M. Ryan, *Inexact cuts in Benders decomposition*, SIAM J. Optim., to appear.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 132–154

EFFICIENT IMPLEMENTATION OF THE TRUNCATED-NEWTON ALGORITHM FOR LARGE-SCALE CHEMISTRY APPLICATIONS*

DEXUAN XIE† AND TAMAR SCHLICK†

Abstract. To efficiently implement the truncated-Newton (TN) optimization method for large-scale, highly nonlinear functions in chemistry, an unconventional modified Cholesky (UMC) factorization is proposed to avoid large modifications to a problem-derived preconditioner, used in the inner loop in approximating the TN search vector at each step. The main motivation is to reduce the computational time of the overall method: large changes in standard modified Cholesky factorizations are found to increase the number of total iterations, as well as computational time, significantly. Since the UMC may generate an indefinite, rather than a positive definite, effective preconditioner, we prove that directions of descent still result. Hence, convergence to a local minimum can be shown, as in classic TN methods, for our UMC-based algorithm. Our incorporation of the UMC also requires changes in the TN inner loop regarding the negative-curvature test (which we replace by a descent direction test) and the choice of exit directions. Numerical experiments demonstrate that the unconventional use of an indefinite preconditioner works much better than the minimizer without preconditioning or other minimizers available in the molecular mechanics package CHARMM. Good performance of the resulting TN method for large potential energy problems is also shown with respect to the limited-memory BFGS method, tested both with and without preconditioning.

Key words. truncated-Newton method, indefinite preconditioner, molecular potential minimization, descent direction, modified Cholesky factorization, unconventional modified Cholesky factorization

AMS subject classifications. 65K10 92E10

PII. S1052623497313642

1. Introduction.

Optimization of highly nonlinear objective functions is an important task in biomolecular simulations. In these chemical applications, the energy of a large molecular system—such as a protein or a nucleic acid, often surrounded by water molecules—must be minimized to find a favorable configuration of the atoms in space. Finding this geometry is a prerequisite to further studies with molecular dynamics simulations or global optimization procedures, for example. An important feature of the potential energy function is its ill conditioning; function evaluations are also expensive, and the Hessian is typically dense. Moreover, a minimum-energy configuration corresponds to a fairly accurate local optimum. Since thousands of atoms are involved as independent variables and, often, the starting coordinates may be far away from a local minimum, this optimization task is formidable and is attracting an increasing number of numerical analysts in this quest, especially for global optimization (see [17, 18], for example).

The practical requirements that chemists and biophysicists face are somewhat different from those of the typical numerical analyst who develops a new algorithm. The computational chemists seek reliable algorithms that produce answers quickly, with as little tinkering of parameters and options as possible. Thus, theoretical performance

*Received by the editors September 9, 1997; accepted for publication (in revised form) July 3, 1998; published electronically October 20, 1999. This work was supported in part by the National Science Foundation through award number ASC-9318159. The second author is an investigator of the Howard Hughes Medical Institute.

http://www.siam.org/journals/siopt/10-1/31364.html

†Departments of Chemistry, Mathematics, and Computer Science, Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, NY 10012 (dexuan@cims.nyu.edu, schlick@nyu.edu).

is not as important as practical behavior, and CPU time is of the utmost importance. A prominent example is the current preference in the biomolecular community for Ewald summation techniques (for periodic systems) over fast-multipole approaches for evaluating the long-range forces in molecular simulations; the latter have smaller complexity in theory ($O(n)$, where n is the system size, rather than the $O(n \log n)$ associated with Ewald), but the Ewald procedure is easy to program and is very fast in practice for a range of molecular sizes.

This paper focuses on implementation details of a truncated-Newton (TN) method that are important in practice for performance efficiency in large-scale potential-energy minimization problems. The algorithmic variations we discuss are motivated by optimization theory but depart from standard notions (e.g., of a positive-definite preconditioner) for the sake of efficiency. Algorithmic stability and convergence properties are still retained in theory, as in the traditional approach, but performance in practice is enhanced by the proposed modifications.

Our interest in such chemistry applications first led to the development of a TN method adapted to potential-energy functions [25]. Our TN package, TNPACK [23, 24], was then adapted [5] for the widely used molecular mechanics and dynamics program CHARMM [1].

In TN methods, the classic Newton equation at step k,

$$H(X^k)P = -g(X^k), \tag{1}$$

where g and H are the gradient and Hessian, respectively, of the objective function E at X^k, is solved iteratively and approximately for the search vector P [4]. The linear conjugate gradient (CG) method is a suitable choice for this solution process for large-scale problems, and preconditioning is necessary to accelerate convergence. A main ingredient of TNPACK is the use of an application-tailored preconditioner M_k. This matrix is a sparse approximation to $H_k \equiv H(X^k)$, formulated at each outer minimization step k. The preconditioner in chemical applications is constructed naturally from the local chemical interactions: bond length, bond angle, and torsional potentials [25]. These terms often contain the elements of largest magnitude and lead to a sparse matrix structure which remains constant (in topology) throughout the minimization process [5]. Since M_k may not be positive definite, our initial implementation applied the modified Cholesky (MC) factorization of Gill and Murray [7] to solve the linear system $M_k z = r$ at each step of PCG (preconditioned CG). Thus, an effective positive-definite preconditioner, $\widetilde{M_k}$, results.

Why is a TN scheme a competitive approach? First, analytic second-derivative information is available in most molecular modeling packages and should be used to improve minimization performance. That is, curvature information can guide the search better toward low-energy regions. Second, the basic idea of not solving the Newton equations exactly for the search vector when far away from a minimum region saves unnecessary work and accelerates the path toward a solution. Third, the iterative TN scheme can be tailored to the application in many ways: handling of the truncated inner loop, application of a preconditioner, incorporating desired accuracy, and so on. These implementation details are crucial to realized performance in practice.

In our previous studies, we have discussed alternative minimization approaches to TN [5, 23, 24, 25]. We showed that modified Newton methods are computationally too expensive to be feasible for large systems [23, 24, 25] since the large Hessian of potential energy function is dense and highly indefinite. Nonlinear CG methods can take excessively long times to reach a solution [5]; this is not only because of the known properties of these methods but also due to the expense of evaluating the objective function at each step, a cost that dominates the CPU time [15]. A competitive approach to TN, however, is the limited-memory BFGS algorithm (LM-BFGS) [11], which also uses curvature information to guide the search. A study by Nash and Nocedal [15] comparing the performance of a discrete TN method[1] to LM-BFGS found both schemes to be effective for large-scale nonlinear problems. They suggested that the former performs better for nearly quadratic functions and also may perform poorly on problems associated with ill-conditioned Hessians. However, as Nash and Nocedal point out, since TN almost always requires fewer iterations than LM-BFGS, TN would be more competitive if the work performed in the inner loop were reduced as much as possible. This is the subject of this article.

Our experiences to date in chemical applications for medium-size problems suggest that the CPU time of the TN approach can be smaller than LM-BFGS since the total number of function evaluations is reduced. Examples shown in the present work, for larger problems as well, reinforce this. Surely, both methods can be efficient tools for large-scale optimization, and superiority of one scheme over another cannot be claimed.

In this paper, we focus on an important aspect of the TN method that affects its performance profoundly: the formulation and handling of the preconditioner in the inner PCG loop that is used to approximate the search vector at each step of the method. The use of a standard modified Cholesky factorization applied to physically constructed preconditioners leads to excessively large modifications, which in turn means many function evaluations and thus a large total CPU time for the minimization method. Pivoting strategies [6, 8] can reduce the size of the modifications but not necessarily the problem condition number, and thus are not a clear solution. The problem we address here is thus a general one, associated with other modified Cholesky factorization methods [3, 6, 7, 8, 26]: how to handle large modifications to matrices that are far from positive definite. However, we address this problem only for the TN minimization context, where the solution of such a linear system is not as important as progress in the overall minimization method.

In chemistry problems, a large negative eigenvalue often corresponds to a transition or saddle point. We argue that in our special context (large-scale computational chemical problems and TN), a standard MC factorization is inappropriate. Rather, it is sufficient to require only that the preconditioner be nonsingular and often positive definite near a minimum point. This leads to the development of our simple unconventional modified Cholesky (UMC) factorization.

We present details of the resulting TN algorithm along with many practical examples that illustrate how the use of an indefinite preconditioner outperforms other variants (e.g., no preconditioning, positive-definite preconditioning) in the TN framework. We detail analysis that shows that the directions produced are still descent directions, and thus the global convergence of the method (to a local minimum) can be proven in the same way as for the "classic" TN scheme [4]. We also offer comparisons with LM-BFGS that suggest the better performance of TN for large potential energy problems.

The remainder of the paper is organized as follows. In the next section, we summarize the structure of a general descent method and describe the new PCG inner loop we develop for the TN method. In section 3, we present the UMC designed for

[1] The discrete TN method computes Hessian and vector products by finite differences of gradients.

our applications. In section 4, we present numerical experiments that demonstrate the overall performance of the modified TNPACK minimizer, along with a comparison to LM-BFGS and other minimizers available in CHARMM (a nonlinear CG and a Newton method). Conclusions are summarized in section 5. For completeness, analyses for the PCG inner loop of TN are presented in Appendix A, and the full algorithm of the TN method is described in Appendix B. The modified package also is described in [28].

2. Descent methods and the truncated Newton approach. We assume that a real-valued function $E(X)$ is twice continuously differentiable in an open set $\mathcal{D}$ of the n-dimensional vector space R^n. Descent methods for finding a local minimum of E from a given starting point generate a sequence $\{X^k\}$ in the form

$$X^{k+1} = X^k + \lambda_k P^k, \tag{2}$$

where the search direction P^k satisfies

$$g(X^k)^T P^k < 0. \tag{3}$$

(The superscript T denotes a vector or matrix transpose.) Equation (3) defines the descent direction P^k which yields function reduction. The steplength λ_k in (2) is chosen to guarantee sufficient decrease, e.g., such that [12]

$$E(X^k + \lambda_k P^k) \le E(X^k) + \alpha \lambda_k g(X^k)^T P^k \tag{4}$$

and

$$|g(X^k + \lambda_k P^k)^T P^k| \le \beta |g(X^k)^T P^k|, \tag{5}$$

where α and β are given constants satisfying $0 < \alpha < \beta < 1$.

Condition (5) is referred to as the strong Wolfe condition. A steplength λ_k satisfying (5) must satisfy the usual Wolfe condition:

$$g(X^k + \lambda_k P^k)^T P^k \ge \beta g(X^k)^T P^k.$$

According to the line search algorithm of Moré and Thuente [12] (used in TNPACK), such a steplength λ_k is guaranteed to be found in a finite number of iterations. Hence, according to the basic theory of descent methods [10], a descent method defined in the form (2) guarantees that

$$\lim_{k \to \infty} g(X^k) = 0.$$

The challenge in developing an efficient descent method is balancing the cost of constructing a descent direction P^k with performance realized in practice.

To reduce the work cost of the classic modified Newton method and develop a globally convergent descent algorithm, Dembo and Steihaug proposed a clever variation known as the truncated Newton method [4]. Since then, several variants have been developed and applied in various contexts; see, e.g., [13, 14, 15, 23, 25, 29]. The linear PCG framework is the most convenient generator of descent directions in the inner TN loop due to its efficiency and economic storage requirements for solving large positive-definite linear systems. Since the PCG method may fail at some step when the matrix H_k is indefinite, a termination strategy is required to guarantee that the resulting search directions are still descent directions. In addition, the PCG inner loop of the TN method can be made more effective by employing a *truncation test.*

We present our PCG inner loop of the TN scheme in Algorithm 1. The changes with respect to a "standard" PCG inner loop include allowing an indefinite preconditioner, the UMC factorization (discussed in the next section), and a new descent direction test.

ALGORITHM 1 (PCG inner loop k of TN for solving $H_k p = -g_k$ with a given preconditioner M_k).

Let p_j represent the jth PCG iterate, d_j the direction vector, and r_j the residual vector satisfying $r_j = -g_k - H_k p_j$. Let $\mathrm{IT}_{\mathrm{PCG}}$ denote the maximum number of allowable PCG iterations at each inner loop.

Set $p_1 = 0$, $r_1 = -g_k$, and $d_1 = z_1$, where z_1 solves a system related to $M_k z_1 = -g_k$ by UMC.

```
For $j = 1, 2, 3, \ldots$,
  1. [SINGULARITY TEST]
       If either $|r_j^T z_j| \le \delta$ or $|d_j^T H_k d_j| \le \delta$ (e.g., $\delta = 10^{-10}$),
       exit PCG loop with search direction $P^k = p_j$ (for $j = 1$, set $P^k = -g_k$).
  2. Compute $\alpha_j = r_j^T z_j / d_j^T H_k d_j$ and $p_{j+1} = p_j + \alpha_j d_j$.
  3. [DESCENT DIRECTION TEST] (replaces *negative curvature test*)
       If $g_k^T p_{j+1} \ge g_k^T p_j + \delta$,
       exit PCG loop with $P^k = p_j$ (for $j = 1$, set $P^k = -g_k$).
  4. Compute $r_{j+1} = r_j - \alpha_j H_k d_j$.
  5. [TRUNCATION TEST]
       If $\|r_{j+1}\| \le \min\{c_r/k\ ,\ \|g_k\|\} \cdot \|g_k\|$, or $j + 1 > \mathrm{IT}_{\mathrm{PCG}}$,
       exit PCG loop with $P^k = p_{j+1}$.
       (By default, $c_r = 0.5$ and $\mathrm{IT}_{\mathrm{PCG}} = 40$).
  6. Compute $\beta_j = r_{j+1}^T z_{j+1} \,/\, r_j^T z_j$, and $d_{j+1} = z_{j+1} + \beta_j d_j$,
       where $z_{j+1}$ solves a system related to $M_k z_{j+1} = r_{j+1}$ by UMC.
```

Since the effective preconditioner $\widetilde{M_k}$ generated by our UMC (see the next section) and the Hessian matrix H_k may be indefinite, it may happen that $r_j^T z_j$ or $d_j^T H_k d_j$ is exactly zero for some j.(So far, we have not encountered this in practice). Hence, to ensure that the PCG recursive formulas are well defined, *the singularity test* has been added in step 1 above.

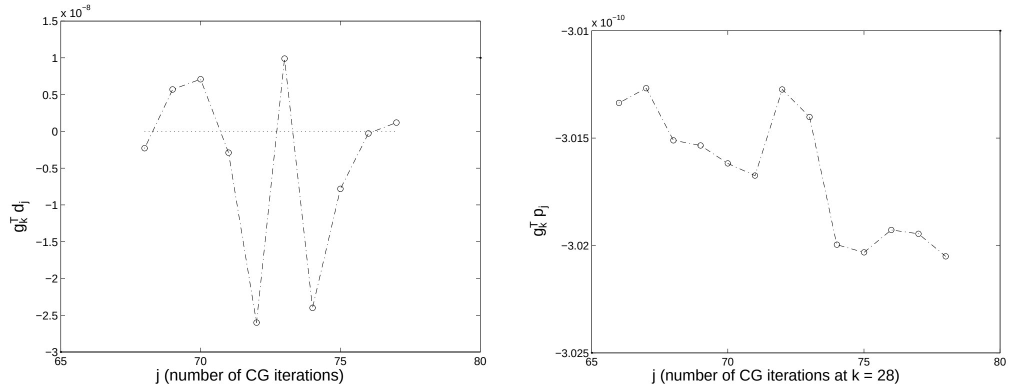
Our descent direction test (step 3) is equivalent in theory to the following *negative curvature test* [4]: if $d_j^T H_k d_j < \delta d_j^T d_j$, halt the PCG loop with exit search direction $P^k = -g_k$ if $j = 1$ or p_j if $j > 1$. We prove this equivalence in Theorem 3 of Appendix A.

In practice, however, due to computer rounding errors, the negative curvature test may not guarantee that the inner product $g_k^T p_j$ decreases monotonically, as theory predicts (see Theorem 2 in Appendix A). See also Box 1 for numerical examples. The descent direction test in step 3 halts the PCG iterative process as soon as the situation $g_k^T p_{j+1} > g_k^T p_j$ (or $|g_k^T p_{j+1}| < |g_k^T p_j|$) is encountered. We have observed better performance in practice for large-scale problems with this modification.

In the standard implementation of the negative curvature test in TN [4], P^k can be set to p_j or d_j for $j > 1$, both directions of descent. We have now removed the option in step 3 of using the auxiliary directions d_j as exit search vectors. We show in Theorem 1

Box 1. *Examples for the negative curvature test in finite precision arithmetic.*

We consider the negative curvature test implemented in TN as originally described [4] for minimizing the alanine dipeptide potential function (22 atoms, 66 Cartesian variables). For simplicity, we do not use a preconditioner (i.e., M_k is the identity matrix); results thus reflect the case of using a positive-definite preconditioner.



The left figure shows that some d_j are ascent directions even if $d_j^T H_k d_j > 0$. This is possible because the basic relation $\{g_k^T r_j\} = 0$ (see (13) in Appendix A) may deteriorate due to computer rounding errors. Namely, the inner product $g_k^T r_j$ may become positive, and thus d_j may be an ascent direction because $g_k^T d_j = g_k^T r_j + \beta_{j-1} g_k^T d_{j-1}$.

The right figure shows that the inner product $g_k^T p_j$ does not necessarily decrease monotonically due to computer rounding errors. This can be understood since, if d_j is an ascent direction for some j but not a direction of negative curvature (i.e., $d_j^T H_k d_j > 0$), then we have $\alpha_j > 0$ and $g_k^T d_j > 0$, so that $g_k^T p_{j+1} = g_k^T p_j + \alpha_j g_k^T d_j \geq g_k^T p_j$ or $g_k^T p_{j+1} > 0$. Hence, the negative curvature test may not guarantee a descent direction or even a "good" descent direction in some sense, as theory predicts for TN in finite precision arithmetic (see Theorem 2 in Appendix A).

of Appendix A that d_j may be an ascent direction when the effective preconditioner is indefinite. Even in the standard implementation (i.e., positive-definite effective preconditioner), we argue that p_j is a better choice than d_j according to Theorem 4 of Appendix A.

Although not used here, we leave the option of using the negative curvature test (with the d_j choice removed) in the general TNPACK package; see the full algorithm in Appendix B.

3. The UMC method. We discuss our motivation for developing the UMC in section 3.1 and describe the factorization in section 3.2.

3.1. Motivation. Recall that our major goal is to reduce the computational effort in the inner loop of the TN method. Therefore, we choose a preconditioner that is sparse (sparsity less than 5% for medium-size molecules) and rapid to compute. The factorization of the linear system involving M is handled efficiently within the framework of the Yale Sparse Matrix Package (YSMP) [19, 20]. YSMP routines use special

Table 1
The CPU time distribution (%) among TNPACK components.

Protein	$E, g,$ & H evals.	M evals.	Solve $Mz = r$	Hd evals.	Other tasks
BPTI	21	0.14	6.8	69	3.06
Lysozyme	22	0.08	2.5	74	1.42

pointer arrays to record data positions and manipulate only the nonzero elements. Efficiency is further enhanced by reordering M at the onset of the minimization method to minimize fill-in. This works because the structure of the preconditioner, or the connectivity structure of our molecular system, remains constant. This reordering is optional.

The main advantage of this sparsity-based factorization is efficiency. As we show in Table 1, the CPU percentage involved in solving $Mz = r$ within YSMP is less than 7% and 3% of the total TN cost of minimization for the proteins BPTI (568 atoms, 1704 Cartesian variables) and lysozyme (2030 atoms, 6090 variables), respectively. Since the Hessian H is dense and we evaluated the Hessian and vector products Hd in PCG directly (i.e., not by finite differences of gradients [23]), this part consumes the majority of the CPU time: about 70%. The finite-differencing approximation of Hd may be more competitive for large systems. In addition, function and derivative evaluations consume about 20% of the total CPU time.

A disadvantage of our approach is the absence of pivoting strategies based on numerical values of the matrix elements. Pivoting would increase the computational time but possibly lead to a lower condition number for the modification $\widetilde{M}$ of M, and a smaller error bound $\|E\|_\infty$ in the MC process. Here $E = \widetilde{M} - M$ is a nonnegative diagonal matrix. Our experiences suggest that pivoting strategies in the context of a standard MC are far less effective than our UMC in the TN context. Namely, our numerical experiments demonstrate that the Gill–Murray–Wright MC (GMW MC) with pivoting [8] can reduce the error bound $\|E\|_\infty$ but far less significantly the condition number of $\widetilde{M}$ (see Box 2). This is a consequence of our highly indefinite preconditioner near regions far away from a local minimum.

Our experiments studied three other MC algorithms: the partial Cholesky (PC) factorization described by Forsgren, Gill, and Murray [6]; the Schnabel and Eskow (SE) MC [26]; and the Cheng and Higham (CH) MC [3] (see Box 2). As the authors state, all methods can produce unacceptably large perturbations in special cases. In our application, these large perturbations typically lead to poor performance when the objective matrix M is highly indefinite; a very large condition number or a very large error bound $\|E\|_\infty$ (much larger than the magnitude of the negative minimum eigenvalue $\lambda_{min}(M)$ of M) can result.

To see this analytically, recall that the GMW MC process modifies a symmetric $n \times n$ matrix M into a positive-definite matrix $\widetilde{M}$ and factors it as $\widetilde{M} = M + E = LDL^T$, where L, D, and E are, respectively, unit lower-triangular, diagonal, and diagonal $n \times n$ matrices. The elements $e_j = d_j - \overline{d_j}$ of E are defined by

$$\overline{d_j} = m_{jj} - \sum_{k=1}^{j-1} l_{jk} c_{jk} \quad \text{and} \quad d_j = \max\left\{ |\overline{d_j}|, \delta, \frac{\theta^2}{\beta^2} \right\}, \tag{6}$$

where $c_{ij} = l_{ij} d_j$, $\theta = \max_{j+1 \le i \le n} |c_{ij}|$, and positive numbers δ and β are introduced to ensure the numerical stability (e.g., $\delta = 10^{-9}$ and $\beta = \xi/\sqrt{n^2 - 1}$, where ξ is the

Box 2. *Examples of large modifications in MC methods.*

We experimented with four MC methods in Matlab for the symmetric 42×42 matrix M constructed from the second partial derivatives of the butane potential-energy terms coming from bond length, bond angle, and torsional potentials [16]. M is indefinite with three negative eigenvalues: $-5.718, -74.703$, and -218.475, and the condition number is 1.553×10^3.

These four MC methods are due to Gill, Murray, and Wright (GMW) [8]; Schnabel and Eskow (SE) [26]; Forsgren, Gill, and Murray [6], a partial Cholesky (PC) factorization; and Cheng and Higham (CH) [3]. The Matlab M-files for GMW, SE, and CH were provided by Wright, Eskow, and Cheng, respectively. We used their default tolerance δ. In addition, we wrote corresponding files for GMW without pivoting and for PC according to [7] and [6]. Note that the error matrix E in PC and CH may not be diagonal.

Modified Cholesky factorization	$\widetilde{M}$ condition number	$\widetilde{M}$ minimum eigenvalue	Error $\|E\|_\infty$
GMW	8.42×10^6	1.70×10^{-3}	1.38×10^4
GMW, no pivoting	1.78×10^7	2.80×10^{-2}	5.13×10^5
PC	6.79×10^6	1.30×10^{-3}	4.59×10^2
SE	3.09×10^1	3.62×10^2	2.33×10^3
CH	7.28×10^9	1.22×10^{-6}	7.86×10^2

For reference, our UMC gives as a function of the control parameter τ the following results.

τ of UMC	$\widetilde{M}$ condition number	$\widetilde{M}$ minimum eigenvalue	$\|E\|_\infty$
40	546.157	−874.032	40.0
120	198.716	−98.475	120.0
200	491.541	−18.475	200.0
240	423.753	21.524	240.0
280	148.903	61.524	280.0

largest magnitude of an element of M. From (6) we see that an element e_j of E has the following expression:

$$e_j = d_j - \overline{d_j} = \begin{cases} \delta - \overline{d_j} & \text{when } \delta \geq \max\{|\overline{d_j}|, \frac{\theta^2}{\beta^2}\}, \\ \frac{\theta^2}{\beta^2} - \overline{d_j} & \text{when } \frac{\theta^2}{\beta^2} \geq \max\{|\overline{d_j}|, \delta\}, \\ |\overline{d_j}| - \bar{d}_j & \text{when } |\overline{d_j}| \geq \max\{\frac{\theta^2}{\beta^2}, \delta\}. \end{cases} \tag{7}$$

For a negative $\overline{d_j}$, e_j may thus be $2|\overline{d_j}|$ or $\frac{\theta^2}{\beta^2} + |\overline{d_j}|$. If $\overline{d_j}$ is a large negative number or $\frac{\theta^2}{\beta^2}$ a large positive number, $\|E\|_\infty$ may be much larger than the value of $|\lambda_{min}(M)|$.

The second author has studied performance of the GMW [8] versus the SE MC [26] for difficult computational chemistry problems in the context of TN [22]. That study showed that no factorization is clearly superior to any other. We have retained the former in TNPACK since it is simple to implement in the context of YSMP. The CH MC [3] is another possibility worth examining in our context since it is easily implemented in existing software. Still, Box 2 suggests that all MC algorithms may exhibit poor performance for a highly indefinite matrix M.

In our TN applications, while we find that pivoting strategies can improve the performance of MC and even reduce the total number of outer (Newton) iterations,

Table 2

Performance of TNPACK based on the GMW MC with pivoting (GMW_P) and without pivoting (GMW) and our UMC with $\tau = 10$.

<table>
<tr><th colspan="6">Butane (42 variables)</th></tr>
<tr><th>MC</th><th>Final E</th><th>Final $\|g\|$</th><th>Outer (inner) Iter.</th><th>E &amp; g evals.</th><th>CPU time</th></tr>
<tr><td>GMW_P</td><td>4.7531</td><td>3.57×10^{-10}</td><td>35 (1329)</td><td>44</td><td>6.3 sec.</td></tr>
<tr><td>GMW</td><td>3.9039</td><td>2.07×10^{-8}</td><td>52 (78)</td><td>93</td><td>0.28</td></tr>
<tr><td>UMC</td><td>3.9039</td><td>1.12×10^{-8}</td><td>21 (62)</td><td>26</td><td>0.17</td></tr>
<tr><td colspan="6">Alanine dipeptide (66 variables)</td></tr>
<tr><td>GMW_P</td><td>-15.245</td><td>1.45×10^{-8}</td><td>31 (5641)</td><td>39</td><td>78</td></tr>
<tr><td>GMW</td><td>-15.245</td><td>1.85×10^{-11}</td><td>6387 (7108)</td><td>15801</td><td>44</td></tr>
<tr><td>UMC</td><td>-15.245</td><td>3.65×10^{-9}</td><td>27 (186)</td><td>39</td><td>1.3</td></tr>
</table>

the total number of inner (PCG) iterations may increase significantly. This may result from the large modification made to M. Consequently, the CPU time of TN is large even when pivoting is used in standard MC schemes. See Table 2 for examples on two small molecular systems. Note that pivoting strategies (for example, Bunch–Kaufman [2] and that used in [3]) require at least $O(n)$ comparisons as well as substantial data movement.

The objective of allowing an indefinite preconditioner in the context of TN is to produce an efficient preconditioner for the inner loop, that is, one that leads to the smallest number of PCG iterations. The original indefinite M_k is a good approximation to H_k, so we do not want to make excessively large (and perhaps artificial) perturbations, as often required by standard MC methods we have experimented with. Since the PCG with an indefinite preconditioner can still generate directions of descent (Theorem 2 of Appendix A), using the UMC to solve the linear system involving M_k in the context of YSMP is one feasible efficient strategy.

In formulating the UMC, we were also guided by the observation that the Hessian matrix itself in our applications is often positive definite near a solution (minimum energy). This led us to construct preconditioners that also exhibit this trend. This can be accomplished by adding a constant matrix τI to M_k, where τ is a problem-size independent small positive number found by experimentation (e.g., $\tau = 10$).

Intuitively, the UMC can be interpreted as follows. When $\tau > |\lambda_{min}(M_k)|$, $M_k + \tau I$ is positive definite and has the standard (stable) LDL^T factorization. To ensure a numerically stable factorization when $M_k + \tau I$ is indefinite, we modify it further by adding a diagonal matrix as in GMW, so as to impose an upper bound on the factors L and D. The difference in our treatment from the standard GMW MC is that our diagonal candidates can be negative (the third situation in (9) below), and thus the resulting UMC matrix may still be indefinite. Certainly, other procedures for solving linear systems involving indefinite matrices exist, but the simple UMC strategy above is most easily incorporated into our current software and is found to work well.

3.2. The UMC factorization. Our UMC effectively applies a standard LDL^T factorization for matrix $M + \tau I$ for a given nonnegative number τ. The simple approach of adding a multiple of the identity matrix to the indefinite matrix has been discussed in Dennis and Schnabel [10]; however, the scalar τ is chosen to make $\widetilde{M}$ safely positive definite on the basis of a diagonal dominance estimate and thus can be much larger than necessary. Our approach effectively sets τ to be a small nonnegative number like 10 (through numerical experiments) that ensures that $M_k + \tau I$ is positive definite at the final steps of the TN minimization process. At other steps, $M_k + \tau I$

may be indefinite, but the modification to the original M is relatively small, and this produces faster convergence overall.

Since $M + \tau I$ may not be positive definite, a similar strategy to the standard GMW strategy [7] (i.e., the use of two bound parameters δ and β in (8) and the dependence of the entries d_j of the factor D on the elements of M as shown in (9)) is employed in UMC to guarantee numerical stability. The following scheme describes our numerically stable process for factoring a symmetric matrix M with small perturbations, with the resultant matrix not necessarily positive definite.

In the jth step of the UMC factorization, suppose that the first $j-1$ columns have been computed, and satisfy

$$|d_k| > \delta, \quad \text{and} \quad |l_{ik}|\sqrt{|d_k|} \le \beta, \quad i > k, \tag{8}$$

for $k = 1, 2, \ldots, j-1$. Here δ is a small positive number used to avoid numerical difficulties when $|d_k|$ is too small and β is a positive number satisfying $\beta^2 = \xi/\sqrt{n(n-1)}$, where ξ is the largest magnitude of an element of M.

We define

$$\overline{d_j} = m_{jj} - \sum_{k=1}^{j-1} l_{jk}c_{jk} \quad \text{and} \quad \theta = \max_{j+1\le i\le n} |c_{ij}|,$$

where $c_{ij} = l_{ij}d_j$ is computed by using

$$c_{ij} = m_{ij} - \sum_{k=1}^{j-1} l_{jk}c_{ik}, \qquad i = j+1, \ldots, n.$$

We then set $\widetilde{d_j} = \overline{d_j} + \tau$ and define

$$d_j = \begin{cases} \max\{\widetilde{d_j}, \frac{\theta^2}{\beta^2}\} & \text{when } \widetilde{d_j} > \delta, \\ \delta & \text{when } |\widetilde{d_j}| \le \delta, \\ \min\{\widetilde{d_j}, -\frac{\theta^2}{\beta^2}\} & \text{when } \widetilde{d_j} < -\delta, \end{cases} \tag{9}$$

where δ and β are given in (8). Note that the above d_j is negative when the third possibility in (9) occurs, resulting in an indefinite matrix.

This definition of d_j implies by induction that the relation (8) holds for all $k = 1, 2, \ldots, n$, and hence the factorization is numerically stable.

The effective $\widetilde{M}$ produced by our UMC satisfies

$$\widetilde{M} = LDL^T = M + E,$$

where E is a diagonal matrix. In particular, $\widetilde{M}$ becomes positive definite with $E = \tau I$ if $\tau > |\lambda_{min}(M)|$; otherwise, the jth element e_j of E can be expressed in the form

$$e_j = d_j - \overline{d_j} = \begin{cases} \tau & \text{when } |\overline{d_j} + \tau| > \max\{\delta, \frac{\theta^2}{\beta^2}\}, \\ \delta - \overline{d_j} & \text{when } \delta \ge \max\{|\overline{d_j} + \tau|, \frac{\theta^2}{\beta^2}\}, \\ \frac{\theta^2}{\beta^2} - \overline{d_j} & \text{when } \frac{\theta^2}{\beta^2} \ge \overline{d_j} + \tau > \delta, \\ -\frac{\theta^2}{\beta^2} - \overline{d_j} & \text{when } -\delta > \overline{d_j} + \tau > -\frac{\theta^2}{\beta^2}, \end{cases} \tag{10}$$

for $j = 1, 2, \ldots, n$. By arguments similar to those used in [7], it can be shown that

$$|e_j| \le \frac{\theta^2}{\beta^2} + |\overline{d_j}| + \delta + \tau,$$

along with $|\overline{d_j}| < \gamma + (n-1)\beta^2$ and $\theta \le \xi + (n-1)\beta^2$, where

$$\gamma = \max_{1\le i\le n} |m_{ii}| \qquad \text{and} \qquad \xi = \max_{1\le j\le n}\ \max_{j+1\le i\le n} |m_{ij}|.$$

Therefore, a worst-case bound of $\|E\|_\infty$ is obtained:

$$\|E\|_\infty \le \left[\frac{\xi}{\beta} + (n-1)\beta\right]^2 + \gamma + (n-1)\beta^2 + \tau + \delta \tag{11}$$

for a τ satisfying $\tau \le |\lambda_{min}(M)|$.

If we denote the above upper bound of $\|E\|_\infty$ as a function $\phi(\beta)$, it can be shown that $\phi(\beta)$ has a minimum at $\beta^2 = \xi/\sqrt{n(n-1)}$, an a priori choice of β in our UMC method.

The upper bound of $\|E\|_\infty$ in (11) is similar to that for GMW [7]. Hence, like the GMW factorization, our UMC can lead to large perturbations when $\tau \le |\lambda_{min}(M)|$. In our numerical experiments, we rarely observe this; instead, we often have $\|E\|_\infty = \tau$ even when $\tau \le |\lambda_{min}(M)|$ (see Figure 4, for example). Note that a large τ satisfying $\tau > |\lambda_{min}(M)|$ reduces UMC to the standard Cholesky factorization.

To avoid perturbing a positive-definite matrix, our algorithm can be divided into two phases (in the spirit of the SE MC [26]). We first apply the standard LDL^T factorization to matrix M, stopping at the first occasion that a diagonal element d_j of D becomes negative or very small. We then switch to the second phase, where the modified matrix $M + \tau I$ is applied.

The performance of our UMC on the 42×42 indefinite matrix M is shown in Box 2, following results of other factorizations. Clearly, as τ increases, $\widetilde{M}$ approaches positive definiteness. This is accompanied by a monotonic reduction of the condition number of $\widetilde{M}$. The error bound $\|E\|_\infty$ is equal to τ.

4. Numerical results. We consider four molecular systems for our tests: butane, alanine dipeptide, BPTI, and lysozyme. Butane is a small, 14-atom hydrocarbon molecule with the chemical formula C_4H_{10}. Alanine dipeptide, a blocked alanine residue, consists of 22 atoms. The 58-residue protein BPTI has 568 atoms and thus 1704 Cartesian variables. It is considered "small" by computational chemists. The larger protein lysozyme has 130 residues, 2030 atoms, and 6090 variables.

All computations were performed in double precision in serial mode on an SGI Power Challenge L computer with R10000 processors of speed 195 MHz at New York University. Parameter files were used from CHARMM version 19, but the TNPACK code was implemented into CHARMM version 23 with default TNPACK parameters of [23], unless otherwise stated. These parameters are also listed in the TN algorithm of the appendix. No cutoffs were used for the nonbonded interactions of the potential energy function, and a distance-dependent dielectric function was used. The vector norm $\|\cdot\|$ in all tables and figures is the standard Euclidean norm divided by $\sqrt{n}$, where n is the number of independent variables of a potential energy function. The inner loop of TN is followed as outlined in Algorithm 1, with $\tau = 10$ and $\mathrm{IT}_{\mathrm{PCG}} = 40$ unless otherwise stated.

4.1. No preconditioning vs. indefinite preconditioning. Figure 1 shows that our TN based on UMC uses far fewer outer iterations than the minimizer without preconditioning for BPTI. We experimented with both $\tau = 0$ and also $\tau = 10$ for the UMC. Since all $\{M_k\}$ were indefinite throughout the TN process, the preconditioner $\widetilde{M_k}$ used by TN was indefinite.

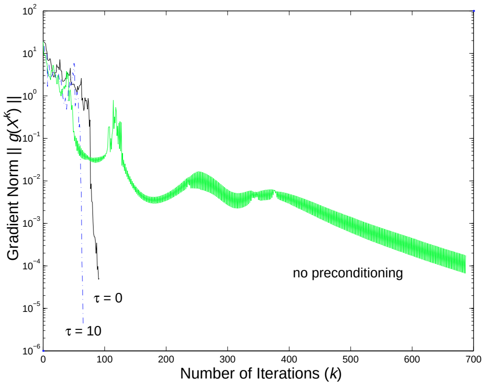


FIG. 1. *TN based on PCG with an indefinite preconditioner performs much better than without preconditioning (even when $\tau = 0$ in UMC) for the minimization of the BPTI potential function.*

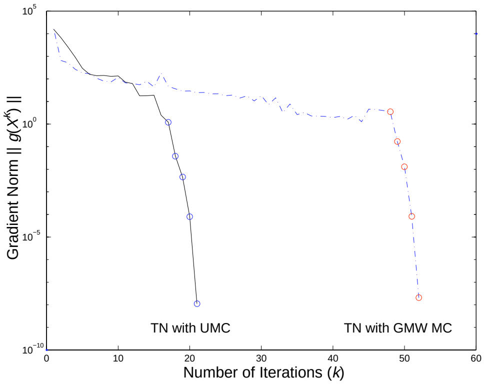


FIG. 2. *The gradient norms generated by TN based on GMW MC vs. UMC for butane minimization. Here circular markers indicate values at the last few steps.*

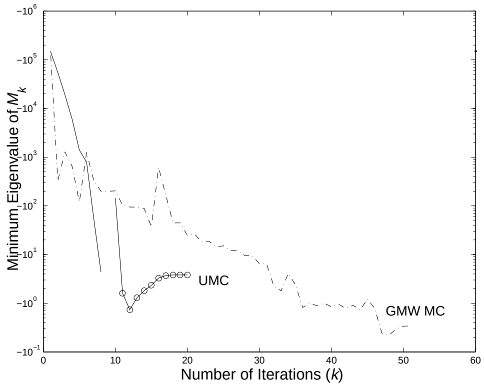


FIG. 3. *The minimum eigenvalue of $\{M_k\}$ resulting from GMW MC vs. UMC for the minimization of a butane molecular system. Circular marks indicate that the modified matrices $\{\widetilde{M_k}\}$ are positive definite. For $k = 9$, M_k is also positive definite for UMC.*

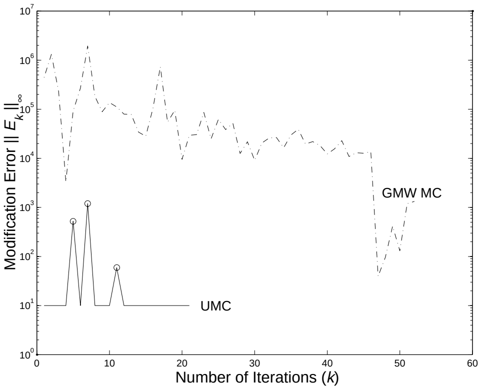


FIG. 4. *The error norms $\{\|E_k\|_\infty\}$ ($E_k = \widetilde{M_k} - M_k$) generated by GMW MC vs. UMC for butane minimization. With $\tau = 10$ for our UMC, $\|E_k\|_\infty = 10$ for all k except the three points indicated by circles.*

Even better, the total CPU time is much smaller for the indefinite preconditioner version. Namely, the indefinite preconditioner variant required only 8 minutes (for 92 TN iterations and a total of 2390 inner PCG iterations) to find a local minimizer. In contrast, without preconditioning, 80 minutes were required for 687 TN iterations and 27347 CG iterations. This behavior is typical for the molecular systems examined.

4.2. Standard MC vs. our UMC. We next compare the performance of TNPACK based on GMW without pivoting [7] and our UMC for butane minimization. Pivoting in GMW was discussed in section 3.1; see Table 2. Efficiency argues for sparsity-based factorization in our context. We further compare our UMC vs. GMW in Figures 2, 3, and 4 for the minimization of the butane potential function.

Table 3

Performance of TNPACK on BPTI based on PCG with an indefinite preconditioner at different values of $\mathrm{IT}_{\mathrm{PCG}}$, the maximum number of allowable PCG iterations at each inner loop. For $\mathrm{IT}_{\mathrm{PCG}} = 300$, the truncation test was used throughout the TN process.

$\mathrm{IT}_{\mathrm{PCG}}$	Final energy	Final $\|g\|$	TN outer loops	Total PCG iterations	CPU time (min.)	
					Total	PCG
2	-2780.47	5.9×10^{-4}	1402	2801	27.24	9.81
5	-2769.33	6.8×10^{-5}	233	1139	6.54	3.49
10	-2755.07	5.7×10^{-5}	92	778	3.65	2.31
20	-2756.40	3.2×10^{-5}	73	1114	4.32	3.24
40	-2769.25	2.1×10^{-5}	71	1456	5.13	4.07
120	-2769.25	1.4×10^{-6}	72	2221	7.32	6.25
200	-2775.14	1.0×10^{-6}	143	4640	15.26	13.13
250	-2775.14	1.3×10^{-6}	151	5937	18.86	16.62
300	-2775.14	3.8×10^{-6}	150	6049	18.96	16.74

Figure 2 shows that TN based on the UMC strategy performs favorably in terms of Newton iterations. It also requires less CPU time (0.17 vs. 0.28 sec.; see Table 2). Further, it has a quadratic convergence rate at the last few iterations, as shown by the circles in the figure.

Figures 3 and 4 plot the minimum eigenvalues of $\{M_k\}$ and the values of $\{\|E_k\|_\infty\}$, respectively, where $E_k = \widetilde{M_k} - M_k$. The UMC leads to much smaller modifications of $\{M_k\}$ than the standard MC. Since the minimum eigenvalue of M_k is less than 10 for $k \geq 12$ (circles in Figure 3), our effective preconditioner $\widetilde{M_k}$ is positive definite for $k \geq 12$ with $\|E_k\|_\infty = 10$.

4.3. The importance of the maximum limit on PCG iterations. Table 3 illustrates how TN performs with different values of $\mathrm{IT}_{\mathrm{PCG}}$ (see Algorithm 1) for BPTI minimization. With $\mathrm{IT}_{\mathrm{PCG}} = 300$ (last row), the truncation test (step 5 of Algorithm 1) was satisfied throughout the TN process. These results can also be visualized in Figures 5 and 6, which show the CPU time and the total number of TN iterations as functions of $\mathrm{IT}_{\mathrm{PCG}}$, respectively. The evolution of the gradient norm from TN minimization, corresponding to $\mathrm{IT}_{\mathrm{PCG}} = 40$ (leading to the fewest outer iterations) and 300, as a function of the number of TN iterations, is shown in Figure 7. Note the quadratic convergence in the last few steps.

There are several interesting observations from the data of Table 3. As Figure 5 shows, an optimal value for $\mathrm{IT}_{\mathrm{PCG}}$ can be associated with the smallest CPU time. Here, about 4 minutes resulted from $\mathrm{IT}_{\mathrm{PCG}} = 10$, much less than about 19 minutes required when $\mathrm{IT}_{\mathrm{PCG}} = 300$.

Figure 6, however, shows that a somewhat larger value of $\mathrm{IT}_{\mathrm{PCG}}$ (namely 40) leads to a minimal value of the total number of TN iterations, 71. In contrast, the $\mathrm{IT}_{\mathrm{PCG}}$ value for optimal CPU time (namely 10) is associated with 92 TN iterations. For reference, a small value, $\mathrm{IT}_{\mathrm{PCG}} = 2$, gives 1402 TN iterations, and a very large $\mathrm{IT}_{\mathrm{PCG}}$ gives 150.

In terms of the final energy value obtained for the different variants, we clearly see that several local minima are reached by varying the minimization procedure (six different energy values noted for the nine runs). This multiple-minima problem is beyond the scope of this work. However, we suggest that a larger $\mathrm{IT}_{\mathrm{PCG}}$ value might be preferred over a lower one (within a small optimal range) in an attempt to reach lower energy values.

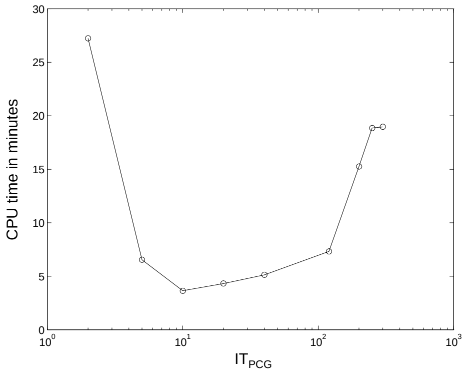


Fig. 5. *An optimal choice for the maximum number of allowable PCG iterations (per TN inner loop) in terms of total CPU time can be seen when TN uses an indefinite preconditioner.*

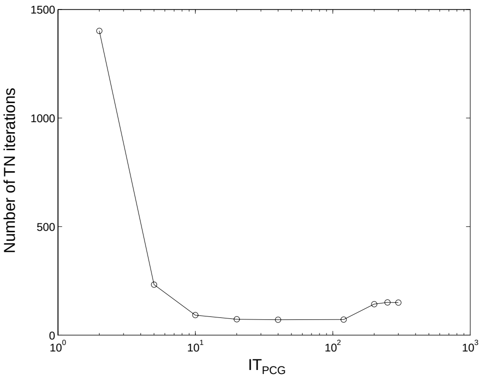


Fig. 6. *An optimal choice for the maximum number of allowable PCG iterations (per TN inner loop) in terms of total number of TN iterations can be seen when TN uses an indefinite preconditioner.*

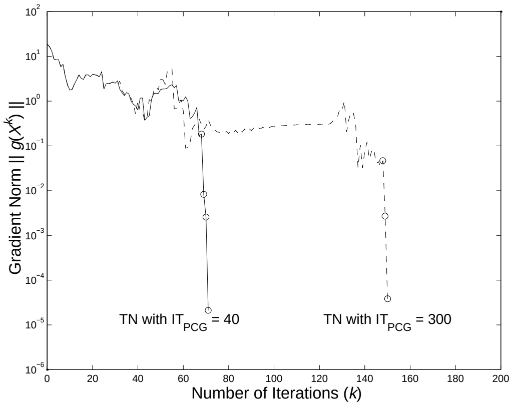


Fig. 7. *TN with a maximum allowable number of* 40 *PCG iterations per inner loop can more efficiently find a local minimum than when* IT_{PCG} *is* 300. *Similar convergence rates are still seen in both paths, as indicated by the circular markers at the last few iterations.*

Figures 5 and 6 also show that the range of IT_{PCG} for which TN performs better than the minimizer based on the truncation test alone is fairly large, here between 10 to 120. Based on this observation and the point above regarding larger IT_{PCG} for smaller final energy values, we set $\text{IT}_{\text{PCG}} = 40$ for the numerical experiments presented below.

4.4. Comparison to other minimization schemes. We now compare, in Table 4, the minimization performance of TNPACK with two other CHARMM minimizers, ABNR (an adopted basis Newton–Raphson method) and CONJ (a nonlinear conjugate gradient method), as well as with LM-BFGS, with $u = 5$ stored updates [11]. For LM-BFGS we test no-preconditioning as well as preconditioning options. The preconditioning strategy used for LM-BFGS was described by Schlick [21]. Briefly, the initial search vector in each sequence of LM-BFGS updates is set as the solution

Table 4

Comparison of TNPACK with two other CHARMM minimizers and LM-BFGS.

Butane (42 variables)					
Minimizer	Iterations	Final E	Final $\|g\|$	E & g evals.	CPU time
TN	21 (62)*	3.904	1.1×10^{-8}	26	0.17 sec.
LM-BFGS	93	3.904	7.7×10^{-7}	101	0.14
LM-BFGS (P)	133	4.753	9.4×10^{-7}	148	0.33
ABNR	368	3.904	9.8×10^{-8}	368	0.32
CONJ	127	3.904	9.1×10^{-7}	307	0.17
Alanine dipeptide (66 variables)					
TN	29 (210)	−15.25	7.67×10^{-11}	44	1.12 sec.
LM-BFGS	711	−15.25	1.4×10^{-6}	740	1.39
LM-BFGS (P)	367	−15.25	1.3×10^{-6}	378	1.97
ABNR	16466	−15.25	9.9×10^{-8}	16467	7.47
CONJ	882	−15.25	9.83×10^{-7}	2507	2.34
BPTI (1704 variables)					
TN	65 (1335)	−2773.70	4.2×10^{-6}	240	5.21 min.
LM-BFGS	4486	−2792.96	6.3×10^{-5}	4622	12.61
LM-BFGS (P)	3929	−2792.92	5.9×10^{-5}	3946	64.2
ABNR	8329	−2792.96	8.9×10^{-6}	8330	25.17
CONJ	12469	−2792.93	9.9×10^{-6}	32661	97.8
Lysozyme (6090 variables)					
TN	79 (1841)	−4631.38	3.7×10^{-6}	244	1.54 hrs.
LM-BFGS	5546	−4617.21	1.4×10^{-4}	5711	4.26
LM-BFGS (P)	3331	−4620.27	1.6×10^{-4}	3374	12.92
ABNR	7637	−4605.94	9.9×10^{-6}	7638	6.11
CONJ	9231	−4628.36	9.9×10^{-5}	24064	19.63
* The number in parentheses is the total number of PCG iterations.					

p_k to the system

$$M_k p_k = -g_k, \tag{12}$$

where M_k is defined as before, so that M_k replaces the initial approximation to the Hessian. To solve (12) in LM-BFGS we use the standard GMW MC. We expect preconditioning in LM-BFGS to reduce the number of function evaluations significantly, but this must be balanced with the added cost involved in evaluating and factoring the preconditioner.

In all computations, we used the default parameters in CHARMM for the minimizers. No cutoffs for the nonbonded terms were used to avoid formation of artificial minima that result when the nonbonded terms are turned off at some distance, even when this is done smoothly. We also used the same convergence test (i.e., inequality (B1d) in the Appendix B with $\epsilon_g = 10^{-6}$) for TNPACK, ABNR, CONJ, and LM-BFGS. Both TNPACK and ABNR can reach much lower gradient norms than CONJ.

For butane and alanine dipeptide, all minimizers (except for one case: LM-BFGS with preconditioning for butane[2]) find the same minimum value, while for BPTI and

[2] For butane, the global minimum corresponds to an open chain configuration ("trans-staggered"), with the central dihedral angle φ, defining the relative orientation of the four carbons, adopting the value −180°; the higher energy minimum corresponds to a more compact configuration, with φ about −65°.

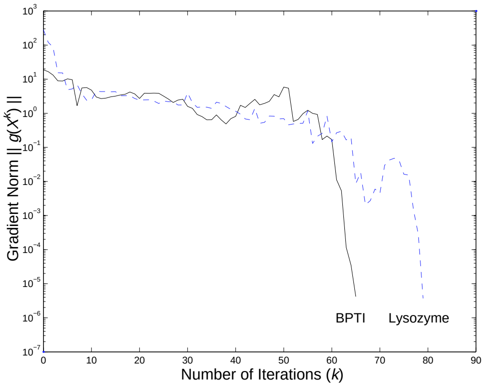


FIG. 8. *Evolution of the gradient norms generated by TNPACK for BPTI (solid) and lysozyme (dashed).*

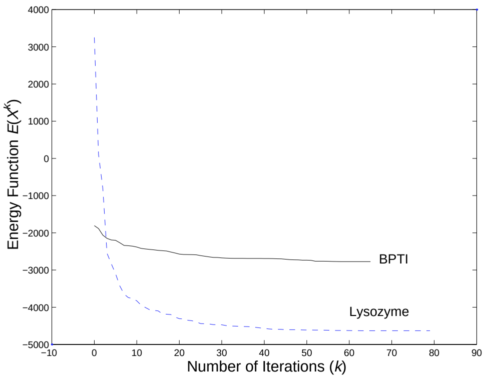


FIG. 9. *The decrease of the potential energy function for BPTI (solid) and lysozyme (dashed) by TNPACK.*

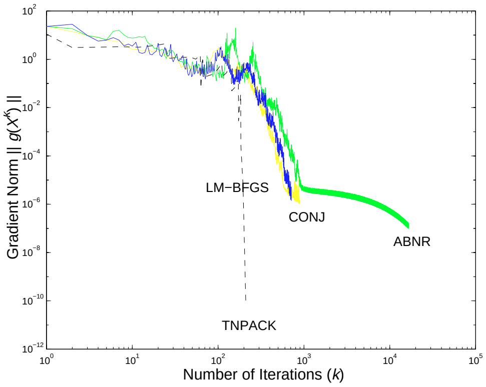


FIG. 10. *A comparison of gradient norms generated by TNPACK, ABNR, CONJ, and LM-BFGS for alanine dipeptide minimization.*

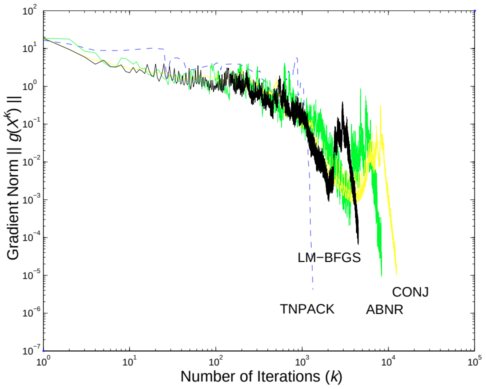


FIG. 11. *A comparison of gradient norms generated by TNPACK, ABNR, CONJ, and LM-BFGS for BPTI minimization.*

lysozyme different minima are obtained. This is a consequence of different paths taken toward a local minimum in each case. The results in Table 4 show that TNPACK requires less CPU time than the other methods and reaches very low gradient norms. The results for LM-BFGS show how preconditioning tends to reduce the total number of iterations but to increase the CPU time. For the proteins, the CPU time of TNPACK is less than that of the best LM-BFGS variant by a factor of 2 to 3.

In Figure 8 we illustrate the evolution of the gradient norm for BPTI and lysozyme molecular systems for TNPACK, along with their energy decreases in Figure 9.

In Figures 10 and 11, we compare the gradient norm evolution for TNPACK, ABNR, CONJ, and LM-BFGS (no preconditioning) for the dipeptide and BPTI. For TNPACK, the "iteration" value in the abscissa corresponds to the accumulated number of PCG iterations.

The relative importance of updating and preconditioning in LM-BFGS was discussed in [21] by testing preconditioning with various numbers of stored updates (i.e.,

$u = 0, 1, 2, 3, 4, 5$). It was found that the relative importance of these factors in generating performance improvement depends on the initial guess for the minimum—preconditioning is more important when the initial guess is better. Our experiments here with different numbers of updates for the LM-BFGS version without preconditioning revealed that $u = 4$ or 5 is optimal in terms of CPU time (data not shown); when preconditioning is used, the optimal u tends to be lower (e.g., $u = 2$ or 3).

5. Conclusions. We have suggested the use of an indefinite rather than a positive-definite preconditioner in the TN optimization method applied to large-scale, highly nonlinear functions with problem-formulated preconditioners. With the UMC applied to solve a linear system involving the preconditioner, we guarantee that the resulting search vectors are directions of descent. Thus, convergence to a local minimum can be derived as in classic TN methods.

An indefinite preconditioner makes sense in our applications for efficiency considerations. Namely, the sparse preconditioner generated from the local chemical interactions [25] can have large negative eigenvalues, and other MC schemes [3, 7, 8, 26] (when used with PCG for solving such preconditioned linear systems) tend to exhibit poor numerical behavior when very large modifications are permitted. This leads to many PCG iterations and large CPU times for the overall minimization method. We overcome this difficulty by proposing the UMC to prescribe matrix modifications τI in a numerically stable manner. The parameter τ is chosen heuristically, so as to lead to positive-definite preconditions near a minimum. This bound appears insensitive to the problem size, and in our application we use $\tau = 10$. Undoubtedly, there are other ways to factor a symmetric matrix M in this way.

The numerical experiments reported here highlight that the unconventional use of an indefinite preconditioner works better than the minimizer without preconditioning, as well as other minimizers available in CHARMM (ABNR and CONJ). A competitive method tested is also LM-BFGS, examined both with and without preconditioning. Although preconditioning reduces the total number of iterations in LM-BFGS, it increases the CPU time because of the added cost of the linear system. Results show that TNPACK requires less CPU time than the other methods tested for large potential energy problems. Very recently, we have updated the program routines of TNPACK/CHARMM to significantly reduce memory requirements by using a specified sparsity pattern for the preconditioner and finite differences for Hessian and vector multiplication. These developments, including applications to problems with up to 35000 variables, will be reported separately.

These algorithmic suggestions may open new opportunities for other large-scale optimization problems in which partial second-derivative information might be exploited in the TN framework. Particularly interesting is the possibility of using TNPACK as a local minimizer in the context of a stochastic global optimization method. The different local minima reached for the proteins in this work suggest that even a simple global aspect added to the local minimizer can be of practical importance.

Appendix A. Analyses for the PCG inner loop of TN. We consider the PCG algorithm of Algorithm 1 for solving Newton equation $H_k p = -g_k$ with a preconditioner M_k, where both H_k and M_k are nonsingular but not necessarily positive definite. We assume that there exists a positive integer l such that $d_j^T H_k d_j \neq 0$ and $r_j^T M_k^{-1} r_j \neq 0$ for $j = 1, 2, \ldots, l$, and thus the PCG iterates $\{p_j\}_{j=1}^{l}$ are well defined. As for the standard case (i.e., M_k is positive definite) [9], it follows that the PCG residual vectors $\{r_j\}_{j=1}^l$ $(r_j \equiv -g_k - H_k p_j)$ satisfy in exact arithmetic

$$r_i^T M_k^{-1} r_j = 0 \quad \text{for} \quad 1 \le i < j \le l. \tag{13}$$

Theorem 1 (motivation for not using d_j as exit search direction). *Let M_k be nonsingular and the initial guess $p_1 = 0$. Then, if $g_k^T M_k^{-1} r_j = 0$ for $1 < j \le l$, all vectors d_j for $1 \le j \le l$ satisfy*

$$g_k^T d_j = -r_j^T M_k^{-1} r_j. \tag{14}$$

Proof. Since $r_1 = -g_k$, from (13) it follows that

$$g_k^T M_k^{-1} r_{j+1} = -r_1^T M_k^{-1} r_{j+1} = 0 \quad \text{for } 1 \le j \le l-1.$$

Thus, for all $j = 1, 2, \ldots, l-1$,

$$\begin{aligned} g_k^T d_{j+1} &= g_k^T (M_k^{-1} r_{j+1} + \beta_j d_j) \\ &= g_k^T M_k^{-1} r_{j+1} + \beta_j g_k^T d_j \\ &= \beta_j g_k^T d_j. \end{aligned}$$

Noting that $\beta_j = r_{j+1}^T M_k^{-1} r_{j+1} / r_j^T M_k^{-1} r_j, r_1 = -g_k$, and $d_1 = -M_k^{-1} g_k$, we obtain

$$\begin{aligned} g_k^T d_{j+1} &= \beta_j \beta_{j-1} \cdots \beta_1 g_k^T d_1 \\ &= -\frac{r_{j+1}^T M_k^{-1} r_{j+1}}{r_1^T M_k^{-1} r_1} g_k^T M_k^{-1} g_k \\ &= -r_{j+1}^T M_k^{-1} r_{j+1}. \qquad \square \end{aligned}$$

From Theorem 1 it follows that d_j may be an ascent direction for the case of an indefinite M_k.

Theorem 2 (motivation for using an indefinite preconditioner in TN). *Let M_k be nonsingular and the initial guess $p_1 = 0$. Then, if $d_j^T H_k d_j > 0$ and $r_j^T M_k^{-1} r_j \ne 0$ for $j = 1, 2, \ldots, l$, all p_j with $2 \le j \le l+1$ are descent directions and satisfy*

$$g_k^T p_{l+1} < \cdots < g_k^T p_j < g_k^T p_{j-1} < \cdots < g_k^T p_2 < 0. \tag{15}$$

Proof. Using $\alpha_j = r_j^T M_k^{-1} r_j / d_j^T H_k d_j$ and (14), we have

$$g_k^T p_{j+1} = g_k^T p_j + \alpha_j g_k^T d_j = g_k^T p_j - \frac{(r_j^T M_k^{-1} r_j)^2}{d_j^T H_k d_j}. \tag{16}$$

Relation (16) together with $d_j^T H_k d_j > 0$ and $(r_j^T M_k^{-1} r_j)^2 > 0$ give $g_k^T p_{j+1} < g_k^T p_j$ for $j = 1, 2, \ldots, l$. In particular, $g_k^T p_2 < g_k^T p_1 = 0$ as $p_1 = 0$. Therefore, it follows that all p_j with $2 \le j \le l+1$ satisfy (15). $\square$

From Theorem 2 it follows that the PCG directions of Algorithm 1 with an indefinite preconditioner M_k are directions of descent.

Theorem 3 (equivalence of the descent direction and negative curvature tests). *Let M_k be nonsingular and $r_j^T M_k^{-1} r_j \ne 0$. Then $g_k^T p_{j+1} > g_k^T p_j$ if and only if $d_j^T H_k d_j < 0$.*

Proof. From (14) and $\alpha_j = r_j^T M_k^{-1} r_j / d_j^T H_k d_j$, we have

$$\alpha_j g_k^T d_j = -(r_j^T M_k^{-1} r_j)^2 / d_j^T H_k d_j.$$

Thus, if the denominator $d_j^T H_k d_j < 0$, then the left-hand side $\alpha_j g_k^T d_j > 0$, implying that

$$g_k^T p_{j+1} = g_k^T p_j + \alpha_j g_k^T d_j > g_k^T p_j.$$

On the other hand, if $g_k^T p_{j+1} > g_k^T p_j$, then

$$-(r_j^T M_k^{-1} r_j)^2 / d_j^T H_k d_j = \alpha_j g_k^T d_j = g_k^T p_{j+1} - g_k^T p_j > 0,$$

which implies that $d_j^T H_k d_j < 0$. □

From Theorem 3 it follows that the descent direction test of Algorithm 1 is equivalent to the negative curvature test.

Theorem 4 (another motivation for using p_j rather than d_j as exit search direction). *Let both H_k and M_k be positive definite. Then there exists an index $j_0 > 0$ such that for all $i \geq 2$*

$$g_k^T p_i < g_k^T d_j < 0 \quad \textit{whenever} \quad j > j_0. \tag{17}$$

Proof. When both H_k and M_k are positive definite, from PCG theory we know that $r_j^T M_k^{-1} r_j$ approaches zero as j increases. Subsequently, there exists $j_0 > 0$ such that $r_j^T M_k^{-1} r_j < |g_k^T p_2|$ whenever $j > j_0$. Together with (14) and (15), we have $g_k^T p_2 < 0$ and

$$g_k^T d_j = -r_j^T M_k^{-1} r_j > g_k^T p_2 > g_k^T p_i \quad \text{for all } i > 2. \qquad \square$$

The steplength λ can have a larger range of feasibility to satisfy

$$E(X^k + \lambda P^k) < E(X^k)$$

with a larger value of $|g_k^T P^k|$, where $g_k^T P^k$ is negative. Thus, the objective function value may be reduced more on a larger range of λ. In this sense, Theorem 4 suggests that p_j is a "better" search direction than d_j because choosing the search direction $P^k = p_j$ for $j \geq 2$ can lead to further reduction than using $P^k = d_j$ for a sufficiently large j. Similarly, Theorem 2 suggests that a PCG iterate p_j is better than p_i when $j > i$.

Appendix B. The TN algorithm. The TN algorithm based on the PCG method consists of an outer and an inner loop. We present these two loops in turn, listing the parameter values used in the numerical examples reported in this paper (unless specified otherwise in text). The new algorithmic components introduced in this paper are marked by asterisks. We denote the objective function to be minimized by E; the gradient vector and Hessian matrix of E by g and H, respectively; and the preconditioner for PCG by M. We omit the subscript k from g, H, and M for clarity.

Outer Loop of the TN Method

1. *Initialization*
   - Set $k = 0$ and evaluate $E(X^0)$ and $g(X^0)$ for a given initial guess X^0.
   - If $||g(X^0)|| < 10^{-8} \max(1, ||X^0||)$, exit algorithm, where $|| \cdot ||$ is the standard Euclidean norm divided by $\sqrt{n}$.
2. *Preparation for UMC*
   - Evaluate the preconditioner M at X^0 by assembling only the local potential energy terms (bond length, bond angle, and dihedral angle components).
   - Determine the sparsity pattern of M. The upper triangle of M is stored in a compressed row format, and the pattern is specified by two integer arrays that serve as row and column pointers [23].
   - Compute the *symbolic factorization* LDL^T of M, that is, the sparsity structure of the factor L.
   - Evaluate the Hessian matrix H at X^0.
3. *Inner loop*

Compute a search vector P^k by solving the Newton equation $HP = -g$ *approximately* using PCG with preconditioner M based on the UMC method (see below).

4*. *Line search*
   - Compute a steplength λ by safeguarded cubic and quadratic interpolation [12] (see also [27] for a minor modification that avoids a too small acceptable steplength λ) so that the update $X^{k+1} = X^k + \lambda P^k$ satisfies

$$E(X^{k+1}) \leq E(X^k) + \alpha\lambda g(X^k)^T P^k \quad \text{and} \quad |g(X^{k+1})^T P^k| \leq \beta |g(X^k)^T P^k|,$$

   where $\alpha = 10^{-4}$ and $\beta = 0.9$.

5. *Convergence tests*
   - Check the following inequalities:

$$E(X^{k+1}) - E(X^k) < \epsilon_f (1 + |E(X^{k+1})|) \ , \tag{B1a}$$

$$||X^{k+1} - X^k|| < \sqrt{\epsilon_f}\ (1 + ||X^{k+1}||)/100 \ , \tag{B1b}$$

$$||g(X^{k+1})|| < \epsilon_f^{1/3} (1 + |E(X^{k+1})|) \ , \tag{B1c}$$

$$||g(X^{k+1})|| < \epsilon_g (1 + |E(X^{k+1})|) \ , \tag{B1d}$$

   where $\epsilon_f = 10^{-10}$ and $\epsilon_g = 10^{-8}$.
   If conditions (B1a), (B1b), (B1c), or (B1d) are satisfied, exit algorithm.

6. *Preparation for the next Newton step*
   - Compute the preconditioner M at X^{k+1} by using the pattern determined originally.
   - Evaluate the Hessian matrix H at X^{k+1}.
   - Set $k \rightarrow k + 1$, and go to step 3.

Inner Loop of the Truncated Newton Method (step 3 of Outer Loop)

The sequence $\{p_j\}$ below represents the PCG vectors used to construct P^k in step 3 of the Outer Loop, above.

1. *Initialization*
    - Set $j = 1$, $p_1 = 0$, and $r_1 = -g$.
    - Set the parameters $\eta_k = \min\{c_r/k\ ,\ \|g\|\}$ and $\mathrm{IT}_{\mathrm{PCG}}$ for the truncation test in step 5. We use $c_r = 0.5$ and $\mathrm{IT}_{\mathrm{PCG}} = 40$.

2*. *The UMC factorization*
- Perform the UMC of M so that the resulting effective preconditioner is $\widetilde{M} = LDL^T$ with a chosen parameter τ (we use $\tau = 10$). The factor L is stored in the same sparse row format used for M.
- Solve for z_j in $\widetilde{M} z_j = r_j$ by using the triangular systems

$$Lx = r_j \quad \text{and} \quad L^T z_j = D^{-1}x.$$

- Set $d_j = z_j$.

3*. *Singularity test*

Compute the matrix–vector product $q_j = Hd_j$.
If either $|r_j^T z_j| \le \delta$ or $|d_j^T q_j| \le \delta$ (e.g., $\delta = 10^{-10}$),
exit PCG loop with $P^k = p_j$ (for $j = 1$, set $P^k = -g_k$).

4*. *Implement one of the following two tests:*

4a. [The descent direction test]
Update the quantities

$$\alpha_j = r_j^T z_j \,/\, d_j^T q_j \quad \text{and} \quad p_{j+1} = p_j + \alpha_j d_j. \tag{B2}$$

If $g^T p_{j+1} \ge g^T p_j + \delta$,
exit inner loop with $P^k = p_j$ (for $j = 1$, set $P^k = -g$).

4b. [The standard negative curvature test]
if $d_j^T q_j \le \delta(d_j^T d_j)$,
exit inner loop with $P^k = p_j$ (for $j = 1$, set $P^k = -g$);
else update α_j and p_{j+1} as in (B2).

5*. *Truncation test*
- Compute $r_{j+1} = r_j - \alpha_j q_j$.
- *If* $\|r_{j+1}\| \le \eta_k \|g\|$ or $j + 1 > \mathrm{IT}_{\mathrm{PCG}}$,
  exit inner loop with search direction $P^k = p_{j+1}$.

6*. *Continuation of PCG*
- Solve for z_{j+1} as in step 2 in $\widetilde{M} z_{j+1} = r_{j+1}$.
- Update the quantities

$$\beta_j = r_{j+1}^T z_{j+1} \,/\, r_j^T z_j \quad \text{and} \quad d_{j+1} = z_{j+1} + \beta_j d_j. \tag{B3}$$

- Set $j \leftarrow j + 1$, and go to step 3

Acknowledgments. We are indebted to the referees and the reviewing editor for valuable comments.

REFERENCES

[1] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus, *CHARMM: A program for macromolecular energy, minimization, and dynamics calculations*, J. Comp. Chem., 4 (1983), pp. 187–217.

[2] J. R. Bunch and L. Kaufman, *Some stable methods for calculating inertia and solving symmetric linear systems*, Math. Comp., 31 (1977), pp. 163–655.

[3] S. H. Cheng and N. J. Higham, *A modified Cholesky algorithm based on a symmetric indefinite factorization*, SIAM J. Matrix Anal. Appl., 19 (1998), pp. 1097–1110.

[4] R. S. Dembo and T. Steihaug, *Truncated-Newton algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.

[5] P. Derreumaux, G. Zhang, B. Brooks, and T. Schlick, *A truncated-Newton method adapted for CHARMM and biomolecular applications,* J. Comp. Chem., 15 (1994), pp. 532–552.

[6] A. Forsgren, P. E. Gill, and W. Murray, *Computing modified Newton directions using a partial Cholesky factorization*, SIAM J. Sci. Comput., 16 (1995), pp. 139–150.

[7] P. E. Gill and W. Murray, *Newton-type methods for unconstrained and linearly constrained optimization,* Math. Programming, 28 (1974), pp. 311–350.

[8] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, London, 1983.

[9] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed., the John Hopkins University Press, Baltimore, MD, 1986.

[10] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice–Hall, Englewood Cliffs, NJ, 1983; reprinted as Classics Appl. Math. 16, SIAM, Philadelphia, PA, 1996.

[11] D. C. Liu and J. Nocedal, *On the limited memory BFGS method for large-scale optimization*, Math. Programming, 45 (1989), pp. 503–528.

[12] J. J. Moré and D. J. Thuente, *Line search algorithms with guaranteed sufficient decrease*, ACM Trans. Math. Software, 20 (1994), pp. 286–307.

[13] S. G. Nash, *Newton-type minimization via the Lanczos method*, SIAM J. Numer. Anal., 21 (1984), pp. 770–788.

[14] S. G. Nash, *Preconditioning of truncated-Newton methods,* SIAM J. Sci. Statist. Comput., 6 (1985), pp. 599–616.

[15] S. G. Nash and J. Nocedal, *A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization*, SIAM J. Optim., 1 (1991), pp. 358–372.

[16] A. Nyberg and T. Schlick, *A computational investigation of dynamic properties with the implicit-Euler scheme for molecular dynamics simulations*, J. Chem. Phys., 95 (1991), pp. 4986–4996.

[17] P. M. Pardalos, D. Shalloway and G. Xue, eds., *Global Minimization of Nonconvex Energy Function: Molecular Conformation and Protein Folding*, DIMACS Ser. Discrete Math. Theoret. Comput. Sci., 23, AMS, Providence, RI, 1996.

[18] P. M. Pardalos and G. Xue, Eds., *Special issue on computer simulations in molecular and protein conformations*, J. Global Optim., 11 (1997).

[19] S. C. Eisenstat, A. H. Sherman, and M. H. Schultz, *Algorithms and data structures for sparse symmetric Gaussian elimination,* SIAM J. Sci. Statist. Comput., 2 (1981), pp. 225–237.

[20] M. H. Schultz, S. C. Eisenstat, M. C. Gursky, and A. H. Sherman, *Yale sparse matrix package,* I. *The symmetric codes,* Internat. J. Numer. Meth. Engrg., 18 (1982), pp. 1145–1151.

[21] T. Schlick, *Optimization methods in computational chemistry,* in Reviews in Computational Chemistry III, pp. 1–71, K. B. Lipkowitz and D. B. Boyd, eds., VCH Publishers, New York, 1992.

[22] T. Schlick, *Modified Cholesky factorizations for sparse preconditioners,* SIAM J. Sci. Comput., 14 (1993), pp. 424–445.

[23] T. Schlick and A. Fogelson, *TNPACK—A truncated Newton minimization package for large-scale problems:* I. *Algorithm and usage*, ACM Trans. Math. Software, 18 (1992), pp. 46–70.

[24] T. Schlick and A. Fogelson, *TNPACK—A truncated Newton minimization package for large-scale problems:* II. *Implementation examples,* ACM Trans. Math. Software, 18 (1992), pp. 71–111.

[25] T. Schlick and M. L. Overton, *A powerful truncated Newton method for potential energy functions.* J. Comp. Chem., 8 (1987), pp. 1025–1039.

[26] R. B. Schnabel and E. Eskow, *A new modified Cholesky factorization*, SIAM J. Sci. Statist. Comput., 11 (1990), pp. 1136–1158.

[27] D. Xie and T. Schlick, *A more lenient stopping rule for line search algorithms*, Optim. Methods. Softw., submitted.

[28] D. Xie and T. Schlick, *Remark on the updated truncated Newton minimization package, Algorithm* 702, ACM Trans. Math. Software, 25 (1999), pp. 108–122.

[29] X. Zou, I. M. Navon, M. Berger, K. H. Phua, T. Schlick, and F. X. Le Dimet, *Numerical experience with limited-memory quasi-Newton and truncated Newton methods*, SIAM J. Optim., 3 (1993), pp. 582–608.

SIAM J. Optim.
Vol. 10, No. 1, pp. 155–176

CONDITION-BASED COMPLEXITY OF CONVEX OPTIMIZATION IN CONIC LINEAR FORM VIA THE ELLIPSOID ALGORITHM[*]

ROBERT M. FREUND[†] AND JORGE R. VERA[‡]

Abstract. A convex optimization problem in conic linear form is an optimization problem of the form

$$\begin{array}{rll} CP(d): & \text{maximize} & c^T x \\ & \text{s.t.} & b - Ax \in C_Y, \\ & & x \in C_X, \end{array}$$

where C_X and C_Y are closed convex cones in n- and m-dimensional spaces X and Y, respectively, and the data for the system is $d = (A, b, c)$. We show that there is a version of the ellipsoid algorithm that can be applied to find an ϵ-optimal solution of $CP(d)$ in at most $O(n^2 \ln(\frac{\mathcal{C}(d)\|c\|_*}{c_1 \epsilon}))$ iterations of the ellipsoid algorithm, where each iteration must either perform a separation cut on one of the cones C_X or C_Y or perform a related optimality cut. The quantity $\mathcal{C}(d)$ is the "condition number" of the program $CP(d)$ originally developed by Renegar and is essentially a scale-invariant reciprocal of the smallest data perturbation $\Delta d = (\Delta A, \Delta b, \Delta c)$ for which the system $CP(d + \Delta d)$ becomes either infeasible or unbounded. The scalar quantity c_1 is a constant that depends only on the simple notion of the "width" of the cones and is independent of the problem data $d = (A, b, c)$ but may depend on the dimensions m and/or n.

Key words. complexity of convex optimization, ellipsoid method, conditioning, error analysis

AMS subject classifications. 90C, 90C05, 90C60

PII. S105262349732829X

1. Introduction.

Consider a convex program in conic linear form:

$$\begin{array}{rll} CP(d): & \text{maximize} & c^T x \\ & \text{s.t.} & b - Ax \in C_Y, \\ & & x \in C_X, \end{array} \tag{1}$$

where $C_X \subset X$ and $C_Y \subset Y$ are each a closed convex cone in the (finite) n-dimensional linear vector space X (with norm $\|x\|$ for $x \in X$) and in the (finite) m-dimensional linear vector space Y (with norm $\|y\|$ for $y \in Y$), respectively. Here $b \in Y$, and $A \in L(X, Y)$, where $L(X, Y)$ denotes the set of all linear operators $A : X \to Y$. Also, $c \in X^*$, where X^* is the space of all linear functionals defined on X; i.e., X^* is the dual space of X. In order to maintain consistency with standard linear algebra notation in mathematical programming, we consider c to be a column vector in the space X^* and we denote the linear function $c(x)$ by $c^T x$. Similarly, for $A \in L(X, Y)$ and $f \in Y^*$, we denote $A(x)$ by Ax and $f(y)$ by $f^T y$. We denote the adjoint of A by A^T.

[*] Received by the editors October 6, 1997; accepted for publication (in revised form) October 30, 1998; published electronically October 20, 1999. This research has been partially supported through National Science Foundation grant INT-9703803 and a grant from Fundación Andes/CONICYT, in the framework of the international cooperation agreement with NSF. The second author also was supported by a grant from CONICYT within the FONDAP program in applied mathematics.

http://www.siam.org/journals/siopt/10-1/32829.html

[†] MIT Operations Research Center, 77 Massachusetts Avenue, Cambridge, MA 02139 (rfreund@mit.edu).

[‡] Department of Industrial and Systems Engineering, Catholic University of Chile, Campus San Joaquín, Vicuña Mackenna 4860, Santiago, Chile (jvera@ing.puc.cl).

The "data" d for problem $CP(d)$ is the array $d = (A, b, c) \in \{L(X, Y), Y, X^*\}$. We call the above program $CP(d)$ rather than simply CP to emphasize the dependence of the optimization problem on the data $d = (A, b, c)$, and we note that the cones C_X and C_Y are not part of the data; that is, they are considered to be given and fixed. At the moment, we make no assumptions on C_X and on C_Y except to note that each is a closed convex cone.

The format of $CP(d)$ is quite general (any convex optimization problem can be cast in the format of $CP(d)$) and has received much attention recently in the context of interior-point algorithms; see Nesterov and Nemirovskii [13] and Renegar [19], [20], as well as Nesterov and Todd [15], [14] and Nesterov, Todd, and Ye [16], among others.

In contrast to interior-point methods, this paper focuses on the complexity of solving $CP(d)$ via the ellipsoid algorithm. The ellipsoid algorithm of Yudin and Nemirovskii [26] and Shor [21] (see also [4], [8], and [9]) and the interior-point algorithm of Nesterov and Nemirovskii [13] are two fundamental theoretically efficient algorithms for solving general convex optimization. The ellipsoid algorithm enjoys a number of important advantages over interior-point algorithms: the ellipsoid algorithm is based on elegantly simple geometric notions, it always has excellent theoretical efficiency in the dimension of the variables n, it requires only the use of a separation oracle for its implementation, and it is important in both continuous and discrete optimization [8]. (Of course, when applied to solving linear programs, interior-point algorithms typically exhibit vastly superior practical performance over the ellipsoid algorithm, but that is not the focus of this study.)

The ellipsoid algorithm belongs to a larger class of efficient volume-reducing cutting-plane algorithms that includes the method of centers of gravity [11], the method of inscribed ellipsoids [10], and the method of volumetric centers [22], among others. We focus herein on the ellipsoid algorithm because of its prominence and history in the complexity analysis of convex optimization, but our analysis is applicable to these other volume-reducing cutting-plane methods as well; see the remarks in section 6.

In analyzing the complexity of the ellipsoid algorithm, we adopt the relatively new concept of the *condition number* $\mathcal{C}(d)$ of the program $CP(d)$, developed by Renegar in the series of papers [17], [18], and [19]. We show (in section 5) that there is a version of the ellipsoid algorithm that can be applied to find an ϵ-optimal solution of $CP(d)$ in at most $O(n^2 \ln(\frac{\mathcal{C}(d)\|c\|_*}{c_1 \epsilon}))$ iterations of the ellipsoid algorithm, where each iteration must perform either a separation cut on one of the cones C_X or C_Y or a related optimality cut. The quantity $\mathcal{C}(d)$ is the condition number of the program $CP(d)$, and $\|c\|_*$ is the norm of c. The scalar quantity c_1 is a constant that depends only on the simple notion of the "width" of the cones, and is independent of the problem data $d = (A, b, c)$, but may depend on the dimensions m and/or n.

Two special cases of $CP(d)$ deserve special mention: linear programming and semidefinite programming. Let $\Re$ and $\Re_+$ denote the set of real numbers and the set of nonnegative real numbers, respectively, and let $\Re^k$ and $\Re^k_+$ denote real k-dimensional space and the nonnegative orthant in $\Re^k$, respectively. Then by setting (i) $C_X = \Re^n_+$ and $C_Y = \Re^m_+$, (ii) $C_X = \Re^n_+$ and $C_Y = \{0\}$, or (iii) $C_X = \Re^n$ and $C_Y = \Re^m_+$, then $CP(d)$ is a linear program of the format (i) $\max\{c^T x \mid Ax \le b, x \ge 0, x \in \Re^n\}$, (ii) $\max\{c^T x \mid Ax = b, x \ge 0, x \in \Re^n\}$, or (iii) $\max\{c^T x \mid Ax \le b, x \in \Re^n\}$, respectively.

The other special case of $CP(d)$ that we mention is semidefinite programming. Semidefinite programming has been shown to be of enormous importance in mathematical programming (see Alizadeh [1] and Nesterov and Nemirovskii [13] as well as

Vandenberghe and Boyd [23]). Let X denote the set of real $k \times k$ symmetric matrices, whereby $n = k(k+1)/2$, and define the Löwner partial ordering "$\succeq$" on X as $x \succeq w$ if and only if the matrix $x - w$ is positive semidefinite. The semidefinite program in standard (primal) form is the problem $\max\{c^T x \mid Ax = b, x \succeq 0\}$. Define $C_X = \{x \in X \mid x \succeq 0\}$. Then C_X is a closed convex cone. Let $Y = \Re^m$ and $C_Y = \{0\} \subset \Re^m$. Then the standard form semidefinite program is easily seen to be an instance of $CP(d)$.

Most studies of the ellipsoid algorithm (for example, [9], [4], [8]) pertain to the case when $CP(d)$ is a linear or convex quadratic program and focus on the complexity of the algorithm in terms of the bit length L of a binary representation of the data $d = (A, b, c)$. However, when the cones C_X and/or C_Y are not polyhedral or when the data $d = (A, b, c)$ are not rational, it makes little or no sense to study the complexity of the ellipsoid algorithm in terms of L. Indeed, a much more natural and intuitive measure that is relevant for complexity analysis and that captures the inherent data-dependent behavior of $CP(d)$ is the "condition number" $\mathcal{C}(d)$ of the problem $CP(d)$, which was developed by Renegar in a series of papers [17], [18], [19]. The quantity $\mathcal{C}(d)$ is essentially a scale invariant reciprocal of the smallest data perturbation $\Delta d = (\Delta A, \Delta b, \Delta c)$ for which the system $CP(d + \Delta d)$ becomes either infeasible or unbounded. (These concepts will be reviewed in detail shortly.)

The paper is organized as follows. The remainder of this introductory section discusses the condition number $\mathcal{C}(d)$ of the optimization problem $CP(d)$. Section 2 contains further notation and a discussion of the width of a cone. In section 3 we demonstrate a ball construction for the set of ϵ-optimal solutions of $CP(d)$, and we review several previous results regarding the geometry of $CP(d)$. Section 4 briefly reviews relevant complexity aspects of the ellipsoid algorithm and reviews a transformation of $CP(d)$ into a homogenized form called $HP(d)$ that is more convenient for the application of the ellipsoid algorithm. Lemma 4.1 contains a key volume-ratio upper bound that is the main tool used in proving the complexity results for the ellipsoid algorithm for solving $CP(d)$, which are presented in section 5. Section 6 discusses related issues: complexity results for other volume-reducing cutting-plane algorithms, testing for ϵ-optimality, the complexity of testing for infeasibility of $CP(d)$, and bounding the skewness of the ellipsoids computed in the ellipsoid algorithm.

The concept of the "distance to ill-posedness" and a closely related condition number for problems such as $CP(d)$ was introduced by Renegar in [17] in a more specific setting but then generalized more fully in [18] and [19]. We now describe these two concepts in detail.

Using the constructs of Lagrangian duality, one obtains the following dual problem of $CP(d)$:

$$(2) \qquad \begin{array}{rll} CD(d): & \text{minimize} & b^T y \\ & \text{s.t.} & A^T y - c \in C_X^*, \\ & & y \in C_Y^*, \end{array}$$

where C_X^* and C_Y^* are the dual convex cones associated with the cones C_X and C_Y, respectively, and where the dual cone of a convex cone K in a linear vector space X is defined by

$$K^* = \{z \in X^* | z^T x \geq 0 \text{ for any } x \in K\}.$$

The data for the program $CD(d)$ is also the array $d = (A, b, c)$.

We denote the space of all data $d = (A, b, c)$ for $CP(d)$ by $\mathcal{D}$. Then $\mathcal{D} = \{d = (A, b, c) \mid A \in L(X, Y), b \in Y, c \in X^*\}$. Because X and Y are normed linear vector spaces, we can define the following product norm on the data space $\mathcal{D}$:

$$\|d\| = \|(A, b, c)\| = \max\{\|A\|, \|b\|, \|c\|_*\} \qquad \text{for any } d \in \mathcal{D},$$

where $\|A\|$ is the operator norm, namely,

$$\|A\| = \max\{\|Ax\| \mid \|x\| \leq 1\},$$

and where $\|c\|_*$ is the dual norm of c induced on $c \in X^*$, defined as

$$\|c\|_* = \max\{c^T x \mid \|x\| \leq 1, x \in X\},$$

with a similar definition holding for $\|v\|_*$ for $v \in Y^*$.

Consider the following subsets of the data set $\mathcal{D}$:

$$\mathcal{F}_P = \{(A, b, c) \in \mathcal{D} \mid \ \text{there exists } x \text{ such that } b - Ax \in C_Y, x \in C_X\},$$

$$\mathcal{F}_D = \{(A, b, c) \in \mathcal{D} \mid \ \text{there exists } y \text{ such that } A^T y - c \in C_X^*, y \in C_Y^*\},$$

and

$$\mathcal{F} = \mathcal{F}_P \cap \mathcal{F}_D.$$

The elements in $\mathcal{F}_P$ correspond to those data instances $d = (A, b, c)$ in $\mathcal{D}$ for which $CP(d)$ is feasible and the elements in $\mathcal{F}_D$ correspond to those data instances $d = (A, b, c)$ in $\mathcal{D}$ for which $CD(d)$ is feasible. Observe that $\mathcal{F}$ is the set of data instances $d = (A, b, c)$ that are both primal and dual feasible. The complement of $\mathcal{F}_P$, denoted by $\mathcal{F}_P^C$, is the set of data instances $d = (A, b, c)$ for which $CP(d)$ is infeasible, and the complement of $\mathcal{F}_D$, denoted by $\mathcal{F}_D^C$, is the set of data instances $d = (A, b, c)$ for which $CD(d)$ is infeasible.

The boundary of $\mathcal{F}_P$ and $\mathcal{F}_P^C$ is the set

$$\mathcal{B}_P = \partial\mathcal{F}_P = \partial\mathcal{F}_P^C = \ \mathrm{cl}(\mathcal{F}_P) \cap \mathrm{cl}(\mathcal{F}_P^C),$$

and the boundary of $\mathcal{F}_D$ and $\mathcal{F}_D^C$ is the set

$$\mathcal{B}_D = \partial\mathcal{F}_D = \partial\mathcal{F}_D^C = \ \mathrm{cl}(\mathcal{F}_D) \cap \mathrm{cl}(\mathcal{F}_D^C),$$

where ∂S denotes the boundary of a set S and $\mathrm{cl}(S)$ is the closure of a set S. Note that $\mathcal{B}_P \neq \emptyset$ since $(0, 0, 0) \in \mathcal{B}_P$. The data instances $d = (A, b, c)$ in $\mathcal{B}_P$ are called the ill-posed data instances for the primal, in that arbitrarily small changes in the data $d = (A, b, c)$ can yield data instances in $\mathcal{F}_P$ as well as data instances in $\mathcal{F}_P^C$. Similarly, the data instances $d = (A, b, c)$ in $\mathcal{B}_D$ are called the ill-posed data instances for the dual.

For $d = (A, b, c) \in \mathcal{D}$, we define the ball centered at d with radius δ as

$$B(d, \delta) = \{\bar{d} \in \mathcal{D} : \|\bar{d} - d\| \leq \delta\}.$$

For a data instance $d \in \mathcal{D}$, the "primal distance to ill-posedness" is defined as follows:

$$\rho_P(d) = \inf\{\|\Delta d\| : d + \Delta d \in \mathcal{B}_P\}$$

(see [17], [18], [19]), and so $\rho_P(d)$ is the distance of the data instance $d = (A, b, c)$ to the set $\mathcal{B}_{\mathcal{P}}$ of ill-posed instances for the primal problem $CP(d)$. It is straightforward to show that

$$\rho_P(d) = \begin{cases} \sup\{\delta : B(d, \delta) \subset \mathcal{F}_P\} & \text{if } d \in \mathcal{F}_P, \\ \sup\{\delta : B(d, \delta) \subset \mathcal{F}_P^C\} & \text{if } d \in \mathcal{F}_P^C, \end{cases} \tag{3}$$

so that we could also define $\rho_P(d)$ by employing (3). In the typical case when $CP(d)$ is feasible, i.e., $d \in \mathcal{F}_P$, $\rho_P(d)$ is the minimum change Δd in the data d needed to create a primal-infeasible instance $d + \Delta d$, and so $\rho_P(d)$ measures how close the data instance $d = (A, b, c)$ is to the set of infeasible instances of $CP(d)$. Put another way, $\rho_P(d)$ measures how close $CP(d)$ is to being infeasible. Note that $\rho_P(d)$ measures the distance of the data d to primal infeasible instances, and so the objective function vector c plays no role in this measure.

The "primal condition number" $\mathcal{C}_P(d)$ of the data instance d is defined as

$$\mathcal{C}_P(d) = \frac{\|d\|}{\rho_P(d)}$$

when $\rho_P(d) > 0$ and $\mathcal{C}_P(d) = \infty$ when $\rho_P(d) = 0$. The primal condition number $\mathcal{C}_P(d)$ can be viewed as a scale-invariant reciprocal of $\rho_P(d)$, as it is elementary to demonstrate that $\mathcal{C}_P(d) = \mathcal{C}_P(\alpha d)$ for any positive scalar α. Observe that since $\bar{d} = (\bar{A}, \bar{b}, \bar{c}) = (0, 0, 0) \in \mathcal{B}_P$ and $\mathcal{B}_P$ is a closed set, then for any $d \notin \mathcal{B}_P$ we have $\|d\| \geq \rho_P(d) > 0$, so that $\mathcal{C}_P(d) \geq 1$. The value of $\mathcal{C}_P(d)$ is a measure of the relative conditioning of the primal *feasibility* problem for the data instance d. For a discussion of the relevance of using $\mathcal{C}_P(d)$ as a condition number for the problem $CP(d)$, see Renegar [17], [18] and Vera [24].

These measures are not nearly as intangible as they might seem at first glance. In [7], it is shown that $\rho_P(d)$ can be computed by solving rather simple convex optimization problems involving the data $d = (A, b, c)$, the cones C_X and C_Y, and the norms $\|\cdot\|$ given for the problem. As in traditional condition numbers for systems of linear equations, the computation of $\rho_P(d)$ and hence of $\mathcal{C}_P(d)$ is roughly as difficult as solving $CP(d)$; see [7].

For a data instance $d \in \mathcal{D}$, the "dual distance to ill-posedness" is defined in a manner exactly analogous to the "primal distance to ill-posedness":

$$\rho_D(d) = \inf\{\|\Delta d\| : d + \Delta d \in \mathcal{B}_D\}$$

or equivalently

$$\rho_D(d) = \begin{cases} \sup\{\delta : B(d, \delta) \subset \mathcal{F}_D\} & \text{if } d \in \mathcal{F}_D, \\ \sup\{\delta : B(d, \delta) \subset \mathcal{F}_D^C\} & \text{if } d \in \mathcal{F}_D^C. \end{cases} \tag{4}$$

The "dual condition number" $\mathcal{C}_D(d)$ of the data instance d is defined as

$$\mathcal{C}_D(d) = \frac{\|d\|}{\rho_D(d)}$$

when $\rho_D(d) > 0$ and $\mathcal{C}_D(d) = \infty$ when $\rho_D(d) = 0$.

The two measures of distances to ill-posed instances and condition numbers are combined as follows. Recalling the definition of $\mathcal{F}$, the elements in $\mathcal{F}$ correspond to those data instances $d = (A, b, c)$ in $\mathcal{D}$ for which both $CP(d)$ and $CD(d)$ are feasible.

The complement of $\mathcal{F}$, denoted by $\mathcal{F}^C$, is the set of data instances $d = (A, b, c)$ for which $CP(d)$ is infeasible or $CD(d)$ is infeasible. The boundary of $\mathcal{F}$ and $\mathcal{F}^C$ is the set

$$\mathcal{B} = \partial\mathcal{F} = \partial\mathcal{F}^C = \operatorname{cl}(\mathcal{F}) \cap \operatorname{cl}(\mathcal{F}^C).$$

The data instances $d = (A, b, c)$ in $\mathcal{B}$ are called the ill-posed data instances in that arbitrarily small changes in the data $d = (A, b, c)$ can yield data instances in $\mathcal{F}$ as well as data instances in $\mathcal{F}^C$. For a data instance $d \in \mathcal{D}$, the "distance to ill-posedness" is defined as follows:

$$\rho(d) = \inf\{\|\Delta d\| : d + \Delta d \in \mathcal{B}\}$$

or equivalently

$$\rho(d) = \begin{cases} \sup\{\delta : B(d, \delta) \subset \mathcal{F}\} & \text{if } d \in \mathcal{F}, \\ \sup\{\delta : B(d, \delta) \subset \mathcal{F}^C\} & \text{if } d \in \mathcal{F}^C. \end{cases} \tag{5}$$

In the typical case when $CP(d)$ and $CD(d)$ are both feasible, i.e., $d \in \mathcal{F}$, $\rho(d)$ is the minimum change Δd in the data d needed to create a data instance $d + \Delta d$ that is either primal infeasible or dual infeasible. The "condition number" $\mathcal{C}(d)$ of the data instance d is defined as

$$\mathcal{C}(d) = \frac{\|d\|}{\rho(d)}$$

when $\rho(d) > 0$ and as $\mathcal{C}(d) = \infty$ when $\rho(d) = 0$. The condition number $\mathcal{C}(d)$ can be viewed as a scale-invariant reciprocal of $\rho(d)$. The value of $\mathcal{C}(d)$ is a measure of the relative conditioning of the problem $CP(d)$ and its dual $CD(d)$ for the data instance d.

It is straightforward to demonstrate that

$$\rho(d) = \min\{\rho_P(d), \rho_D(d)\} \quad \text{if } d \in \mathcal{F},$$

and so

$$\mathcal{C}(d) = \max\{\mathcal{C}_P(d), \mathcal{C}_D(d)\} \quad \text{if } d \in \mathcal{F}. \tag{6}$$

We offer the following interpretation of $\rho(d)$ and $\mathcal{C}(d)$ in terms of the primal problem when both the primal problem and the dual problem are feasible. Because $\rho_P(d)$ measures how close the data instance $d = (A, b, c)$ is to being an infeasible instance of the primal, and the $\rho_D(d)$ measures how close the data instance $d = (A, b, c)$ is to being an unbounded instance of the primal (in the primal objective function value), then $\rho(d)$ measures how close the data instance $d = (A, b, c)$ is to being either a primal infeasible or a primal unbounded data instance. The larger the value of condition number $\mathcal{C}(d)$ is, the closer the primal problem is to either an infeasible or an unbounded instance of the primal.

2. Further notation, coefficient of linearity, and width of a cone. We will say that a cone C is *regular* if C is a closed convex cone, has a nonempty interior, and is pointed (i.e., contains no line).

REMARK 2.1. *If C is a closed convex cone, then C is regular if and only if C^* is regular.*

Let C be a regular cone in the normed linear vector space X. Let $B(x, r)$ denote the ball centered at x with radius r. We will use the following definition of the width of C.

Definition 2.1. *If C is a regular cone in the normed linear vector space X, the width of C is given by*

$$\tau = \max \left\{ \frac{r}{\|x\|} \mid B(x, r) \subset C \right\}.$$

We remark that τ measures the maximum ratio of the radius to the norm of the center of an inscribed ball in C, and so larger values of τ correspond to an intuitive notion of greater width of C. Note that $\tau \in (0, 1]$, since C has a nonempty interior and C is pointed, and τ is attained for some $(\bar{x}, \bar{r})$ as well as along the ray $(\alpha\bar{x}, \alpha\bar{r})$ for all $\alpha > 0$.

In previous work [7], we employed the "coefficient of linearity" for a cone C.

Definition 2.2. *If C is a regular cone in the normed linear vector space X, the coefficient of linearity for the cone C is given by*

$$\beta = \begin{array}{ll} \sup & \inf u^T x \\ u \in X^*, & x \in C, \\ \|u\|_* = 1, & \|x\| = 1. \end{array} \tag{7}$$

The coefficient of linearity β for the regular cone C is essentially the same as the scalar α defined in Renegar [19, p. 328]. In [7], the coefficient of linearity was used as part of an analysis of geometric properties of the feasible region of $CP(d)$ that are implied by the condition number $\mathcal{C}_P(d)$. The following proposition shows that the width of C is equal to the coefficient of linearity of C^*.

Proposition 2.1. *Suppose that C is a regular cone in the normed linear vector space X, τ denote the width of C, and β^* denote the coefficient of linearity for C^*. Then $\tau = \beta^*$.*

Proof. From the definition of the coefficient of linearity for C^*, we have

$$\beta^* = \begin{array}{ll} \sup & \inf x^T w \\ x \in X, & w \in C^*, \\ \|x\| = 1, & \|w\|_* = 1. \end{array} \tag{8}$$

From the outer optimization problem above, there exists $\bar{x} \in X$ for which $\|\bar{x}\| = 1$ and $w^T\bar{x} \geq \beta^*$ for any $w \in C^*$ satisfying $\|w\|_* = 1$. Let $x \in B(\bar{x}, \beta^*)$, i.e., $x = \bar{x} + \beta^* v$, where $\|v\| \leq 1$. For any $w \in C^*$ satisfying $\|w\|_* = 1$, we have $w^T x = w^T\bar{x} + \beta^* w^T v \geq w^T\bar{x} - \beta^*\|w\|_*\|v\| \geq \beta^* - \beta^* = 0$, and so $B(\bar{x}, \beta^*) \subset C$. Therefore, $\tau \geq \frac{\beta^*}{\|\bar{x}\|} = \beta^*$.

From the definition of the width of C, there exists $\tilde{x}$ satisfying $\|\tilde{x}\| = 1$ and $B(\tilde{x}, \tau) \subset C$. Let $w \in C^*$ satisfying $\|w\|_* = 1$ be given. Then, from the duality properties of norms, there exists $\bar{v} \in X$ satisfying $\|\bar{v}\| \leq 1$ for which $\|w\|_* = w^T\bar{v}$. Since $B(\tilde{x}, \tau) \subset C$, $\tilde{x} - \tau\bar{v} \in C$, and so $w^T(\tilde{x} - \tau\bar{v}) \geq 0$, whereby $w^T\tilde{x} \geq \tau w^T\bar{v} = \tau\|w\|_* = \tau$. As this is true for any given $w \in C^*$ satisfying $\|w\|_* = 1$, it follows that $\beta^* \geq \tau$, completing the proof. □

We illustrate the width construction on two families of cones, the nonnegative orthant $\Re^k_+$ and the positive semidefinite cone $S^{k\times k}_+$. First consider the nonnegative orthant. Let $X = \Re^k$ with Euclidean norm $\|x\| = \|x\|_2 = \sqrt{x^T x}$, and $C = \Re^k_+ = \{x \in \Re^k \mid x \geq 0\}$. Then it is straightforward to show, by setting $x = e = (1, \ldots, 1)^T$, that the width of $\Re^k_+$ is $\tau = 1/\sqrt{k}$. Next consider the positive semidefinite cone. Let $X = S^{k\times k}$ denote the set of real $k \times k$ symmetric matrices with Frobenius norm $\|x\| := \sqrt{\text{trace}(x^T x)}$, and let $C = S^{k\times k}_+ = \left\{x \in S^{k\times k} \mid x \succeq 0\right\}$. Then $S^{k\times k}_+$ is a closed convex cone, and it is easy to show by setting $x = I$ that the width of $S^{k\times k}_+$ is $\tau = \frac{1}{\sqrt{k}}$.

For the remainder of this paper, we amend our notation as follows.

DEFINITION 2.3. *Whenever the cone C_X is regular, the width of C_X is denoted by τ, and the width of C^*_X is denoted by τ^*. Whenever the cone C_Y is regular, the width C_Y is denoted by $\bar{\tau}$, and the width of C^*_Y is denoted by $\bar{\tau}^*$.*

3. A ball construction for the ϵ-optimal set for $CP(d)$. In this section we demonstrate some valuable geometric properties of the set of ϵ-optimal solutions of $CP(d)$ that will be used later to obtain complexity bounds for the ellipsoid algorithm. Let X_d denote the feasible region of $CP(d)$ and let $z^*(d)$ denote the optimal objective function value of $CP(d)$. For any $\epsilon > 0$, denote the set of ϵ-optimal solutions of $CP(d)$ by X^ϵ_d, i.e., $X^\epsilon_d = \{x \in X \mid x \in X_d \text{ and } c^T x \geq z^*(d) - \epsilon\}$.

Let $\epsilon > 0$ be given. The following lemma asserts the existence of a ball in the set of ϵ-optimal solutions of $CP(d)$ that has certain geometric properties, under the condition that the feasible region contains a ball $B(\hat{x}, r)$.

LEMMA 3.1. *Suppose that the feasible region X_d contains the ball $B(\hat{x}, r)$, where $r > 0$. Let x^* be an optimal solution of $CP(d)$, and let $\epsilon > 0$ be given. Then there exists a ball $B(\bar{x}, \bar{r})$ with the following properties:*

$$\text{(i)} \quad B(\bar{x}, \bar{r}) \subset X^\epsilon_d \ ,$$

$$\text{(ii)} \quad \bar{r} \geq \frac{\epsilon r}{\max\{\epsilon, z^*(d) - c^T\hat{x} + r\|c\|_*\}} \ ,$$

$$\textit{and} \quad \text{(iii)} \quad \|\bar{x}\| \leq \max\left\{\|\hat{x}\|, \|x^*\|\right\}.$$

Proof. We have $B(\hat{x}, r) \subset X_d$ and $x^* \in X_d$. Therefore, from the convexity of X_d, we have

$$B(\alpha\hat{x} + (1-\alpha)x^*, \alpha r) \subset X_d \quad \text{for any } \alpha \in [0, 1]. \tag{9}$$

We have two cases.

Case 1. $\epsilon \leq z^*(d) - c^T\hat{x} + r\|c\|_*$. Define

$$\alpha = \frac{\epsilon}{z^*(d) - c^T\hat{x} + r\|c\|_*} \ , \quad \bar{x} = \alpha\hat{x} + (1-\alpha)x^* \ , \quad \text{and} \quad \bar{r} = \alpha r.$$

Then $\alpha \in [0, 1]$ and so $B(\bar{x}, \bar{r}) \subset X_d$ from (9). Furthermore, for any $x \in B(\bar{x}, \bar{r})$, we have

$$c^T x \geq \alpha c^T\hat{x} + (1-\alpha)c^T x^* - \alpha r\|c\|_* = z^*(d) - \alpha\left(z^*(d) - c^T\hat{x} + r\|c\|_*\right) = z^*(d) - \epsilon,$$

whereby (i) is satisfied. For (ii), note that

$$\bar{r} = \alpha r = \frac{\epsilon r}{z^*(d) - c^T\hat{x} + r\|c\|_*} = \frac{\epsilon r}{\max\{\epsilon, z^*(d) - c^T\hat{x} + r\|c\|_*\}}.$$

Part (iii) follows since $\|\bar{x}\| = \|\alpha\hat{x} + (1-\alpha)x^*\| \leq \max\left\{\|\hat{x}\|, \|x^*\|\right\}$.

Case 2. $\epsilon > z^*(d) - c^T\hat{x} + r\|c\|_*$. Define

$$\bar{x} = \hat{x} \quad \text{and} \quad \bar{r} = r.$$

To prove (i), note that for any $x \in B(\hat{x}, r)$, we have

$$c^T x \geq c^T\hat{x} - r\|c\|_* = z^*(d) - \big(z^*(d) - c^T\hat{x} + r\|c\|_*\big) > z^*(d) - \epsilon,$$

whereby (i) is satisfied. Parts (ii) and (iii) follow trivially. □

We would like to apply Lemma 3.1 to obtain a lower bound on the volume of the set of ϵ-optimal solutions of $CP(d)$. However, in order to obtain such a lower bound via Lemma 3.1, we need the following ingredients:

(i) an upper bound on the optimal objective function value $z^*(d)$ of $CP(d)$,

(ii) an upper bound on the norm of an optimal solution x^* of $CP(d)$, and

(iii) the existence of a ball $B(\hat{x}, r)$ in the feasible region for which there is an upper bound on $\|\hat{x}\|$ and a lower bound on r.

The following previously derived results pertain to the first two conditions above.

THEOREM 1 OF [17]. *Suppose that $d \in \mathcal{F}$ and $\mathcal{C}(d) < +\infty$. Then*

$$|z^*(d)| \leq \|c\|_*\mathcal{C}(d). \tag{10}$$

Furthermore, $CP(d)$ attains its optimum and every optimal solution x^* satisfies

$$\|x^*\| \leq \mathcal{C}(d)^2 \ . \quad □ \tag{11}$$

The third condition above is treated with the following previously known results.

THEOREM 5.1 OF [7] *Suppose that C_X is a regular cone and C_Y is a regular cone and that $d \in \mathcal{F}$ and that $\mathcal{C}(d) < +\infty$. Then there exists $\hat{x} \in X_d$ and a scalar $r > 0$ such that $B(\hat{x}, r) \subset X_d$, and*

$$r \geq \frac{\min\{\tau, \bar{\tau}\}}{6\mathcal{C}(d)} \ , \quad \|\hat{x}\| \leq \frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \ , \quad \textit{and} \quad \frac{\|\hat{x}\|}{r} \leq \frac{6\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \ . \quad □ \tag{12}$$

THEOREM 5.3 OF [7] *Suppose that C_X is a regular cone and $C_Y = \{0\}$ and that $d \in \mathcal{F}$ and that $\mathcal{C}(d) < +\infty$. Then there exists $\hat{x} \in X_d$ and a scalar $r > 0$ such that $\{x \in X \mid \|x - \hat{x}\| \leq r, Ax = b\} \subset X_d$, and*

$$r \geq \frac{\tau}{3\mathcal{C}(d)} \ , \quad \|\hat{x}\| \leq \frac{4\mathcal{C}(d)}{\tau} \ , \quad \textit{and} \quad \frac{\|\hat{x}\|}{r} \leq \frac{3\mathcal{C}(d)}{\tau} \ . \quad □ \tag{13}$$

THEOREM 5.5 OF [7] *Suppose that $C_X = X$ and C_Y is a regular cone, that $d \in \mathcal{F}$, and that $\mathcal{C}(d) < +\infty$. Then there exists $\hat{x} \in X_d$ and a scalar $r > 0$ such that $B(\hat{x}, r) \subset X_d$, and*

$$r \geq \frac{\bar{\tau}}{3\mathcal{C}(d)} \ , \quad \|\hat{x}\| \leq \frac{3\mathcal{C}(d)}{\bar{\tau}} \ , \quad \textit{and} \quad \frac{\|\hat{x}\|}{r} \leq \frac{2\mathcal{C}(d)}{\bar{\tau}} \ . \quad □ \tag{14}$$

(These three results are slightly altered from their presentation in [7], which uses the notation of coefficients of linearity. In the notation of [7], we have from Proposition 2.1 that $\tau = \beta^*, \tau^* = \beta, \bar{\tau} = \bar{\beta}^*$, and $\bar{\tau}^* = \bar{\beta}$. The above statements follow by noticing from [7] that $\mathcal{C}(d) \geq 1, \tau \leq 1, \bar{\tau} \leq 1$, and $\frac{\|\hat{x}\|}{r} \leq \frac{R}{r} - 1$.)

4. The ellipsoid algorithm and a homogenizing transformation.

We review a few basic results regarding the ellipsoid algorithm for solving an optimization problem; see [26], [21], [9], [4], [8], [3]. We will consider the following optimization problem:

$$P: \quad \underset{x}{\text{maximize}} \quad f(x) \qquad \text{s.t.} \quad x \in S, \tag{15}$$

where S is a convex set (closed or not) in $\Re^k$, $f(x)$ is a quasi-concave function, and $\|x\|_2 := \sqrt{x^T x}$ is the Euclidean norm. Actually, the ellipsoid algorithm is more usually associated with the assumption that S is a closed convex set and also that $f(x)$ is a concave function, but these assumptions can be relaxed slightly. It is only necessary that S be a convex set, that the upper level sets of $f(x)$ be convex sets on S (which is equivalent to the statement that $f(x)$ is a quasi-concave function on S; see [2], for example), and that a separation oracle be available for S as well as for each of the upper level sets of $f(x)$. (Note that if $f(x)$ is a differentiable quasi-concave function, then $\nabla f(x)$ furnishes a separation oracle for the upper level sets of $f(x)$, provided that $\nabla f(x)$ does not vanish at any nonmaximizing points.)

In order to implement the ellipsoid algorithm to approximately solve P, it is necessary that one has available a separation oracle for the set S, i.e., that for any $\bar{x} \notin S$, one can perform a feasibility cut for the set S, which consists of computing a vector $v \neq 0$ for which $S \subset \{x \mid v^T x \geq v^T \bar{x}\}$. Suppose that T_1 is an upper bound on the number of operations needed to perform a feasibility cut for the set S. It is also necessary that one has available a support oracle for the upper level sets $U_\alpha = \{x \in S \mid f(x) \geq \alpha\}$ of the quasi-concave function $f(x)$. That is, for any $\bar{x} \in S$, it is necessary to be able to perform an optimality cut for the objective function $f(x)$ at any point $\bar{x} \in S$, which consists of computing a vector $v \neq 0$ for which $U_{f(\bar{x})} \subset \{x \in \Re^k \mid v^T x \geq v^T \bar{x}\}$. Suppose that T_2 is an upper bound on the number of operations needed to compute an optimality cut for the function $f(x)$ on the set S.

Let z^* denote the optimal value of P, and denote the set of ϵ-optimal solutions of P by S^ϵ, i.e., $S^\epsilon = \{x \in \Re^k \mid x \in S \text{ and } f(x) \geq z^* - \epsilon\}$. In a typical application of the ellipsoid algorithm, we wish to find an ϵ-optimal solution of P. Suppose that we know a priori a positive scalar R with the property that

$$B(0,R) \cap S^\epsilon$$

has positive volume, where $B(\bar{x}, r) := \{x \in \Re^k \mid \|x - \bar{x}\|_2 \leq r\}$ is the Euclidean ball centered at $\bar{x}$ with radius r. Then the ellipsoid algorithm for solving P can be initiated with the Euclidean ball $B(0,R)$. The following is a generic result about the performance of the ellipsoid algorithm, where in the statement of the theorem, "vol(Q)" denotes the volume of a set Q.

Ellipsoid Algorithm Theorem with Known ***R*** (from [26], [21]). *Suppose that a positive scalar R is known with the property that the set*

$$F := B(0,R) \cap S^\epsilon$$

has positive volume. Then, if the ellipsoid algorithm is initiated with the Euclidean ball $B(0,R)$, the algorithm will compute an ϵ-optimal solution of P in at most

$$\left\lceil 2(k+1) \ln\left(\frac{\text{vol}(B(0,R))}{\text{vol}(B(0,R) \cap S^\epsilon)}\right)\right\rceil \tag{16}$$

iterations, where each iteration must perform at most $\left(k^2 + \max\{T_1, T_2\}\right)$ *operations, and where* T_1 *and* T_2 *are the numbers of operations needed to perform a feasibility cut on* S *and an optimality cut on* $f(x)$*, respectively.*

We note that the bound on the number of operations per iteration arises from performing either a feasibility or an optimality cut (which takes $\max\{T_1, T_2\}$ operations), and then performing a rank-one update of the positive definite matrix defining the ellipsoid (see [3], for example), which takes k^2 operations.

Because an a priori bound on R is typically not known except in very special cases of P, we employ a standard homogenizing transformation to convert P to the homogenized fractional program:

$$\text{(17)} \qquad \begin{array}{lll} HP: & \underset{w,\theta}{\text{maximize}} & g(w,\theta) := f(w/\theta) \\ & \text{s.t.} & w \in \theta S, \\ & & \theta > 0 \end{array}$$

(see, for example, [5] and [6]). It is trivial to show that z^* is the common optimal objective function value of P and HP. Let H and H^ϵ denote the set of feasible and ϵ-optimal solutions of HP, respectively, i.e.,

$$\text{(18)} \qquad H = \{(w,\theta) \in \Re^{k+1} \mid w \in \theta S,\ \theta > 0\}$$

and

$$\text{(19)} \qquad H^\epsilon = \{(w,\theta) \in \Re^{k+1} \mid w \in \theta S,\ \theta > 0,\ g(w,\theta) \geq z^* - \epsilon\}.$$

Then H and H^ϵ are both convex sets. Furthermore, the objective function $g(w,\theta) := f(w/\theta)$ of HP is easily seen to be a quasi-concave function over the feasible region H whenever $f(x)$ is a quasi-concave function over the feasible region S. The following (obvious) transformations $h(\cdot)$ and $h^{-1}(\cdot)$ map the feasible regions and ϵ-optimal regions of P and HP onto one another:

$$\text{(20)} \qquad h(T) = \{(w,\theta) \in \Re^{k+1} \mid w/\theta \in T \text{ and } \theta > 0\} \qquad \text{for any } T \subset S$$

and

$$\text{(21)} \quad h^{-1}(W) = \{x \in \Re^k \mid x = w/\theta \text{ for some } (w,\theta) \in W\} \qquad \text{for any } W \subset H\ .$$

Because any feasible solution of HP can be scaled by an arbitrary positive scalar without changing its objective function value or affecting its feasibility, the feasible region and all upper level sets of the objective function $g(w,\theta)$ of HP contain points in the $(k+1)$-dimensional unit Euclidean ball. This allows us to conveniently start the ellipsoid algorithm for solving HP with the $(k+1)$-dimensional unit Euclidean ball.

The following result concerns volumes of subsets of S under the projective transformation $h(\cdot)$ and provides the final ingredient we will need for our analysis of the ellipsoid algorithm. Let B^{k+1} denote the $(k+1)$-dimensional unit Euclidean ball, namely,

$$B^{k+1} := \left\{(w,\theta) \in \Re^{k+1} \mid \sqrt{w^T w + \theta^2} \leq 1\right\}.$$

Lemma 4.1. *Suppose that S is a convex set in $\Re^k$, that $T \subset S$ is given, that there exists $\bar{r} > 0$ and $\bar{x}$ for which $B(\bar{x}, \bar{r}) \subset T$, and that $\bar{r} \le 1$. Let $W = h(T)$, where $h(\cdot)$ is defined as in* (20)*. Then*

$$\ln\left(\frac{\operatorname{vol}\left(B^{k+1}\right)}{\operatorname{vol}\left(B^{k+1} \cap W\right)}\right) \le (k+1)\ln\left(2 + \frac{3(\|\bar{x}\| + 1)}{\bar{r}}\right) + \left[\ln\left(\|\bar{x}\|\right)\right]^+ .$$

Proof. We first define two constants,

$$\delta = \max\{\|\bar{x}\|, 1\}$$

and

$$\gamma = 1 + \frac{\bar{r}}{3} + \frac{\bar{r}}{3\delta} + \|\bar{x}\|,$$

and we define the following ellipsoid centered at $(\bar{x}, 1) \in \Re^{k+1}$:

$$E = \left\{(w, \theta) \in \Re^{k+1} \mid \sqrt{(w - \bar{x})^T(w - \bar{x}) + \delta^2(\theta - 1)^2} \le \frac{\bar{r}}{3}\right\} .$$

We prove below that

$$E \subset W, \tag{22}$$

$$E \subset \gamma B^{k+1}. \tag{23}$$

It then follows that

$$\gamma^{-1}E \subset B^{k+1} \quad \text{and} \quad \gamma^{-1}E \subset W, \tag{24}$$

and so

$$\gamma^{-1}E \subset B^{k+1} \cap W, \tag{25}$$

since in particular W is closed under positive scalings. Then

$$\begin{aligned}
\ln\left(\frac{\operatorname{vol}\left(B^{k+1}\right)}{\operatorname{vol}\left(B^{k+1} \cap W\right)}\right) &\le \ln\left(\frac{\operatorname{vol}\left(B^{k+1}\right)}{\operatorname{vol}\left(\gamma^{-1}E\right)}\right) \\
&= (k+1)\ln(\gamma) + \ln\left(\frac{\operatorname{vol}\left(B^{k+1}\right)}{\operatorname{vol}\left(E\right)}\right) \\
&= (k+1)\ln(\gamma) + \ln\left(\frac{1}{\left(\frac{\bar{r}}{3}\right)^{k+1}\left(\frac{1}{\delta}\right)}\right) \\
&= (k+1)\ln\left(\frac{3\gamma}{\bar{r}}\right) + \ln\left(\delta\right) \\
&= (k+1)\ln\left(\frac{3}{\bar{r}} + 1 + \frac{1}{\delta} + \frac{3\|\bar{x}\|}{\bar{r}}\right) + \ln\left(\delta\right) \\
&\le (k+1)\ln\left(2 + \frac{3(\|\bar{x}\| + 1)}{\bar{r}}\right) + \left[\ln\left(\|\bar{x}\|\right)\right]^+ ,
\end{aligned}$$

since $\delta \geq 1$. We therefore need to demonstrate (22) and (23) to complete the proof.

For any $(w,\theta) \in E$, $(w,\theta) = (\bar{x}+q, 1+v)$, where $\|q\| \leq \frac{\bar{r}}{3}$ and $|v| \leq \frac{\bar{r}}{3\delta} \leq \frac{1}{3}$, since $\bar{r} \leq 1$ and $\delta \geq 1$, and so $\theta \geq \frac{2}{3} > 0$. We also have

$$\frac{w}{\theta} = \frac{\bar{x}+q}{1+v} = \bar{x} + \frac{q - v\bar{x}}{1+v} \ ,$$

and so

$$\left\| \frac{w}{\theta} - \bar{x} \right\| = \frac{\|q - v\bar{x}\|}{1+v} \leq \frac{3}{2}\left(\|q\| + |v|\|\bar{x}\|\right) \leq \frac{3}{2}\left(\frac{\bar{r}}{3} + \frac{\bar{r}}{3\delta}\|\bar{x}\|\right) \leq \bar{r} \ .$$

Therefore, $\frac{w}{\theta} \in B(\bar{x}, \bar{r})$, whereby $\frac{w}{\theta} \in T$, and so $w \in \theta T$ or, equivalently, $(w,\theta) \in h(T)$. Therefore, $E \subset h(T) = W$, proving (22).

To prove (23), let $(w,\theta) \in E$. Then $\|w - \bar{x}\|_2 \leq \frac{\bar{r}}{3}$ and $|\theta - 1| \leq \frac{\bar{r}}{3\delta}$. Therefore,

$$\begin{aligned}
\|(w,\theta)\|_2 &\leq \|(w-\bar{x}, \theta - 1)\|_2 + \|(\bar{x},1)\|_2 \\
&\leq \|w - \bar{x}\|_2 + |\theta - 1| + \|\bar{x}\|_2 + 1 \\
&\leq \frac{\bar{r}}{3} + \frac{\bar{r}}{3\delta} + \|\bar{x}\|_2 + 1 \\
&= \gamma \ ,
\end{aligned}$$

and so $(w,\theta) \in \gamma B^{k+1}$, which proves (23) and thus the proof of the lemma is complete. □

It is trivial to show that a separation oracle for S can be readily converted to a separation oracle for H. If T_1 is the number of operations needed to compute a feasibility cut for S, then one needs $O(T_1 + k)$ operations to compute a feasibility cut for H. Furthermore, any support oracle for the upper level sets of $f(x)$ over S can be readily converted to a support oracle for the upper level sets of $g(w,\theta)$ over H. To see why this is true, suppose that $(\bar{w},\bar{\theta})$ is a feasible solution of HP, and define $\bar{x} = \bar{w}/\bar{\theta}$. Then $\bar{x}$ is feasible for P and let v be the vector produced by the support oracle for $f(x)$ at $x = \bar{x}$. Then

$$\{x \in S \mid f(x) \geq f(\bar{x})\} \subset \left\{x \in \Re^k \mid v^T x \geq v^T \bar{x}\right\},$$

which implies that

$$\left\{(w,\theta) \in H \mid g(w,\theta) \geq g(\bar{w},\bar{\theta})\right\} \subset \left\{(w,\theta) \in \Re^{k+1} \mid v^T w - ((v^T\bar{w})/\bar{\theta})\theta \geq 0\right\},$$

and so the concatenated vector $(v, -(v^T\bar{w}/\bar{\theta}))$ is a support vector for the upper level set of the function $g(w,\theta)$ at the feasible point $(\bar{w},\bar{\theta})$. If T_2 is the number of operations needed to compute an optimality cut on $f(x)$ over S, then one needs $O(T_2 + k)$ operations to compute an optimality cut on $g(w,\theta)$ over H.

Finally, returning to the problem $CP(d)$, note that the homogenized problem corresponding to $CP(d)$ is

$$(26) \qquad \begin{array}{rll}
HP(d): & \text{maximize}_{w,\theta} & g(w,\theta) := \dfrac{c^T w}{\theta} \\
& \text{s.t.} & b\theta - Aw \in C_Y, \\
& & w \in C_X, \\
& & \theta > 0 \ ,
\end{array}$$

which we refer to as $HP(d)$.

5. Complexity results. In this section, we assume that $X = \Re^n$ is endowed with the Euclidean norm $\|x\| = \|x\|_2 = \sqrt{x^T x}$. For the purpose of developing complexity results, we focus on three different classes of instances of $CP(d)$, namely,

Class (i): C_X and C_Y are both regular;

Class (ii): C_X is regular and $C_Y = \{0\}$;

Class (iii): $C_X = X$ and C_Y is regular.

For these three classes of instances, $CP(d)$ can be written as (i) $\max\{c^T x \mid b - Ax \in C_Y, x \in C_X\}$, (ii) $\max\{c^T x \mid Ax = b, x \in C_X\}$, and (iii) $\max\{c^T x \mid b - Ax \in C_Y, x \in X\}$, respectively.

The following three theorems contain iteration complexity bounds on the ellipsoid algorithm for these three classes of instances of $CP(d)$, respectively. The proofs of the theorems are deferred to the end of the section.

THEOREM 5.1. *Suppose that C_X is a regular cone with width τ, that C_Y is a regular cone with width $\bar{\tau}$, and that $d \in \mathcal{F}$ and $\mathcal{C}(d) < +\infty$. Let ϵ satisfying $0 < \epsilon < \|c\|_*$ be given. Suppose that the ellipsoid algorithm is applied to solve $HP(d)$ and is initiated with the Euclidean unit ball centered at $(w^0, \theta^0) = (0, 0)$. Then the ellipsoid algorithm will compute an ϵ-optimal solution of $HP(d)$ (and hence, by transformation, to $CP(d)$) in at most*

$$\left\lceil 8(n+2)^2 \ln\left(\frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \frac{\|c\|_*}{\epsilon}\right)\right\rceil$$

iterations, where each iteration must perform at most $\left((n+1)^2 + \max\{2n, S_1, m + mn + S_2\}\right)$ operations, and where S_1 and S_2 are the number of operations needed to perform a feasibility cut on C_X and C_Y, respectively.

THEOREM 5.2. *Suppose that C_X is a regular cone with width τ, that $C_Y = \{0\}$, and that $d \in \mathcal{F}$ and $\mathcal{C}(d) < +\infty$. Let ϵ satisfying $0 < \epsilon < \|c\|_*$ be given. Suppose that the ellipsoid algorithm is applied to solve $HP(d)$ and is initiated with the Euclidean unit disk centered at $(w^0, \theta^0) = (0, 0)$ in the subspace $\{(w, \theta) \in \Re^{n+1} \mid Aw - b\theta = 0\}$. Then the ellipsoid algorithm will compute an ϵ-optimal solution of $HP(d)$ (and hence, by transformation, to $CP(d)$) in at most*

$$\left\lceil 8(n-m+2)^2 \ln\left(\frac{3\mathcal{C}(d)}{\tau} \frac{\|c\|_*}{\epsilon}\right)\right\rceil$$

iterations, where each iteration must perform at most $\left((n-m+1)^2 + \max\{2n, S_1\}\right)$ operations, and where S_1 is the number of operations needed to perform a feasibility cut on C_X.

THEOREM 5.3. *Suppose that $C_X = X$ and C_Y is a regular cone with width $\bar{\tau}$, and that $d \in \mathcal{F}$ and $\mathcal{C}(d) < +\infty$. Let ϵ satisfying $0 < \epsilon < \|c\|_*$ be given. Suppose that the ellipsoid algorithm is applied to solve $HP(d)$, and is initiated with the Euclidean unit ball centered at $(w^0, \theta^0) = (0, 0)$. Then the ellipsoid algorithm will compute an ϵ-optimal solution of $HP(d)$ (and hence, by transformation, to $CP(d)$) in at most*

$$\left\lceil 8(n+2)^2 \ln\left(\frac{3\mathcal{C}(d)}{\bar{\tau}} \frac{\|c\|_*}{\epsilon}\right)\right\rceil$$

iterations, where each iteration must perform at most $\big((n+1)^2 + \max\{2n, m + mn + S_2\}\big)$ *operations and where* S_2 *is the number of operations needed to perform a feasibility cut on* C_Y*.*

Proof of Theorem 5.1. Let

$$a_1 = \frac{6}{\min\{\tau, \bar{\tau}\}}, \qquad a_2 = \frac{4}{\min\{\tau, \bar{\tau}\}}, \qquad \text{and} \quad a_3 = \frac{6}{\min\{\tau, \bar{\tau}\}} \ . \tag{27}$$

Then, from (12), we have that there exists $\hat{x}$ and $r > 0$ such that $B(\hat{x}, r) \subset X_d$, and

$$\frac{1}{r} \le a_1 \mathcal{C}(d), \quad \|\hat{x}\| \le a_2 \mathcal{C}(d) \ , \quad \text{and} \quad \frac{\|\hat{x}\|}{r} \le a_3 \mathcal{C}(d) \ . \tag{28}$$

Applying Lemma 3.1, X_d^ϵ contains a ball $B(\bar{x}, \bar{r})$ with the following properties:

$$\frac{1}{\bar{r}} \le \frac{\max\{\epsilon, z^*(d) - c^T\hat{x} + r\|c\|_*\}}{\epsilon r} \quad \text{and} \quad \|\bar{x}\| \le \max\left\{\|\hat{x}\|, \|x^*\|\right\}, \tag{29}$$

where x^* is any optimal solution of $CP(d)$. Furthermore, from (10) and (11), we have $|z^*(d)| \le \|c\|_* \mathcal{C}(d)$ and $\|x^*\| \le \mathcal{C}(d)^2$.

Examining the first inequality of (29), notice that

$$\frac{\max\{\epsilon, z^*(d) - c^T\hat{x} + r\|c\|_*\}}{\epsilon r} \ge \frac{\|c\|_*}{\epsilon} \ge 1 \ .$$

If $\bar{r} > 1$, we can reset $\bar{r} = 1$ and (29) will still hold. Therefore, there is no loss of generality in assuming that $\bar{r} \le 1$.

The dimension in which the ellipsoid algorithm is implemented is $n+1$. Let H_d^ϵ denote the set of ϵ-optimal solutions of $HP(d)$, and so H_d^ϵ is the image of X_d^ϵ under the transformation $h(\cdot)$ of (20). Then, from the ellipsoid algorithm theorem (16), the algorithm will compute an ϵ-optimal solution of $HP(d)$ in at most

$$\left\lceil 2(n+2) \ln\left(\frac{\mathrm{vol}(B^{n+1})}{\mathrm{vol}(B^{n+1} \cap H_d^\epsilon)}\right)\right\rceil \tag{30}$$

iterations, where B^{n+1} is the $(n+1)$-dimensional Euclidean unit ball.

Now let $T = X_d^\epsilon$. Then $H_d^\epsilon = h(T)$ and $B(\bar{x}, \bar{r}) \subset X_d^\epsilon$. Furthermore, $\bar{r} \le 1$ from the comments above. We therefore can apply Lemma 4.1 to bound the logarithm term of (30):

$$\ln\left(\frac{\mathrm{vol}\left(B^{n+1}\right)}{\mathrm{vol}\left(B^{n+1} \cap H_d^\epsilon\right)}\right) \le (n+1) \ln\left(2 + \frac{3(\|\bar{x}\| + 1)}{\bar{r}}\right) + \left[\ln\left(\|\bar{x}\|\right)\right]^+ . \tag{31}$$

We now bound the relevant quantities in (31) in order to obtain the desired bound on (30).

From (29) we have

$$\begin{aligned}
\frac{\|\bar{x}\|}{\bar{r}} \quad &\le \quad \frac{1}{\epsilon} \max\left\{\|\hat{x}\|, \|x^*\|\right\} \max\left\{\frac{\epsilon}{r} \ , \frac{z^*(d) - c^T\hat{x}}{r} + \|c\|_*\right\} \\
&\le \quad \frac{1}{\epsilon} \max\left\{\|\hat{x}\|, \|x^*\|\right\} \left(\max\left\{\frac{\epsilon}{r} \ , \frac{z^*(d) - c^T\hat{x}}{r}\right\} + \|c\|_*\right) \\
&\le \quad \frac{1}{\epsilon} \max\left\{\frac{\|\hat{x}\|}{r} \ , \frac{\|x^*\|}{r}\right\} \max\left\{\epsilon, z^*(d) - c^T\hat{x}\right\} + \frac{1}{\epsilon} \max\left\{\|\hat{x}\|, \|x^*\|\right\} \|c\|_* \ .
\end{aligned}$$

Substituting in the bounds from (28), (10), and (11) and recalling that $\epsilon \le \|c\|_*$, we obtain from the above inequality

$$\begin{aligned}\frac{\|\bar{x}\|}{\bar{r}} &\le \frac{1}{\epsilon}\max\left\{a_3\mathcal{C}(d), a_1\mathcal{C}(d)^3\right\}\max\left\{\|c\|_*, \|c\|_*\mathcal{C}(d) + \|c\|_* a_2\mathcal{C}(d)\right\} \\ &+ \frac{1}{\epsilon}\max\left\{a_2\mathcal{C}(d), \mathcal{C}(d)^2\right\}\|c\|_* \ ,\end{aligned}$$

whereby we obtain

$$\frac{\|\bar{x}\|}{\bar{r}} \le \frac{\|c\|_*}{\epsilon}\mathcal{C}(d)^4\left[(1+a_2)\left(\max\left\{a_1, a_3\right\}\right) + a_2\right] \ . \tag{32}$$

From (29) we have

$$\begin{aligned}\frac{1}{\bar{r}} &\le \frac{\max\{\epsilon, z^*(d) - c^T\hat{x} + r\|c\|_*\}}{\epsilon r} \\ &= \frac{1}{\epsilon}\max\left\{\frac{\epsilon}{r}\ , \frac{z^*(d) - c^T\hat{x}}{r} + \|c\|_*\right\} \\ &\le \frac{1}{\epsilon}\left(\max\left\{\frac{\|c\|_*}{r}\ , \frac{z^*(d) - c^T\hat{x}}{r}\right\} + \|c\|_*\right) \\ &\le \frac{1}{\epsilon}\left[\max\left\{\frac{\|c\|_*}{r}\ , \frac{\|c\|_*\mathcal{C}(d)}{r} + \frac{\|c\|_*\|\hat{x}\|}{r}\right\} + \|c\|_*\right] \qquad \text{(from (10))} \\ &\le \frac{\|c\|_*}{\epsilon}\left[\max\left\{a_1\mathcal{C}(d), a_1\mathcal{C}(d)^2 + a_3\mathcal{C}(d)\right\} + 1\right] \qquad \text{(from (28))},\end{aligned}$$

and so

$$\frac{1}{\bar{r}} \le \frac{\|c\|_*}{\epsilon}\mathcal{C}(d)^2\left(a_1 + a_3 + 1\right) \ . \tag{33}$$

We also have from (29) that

$$\|\bar{x}\| \le \max\left\{\|\hat{x}\|, \|x^*\|\right\} \le \max\left\{a_2\mathcal{C}(d), \mathcal{C}(d)^2\right\} \le a_2\mathcal{C}(d)^2 \ . \tag{34}$$

Substituting (32), (33), and (34) into (31) and then substituting (31) into (30) yields the following iteration bound on the ellipsoid algorithm:

(35)

$$\left\lceil 2(n+2)\left[(n+1)\ln\left(2 + \frac{3\|c\|_*}{\epsilon}\mathcal{C}(d)^4\left(1 + a_1 + a_2 + a_3 + (1+a_2)\max\left\{a_1, a_3\right\}\right)\right) + \ln(a_2\mathcal{C}(d)^2)\right]\right\rceil \ .$$

Substituting (27) into (35), we obtain the following chain of upper bounds on the iteration bound:

$$\left\lceil 2(n+2)\left\{(n+1)\ln\left(2+\frac{141\|c\|_*}{\epsilon(\min\{\tau,\bar{\tau}\})^2}\mathcal{C}(d)^4\right)+\ln\left(\frac{4}{\min\{\tau,\bar{\tau}\}}\mathcal{C}(d)^2\right)\right\}\right\rceil$$

$$\leq \left\lceil 2(n+2)^2\ln\left(\frac{143\|c\|_*}{\epsilon}\left(\frac{\mathcal{C}(d)}{\min\{\tau,\bar{\tau}\}}\right)^4\right)\right\rceil$$

$$\leq \left\lceil 8(n+2)^2\ln\left(\frac{4\mathcal{C}(d)}{\min\{\tau,\bar{\tau}\}}\frac{\|c\|_*}{\epsilon}\right)\right\rceil .$$

The number of operations needed to perform an optimality cut in $HP(d)$ is at most $2n$, since an upper level set support vector for $g(w,\theta)$ at a feasible point $(\bar{w},\bar{\theta})$ of $HP(d)$ is computed as $(c, -(c^T\bar{w}/\bar{\theta}))$, and the number of operations needed to compute and test for feasibility of $b\theta - Aw \in C_Y$ is $(m + mn + S_2)$. □

The proofs of Theorems 5.2 and 5.3 are accomplished by slightly modifying the analysis in the proof of Theorem 5.1 as per the following remark.

REMARK 5.1. *Note in the proof of Theorem 5.1 that the ellipsoid algorithm iteration bound in* (35) *was derived based only on the following facts: the feasible region of $CP(d)$ contains a ball $B(\hat{x},r)$ satisfying $\frac{1}{r} \leq a_1\mathcal{C}(d)$, $\|\hat{x}\| \leq a_2\mathcal{C}(d)$, and $\frac{\|\hat{x}\|}{r} \leq a_3\mathcal{C}(d)$; $|z^*(d)| \leq \|c\|_*\mathcal{C}(d)$; and there exists an optimal solution x^* of $CP(d)$ satisfying $\|x^*\| \leq \mathcal{C}(d)^2$.*

This remark will be used in the proofs of Theorems 5.2 and 5.3, which we now do in reverse order.

Proof of Theorem 5.3. Let

$$a_1 = \frac{3}{\bar{\tau}}, \quad a_2 = \frac{3}{\bar{\tau}}, \quad \text{and} \quad a_3 = \frac{2}{\bar{\tau}}\ . \tag{36}$$

Then from (14) we know that the feasible region of $CP(d)$ contains a ball $B(\hat{x},r)$ satisfying $\frac{1}{r} \leq a_1\mathcal{C}(d)$, $\|\hat{x}\| \leq a_2\mathcal{C}(d)$, and $\frac{\|\hat{x}\|}{r} \leq a_3\mathcal{C}(d)$. Also, from (10), we have $|z^*(d)| \leq \|c\|_*\mathcal{C}(d)$. Furthermore, from (11), there exists an optimal solution x^* of $CP(d)$ satisfying $\|x^*\| \leq \mathcal{C}(d)^2$. Then, from Remark 5.1, the iteration bound of (35) is valid with values of a_1, a_2, and a_3 from (36). Substituting (36) into (35) yields the following iteration bound:

$$\left\lceil 2(n+2)\left\{(n+1)\ln\left(2+\frac{63\|c\|_*}{\epsilon\bar{\tau}^2}\mathcal{C}(d)^4\right)+\ln\left(\frac{3}{\bar{\tau}}\mathcal{C}(d)^2\right)\right\}\right\rceil$$

$$\leq \left\lceil 2(n+2)^2\ln\left(\frac{65\|c\|_*}{\epsilon}\left(\frac{\mathcal{C}(d)}{\bar{\tau}}\right)^4\right)\right\rceil$$

$$\leq \left\lceil 8(n+2)^2\ln\left(\frac{3\mathcal{C}(d)}{\bar{\tau}}\frac{\|c\|_*}{\epsilon}\right)\right\rceil . \qquad □$$

Proof of Theorem 5.2. The feasible region of $CP(d)$ lies in the affine set $\{x \in \Re^n \mid Ax = b\}$. In order to apply the ellipsoid algorithm conveniently, we construct a Euclidean-norm-preserving linear transformation to $\Re^{(n-m)}$. For concreteness, we assume with no loss of generality that A is an $m \times n$ real matrix. Let F be an $(n-m) \times n$ matrix whose rows form an orthonormal basis for the null space of A, and let $g = A^T(AA^T)^{-1}b$, where $\mathcal{C}(d) < +\infty$ implies that $\operatorname{rank}(A) = m$ and so F and g are well defined. Then the following problems are equivalent under the invertible linear transformations $s = Fx, x = F^T s + g$ between $\{x \in \Re^n \mid Ax = b\}$ and $\Re^{(n-m)}$:

$$CP(d): \quad \begin{array}{ll} \text{maximize} & c^T x \\ \text{s.t.} & Ax = b, \\ & x \in C_X, \end{array} \qquad Q: \quad \begin{array}{ll} \text{maximize} & c^T F^T s + c^T g \\ \text{s.t.} & F^T s + g \in C_X. \end{array}$$

Let

$$a_1 = \frac{3}{\tau}, \quad a_2 = \frac{4}{\tau}, \quad \text{and} \quad a_3 = \frac{3}{\tau}\ . \tag{37}$$

Then, from (13), we know that there exists $\hat{x}$ and r for which $A\hat{x} = b$ and $B(\hat{x}, r) \subset C_X$, and that satisfies $\frac{1}{r} \le a_1\mathcal{C}(d)$, $\|\hat{x}\| \le a_2\mathcal{C}(d)$, and $\frac{\|\hat{x}\|}{r} \le a_3\mathcal{C}(d)$. If we let $\hat{s} := F\hat{x}$, then it is straightforward to show that $B(\hat{s}, r)$ is contained in the feasible region of Q and that $\|\hat{s}\| \le \|\hat{x}\| \le a_2\mathcal{C}(d)$, and $\frac{\|\hat{s}\|}{r} \le a_3\mathcal{C}(d)$, where $\|s\| = \|s\|_2$ for $s \in \Re^{n-m}$ and $B(s, r)$ is the Euclidean ball centered at $s \in \Re^{n-m}$ with radius r. Let z_Q denote the optimal objective function value of Q, and let x^* denote an optimal solution of $CP(d)$. Then one can also easily show that $z_Q = z^*(d)$, and so $|z_Q| = |z^*(d)| \le \|c\|_*\mathcal{C}(d)$ from (10). Furthermore, let $s^* := Fx^*$. Then it is easy to show that s^* is an optimal solution of Q and $\|s^*\| \le \|x^*\| \le \mathcal{C}(d)^2$ from (11). Then, from Remark 5.1, the iteration bound of (35) is valid for the program Q with values of a_1, a_2, and a_3 from (37) and with the dimension n replaced by $n - m$. Substituting (37) into (35) yields the following iteration bound:

$$\begin{aligned} &\left\lceil 2(n-m+2)\left\{(n-m+1)\ln\left(2 + \frac{78\|c\|_*}{\epsilon\tau^2}\mathcal{C}(d)^4\right) + \ln\left(\frac{4}{\tau}\mathcal{C}(d)^2\right)\right\}\right\rceil \\ &\qquad \le \left\lceil 2(n-m+2)^2 \ln\left(\frac{80\|c\|_*}{\epsilon}\left(\frac{\mathcal{C}(d)}{\tau}\right)^4\right)\right\rceil \\ &\qquad \le \left\lceil 8(n-m+2)^2 \ln\left(\frac{3\mathcal{C}(d)}{\tau}\frac{\|c\|_*}{\epsilon}\right)\right\rceil. \qquad \square \end{aligned}$$

6. Further issues: Applications to other volume-reducing cutting-plane algorithms; testing for ϵ-optimality; testing for infeasibility; skewness of the ellipsoids.

Applications to other volume-reducing cutting-plane algorithms. The ellipsoid algorithm belongs to a larger class of efficient volume-reducing cutting-plane algorithms that includes the method of centers of gravity [11], the method of inscribed ellipsoids [10], and the method of volumetric centers [22], among others. Here we discuss how our analysis of the ellipsoid algorithm can be easily extended to these other methods. To keep the discussion simple, we focus on the class of instances of $CP(d)$, where C_X and C_Y are both regular cones.

Consider the strategy of applying either the method of centers of gravity or the method of inscribed ellipsoids to solve $CP(d)$ by solving $HP(d)$, starting with the unit ball B^{n+1} in $\Re^{n+1}$ (centered at the origin) and with the goal of computing an ϵ-optimal solution to $HP(d)$ and hence to $CP(d)$ as well. Because both of these methods achieve an (absolute) constant reduction in volume at each iteration, the iteration complexity of each of these methods will be $O(\ln(\frac{\mathrm{vol}\left(B^{n+1}\right)}{\mathrm{vol}\left(B^{n+1}\cap H_d^\epsilon\right)}))$ in order to find an ϵ-optimal solution of $CP(d)$. Now notice that a slight rearrangement of the proof of Theorem 5.1 yields the following inequality:

$$\ln\left(\frac{\mathrm{vol}\left(B^{n+1}\right)}{\mathrm{vol}\left(B^{n+1}\cap H_d^\epsilon\right)}\right) \le 4(n+2)\ln\left(\frac{4\mathcal{C}(d)}{\min\{\tau,\bar{\tau}\}}\frac{\|c\|_*}{\epsilon}\right). \tag{38}$$

Therefore, the iteration complexity of these two methods is $O(n\ln(\frac{\mathcal{C}(d)}{\min\{\tau,\bar{\tau}\}}\frac{\|c\|_*}{\epsilon}))$.

The analysis of the method of volumetric centers is roughly the same as above; this method also achieves a constant reduction in volume at each iteration. However, the volumetric centers method must be initiated with a polytope as opposed to a Euclidean ball. Suppose we endow $X = \Re^n$ with the L_∞ norm rather than the Euclidean norm and that we apply the method of volumetric centers to solve $HP(d)$ initiated at the unit cube C^{n+1} in $\Re^{n+1}$. Then an identical version of (38) can be proved with B^{n+1} replaced by C^{n+1}, and so one can prove that the method of volumetric centers also has iteration complexity $O(n\ln(\frac{\mathcal{C}(d)}{\min\{\tau,\bar{\tau}\}}\frac{\|c\|_*}{\epsilon}))$. We also point out that the method of volumetric centers requires fewer total arithmetic operations than the ellipsoid algorithm.

Similar results can be derived for the two other classes of instances of $CP(d)$. For a more thorough discussion of the complexity of volume-reducing cutting-plane methods, see [12].

Testing for ϵ-optimality by solving the dual problem. One uncomfortable fact about Theorems 5.1, 5.2, and 5.3 is that while the ellipsoid algorithm is guaranteed to find an ϵ-approximate solution of $CP(d)$ in the stated complexity bounds of these theorems, the quantities in the bounds may be unknown (one may know the relevant widths of the cones, but in all likelihood the condition number $\mathcal{C}(d)$ is unknown), and so one does not know when an ϵ-approximate solution of $CP(d)$ has been found. An obvious strategy for overcoming this difficulty is to solve the primal and the dual problems in parallel, and then test at each iteration (of each algorithm) if the best primal and dual solutions obtained so far satisfy a duality gap of at most ϵ. Because of the natural symmetry in format of the dual pair of problems $CP(d)$ and $CD(d)$, one can obtain complexity results for solving the dual problem $CD(d)$ that exactly parallel those of Theorems 5.1, 5.2, and 5.3, where the quantities $\|c\|_*, n, \tau$, and $\bar{\tau}$ are replaced by $\|b\|, m, \bar{\tau}^*$, and τ^*, respectively, and where the cones C_X and C_Y are replaced by C_Y^* and C_X^* in the statements of the complexity results. One also must assume that $Y^* = \Re^m$ and that the norm $\|y\|_*$ on $\Re^m$ is the Euclidean norm.

Testing for infeasibility. If one is not sure whether $CP(d)$ has a feasible solution, the ellipsoid algorithm can be run to test for infeasibility of the primal problem (in parallel with attempting to solve $CP(d)$). This can be accomplished as follows. First, assume that the dual space $Y^* = \Re^m$ is endowed with the Euclidean norm $\|y\|_2$. Second, note that $CP(d)$ has no feasible solution if the "alternative" system,

$$AP(d): \quad \begin{aligned} A^T y &\in C_X^*, \\ y &\in C_Y^*, \\ y^T b &< 0, \end{aligned}$$

has a solution. Define the following "alternative" set:

$$Y_d = \{y \in Y^* \mid A^T y \in C_X^*, y \in C_Y^*, y^T b \le 0\}. \tag{39}$$

Suppose $CP(d)$ has no feasible solution. Then, as special cases of Theorems 5.2, 5.4, and 5.6 of [7], Y_d must contain an inscribed Euclidean ball $B_2(\hat{y}, r)$ (or a disk in the vector subspace $\{y \in \Re^m \mid A^T y = 0\}$ if $C_X = X$) such that $\|\hat{y}\|_2 + r \leq 1$ (and so $B_2(\hat{y}, r)$ is contained in the unit Euclidean ball) and such that

$$\text{(i)} \quad r \geq \frac{\min\{\tau^*, \bar{\tau}^*\}}{4\mathcal{C}_P(d)} \qquad \text{when } C_X \text{ and } C_Y \text{ are both regular,}$$

$$\text{(ii)} \quad r \geq \frac{\tau^*}{2\mathcal{C}_P(d)} \qquad \text{when } C_X \text{ is regular and } C_Y = \{0\},$$

$$\text{(iii)} \quad r \geq \frac{\min\{\bar{\tau}^*, \bar{\tau}\}}{4\mathcal{C}_P(d)} \qquad \text{when } C_X = X \text{ and } C_Y \text{ is regular.}$$

These results can then be used to demonstrate that an upper bound on the number of iterations needed to find a solution of $AP(d)$ using the ellipsoid algorithm starting with the Euclidean unit ball in $\Re^m$ (or the unit disk in the vector subspace $\{y \in \Re^m \mid A^T y = 0\}$ if $C_X = X$) is

$$\text{(i):} \quad O\left(m^2 \ln\left(\frac{\mathcal{C}_P(d)}{\min\{\tau^*, \bar{\tau}^*\}}\right)\right) \qquad \text{when } C_X \text{ and } C_Y \text{ are both regular,}$$

$$\text{(ii):} \quad O\left(m^2 \ln\left(\frac{\mathcal{C}_P(d)}{\tau^*}\right)\right) \qquad \text{when } C_X \text{ is regular and } C_Y = \{0\},$$

$$\text{(iii):} \quad O\left((m-n)^2 \ln\left(\frac{\mathcal{C}_P(d)}{\min\{\bar{\tau}^*, \bar{\tau}\}}\right)\right) \qquad \text{when } C_X = X \text{ and } C_Y \text{ is regular.}$$

Bounding the skewness of the ellipsoids in the ellipsoid algorithm. Let $E_{\bar{x},Q} = \{x \in X \mid (x - \bar{x})^T Q^{-1} (x - \bar{x}) \leq 1\}$ be an ellipsoid centered at the point $\bar{x}$, where Q is a positive definite matrix. The skewness of $E_{\bar{x},Q}$ is defined to be the ratio of the largest to the smallest eigenvalue of the matrix Q defining $E_{\bar{x},Q}$, and so the skewness also corresponds to the traditional condition number of the matrix Q. The skewness of the ellipsoids generated in an application of the ellipsoid algorithm determines the numerical stability of the ellipsoid algorithm, since each iteration of the ellipsoid algorithm uses the current value of Q^{-1} to update the center $\bar{x}$ of the ellipsoid and to perform a rank-one update of Q^{-1}; see [3], for example. Furthermore, one can show that the logarithm of the skewness of the ellipsoid computed at a given iteration is sufficient to specify the numerical precision requirements of the ellipsoid algorithm at that iteration. Herein, we provide an upper bound on the skewness of all of the ellipsoids computed in the ellipsoid algorithm as a function of the condition number $\mathcal{C}(d)$ of $CP(d)$.

The skewness of the unit ball (which is used to initiate the ellipsoid algorithm herein) is 1. From the formula for updating the ellipsoids encountered in the ellipsoid algorithm at each iteration, the skewness increases by at most $(1 + \frac{2}{k-1})$ at each iteration, where k is the dimension of the space in which the ellipsoid algorithm is implemented. Therefore, the skewness of the ellipsoid at iteration j is bounded above by $(1 + \frac{2}{k-1})^j$. Let us consider the class of instances defined for Theorem 5.1, for example, and let J be the (unrounded) iteration bound for the ellipsoid algorithm from Theorem 5.1, namely,

$$J = 8(n+2)^2 \ln\left(\frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \frac{\|c\|_*}{\epsilon}\right), \tag{40}$$

and assume for simplicity of exposition that J is an integer. Let $(\text{Skew})_j$ denote the skewness of the ellipsoid computed in the ellipsoid algorithm at iteration j. Then, for this class of instances, we have $k = n + 1$, whereby

$$(41) \qquad (\text{Skew})_J \le \left(1 + \frac{2}{n}\right)^J = \left(e^{\left(\ln\left(1+\frac{2}{n}\right)\right)}\right)^J = e^{J\left(\ln\left(1+\frac{2}{n}\right)\right)} = \left(e^J\right)^{\left(\ln\left(1+\frac{2}{n}\right)\right)}.$$

Substituting for (40) in (41), we obtain

$$(\text{Skew})_J \le \left(\frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \frac{\|c\|_*}{\epsilon}\right)^{8(n+2)^2 \ln\left(1+\frac{2}{n}\right)}.$$

However, the exponent in the above expression is bounded above by $45n$ for $n \ge 2$ (actually, it is bounded above by $17n$ for large $n \ge 49$), and we have

$$(\text{Skew})_J \le \left(\frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \frac{\|c\|_*}{\epsilon}\right)^{45n}.$$

Taking logarithms, we can rewrite this bound as

$$(42) \qquad \ln (\text{Skew})_J \le 45n \ln \left(\frac{4\mathcal{C}(d)}{\min\{\tau, \bar{\tau}\}} \frac{\|c\|_*}{\epsilon}\right).$$

Therefore, the logarithm of the skewness of the ellipsoids encountered in the ellipsoid algorithm grows at most linearly in the logarithm of the condition number $\mathcal{C}(d)$. Also, the bound in (42) specifies the sufficient numerical precision requirements for the ellipsoid algorithm (in terms of $\ln(\mathcal{C}(d))$ and other quantities) because the logarithm of the skewness is sufficient to specify such requirements. This is similar to the results on numerical precision presented in [25] for an interior-point method for linear programming.

Finally, the above reasoning can be used to obtain similar bounds on the skewness for the other two classes of instances of $CP(d)$.

REFERENCES

[1] F. Alizadeh, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.

[2] M. Bazaraa, H. Sherali, and C. M. Shetty, *Nonlinear Programming, Theory and Algorithms*, 2nd ed., John Wiley, New York, 1993.

[3] D. Bertsimas and J. Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific, Belmont, MA, 1997.

[4] R. Bland, D. Goldfarb, and M. Todd, *The ellipsoid method: A survey*, Oper. Res., 29 (1981), pp. 1039–1091.

[5] A. Charnes and W. Cooper, *Programming with linear fractionals*, Naval Res. Quarterly, 9 (1962), pp. 181–186.

[6] B. Craven and B. Mond, *The dual of a fractional linear program*, J. Math. Anal. Appl., 42 (1973), pp. 507–512.

[7] R. Freund and J. Vera, *Some Characterizations and Properties of the "Distance to Ill-Posedness" and the Condition Number of a Conic Linear System*, Technical Report W.P. #3862-95-MSA, MIT Sloan School of Management, 1995; Math. Programming, to appear.

[8] M. Grötschel, L. Lovasz, and A. Schrijver, *The ellipsoid method and its consequences in combinatorial optimization*, Combinatorica, 1 (1981), pp. 169–197.

[9] L. Khachiyan, *A polynomial algorithm in linear programming*, Soviet Math. Dokl., 20 (1979), pp. 191–194.

[10] L. Khachiyan, S. Tarasov, and I. Erlikh, *The method of inscribed ellipsoids*, Soviet Math. Dokl., 37 (1988), pp. 226–230.

[11] A. Levin, *On an algorithm for the minimization of convex functions*, Soviet Math. Dokl., 6 (1965), pp. 286–290.

[12] T. Magnanti and G. Perakis, *A unifying geometric solution framework and complexity analysis for variational inequalities*, Math. Programming, 71 (1995), pp. 327–351.

[13] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[14] Y. Nesterov and M. Todd, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

[15] Y. Nesterov and M. Todd, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.

[16] Y. Nesterov, M. Todd, and Y. Ye, *Primal-Dual Methods and Infeasibility Detectors for Nonlinear Programming Problems*, Technical Report TR1156, Dept. of IE&OR, Cornell University, Ithaca, NY, 1996.

[17] J. Renegar, *Some perturbation theory for linear programming*, Math. Programming, 65 (1994), pp. 73–91.

[18] J. Renegar, *Incorporating condition measures into the complexity theory of linear programming*, SIAM J. Optim., 5 (1995), pp. 506–524.

[19] J. Renegar, *Linear programming, complexity theory, and elementary functional analysis*, Math. Programming, 70 (1995), pp. 279–351.

[20] J. Renegar, *Condition numbers, the barrier method, and the conjugate gradient method*, SIAM J. Optim., 6 (1996), pp. 879–912.

[21] N. Shor, *Cut off methods with space extension in convex programming problems*, Cybernetics, 13 (1977), pp. 94–96.

[22] P. Vaidya, *A new algorithm for minimizing convex functions over convex sets*, in Proceedings of the 30th IEEE Symposium on Foundations of Computer Science, IEEE Computer Soc. Press, Los Alamitos, CA, 1989, pp. 338–343.

[23] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Review, 38 (1996), pp. 49–95.

[24] J. Vera, *Ill-posedness and the complexity of deciding existence of solutions to linear programs*, SIAM J. Optim., 6 (1996), pp. 549–569.

[25] J. Vera, *On the complexity of linear programming under finite precision arithmetic*, Math. Programming, 80 (1998), pp. 91–123.

[26] D. Yudin and A. Nemirovskii, *Informational complexity and efficient methods for solving complex extremal problems*, Ekonom. i Matem. Metody, 12 (1976), pp. 357–369.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 177–182

A NONLINEAR CONJUGATE GRADIENT METHOD WITH A STRONG GLOBAL CONVERGENCE PROPERTY*

Y. H. DAI† AND Y. YUAN†

Abstract. Conjugate gradient methods are widely used for unconstrained optimization, especially large scale problems. The strong Wolfe conditions are usually used in the analyses and implementations of conjugate gradient methods. This paper presents a new version of the conjugate gradient method, which converges globally, provided the line search satisfies the standard Wolfe conditions. The conditions on the objective function are also weak, being similar to those required by the Zoutendijk condition.

Key words. unconstrained optimization, new conjugate gradient method, Wolfe conditions, global convergence

AMS subject classifications. 65K, 90C

PII. S1052623497318992

1. Introduction.

Our problem is to minimize a function of n variables

$$f(x), \tag{1.1}$$

where f is smooth and its gradient $g(x)$ is available. Conjugate gradient methods for solving (1.1) are iterative methods of the form

$$x_{k+1} = x_k + \alpha_k d_k, \tag{1.2}$$

where $\alpha_k > 0$ is a steplength and d_k is a search direction. Normally the search direction at the first iteration is the steepest descent direction, namely, $d_1 = -g_1$. The other search directions can be defined recursively:

$$d_{k+1} = -g_{k+1} + \beta_k d_k. \tag{1.3}$$

$\beta_k \in \Re$ is so chosen that (1.2)–(1.3) reduces to the linear conjugate gradient method if $f(x)$ is a strictly convex quadratic function and if α_k is the exact one-dimensional minimizer. Well-known formulas for β_k are the Fletcher–Reeves (FR), Polak–Ribière–Polyak (PRP), and Hestenes–Stiefel (HS) formulas (see [6]; [10], [11]; and [7], respectively) and are given by

$$\beta_k^{FR} = \|g_{k+1}\|^2/\|g_k\|^2, \tag{1.4}$$

$$\beta_k^{PRP} = g_{k+1}^T y_k/\|g_k\|^2, \tag{1.5}$$

$$\beta_k^{HS} = g_{k+1}^T y_k/d_k^T y_k, \tag{1.6}$$

where $y_k = g_{k+1} - g_k$ and $\|\cdot\|$ denotes the Euclidean norm.

*Received by the editors March 26, 1997; accepted for publication (in revised form) March 26, 1999; published electronically November 29, 1999. This work was supported by Chinese National Natural Science Foundation grants 19525101 and 19731010.

http://www.siam.org/journals/siopt/10-1/31899.html

†State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, P.O. Box 2719, Beijing 100080, China (dyh@lsec.cc.ac.cn, yyx@lsec.cc.ac.cn).

The global convergence properties of the FR, PRP, and HS methods without regular restarts have been studied by many authors, including Zoutendijk [15], Al-Baali [1], Liu, Han, and Yin [9], Dai and Yuan [2], Powell [12], Gilbert and Nocedal [8], and Dai and Yuan [4]. To establish the convergence results of these methods, it is normally required that the steplength α_k satisfy the following strong Wolfe conditions:

$$f(x_k) - f(x_k + \alpha_k d_k) \geq -\delta \alpha_k g_k^T d_k, \tag{1.7}$$

$$|g(x_k + \alpha_k d_k)^T d_k| \leq -\sigma g_k^T d_k, \tag{1.8}$$

where $0 < \delta < \sigma < 1$. Some convergence analyses even require the α_k be computed by the exact line search, namely,

$$f(x_k + \alpha_k d_k) = \min_{\alpha > 0} f(x_k + \alpha d_k). \tag{1.9}$$

On the other hand, many other numerical methods for unconstrained optimization are proved to be convergent under the standard Wolfe conditions (1.7) and

$$g(x_k + \alpha_k d_k)^T d_k > \sigma g_k^T d_k. \tag{1.10}$$

For example, see Fletcher [5]. Hence it is interesting to investigate whether there exists a conjugate gradient method that converges under the standard Wolfe conditions.

In this paper, we give a new formula for β_k. It is shown that this new conjugate gradient method is globally convergent as long as the standard Wolfe conditions (1.7) and (1.10) are satisfied. Moreover, the conditions on the objective function are also weaker than the usual ones.

2. New formula for $\boldsymbol{\beta_k}$.

One motivation for our new formula for β_k is the descent property of the conjugate descent method (see Fletcher [5]), which uses

$$\beta_k^{CD} = \|g_{k+1}\|^2 / (-d_k^T g_k). \tag{2.1}$$

It can be shown that the conjugate descent method always produces a descent direction if the strong Wolfe conditions are satisfied. We try to find a conjugate gradient method which generates descent directions provided the standard Wolfe conditions are satisfied. Suppose the current search direction d_k is a descent direction, namely, $d_k^T g_k < 0$. Now we need to find a β_k that defines a descent direction d_{k+1}. This requires that

$$-\|g_{k+1}\|^2 + \beta_k g_{k+1}^T d_k < 0. \tag{2.2}$$

We assume that $\beta_k > 0$. Denote $\tau_k = \|g_{k+1}\|^2 / \beta_k$. The above inequality is equivalent to

$$\tau_k > g_{k+1}^T d_k. \tag{2.3}$$

Therefore, we can let $\tau_k = d_k^T y_k$, giving our new formula

$$\beta_k = \|g_{k+1}\|^2 / d_k^T y_k. \tag{2.4}$$

This formula is well defined because line search condition (1.10) implies $d_k^T y_k > 0$. If line searches are exact, the above formula is the same as the FR formula (1.4). Therefore we see that (2.4) corresponds to a nonlinear conjugate gradient method. It is interesting to note that (2.4) has the same numerator as the FR formula (1.4) and has the same denominator as the HS formula (1.6). Now we can define the new method, as follows.

Algorithm 2.1 (A new CG method).

```
Step 1. Given $x_1 \in \Re^n$, $d_1 = -g_1$, $k := 1$, if $g_1 = 0$, then stop.
Step 2. Compute an $\alpha_k > 0$ satisfying (1.7) and (1.10).
Step 3. Let $x_{k+1} = x_k + \alpha_k d_k$. If $g_{k+1} = 0$, then stop.
Step 4. Compute $\beta_k$ by (2.4) and generate $d_{k+1}$ by (1.3),
        $k := k + 1$, go to Step 2.
```

It follows from (1.3) and (2.4) that

$$g_{k+1}^T d_{k+1} = \frac{\|g_{k+1}\|^2}{d_k^T y_k} g_k^T d_k = \beta_k g_k^T d_k. \tag{2.5}$$

The above relation can be rewritten as

$$\beta_k = \frac{g_{k+1}^T d_{k+1}}{g_k^T d_k}. \tag{2.6}$$

This formula is very important in our convergence analysis.

3. Convergence of the new method. In this section, we establish a convergence theorem for Algorithm 2.1. We assume that the objective function satisfies the following conditions.

Assumption 3.1. (1) f is bounded below on $\Re^n$ and is continuously differentiable in a neighborhood $\mathcal{N}$ of the level set $\mathcal{L} = \{x \in \Re^n : f(x) \le f(x_1)\}$; (2) the gradient $\nabla f(x)$ is Lipschitz continuous in $\mathcal{N}$, i.e., there exists a constant $L > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x - y\| \text{ for any } x,\ y \in \mathcal{N}. \tag{3.1}$$

Under Assumption 3.1, we give a useful lemma which was essentially proved by Zoutendijk [15] and Wolfe [13, 14].

Lemma 3.2. *Suppose that x_1 is a starting point for which Assumption* 3.1 *is satisfied. Consider any method of the form* (1.2), *where d_k is a descent direction and α_k satisfies the standard Wolfe conditions* (1.7) *and* (1.10). *Then we have that*

$$\sum_{k \ge 1} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty. \tag{3.2}$$

Proof. It follows from (1.10) that

$$d_k^T y_k = d_k^T (g_{k+1} - g_k) \ge (\sigma - 1) g_k^T d_k. \tag{3.3}$$

On the other hand, the Lipschitz condition (3.1) implies

$$(g_{k+1} - g_k)^T d_k \le \alpha_k L \|d_k\|^2. \tag{3.4}$$

The above two inequalities give

$$\alpha_k \ge \frac{\sigma - 1}{L} \cdot \frac{g_k^T d_k}{\|d_k\|^2}, \tag{3.5}$$

which with (1.7) implies that

$$f_k - f_{k+1} \ge c \frac{(g_k^T d_k)^2}{\|d_k\|^2}, \tag{3.6}$$

where $c = \delta(1-\sigma)/L$. Summing (3.6) and noting that f is bounded below, we see that (3.2) holds, which concludes the proof. □

THEOREM 3.3. *Suppose that x_1 is a starting point for which Assumption* 3.1 *holds. Let $\{x_k,\, k = 1, 2 \dots\}$ be generated by Algorithm* 2.1. *Then the algorithm either terminates at a stationary point or converges in the sense that*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{3.7}$$

Proof. If the algorithm does not terminate after finite many iterations, we have that

$$\|g_k\| > 0 \quad \text{for all } k. \tag{3.8}$$

First we show all search directions are descent, namely,

$$g_k^T d_k < 0 \tag{3.9}$$

for all k. The above inequality is obvious for $k = 1$. Now we prove it for all $k \geq 1$ by induction. Assume (3.9) holds for k. It follows from the line search conditions that

$$d_k^T y_k \geq (\sigma - 1) d_k^T g_k > 0. \tag{3.10}$$

The above inequality and (2.5) imply that (3.9) holds for $k+1$. This shows that (3.9) is true for all $k \geq 1$.

We now rewrite (1.3) as

$$d_{k+1} + g_{k+1} = \beta_k d_k. \tag{3.11}$$

Squaring both sides of the above equation, we get

$$\|d_{k+1}\|^2 = \beta_k^2 \|d_k\|^2 - 2 g_{k+1}^T d_{k+1} - \|g_{k+1}\|^2. \tag{3.12}$$

Dividing both sides by $(g_{k+1}^T d_{k+1})^2$ and applying (2.6), we obtain that

$$\begin{aligned} \frac{\|d_{k+1}\|^2}{(g_{k+1}^T d_{k+1})^2} &= \frac{\|d_k\|^2}{(g_k^T d_k)^2} - \frac{2}{g_{k+1}^T d_{k+1}} - \frac{\|g_{k+1}\|^2}{(g_{k+1}^T d_{k+1})^2} \\ &= \frac{\|d_k\|^2}{(g_k^T d_k)^2} - \left(\frac{1}{\|g_{k+1}\|} + \frac{\|g_{k+1}\|}{g_{k+1}^T d_{k+1}} \right)^2 + \frac{1}{\|g_{k+1}\|^2} \\ &\leq \frac{\|d_k\|^2}{(g_k^T d_k)^2} + \frac{1}{\|g_{k+1}\|^2}. \end{aligned} \tag{3.13}$$

Because $\|d_1\|^2/(g_1^T d_1)^2 = 1/\|g_1\|^2$, (3.13) shows that

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \leq \sum_{i=1}^{k} \frac{1}{\|g_i\|^2} \tag{3.14}$$

for all k. If the theorem is not true, there exists a constant $c > 0$ such that

$$\|g_k\| \geq c \quad \text{for all } k. \tag{3.15}$$

Therefore it follows from (3.14) and (3.15) that

$$\frac{\|d_k\|^2}{(g_k^T d_k)^2} \le \frac{k}{c^2}, \tag{3.16}$$

which implies that

$$\sum_{k\ge 1} \frac{(g_k^T d_k)^2}{\|d_k\|^2} = \infty. \tag{3.17}$$

Relation (3.17) contradicts the Zoutendijk condition (3.2). This contradiction shows that the theorem is true. □

4. Discussion. It is shown in the previous section that the new conjugate gradient method converges under the standard Wolfe line search conditions. It should be noted that our assumption that the objective function is bounded below is weaker than the usual assumption that the level set

$$\{x \in \Re^n : f(x) \le f(x_1)\} \tag{4.1}$$

is bounded.

From the proof of Theorem 3.3, we can see that the equivalent form (2.6) of the formula (2.4) plays an important role in the convergence analysis. Relation (2.6) enables us to establish the recurrence relation (3.13), which is about the sequence of the reciprocal $\{(g_k^T d_k)^2/\|d_k\|^2\}$. The term $(g_k^T d_k)^2/\|d_k\|^2$ is exactly the one that appears in the Zoutendijk condition (3.2). This makes our convergence analysis very simple. It is known that to obtain the convergence of the FR, PRP, and HS methods, one normally has to consider two sequences. For example, Al-Baali [1] considered the sequences $\{\|d_k\|^2\}$ and $\{g_k^T d_k/\|g_k\|^2\}$ for the FR method, and Gilbert and Nocedal [8] considered $\{\|d_k\|^2\}$ and $\{g_k^T d_k\}$ for the PRP and HS methods.

It is also worth noting that Al-Baali [1] and Gilbert and Nocedal [8] proved or required the sufficient descent condition, namely,

$$g_k^T d_k \le -c\|g_k\|^2 \quad \text{for some } c > 0 \text{ and for all } k \ge 1. \tag{4.2}$$

However, our method does not guarantee this inequality. But if the strong Wolfe line search conditions are satisfied at every iteration, we have that

$$l_k = \frac{g_{k+1}^T d_k}{g_k^T d_k} \in [-\sigma, \sigma]. \tag{4.3}$$

Formula (2.5) can be rewritten as

$$g_{k+1}^T d_{k+1} = \frac{1}{l_k - 1}\|g_{k+1}\|^2. \tag{4.4}$$

The above two relations show that (4.2) holds with $c = 1/(1+\sigma)$. This indicates that our method also has the sufficient descent property (4.2) if the strong Wolfe line search conditions are used.

Dai and Yuan [3] considered a class of methods that use

$$\beta_k \in [(\sigma - 1)/(1+\sigma), 1]\bar{\beta}_k, \tag{4.5}$$

where $\bar{\beta}_k$ is given by (2.4). It is shown in [3] that Algorithm 2.1 is still convergent if, in Step 4, β_k computed by (2.4) is replaced by any β_k, satisfying (4.5).

REFERENCES

[1] M. Al-Baali, *Descent property and global convergence of the Fletcher-Reeves method with inexact line search*, IMA J. Numer. Anal., 5 (1985), pp. 121–124.

[2] Y. H. Dai and Y. Yuan, *Convergence properties of the Fletcher-Reeves method*, IMA J. Numer. Anal., 16 (1996), pp. 155–164.

[3] Y. H. Dai and Y. Yuan, *A nonlinear conjugate gradient method with a nice global convergence property*, Research report ICM-95-038, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing, 1995.

[4] Y. H. Dai and Y. Yuan, *Further studies on the Polak-Ribière-Polyak method*, Research report ICM-95-040, Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, Beijing, 1995.

[5] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley, New York, 1989.

[6] R. Fletcher and C. Reeves, *Function minimization by conjugate gradients*, Comput. J., 7 (1964), pp. 149–154.

[7] M. R. Hestenes and E. L. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, 49 (1952), pp. 409–436.

[8] J. C. Gilbert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM. J. Optim., 2 (1992), pp. 21–42.

[9] G. H. Liu, J. Y. Han, and H. X. Yin, *Global convergence of the Fletcher-Reeves algorithm with an inexact line search*, Report, Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, 1993.

[10] E. Polak and G. Ribière, *Note sur la convergence de directions conjugées*, Rev. Francaise Informat Recherche Opertionelle, 3e année, 16 (1969), pp. 35–43.

[11] B. T. Polyak, *The conjugate gradient method in extremem problems*, USSR Comp. Math. Math. Phys., 9 (1969), pp. 94–112.

[12] M. J. D. Powell, *Nonconvex Minimization Calculations and the Conjugate Gradient Method*, Lecture Notes in Math. 1066, Springer-Verlag, Berlin, 1984, pp. 122–141.

[13] P. Wolfe, *Convergence conditions for ascent methods*, SIAM Rev., 11 (1969), pp. 226–235.

[14] P. Wolfe, *Convergence conditions for ascent methods.* II: *Some corrections*, SIAM Rev., 13 (1969), pp. 185–188.

[15] G. Zoutendijk, *Nonlinear programming, computational methods*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, 1970, pp. 37–86.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 183–194

COMPLEXITY OF PREDICTOR-CORRECTOR ALGORITHMS FOR LCP BASED ON A LARGE NEIGHBORHOOD OF THE CENTRAL PATH[*]

CLOVIS C. GONZAGA[†]

Abstract. The predictor-corrector approach for following the central path of monotone linear complementarity and linear programming problems is simple, elegant, and efficient. Although it has excellent theoretical properties when working in narrow neighborhoods of the central path, its proved complexity assumes a frustratingly high value of $O(n^{1.5}L)$ iterations when based on an l_∞ neighborhood and several Newton corrector steps per iteration. This paper shows that by carefully specifying the line searches in each step, the complexity assumes the value $O(nL)$, as should be expected for a method based on this neighborhood.

Key words. linear complementarity problem, primal-dual interior point algorithm, predictor-corrector algorithms

AMS subject classifications. 49M15, 65K05, 90C33

PII. S1052623496304141

1. Introduction.

Predictor-corrector schemes are very common in homotopy methods. Soon after the publication of Karmarkar's algorithm [7], Barnes proposed for the first time an algorithm which alternated cost reduction steps (primal affine scaling steps) and centering steps. But the study of such methods in a primal setting was not easy, because there was no simple way to choose a length for the predictor step. Barnes's result was published long after the first description of the idea, in a joint paper with Chopra and Jensen [2].

This difficulty was eliminated by the development of primal-dual algorithms, which started with Megiddo's description of the central path [11] and the path following algorithms by Kojima, Mizuno, and Yoshise [10, 9] and Monteiro and Adler [15, 16]. The primal-dual predictor-corrector method was first described by Mizuno, Todd, and Ye [14] for linear programming, using what we shall call a "small neighborhood" of the central path. Their algorithm alternates tangent and centering steps in such a way that all points belong to a 2-norm neighborhood of the central path. The algorithm has a complexity of $O(\sqrt{n}L)$ Newton iterations, the duality gap converges to zero Q-quadratically in the number of predictor-corrector steps [18], and the iterates converge to the analytic center of the optimal face [3].

Primal-dual algorithms originally designed for linear programs were extended to monotone linear complementarity problems keeping the same complexity and asymptotic convergence properties (under a strict complementarity hypothesis). See, for instance, Ye and Anstreicher [17]. A detailed treatment of the resulting algorithms is found in Kojima, Megiddo, Noma, and Yoshise [8]. This reference describes several different neighborhoods of the central path and shows that the complexity of most algorithms becomes $O(nL)$ when large neighborhoods are used.

The predictor-corrector method based on l_∞-norm proximity measures[1] can sim-

[*] Received by the editors May 23, 1996; accepted for publication (in revised form) February 16, 1999; published electronically November 29, 1999.
http://www.siam.org/journals/siopt/10-1/30414.html

[†] Department of Mathematics, Federal University of Santa Catarina, Cx. Postal 5210, 88040-970 Florianópolis, SC, Brazil (clovis@mtm.ufsc.br).

[1] The concepts discussed here will be formally defined later in the paper.

ply take a predictor step, which generates a point on the boundary of the neighborhood, and then take one Newton centering step with a line search to reduce the proximity measure. The centering iteration may be very poor, and the steps may be very short, at least theoretically. This method can be modified to use a first-order correction, based on the same Hessian matrix as the predictor step: this modification with some clever heuristics added was proposed by Mehrotra [12], with excellent practical results. Mehrotra's algorithm was enhanced by Zhang and Zhang [19], who proved a complexity of $O(n^{3/4}L)$ iterations. A further increase in the order of the corrections can lower the complexity to a value arbitrarily near $O(\sqrt{n}L)$. See [5].

Although these methods based on a single correction are very efficient, they diverge very much from the original idea of a predictor-corrector method, since they do not really approach the central path. A practical study by Gondzio [4] uses multiple centering (simplified) Newton iterations, with promising results.

The motivation for this paper is mostly theoretical. The predictor-corrector algorithm based on a sequence of tangent steps followed by the application of Newton's method for centering is very simple and elegant. But the existing results are frustrating: in the worst case, each predictor step reduces the duality gap by a factor of $1-\nu/\sqrt{n}$, where $\nu \in (0,1)$. The reduction of the l_∞ proximity measure to, say, $\frac{1}{2}$ needs $O(n)$ centering steps. The resulting complexity is then $O(n^{1.5}L)$ iterations. These facts are discussed by Anstreicher and Bosch [1]. This reference tries to relieve this bad behavior by taking large steps without a tangent predictor. See also Jansen, Roos, Terlaky, and Vial [6].

In this paper we take the following approach: the centering iterations work always in the l_∞-norm neighborhood of the central path. Each centering iteration computes a Newton direction and does a line search along it. The essential feature is that the merit function for the search is the Euclidean norm proximity measure, and not the l_∞ proximity.

The properties of predictor and corrector when seen isolated are the same as we discussed above. Seen together, the following is true: the increase in Euclidean proximity at the predictor step depends on the gap reduction; the work of the corrector depends on the Euclidean proximity: short or large predictor steps require, respectively, a small or large number of centering steps. This results in the desired overall bound of $O(nL)$ Newton steps for the predictor-corrector algorithm.

Conventions. Given a vector d, the corresponding upper case symbol denotes as usual the diagonal matrix D defined by the vector. The symbol e will represent the vector of all ones, with dimension given by the context.

We shall denote componentwise operations on vectors by the usual notations for real numbers. Thus, given two vectors u, v of the same dimension, uv, u/v, etc., will denote the vectors with components u_iv_i, u_i/v_i, etc. This notation is consistent as long as componentwise operations always have precedence in relation to matrix operations. Note that $uv \equiv Uv$, and if A is a matrix, then $Auv \equiv AUv$, but in general $Auv \neq (Au)v$.

2. The problem.

The monotone linear complementarity problem will be stated in the following format: Find $(x,s) \in \mathbb{R}^{2n}$ such that

$$\text{(P)} \qquad \begin{aligned} xs &= 0, \\ Qx + Rs &= b, \\ x, s &\geq 0, \end{aligned}$$

where $b \in \mathbb{R}^n$, and $Q, R \in \mathbb{R}^{n\times n}$ are such that for any $u, v \in \mathbb{R}^n$,

$$\text{if } Qu + Rv = 0, \qquad \text{then } u^T v \geq 0.$$

The feasible set for (P) and the set of interior solutions are, respectively,

$$\begin{aligned} F &:= \{(x,s) \in \mathbb{R}^{2n} \mid Qx + Rs = b \ , \ x, s \geq 0\}, \\ F^0 &:= \{(x,s) \in F \mid x > 0, s > 0\}. \end{aligned}$$

We assume that $F^0 \neq \emptyset$. This ensures that the optimal set is nonempty and bounded.

Central points. A feasible solution (x, s) is the central point associated with the parameter $\mu > 0$ if and only if $xs = \mu e$.

It is well known (see, for instance, Kojima et al. [8]) that the set of central points defines a differentiable curve $\mu > 0 \mapsto (x(\mu), s(\mu))$, which ends at the analytic center of the optimal face.

Given $(x, s) \in F$ and $\mu > 0$, the proximity of (x, s) to $(x(\mu), s(\mu))$ is estimated by the following measures:

$$\begin{aligned} \delta(x,s,\mu) &:= \left\| \frac{xs}{\mu} - e \right\|, \\ \delta_\infty(x,s,\mu) &:= \left\| \frac{xs}{\mu} - e_\infty \right\|. \end{aligned}$$

These proximity measures define neighborhoods of the central path. Given $\alpha \in (0, 1)$, the *small neighborhood* is defined as

$$\mathcal{N} = \{(x,s,\mu) \mid \mu > 0, (x,s) \in F, \delta(x,s,\mu) \leq \alpha\}.$$

The other proximity measure defines similarly the *large neighborhood* $\mathcal{N}_\infty$.

We shall treat feasible interior point algorithms based on the large neighborhood. We assume that an initial interior point (x^0, s^0) is given, as well as a parameter value $\mu^0 > 0$, such that (x^0, s^0, μ^0) lies in the neighborhood.

The Newton step. Let (x, s) be a given interior primal-dual pair. Given $\theta > 0$, the Newton step for solving the equation $x^* s^* = \theta e$ from (x, s) is given by the unique solution of

$$\begin{aligned} su + xv &= -xs + \theta e, \\ Qu + Rv &= 0. \end{aligned} \tag{2.1}$$

Scaling. In the analysis of a Newton step it is common practice to scale the equations above by the change of variables

$$\bar{x} = d^{-1}x, \quad \bar{u} = d^{-1}u, \quad \bar{s} = ds, \quad \bar{v} = dv,$$

where $d = \sqrt{xs^{-1}}$. A nice feature of this scaling is that

$$\bar{x} \ = \ \bar{s} \ = \ \sqrt{xs} \ > \ 0.$$

The Newton equations after scaling and dividing by $\bar{x}$ become

$$\begin{aligned} \bar{u} + \bar{v} &= -\bar{x} + \theta \bar{x}^{-1}, \\ QD\bar{u} + RD^{-1}\bar{v} &= 0. \end{aligned}$$

This simplifies the analysis very much. To keep the notation simple, we shall assume in our proofs that $x = s$, without loss of generality: this situation can always be reached by the scaling above.

3. The algorithm. The algorithm will work in the large neighborhood $\mathcal{N}_\infty$, defined with a fixed radius $\alpha \in (0, 1)$. Each iteration is composed of a predictor step and a corrector algorithm, which we now describe separately. The input and output data for our procedures will be triplets (x, s, μ), where (x, s) is a feasible primal-dual pair and $\mu > 0$.

The predictor step. The predictor step starts from a given $(x_-, s_-, \mu_-) \in \mathcal{N}_\infty$ and takes a step along the Newton direction (2.1) associated with $\theta = 0$. This is known as the affine-scaling or tangent direction. The steplength is such that the result is on the boundary of $\mathcal{N}_\infty$. For completeness, we shall allow "null steps" and comment on them after presenting the complete algorithm.

ALGORITHM 3.1. *Data: (x_-, s_-, μ_-) such that $\delta_\infty(x_-, s_-, \mu_-) \le \alpha$.*

```
If $\delta_\infty(x_-, s_-, \mu_-) = \alpha$ then
    Null step: Set $(x, s, \mu) = (x_-, s_-, \mu_-)$ and exit.
Compute the affine-scaling direction $(u_-, v_-)$ by solving the Newton equations
```

$$\begin{array}{rcl} s_-u_- + x_-v_- & = & -x_-s_- \\ Qu_- + Rv_- & = & 0. \end{array} \tag{3.1}$$

```
Steplength: Compute
```

$$\lambda = \min\{\theta \in (0, 1] \mid \delta_\infty(x_- + \theta u_-, s_- + \theta v_-, (1-\theta)\mu_-) \ge \alpha\}.$$

```
Result: $x = x_- + \lambda u_-$ , $s = s_- + \lambda v_-$ , $\mu = (1-\lambda)\mu_-$.
```

The corrector algorithm. The corrector starts from a point on the boundary of $\mathcal{N}_\infty$ and takes a sequence of Newton centering iterations, i.e., a sequence of Newton steps that each consists of the solution of (2.1) with a fixed value $\theta = \mu$, followed by a line search.

Two aspects of this algorithm must be discussed: the line search and the stopping rule.

There are several possibilities for the stopping rule: one can stop whenever $\delta_\infty(x, s, \mu) \le \beta < \alpha$, when $\delta(x, s, \mu) \le \beta < \alpha$, or simply after a fixed number of centering steps. Any of these possibilities is allowed in our treatment (provided that one does not impose an exceedingly large fixed number of iterations). We shall state the algorithm with the following very general rule: given a fixed integer J and a fixed real $\beta < \alpha$, stop at (x, s, μ) whenever the iteration count equals J or $\delta_\infty(x, s, \mu) \le \beta$.

The line search is an essential feature of this paper. The l_∞ proximity measure does not provide a good merit function for a search procedure because of its severe nonsmoothness. We shall use as merit function the Euclidean proximity measure. The large neighborhood can then be interpreted as a trust region for the Newton steps.

ALGORITHM 3.2. *Data: (x, s, μ) such that $\delta_\infty(x, s, \mu) = \alpha$.*

```
Choose an integer $J \ge 1$ ($J = +\infty$ is fine.)
$j := 0$
REPEAT
    $j := j + 1$.
    Newton: Compute the centering direction $(u, v)$ by solving
```

$$\begin{array}{rcl} su + xv & = & -xs + \mu e, \\ Qu + Rv & = & 0. \end{array}$$

```
        Maximum steplength: Compute
            $\lambda_{\max} = \max\{\theta \in [0,1] \mid \delta_\infty(x + \bar{\theta}u, s + \bar{\theta}v, \mu) \leq \alpha,\ 0 \leq \bar{\theta} \leq \theta\}.$
        Line search: Compute
            $\lambda = \operatorname{argmin}\{\delta(x + \theta u, s + \theta v, \mu) \mid \theta \in [0, \lambda_{\max}]\}.$
        $(x, s) = (x + \lambda u, s + \lambda v).$
    UNTIL $j = J$ or $\delta_\infty(x, s, \mu) \leq \beta$.
    Result: $(x_+, s_+, \mu_+) = (x, s, \mu).$
```

The complete predictor-corrector algorithm is obtained by alternating these two procedures as follows.

ALGORITHM 3.3. *Data:* (x^0, s^0, μ^0) *such that* $\delta_\infty(x^0, s^0, \mu^0) < \alpha$, $\epsilon > 0$.

```
    $k := 0.$
    REPEAT
        $(x_-, s_-, \mu_-) := (x^k, s^k, \mu^k).$
        Predictor: Use Algorithm 3.1 to compute $(x, s, \mu)$ such that
        $\delta_\infty(x, s, \mu) = \alpha.$
        Corrector: Use Algorithm 3.2 to compute $(x_+, s_+, \mu_+)$.
        $(x^{k+1}, s^{k+1}, \mu^{k+1}) := (x_+, s_+, \mu_+).$
        $k := k + 1.$
    UNTIL $\mu^k \leq \epsilon$.
```

We are assuming that an initial nearly central solution is available, and the stopping rule is based on the value of the parameter μ. These initialization and termination steps are usual in interior point methods and extensively discussed, for instance, in [8].

In the rest of this paper we show the following fact: for any choice of the stopping rule in the corrector, Algorithm 3.3 stops after computing no more than $O(n \log(\mu^0/\epsilon))$ Newton steps.

Remark. When a bound $J < +\infty$ is used, the corrector algorithm may end with a point on the boundary of $\mathcal{N}_\infty$. Then the next predictor step will be null, and the corrector algorithm will continue centering. The presence of null steps introduces no extra computations and can be totally avoided with a slightly different stopping rule in the corrector algorithm. The possibility of having a fixed number of centering steps per iteration is interesting: this is done by Gondzio [4], who uses simplified centering steps and chooses J based on an estimate of the time consumed by each step. This reference also proves that one centering step per iteration is enough to ensure a complexity of $O(nL)$ Newton steps.

4. The predictor step.

The predictor step, Algorithm 3.1, starts from (x_-, s_-, μ_-) such that $\delta_\infty(x_-, s_-, \mu_-) \leq \alpha$ and finds a point (x, s) and $\mu = (1-\lambda)\mu_-$, $\lambda \in [0, 1)$, such that $\delta_\infty(x, s, \mu) = \alpha$.

Note that $\log \mu = \log \mu_- + \log(1 - \lambda)$, and since $\log(1 - \lambda) \leq -\lambda$,

$$\log \mu \leq \log \mu_- - \lambda.$$

This entitles us to call λ the improvement of μ in the iteration. In this section we show that the variation in the Euclidean proximity due to the predictor step is closely related to the improvement of μ.

The next section will study the corrector step and show that the number of Newton centering steps will be related to the variation of Euclidean proximity in the corrector step. Summing up these two facts, we have roughly the following: if the improvement in the predictor step is large, then a large number of steps will be needed to recenter; if the predictor step results in a small improvement of the parameter μ, then the corrector steps will recenter quickly. The complexity result will be obtained in the last section by summing the series of Euclidean proximity variations during the complete algorithm.

LEMMA 4.1. *Consider an application of Algorithm* 3.1 *from* (x_-, s_-, μ_-) *such that* $\delta_\infty(x_-, s_-, \mu_-) = \delta_\infty^- \leq \alpha$. *Then*

$$\delta(x, s, \mu) \leq \delta(x_-, s_-, \mu_-) + O(\sqrt{n}\lambda). \tag{4.1}$$

Proof. If $\lambda = 0$ (null step), then the result is trivial. Assume that $\lambda > 0$.

Assume without loss of generality that $x_- = s_- = \phi > 0$ (see the end of section 3). Then the Newton equations can be written as

$$\begin{aligned} u + v &= -\phi, \\ Qu + Rv &= 0. \end{aligned} \tag{4.2}$$

The proximity after the predictor step is computed as follows, using (3.1):

$$\begin{aligned} xs = (\phi + \lambda u)(\phi + \lambda v) &= \phi^2 + \lambda\phi(u + v) + \lambda^2 uv \\ &= (1 - \lambda)\phi^2 + \lambda^2 uv, \end{aligned}$$

so that with $\mu = (1 - \lambda)\mu_-$,

$$\frac{xs}{\mu} - e = \frac{x_- s_-}{\mu_-} - e + \frac{\lambda^2}{1 - \lambda}\frac{uv}{\mu_-}. \tag{4.3}$$

Taking Euclidean norms,

$$\delta(x, s, \mu) \leq \delta(x_-, s_-, \mu_-) + \frac{\lambda^2}{1 - \lambda}\left\|\frac{uv}{\mu_-}\right\|.$$

Our task is proving that

$$\frac{\lambda^2}{1 - \lambda}\left\|\frac{uv}{\mu_-}\right\| = O(\sqrt{n}\lambda). \tag{4.4}$$

We shall study the vector

$$w = \frac{uv}{\mu_-}.$$

Let us first dismiss two simple situations:

—If λ is large, say, $\lambda > 0.1$, then (4.1) is trivially true because for any (x, s, μ) such that $\delta_\infty(x, s, \mu) < \alpha$ we have $\delta(x, s, \mu) < \alpha\sqrt{n}$. Assume then that $\lambda \leq 0.1$.

—If w is small, say, $\|w\|_\infty \leq 1$, then (4.4) is true because for $\lambda \leq 0.1$,

$$\frac{\lambda^2}{1 - \lambda}\|w\| \leq \frac{0.1}{0.9}\lambda\sqrt{n}.$$

Assume then that $\|w\|_\infty > 1$.

Let us analyze the vector w. Multiplying (4.2) by u, for $i = 1, \ldots, n$,

$$u_i v_i = -u_i^2 - u_i \phi_i.$$

The right-hand side of this expression is a concave function of u_i with a maximum at $u_i = -\phi_i/2$ with value $\phi_i^2/4$. Hence

$$u_i v_i \le \phi_i^2/4. \tag{4.5}$$

Since $\delta_\infty(\phi, \phi, \mu_-) < \alpha$, we have for $i = 1, \ldots, n$ that $|\phi_i^2/\mu_- \, - 1| < \alpha$, or

$$\phi_i^2 < (1 + \alpha)\mu_-. \tag{4.6}$$

Combining (4.5) and (4.6), we conclude that

$$u_i v_i < \frac{\mu_-}{2}, \quad w_i < \frac{1}{2}. \tag{4.7}$$

From (4.7) and our assumptions we conclude that w has large negative components.

Remark. This is the key point in the proof: since $\sum w_i \ge 0$ by monotonicity, only a few components of w can have large negative values; i.e., for each $w_i << 0$, a large number of components $w_i \in [0, 1]$ will be needed. This restrains the relation between $\|w\|$ and $\|w\|_\infty$, and consequently between the variations of δ and δ_∞.

We have $\|w\|^2 = w^T w \le \|w\|_\infty \|w\|_1$, and

$$\|w\|_1 = \sum_{i|w_i \ge 0} w_i - \sum_{i|w_i < 0} w_i \le 2 \sum_{i|w_i \ge 0} w_i \le n,$$

where the first inequality uses monotonicity and the second uses (4.7). It follows that

$$\|w\|^2 \le n \, \|w\|_\infty \, . \tag{4.8}$$

Taking norms in (4.3),

$$\frac{\lambda^2}{1-\lambda} \, \|w\|_\infty \; \le \; \delta_\infty(x, s, \mu) + \delta_\infty(x_-, s_-, \mu_-) \; \le \; 2\alpha,$$

because both points are in the large neighborhood. We can finally prove (4.4) using the two last expressions. From (4.8),

$$\begin{aligned} \frac{\lambda^4}{(1-\lambda)^2} \|w\|^2 &\le n \frac{\lambda^2}{1-\lambda} \frac{\lambda^2}{1-\lambda} \, \|w\|_\infty \\ &\le 2n\alpha \frac{\lambda^2}{1-\lambda} \\ &\le 3n\alpha\lambda^2 \end{aligned}$$

for $\lambda \le 0.1$. It follows that

$$\frac{\lambda^2}{1-\lambda} \|w\| \le \sqrt{3\alpha} \, \lambda \sqrt{n},$$

completing the proof. $\square$

5. The corrector steps. The corrector algorithm is composed of j centering steps with a fixed value of μ. We shall study the effect of each centering step on the Euclidean proximity. A centering step starts from (x, s, μ) such that $0 < \beta < \delta_\infty(x, s, \mu) \leq \alpha$ and finds a point (x_+, s_+) such that $\delta_\infty(x_+, s_+, \mu) \leq \alpha$ and $\delta(x_+, s_+, \mu) < \delta(x, s, \mu)$.

LEMMA 5.1. *Consider an iteration of Algorithm* 3.2 *(a centering step). Then*

$$\delta(x + \lambda u, s + \lambda v, \mu) \leq \delta(x, s, \mu) - \frac{K_2}{\sqrt{n}}, \tag{5.1}$$

where

$$K_2 = \min\left\{\frac{1-\alpha}{2}, \frac{\sqrt{n}\,\beta}{2}\right\}.$$

Remark. Usually $\sqrt{n}\,\beta >> 1 - \alpha$, and this could be used as a hypothesis. But it is interesting to notice that the complexity does not change if the centralization algorithm is made very precise, with $\beta = \alpha/\sqrt{n}$, for example, or if the stopping rule uses the 2-norm proximity.

Proof. Let (x, s, μ) be given with $\delta_\infty(x, s, \mu) = \delta_\infty \in [\beta, \alpha]$, and consider an iteration of Algorithm 3.2. Without loss of generality, assume that $x = s = \phi > 0$ (see the end of section 3). The Newton equations become

$$u + v = \frac{1}{\phi}(-\phi^2 + \mu e), \quad Qu + Rv = 0. \tag{5.2}$$

Since $\left\|\phi^2/\mu \; - e\right\|_\infty = \delta_\infty \in [\beta, \alpha]$, we have for $i = 1, \ldots, n$,

$$1 - \delta_\infty \;\leq\; \frac{\phi_i^2}{\mu} \;\leq\; 1 + \delta_\infty,$$

and hence

$$\frac{1}{1-\delta_\infty} \;\geq\; \frac{\mu}{\phi_i^2} \;\geq\; \frac{1}{1+\delta_\infty}. \tag{5.3}$$

From the monotonicity hypothesis, $u^T v \geq 0$. From a well-known lemma by Mizuno [13], $\|uv\| \leq \|u+v\|^2/\sqrt{8}$. Using (5.2), we obtain

$$\|uv\| \leq \frac{1}{\sqrt{8}} \left\|\frac{1}{\phi^2}\right\|_\infty \|\phi^2 - \mu e\|^2,$$

and using (5.3),

$$\left\|\frac{uv}{\mu}\right\| \leq \frac{1}{\sqrt{8}} \frac{1}{1-\delta_\infty} \delta^2(x, s, \mu). \tag{5.4}$$

Let us study the variation of the Euclidean proximity along the direction (u, v), using the Newton equations. Denote $\delta = \delta(x, s, \mu)$, and for $\theta \geq 0$, $\delta(\theta) = \delta(x + \theta u, s + \theta v, \mu)$, $\delta_\infty(\theta) = \delta_\infty(x + \theta u, s + \theta v, \mu)$. We have

$$\begin{aligned}(x + \theta u)(s + \theta v) &= xs + \theta(xv + su) + \theta^2\, uv \\ &= xs + \theta(-xs + \mu e) + \theta^2\, uv,\end{aligned}$$

and it follows that

$$\frac{(x+\theta u)(s+\theta v)}{\mu} - e = (1-\theta)\left(\frac{xs}{\mu} - e\right) + \theta^2 \frac{uv}{\mu}.$$

Taking norms, for $\theta \in [0,1]$,

$$\delta(\theta) \le (1-\theta)\delta + \theta^2 \left\| \frac{uv}{\mu} \right\|. \tag{5.5}$$

Using (5.4),

$$\delta(\theta) \le (1-\theta)\delta + \theta^2 \frac{\delta^2}{\sqrt{8}(1-\delta_\infty)}.$$

Denote the right-hand side of this expression by $g(\theta)$. Then $g(\cdot)$ is a convex quadratic function with derivative

$$g'(\theta) = \frac{2\delta^2}{\sqrt{8}(1-\delta_\infty)}\theta - \delta.$$

The following facts are easily seen: $g(\cdot)$ assumes a minimum at $\theta^* = \sqrt{2}(1-\delta_\infty)/\delta$ with value

$$g(\theta^*) = \delta - (1-\delta_\infty)/\sqrt{2}, \tag{5.6}$$

and for $\theta \in [0, \theta^*]$,

$$g(\theta) \ \le \ g(0) + \frac{g'(0)}{2}\theta \ = \ \delta - \frac{\delta}{2}\theta. \tag{5.7}$$

Now there are three possibilities concerning θ^* and $\lambda_{\max}$: $\lambda_{\max} \ge \theta^*$, $\lambda_{\max} < \theta^*$, and $\lambda_{\max} = 1$, or $\lambda_{\max} < \theta^*$ and $\delta_\infty(\lambda_{\max}) = \alpha$. Let us develop each of them:

(i) If $\lambda_{\max} \ge \theta^*$, then $\delta(\lambda) \le g(\theta^*)$, and from (5.6), $\delta(\lambda) \le \delta - (1-\alpha)/2 \le \delta - K_2 \le \delta - K_2/\sqrt{n}$.

(ii) If $1 = \lambda_{\max} < \theta^*$, then from (5.7), $g(1) \le \delta - \delta/2$. By construction, $\delta \ge \delta_\infty \ge \beta$ and hence

$$\frac{\delta}{2} \ge \frac{\sqrt{n}\,\beta}{2\sqrt{n}} = \frac{K_2}{\sqrt{n}}.$$

It follows that $\delta(\lambda) \le g(1) \le \delta - K_2/\sqrt{n}$.

(iii) Assume that $\lambda_{\max} < \theta^*$ and $\delta_\infty(\lambda_{\max}) = \alpha$. From (5.5) with $\theta = \lambda_{\max} \in [0,1]$ and using the l_∞-norm,

$$\delta_\infty \ \le \ \alpha \ = \ \delta_\infty(\lambda_{\max}) \ \le \ (1-\lambda_{\max})\delta_\infty + \lambda_{\max}^2 \left\| \frac{uv}{\mu} \right\|_\infty.$$

Simplifying this expression, we obtain

$$\lambda_{\max} \ge \delta_\infty \left\| \frac{uv}{\mu} \right\|_\infty^{-1}.$$

Combining this with (5.4), we obtain

$$\lambda_{\max} \geq \frac{\sqrt{8}\,\delta_\infty(1-\delta_\infty)}{\delta^2}.$$

Using (5.7), we see that

$$g(\lambda_{\max}) \leq \delta - (\sqrt{2}\,(1-\delta_\infty))\frac{\delta_\infty}{\delta}.$$

But $\delta \leq \sqrt{n}\delta_\infty$ by the relationship between Euclidean and l_∞-norms. Using this and the fact that $\delta_\infty \leq \alpha$, we conclude that

$$\delta(\lambda) \;\leq\; g(\lambda_{\max}) \;\leq\; \delta - \frac{\sqrt{2}\,(1-\alpha)}{\sqrt{n}} \;\leq\; \delta - \frac{1-\alpha}{\sqrt{2n}} \leq \delta - \frac{K_2}{\sqrt{n}},$$

completing the proof. □

6. The predictor-corrector algorithm. Now we study the predictor-corrector Algorithm 3.3.

THEOREM 6.1. *Algorithm* 3.3 *stops after computing no more than* $O(n\log(\mu^0/\epsilon))$ *Newton steps.*

Proof. Consider an application of the algorithm. Each iteration starts with data (x^k, s^k, μ^k) with $\delta(x^k, s^k, \mu^k) \leq \alpha$. The predictor step computes a steplength λ^k (with $\lambda^k = 0$ for a null step) and constructs (x, s, μ^{k+1}) with $\mu^{k+1} = (1-\lambda^k)\mu^k$. This implies that

$$\log\mu^{k+1} \leq \log\mu^k - \lambda^k. \tag{6.1}$$

Using Lemma 4.1,

$$\delta(x, s, \mu^{k+1}) \leq \delta(x^k, s^k, \mu^k) + K_1\sqrt{n}\lambda^k,$$

where K_1 is a constant dependent on α. The corrector starts from (x, s, μ^{k+1}) and performs j^k centering steps, obtaining $(x^{k+1}, s^{k+1}, \mu^{k+1})$. By Lemma 5.1,

$$\delta(x^{k+1}, s^{k+1}, \mu^{k+1}) \leq \delta(x, s, \mu^{k+1}) - j^k K_2\frac{1}{\sqrt{n}},$$

where $K_2 = (1-\alpha)/\sqrt{2}$.

Combining the last two inequalities,

$$\delta(x^{k+1}, s^{k+1}, \mu^{k+1}) \leq \delta(x^k, s^k, \mu^k) + K_1\sqrt{n}\lambda^k - j^k K_2\frac{1}{\sqrt{n}}.$$

It follows that

$$\delta(x^k, s^k, \mu^k) \leq \delta(x^0, s^0, \mu^0) + K_1\sqrt{n}\sum_{i=0}^{k-1}\lambda^i - \frac{K_2}{\sqrt{n}}\sum_{i=0}^{k-1} j^i.$$

Hence

$$\sum_{i=0}^{k-1}\lambda^i \geq \frac{1}{K_1\sqrt{n}}\left(\frac{K_2}{\sqrt{n}}\sum_{i=0}^{k-1} j^i + \delta(x^k, s^k, \mu^k) - \delta(x^0, s^0, \mu^0)\right).$$

But $\delta(x^i, s^i, \mu^i) \le \alpha\sqrt{n}$ for all $i = 0, 1, \ldots, k$ and hence

$$\sum_{i=0}^{k-1} \lambda^i \ge \frac{K_2}{K_1\, n} \sum_{i=0}^{k-1} j^i - \frac{\alpha}{K_1}.$$

From (6.1),

$$\log \frac{\mu^k}{\mu^0} \le -\sum_{i=0}^{k-1} \lambda^i \le -\frac{K_2}{K_1\, n} \sum_{i=0}^{k-1} j^i + \frac{\alpha}{K_1}.$$

The algorithm stops when $\mu^k \le \epsilon$. Hence at all iterations, $\log(\mu^k/\mu^0) \ge \log(\epsilon/\mu^0)$, and consequently

$$-\frac{K_2}{K_1\, n} \sum_{i=0}^{k-1} j^i + \frac{\alpha}{K_1} \ge \log \frac{\epsilon}{\mu^0}.$$

It follows that at all iterations

$$\sum_{i=0}^{k-1} j^i \le \frac{K_1}{K_2} n \left(\log \frac{\mu^0}{\epsilon}\right) + \frac{\alpha}{K_1} \ = \ O\left(n \log \frac{\mu^0}{\epsilon}\right).$$

Thus the total number of centering steps is bounded by $O(n \log \frac{\mu^0}{\epsilon})$. The number of predictor steps has the same bound, since each predictor step is followed by at least one corrector step, and this completes the proof. □

REFERENCES

[1] K. M. Anstreicher and R. A. Bosch, *A new infinity-norm path following algorithm for linear programming*, SIAM J. Optim., 5 (1995), pp. 236–246.

[2] E. R. Barnes, S. Chopra, and D. J. Jensen, *The Affine Scaling Method with Centering*, Technical Report, Department of Mathematical Sciences, IBM T. J. Watson Research Center, Yorktown Heights, NY, 1988.

[3] C. C. Gonzaga and R. A. Tapia, *On the convergence of the Mizuno–Todd–Ye algorithm to the analytic center of the solution set*, SIAM J. Optim., 7 (1997), pp. 47–65.

[4] J. Gondzio, *Multiple centrality corrections in a primal-dual method for linear programming*, Comput. Optim. Appl., 6 (1996), pp. 137–156.

[5] P. F. Hung and Y. Ye, *An asymptotical $O(\sqrt{n}L)$-iteration path-following linear programming algorithm that uses wide neighborhoods*, SIAM J. Optim., 6 (1996), pp. 570–586.

[6] B. Jansen, C. Roos, T. Terlaky, and J.-Ph. Vial, *Primal-dual algorithms for linear programming based on the logarithmic barrier method*, J. Optim. Theory Appl., 83 (1994), pp. 1–26.

[7] N. Karmarkar, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.

[8] M. Kojima, N. Megiddo, T. Noma, and A. Yoshise, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Lecture Notes in Comput. Sci. 538, Springer-Verlag, Berlin, 1991.

[9] M. Kojima, S. Mizuno, and A. Yoshise, *A polynomial-time algorithm for a class of linear complementarity problems*, Math. Programming, 44 (1989), pp. 1–26.

[10] M. Kojima, S. Mizuno, and A. Yoshise, *A primal-dual interior point algorithm for linear programming*, in Progress in Mathematical Programming: Interior Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 29–47.

[11] N. Megiddo, *Pathways to the optimal set in linear programming*, in Progress in Mathematical Programming: Interior Point and Related Methods, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.

[12] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

[13] S. Mizuno, *A new polynomial time method for a linear complementarity problem*, Math. Programming, 56 (1992), pp. 31–43.

[14] S. Mizuno, M. J. Todd, and Y. Ye, *On adaptive step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.

[15] R. D. C. Monteiro and I. Adler, *Interior path following primal-dual algorithms Part* I*: Linear programming*, Math. Programming, 44 (1989), pp. 27–41.

[16] R. D. C. Monteiro and I. Adler, *Interior path following primal-dual algorithms: Part* II*: Convex quadratic programming*, Math. Programming, 44 (1989), pp. 43–66.

[17] Y. Ye and K. M. Anstreicher, *On quadratic and $O(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for the linear complementary problem*, Math. Programming, 62 (1993), pp. 537–551.

[18] Y. Ye, O. Güler, R. A. Tapia, and Y. Zhang, *A quadratically convergent $O(\sqrt{n}L)$-iteration algorithm for linear programming*, Math. Programming, 59 (1993), pp. 151–162.

[19] Y. Zhang and D. Zhang, *On polynomiality of the Mehrotra-type predictor-corrector interior-point algorithms*, Math. Programming, 68 (1995), pp. 303–317.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 195–210

ON THE LOCAL CONVERGENCE OF A PREDICTOR-CORRECTOR METHOD FOR SEMIDEFINITE PROGRAMMING[*]

JUN JI[†], FLORIAN A. POTRA[‡], AND RONGQIN SHENG[§]

Abstract. We study the local convergence of a predictor-corrector algorithm for semidefinite programming problems based on the Monteiro–Zhang unified direction whose polynomial convergence was recently established by Monteiro. Under strict complementarity and nondegeneracy assumptions superlinear convergence with Q-order 1.5 is proved if the scaling matrices in the corrector step have bounded condition number. A version of the predictor-corrector algorithm enjoys quadratic convergence if the scaling matrices in both predictor and corrector steps have bounded condition numbers. The latter results apply in particular to algorithms using the Alizadeh–Haeberly–Overton (AHO) direction since there the scaling matrix is the identity matrix.

Key words. semidefinite programming, interior point method, superlinear convergence

AMS subject classifications. 65K05, 90C25

PII. S1052623497316828

1. Introduction.

The study of superlinear convergence of interior point methods for linear programming (LP) was initiated in the early 1990s in an effort to explain the fact that interior point methods tend to perform significantly better in practice than indicated by the polynomial complexity bounds. This discrepancy is due to the limitation of the worst-case analysis used in deriving polynomial complexity bounds and reflects the inherent conflict between the requirements of global convergence and fast local convergence. Superlinear convergence is especially important for semidefinite programming (SDP) since no finite termination schemes exist for such problems. As predicted by theory and confirmed by numerical experiments the condition number of the linear systems defining the search directions increases as $1/\mu$, where μ is the normalized duality gap, so that the respective systems become very ill conditioned as we approach the solution. Therefore, an interior point method that is not superlinearly convergent is unlikely to obtain high accuracy in practice despite its theoretical "polynomial complexity." On the other hand a superlinearly convergent interior point method will achieve good accuracy (e.g., 10^{-10} or better) in substantially fewer iterations than indicated by its worse-case global linear convergence rate that is related to polynomial complexity.

The local convergence analysis for interior point algorithms for SDP is much more challenging than those for LP. The first two papers investigating superlinear convergence of interior point algorithms were written independently by Kojima, Shida, and Shindoh [4] and by Potra and Sheng [13]. The algorithm investigated in these papers

[*] Received by the editors February 25, 1997; accepted for publication (in revised form) January 22, 1999; published electronically November 29, 1999.

http://www.siam.org/journals/siopt/10-1/31682.html

[†] Department of Mathematics and Computer Science, Valdosta State University, Valdosta, GA 31698 (junji@goldendog.valdosta.peachnet.edu). The work of this author was supported in part by a grant from the Center for Faculty Development and Research at Valdosta State University.

[‡] Department of Mathematics and Statistics, University of Maryland Baltimore County, Baltimore, MD 21250 (potra@pc1.math.umbc.edu). The work of this author was supported in part by NSF grants DMS-9305760 and DMS-9706894.

[§] Lucent Technologies, Inc., 2443 Warrenville Road, Lisle, IL 60532 (rsheng@mail.dls.net). The work of this author was performed at the University of Iowa and was supported in part by NSF grant DMS-9305760.

is an extension of the Mizuno–Todd–Ye predictor-corrector algorithm for LP and uses the Kojima–Shindoh–Hara/Helmberg–Rendl–Vanderbei–Wolkowicz/Monteiro (KSH /HRVW/M) search direction. (See the next section for a definition of this search direction.) Kojima, Shida, and Shindoh [4] established the superlinear convergence under the following three assumptions:

(A) SDP has a strictly complementary solution.

(B) SDP is nondegenerate in the sense that the Jacobian matrix of its KKT system is nonsingular.

(C) The iterates converge tangentially to the central path in the sense that the size of the neighborhood containing the iterates must approach zero, namely,

$$\lim_{k\to\infty} \|(X^k)^{1/2}S^k(X^k)^{1/2} - (X^k \bullet S^k/n)I\|_F/(X^k \bullet S^k/n) = 0.$$

Here $\|.\|_F$ denotes the Frobenius norm of a matrix and $\bullet$ denotes the corresponding scalar product. (See the end of this section for precise definitions.) In [13] assumptions (B) and (C) were not used. Instead a sufficient condition for superlinear convergence, which is implied by the above assumptions, was proposed. In [14] Potra and Sheng improved this result and obtained superlinear convergence under assumption (A) and the condition

$$\text{(D)} \qquad \lim_{k\to\infty} X^k S^k/\sqrt{X^k \bullet S^k} = 0,$$

which is clearly weaker than (C). Of course both (C) and (D) can be enforced by the algorithm, but the practical efficiency of such an approach is questionable. However, from a theoretical point of view it is proved in [14] that the modified algorithm in [4] that uses several corrector steps in order to enforce (C) has polynomial complexity and is superlinearly convergent under assumption (A) only. It is well known that assumption (A) is necessary for superlinear convergence of standard interior point methods even in the QP case (see [10]).

Kojima, Shida, and Shindoh [4] also gave an example suggesting that interior point algorithms for SDP based on the KSH/HRVW/M search direction are unlikely to be superlinearly convergent without imposing a condition like (C). In [5] the same authors showed that a predictor-corrector algorithm using the AHO direction is quadratically convergent under assumptions (A) and (B). (See the next section for a definition of the AHO search direction.) They also proved that the algorithm is globally convergent, but no polynomial complexity bounds have been found for this algorithm. It is shown that condition (C) is automatically satisfied by the iteration sequence generated by the algorithm. It appears that the use of the AHO direction in the corrector step has a strong effect on centering. Potra and Sheng exploited this property in [15], showing that a direct extension of the Mizuno–Todd–Ye algorithm, based on the KSH/HRVW/M direction in the predictor step and the AHO direction in the corrector step, has polynomial complexity and is superlinearly convergent with Q-order 1.5 under assumptions (A) and (B).

An interesting superlinearly convergent predictor-corrector algorithm based on the Nesterov–Todd (NT) search direction was proposed by Luo, Sturm, and Zhang [7]. The algorithm depends on a parameter $\epsilon > 0$. It produces points $(X^k, y^k, S^k) \in \mathcal{N}_F(\gamma_k)$, where the neighborhood $\mathcal{N}_F(\gamma)$ is defined in (2.7), $\gamma_k = 1/4$ if $\mu_k := X^k \bullet S^k/n \geq \epsilon/4$, and $\gamma_k = \mu_k/\epsilon$ if $\mu_k < \epsilon/4$. The algorithm starts from a feasible point $(X^0, y^0, S^0) \in \mathcal{N}_F(1/4)$ and for any given $\tilde{\epsilon} \geq \epsilon/4$ finds a feasible point (X^k, y^k, S^k) with $\mu_k \leq \tilde{\epsilon}$ in at most $O(\sqrt{n}\ln(\mu_0/\tilde{\epsilon}))$ iterations. However, this bound on the number of iterations is not proved to hold for $0 < \tilde{\epsilon} < \epsilon/4$, hence the algorithm is not polynomial in the usual sense. The algorithm is superlinearly convergent under assumption (A). It turns out that (C) is enforced by the algorithm since it is proved in [7] that for sufficiently large k

$$\|(X^k)^{1/2}S^k(X^k)^{1/2} - (X^k \bullet S^k/n)I\|_F/(X^k \bullet S^k/n) \le (X^k \bullet S^k)/(4n).$$

It is also proved that if one uses one predictor and r correctors per iteration, then μ_k converges to zero with Q-order $2/(1+2^{-2r})$.

In this paper we investigate the local behavior of the predictor-corrector algorithm considered by Monteiro [9] for SDP using the Monteiro–Zhang (MZ) family of search directions. We show that under the assumptions (A) and (B), superlinear convergence with Q-order 1.5 is obtained if the scaling matrices in the corrector step have bounded condition number. Finally, we propose a new version of the predictor-corrector algorithm which enjoys quadratic convergence if the scaling matrices in both predictor and corrector steps have bounded condition numbers and (A) and (B) are satisfied.

The following notation and terminology are used throughout the paper:
$\mathbb{R}^p$: the p-dimensional Euclidean space;
$\mathbb{R}^p_+$: the nonnegative orthant of $\mathbb{R}^p$;
$\mathbb{R}^p_{++}$: the positive orthant of $\mathbb{R}^p$;
$\mathbb{R}^{p\times q}$: the set of all $p \times q$ matrices with real entries;
$\mathcal{S}^p$: the set of all $p \times p$ symmetric matrices;
$\mathcal{S}^p_+$: the set of all $p \times p$ symmetric positive semidefinite matrices;
$\mathcal{S}^p_{++}$: the set of all $p \times p$ symmetric positive matrices;
$[M]_{ij}$: the (i,j)th entry of a matrix M;
$\mathrm{Tr}(M)$: the trace of a $p \times p$ matrix, equals $\sum_{i=1}^p [M]_{ii}$;
$M \succeq 0$: M is positive semidefinite;
$M \succ 0$: M is positive definite;
$\lambda_i(M),\ i = 1, \dots, n$: the eigenvalues of $M \in \mathcal{S}^n$;
$\lambda_{\max}(M),\ \lambda_{\min}(M)$: the largest, smallest, eigenvalue of $M \in \mathcal{S}^n$;
$G \bullet H \equiv \mathrm{Tr}(G^T H)$;
$\|\cdot\|$: Euclidean norm of a vector and the corresponding norm of a matrix, i.e.,
$\|y\| \equiv \sqrt{\sum_{i=1}^p y_i^2}, \quad \|M\| \equiv \max\{\|My\| : \|y\| = 1\}$;
$\|M\|_F \equiv \sqrt{M \bullet M},\ M \in \mathbb{R}^{p\times q}$: Frobenius norm of a matrix;
$\|(G,H)\|_F \equiv \sqrt{G \bullet G + H \bullet H},\ G, H \in \mathbb{R}^{p\times q}$;
$M^k = o(1)$: $\|M^k\| \to 0$ as $k \to \infty$;
$M^k = O(1)$: $\|M^k\|$ is bounded;
$M^k = o(\nu_k)$: $M^k/\nu_k = o(1)$;
$M^k = O(\nu_k)$: $M^k/\nu_k = O(1)$.

2. The predictor-corrector algorithm for SDP. We consider the SDP problem,

$$\min\{C \bullet X : A_i \bullet X = b_i,\ i = 1, \dots, m,\ X \succeq 0\}, \tag{2.1}$$

and its associated dual problem,

$$\max\left\{b^T y : \sum_{i=1}^m y_i A_i + S = C,\ S \succeq 0\right\}, \tag{2.2}$$

where $C \in \mathcal{S}^{n\times n}$, $A_i \in \mathcal{S}^{n\times n}$, $i = 1, \dots, m$, $b = (b_1, \dots, b_m)^T \in \mathbb{R}^m$ are given data, and $X \in \mathcal{S}^n_+$, $(y, S) \in \mathbb{R}^m \times \mathcal{S}^n_+$ are the primal and dual variables, respectively. Also, for simplicity we assume that A_i, $i = 1, \dots, m$, are linearly independent.

Throughout this paper we assume that both (2.1) and (2.2) have finite solutions and their optimal values are equal. Under this assumption, X^* and (y^*, S^*) are solutions of (2.1) and (2.2) if and only if they are solutions of the following nonlinear system:

$$A_i \bullet X = b_i,\ i = 1, \ldots, m, \tag{2.3a}$$

$$\sum_{i=1}^{m} y_i A_i + S = C, \tag{2.3b}$$

$$XS = 0,\quad X \succeq 0,\quad S \succeq 0. \tag{2.3c}$$

We denote the feasible set of the problem (2.3) by

$$\mathcal{F} = \{(X, y, S) \in \mathcal{S}^n_+ \times \mathbb{R}^m \times \mathcal{S}^n_+ : (X, y, S) \text{ satisfies (2.3a) and (2.3b)}\}$$

and its solution set by $\mathcal{F}^*$, i.e.,

$$\mathcal{F}^* = \{(X, y, S) \in \mathcal{F} : X \bullet S = 0\}.$$

We consider the symmetrization operator [17]

$$H_P(M) = \frac{1}{2}\left[PMP^{-1} + (PMP^{-1})^T\right] \quad \forall M \in \mathbb{R}^{n\times n}.$$

Since, as observed by Zhang [17],

$$H_P(M) = \tau I \ \text{ iff } \ M = \tau I,$$

for any nonsingular matrix P, any matrix M with real spectrum, and any $\tau \in \mathbb{R}$, it follows that for any given nonsingular matrix P, (2.3) is equivalent to

$$A_i \bullet X = b_i,\ i = 1, \ldots, m, \tag{2.4a}$$

$$\sum_{i=1}^{m} y_i A_i + S = C, \tag{2.4b}$$

$$H_P(XS) = 0,\quad X \succeq 0,\quad S \succeq 0. \tag{2.4c}$$

A perturbed Newton method applied to the system (2.4) leads to the following linear system:

$$H_P(XV + US) = \xi\mu I - H_P(XS), \tag{2.5a}$$

$$A_i \bullet U = 0,\quad i = 1, \ldots, m, \tag{2.5b}$$

$$\sum_{i=1}^{m} w_i A_i + V = 0, \tag{2.5c}$$

where $(U, w, V) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ is the unknown search direction, $\xi \in [0, 1]$ is the centering parameter, and $\mu = (X \bullet S)/n$ is the normalized duality gap corresponding to (X, y, S).

The search direction obtained through (2.5) is called the MZ unified direction [17, 11]. The matrix P used in (2.5) is called the scaling matrix for the search direction. It is well known that taking $P = I$ results in the AHO search direction [1], $P = S^{1/2}$ corresponds to the KSH/HRVW/M search direction [6, 3, 8], and the case of $P^TP =$

$X^{-1/2}[X^{1/2}SX^{1/2}]^{1/2}X^{-1/2}$ coincides with the NT search direction [12]. Monteiro and Zhang [11] established the polynomiality of a long-step path-following method based on search directions defined by scaling matrices belonging to the class

$$\{W^{1/2} \ : \ W \in \mathcal{S}^n_{++} \text{ such that } WXS = SXW\}.$$

Following [11], Sheng, Potra, and Ji [16] proved the polynomiality of a Mizuno–Todd–Ye type predictor-corrector algorithm for SDP by imposing the scaling matrices to be chosen from the class

$$\{P \ : \ P \in \mathbb{R}^{n\times n} \text{ is nonsingular and } \quad PXSP^{-1} \in \mathcal{S}^n\}.$$

Moreover, its superlinear convergence was proved under an additional simple condition. The primal-dual algorithms considered by Monteiro [9] are based on the centrality measure

$$d(X,S) \equiv \|X^{1/2}SX^{1/2} - \mu I\|_F = \left(\sum_{i=1}^n (\lambda_i(XS) - \mu)^2\right)^{1/2}, \tag{2.6}$$

where $(X,S) \in \mathcal{S}^n_+ \times \mathcal{S}^n_+$, $\mu = (X \bullet S)/n = (\sum_{i=1}^n \lambda_i(XS))/n$. Given $\gamma \in (0,1)$, we denote by $\mathcal{N}(\gamma)$ the following neighborhood of the central path:

$$\mathcal{N}(\gamma) = \{(X,y,S) \in \mathcal{F} : d(X,S) \le \gamma\mu, \ X \succ 0, \ S \succ 0, \ \mu = (X \bullet S)/n\}. \tag{2.7}$$

Monteiro's generalized predictor-corrector algorithm for semidefinite programming based on the MZ family of directions consists of a predictor step and a corrector step at each iteration. Starting from a strictly feasible pair (X^0,y^0,S^0) in $\mathcal{N}(\alpha)$, it generates a sequence of iterates $\{(X^k,y^k,S^k)\}$ in $\mathcal{N}(\alpha)$. An iteration of Monteiro's generalized predictor-corrector algorithm can be described as follows.

Predictor-Corrector Algorithm.

Given $(X^k,y^k,S^k) \in \mathcal{N}(\alpha)$, choose nonsingular $n \times n$ matrices P^k and $\overline{P}^k$.

• **Predictor Step.** Solve the system (2.5) with $(X,y,S) = (X^k,y^k,S^k)$, $\xi = 0$, and $P = P^k$. Denote the solution $(U,w,V) \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n$ and set

$$X^k(\theta) = X^k + \theta U, \quad y^k(\theta) = y^k + \theta w, \quad S^k(\theta) = S^k + \theta V. \tag{2.8}$$

Compute the step length

$$\overline{\theta}_k = \max\left\{\tilde{\theta} \in [0,1] : (X^k(\theta), S^k(\theta)) \in \mathcal{N}(\beta) \ \ \forall\, \theta \in [0,\tilde{\theta}]\right\}. \tag{2.9}$$

• **Corrector Step.** Solve the system (2.5) with $(X,y,S) = (X^k(\overline{\theta}_k), y^k(\overline{\theta}_k), S^k(\overline{\theta}_k))$, $\xi = 1$, and $P = \overline{P}^k$. Let $(\overline{U}, \overline{w}, \overline{V})$ be the solution and set

$$X^{k+1} = X^k(\overline{\theta}_k) + \overline{U}, \quad y^{k+1} = y^k(\overline{\theta}_k) + \overline{w}, \quad S^{k+1} = S^k(\overline{\theta}_k) + \overline{V}. \tag{2.10}$$

End of iteration.

Using an elegant analysis, Monteiro [9] proved that the predictor-corrector algorithm defined above with properly chosen parameters α and β $(0 < \alpha < \beta < 1)$ is well defined and that it needs at most $O(\sqrt{n}\ln(\epsilon_0/\epsilon))$ iterations for producing a pair (X^k,y^k,S^k) such that $X^k \bullet S^k \le \epsilon$, where $\epsilon_0 = X^0 \bullet S^0$ is the initial gap. More precisely, Monteiro showed that

$$(X^k,y^k,S^k) \in \mathcal{N}(\alpha) \ \text{ and } \ (X^k(\overline{\theta}_k), y^k(\overline{\theta}_k), S^k(\overline{\theta}_k)) \in \mathcal{N}(\beta), \tag{2.11}$$

$$X^{k+1} \bullet S^{k+1} = (1 - 1/O(\sqrt{n}))X^k \bullet S^k \tag{2.12}$$

$\forall\, k \ge 0$.

3. Technical results. This section states some technical results that play an important role in analyzing the local behavior of the predictor-corrector algorithm of Monteiro presented in the last section.

We begin with some useful facts about matrices. It has been proved by Monteiro ([8, Lemma 3.3] and [9, Lemmas 2.1 and 3.5]) that

$$\lambda_{\min}(E) \geq \lambda_{\min}(H_M(E)) \ \forall E \in \mathcal{S}^p \ \text{ and nonsingular } M \in \mathbb{R}^{p\times p}, \tag{3.1}$$

$$d(G,J) \leq \|H_M(GJ - \tfrac{G\bullet J}{p}I)\|_F \ \ \forall G,J \in \mathcal{S}^p_{++} \ \text{ and nonsingular } M \in \mathbb{R}^{p\times p}, \tag{3.2}$$

$$\|E\|_F \leq \frac{\sqrt{2}}{2}\|E - E^T\|_F \ \text{ if } \ H_M(E) = 0 \ \text{ for a nonsingular } M \in \mathbb{R}^{p\times p}. \tag{3.3}$$

Using the above results, we can now prove the following lemmas.

LEMMA 3.1. *Let $(X,y,S) \in \mathcal{N}(\gamma)$ for some $\gamma \in [0, \ \sqrt{2}-1)$. Suppose that $(D_x, \Delta y, D_s) \in \mathcal{S}^{n\times n} \times \mathbb{R}^m \times \mathcal{S}^{n\times n}$ is a solution of the linear system,*

$$H_P(XD_s + D_xS) = H_P(K), \tag{3.4a}$$

$$A_i \bullet D_x = 0, \ \ i = 1,\ldots,m, \tag{3.4b}$$

$$\sum_{i=1}^m \Delta y_i A_i + D_s = 0, \tag{3.4c}$$

for some $K \in \mathbb{R}^{n\times n}$. Then we have

(i) $\|X^{1/2}D_sX^{1/2}\|_F^2 + \|\mu X^{-1/2}D_xX^{-1/2}\|_F^2 \leq \gamma_1^2\|X^{-1/2}KX^{1/2}\|_F^2$,

(ii) $\|X^{-1/2}(XD_s + D_xS - K)X^{1/2}\|_F \leq \gamma_2\|X^{-1/2}KX^{1/2}\|_F$,

where

$$\gamma_1 = \frac{\sqrt{2}+1}{(1-(\sqrt{2}+1)\gamma)}, \quad \gamma_2 = \sqrt{2} + \sqrt{2}\,\gamma_1\gamma.$$

Proof. By denoting

$$X' = PXP^T, \ \ S' = (P^{-1})^TSP^{-1},$$

$$D'_x = PD_xP^T, \ \ D'_s = (P^{-1})^TD_sP^{-1}, \ \ K' = PKP^{-1},$$

$$\delta_x = \|\mu X^{-1/2}D_xX^{-1/2}\|_F, \quad \delta_s = \|X^{1/2}D_sX^{1/2}\|_F,$$

we can write

$$X'D'_s + D'_sX' + D'_xS' + S'D'_x = K' + K'^T \tag{3.5}$$

and

$$D'_x \bullet D'_s = 0.$$

It is easily seen that $\hat{Q} = (X')^{1/2}(P^{-1})^TX^{-1/2}$ is orthogonal. Then

$$(X')^{1/2} = \hat{Q}X^{1/2}P^T = PX^{1/2}\hat{Q}^T$$

and

$$(X')^{-1/2} = (P^{-1})^TX^{-1/2}\hat{Q}^T = \hat{Q}X^{-1/2}P^{-1}.$$

Using the notation

$$B = X'D_s' + D_x'S' - K',$$

it follows from (3.5) that

$$H_M((X')^{-1/2}B(X')^{1/2}) = 0 \quad \text{for} \quad M = (X')^{1/2}, \tag{3.6}$$

$$\begin{aligned}(X')^{-1/2}B(X')^{1/2} &= (X')^{1/2}D_s'(X')^{1/2} + (X')^{-1/2}D_x'S'(X')^{1/2} - (X')^{-1/2}K'(X')^{1/2} \\ &= \hat{Q}[X^{1/2}D_sX^{1/2} + X^{-1/2}D_xSX^{1/2} - X^{-1/2}KX^{1/2}]\hat{Q}^T. \end{aligned} \tag{3.7}$$

Using (3.6), (3.7), and (3.3) with $E = (X')^{-1/2}B(X')^{1/2}$, we have

$$\begin{aligned}&\|(X')^{-1/2}B(X')^{1/2}\|_F \\ &\quad \le \frac{\sqrt{2}}{2}\|(X')^{-1/2}B(X')^{1/2} - [(X')^{-1/2}B(X')^{1/2}]^T\|_F \\ &\quad \le \frac{\sqrt{2}}{2}\|X^{-1/2}D_xSX^{1/2} - [X^{-1/2}D_xSX^{1/2}]^T\|_F \\ &\qquad + \frac{\sqrt{2}}{2}\|X^{-1/2}KX^{1/2} - [X^{-1/2}KX^{1/2}]^T\|_F \\ &\quad \le \frac{\sqrt{2}}{2}\|X^{-1/2}D_xX^{-1/2}(X^{1/2}SX^{1/2} - \mu I) - [X^{-1/2}D_xX^{-1/2}(X^{1/2}SX^{1/2} - \mu I)]^T\|_F \\ &\qquad + \sqrt{2}\|X^{-1/2}KX^{1/2}\|_F \\ &\quad \le \sqrt{2}\gamma\delta_x + \sqrt{2}\|X^{-1/2}KX^{1/2}\|_F. \end{aligned} \tag{3.8}$$

On the other hand, using (3.7) again, we obtain

$$\begin{aligned}&\|X^{-1/2}KX^{1/2}\|_F \\ &\quad = \|\hat{Q}X^{-1/2}KX^{1/2}\hat{Q}^T\|_F \\ &\quad \ge \|X^{1/2}D_sX^{1/2} + X^{-1/2}D_xSX^{1/2}\|_F - \|(X')^{-1/2}B(X')^{1/2}\|_F \\ &\quad = \|X^{1/2}D_sX^{1/2} + \mu X^{-1/2}D_xX^{-1/2} + X^{-1/2}D_xX^{-1/2}(X^{1/2}SX^{1/2} - \mu I)\|_F \\ &\qquad - \|(X')^{-1/2}B(X')^{1/2}\|_F \\ &\quad \ge \|X^{1/2}D_sX^{1/2} + \mu X^{-1/2}D_xX^{-1/2}\|_F \\ &\qquad - \|X^{-1/2}D_xX^{-1/2}(X^{1/2}SX^{1/2} - \mu I)\|_F - \|(X')^{-1/2}B(X')^{1/2}\|_F \\ &\quad \ge \left(\delta_x^2 + \delta_s^2\right)^{1/2} - \gamma\delta_x - \|(X')^{-1/2}B(X')^{1/2}\|_F \\ &\quad \ge \left(\delta_x^2 + \delta_s^2\right)^{1/2} - (\sqrt{2}+1)\gamma\delta_x - \sqrt{2}\|X^{-1/2}KX^{1/2}\|_F \\ &\quad \ge \left(\delta_x^2 + \delta_s^2\right)^{1/2}(1 - (\sqrt{2}+1)\gamma) - \sqrt{2}\|X^{-1/2}KX^{1/2}\|_F, \end{aligned}$$

which implies (i). Then (ii) follows from (i), (3.8), and the fact that

$$\|X^{-1/2}(XD_s + D_xS - K)X^{1/2}\|_F = \|(X')^{-1/2}B(X')^{1/2}\|_F. \qquad \square$$

It is interesting to note that the inequalities in the above lemma are independent of the scaling matrix P due to the centrality of the iterates. In the next lemma we establish a lower bound for the stepsize $\overline{\theta}_k$, which together with Lemma 3.1 enables us to analyze the asymptotic behavior of the predictor-corrector algorithm.

Lemma 3.2. *Let (X^k, S^k), (U, V), and $\overline{\theta}_k$ be generated by the predictor-corrector algorithm. Then*

$$\overline{\theta}_k \geq \hat{\theta}_k,$$

where

$$\begin{aligned} \omega_k &= \frac{1}{\mu_k} \|(X^k)^{-1/2}(X^k V + U S^k + X^k S^k)(X^k)^{1/2}\|_F, \\ \delta_k &= \frac{1}{\mu_k} \|(X^k)^{-1/2} U V (X^k)^{1/2}\|_F, \\ \hat{\theta}_k &= \frac{2}{\sqrt{\left(\frac{\omega_k+\beta-\alpha}{\beta-\alpha}\right)^2 + \frac{4\delta_k}{\beta-\alpha}} + \frac{\omega_k+\beta-\alpha}{\beta-\alpha}}. \end{aligned}$$

Proof. For simplicity, let us omit the index k. By (2.8), we have

$$X(\theta)S(\theta) = XS + \theta(XV + US) + \theta^2 UV,$$

which together with the linearity of $H_P(\cdot)$, the fact that $\mathrm{Tr}[H_P(M)] = \mathrm{Tr} M$ for $M \in \mathbb{R}^{n\times n}$, and (2.5a) with $\xi = 0$ implies that

$$\begin{aligned} X(\theta) \bullet S(\theta) &= \mathrm{Tr}[X(\theta)S(\theta)] \\ &= \mathrm{Tr}[H_P(X(\theta)S(\theta))] \\ &= \mathrm{Tr}[(1-\theta)H_P(XS) + \theta^2 H_P(UV)] \\ &= (1-\theta)X \bullet S + \theta U \bullet V. \end{aligned}$$

Using the fact that $U \bullet V = 0$ we have

$$\mu(\theta) = (X(\theta) \bullet S(\theta))/n = (1-\theta)(X \bullet S)/n = (1-\theta)\mu.$$

Therefore,

$$X(\theta)S(\theta) - \mu(\theta)I = (1-\theta)(XS - \mu I) + \theta(XV + US + XS) + \theta^2 UV$$

and

$$\begin{aligned} &\|H_{X^{-1/2}}(X(\theta)S(\theta) - \mu(\theta)I)\|_F \\ &\qquad = (1-\theta)\|X^{1/2}SX^{1/2} - \mu I\|_F + \theta\|X^{-1/2}(XV + US + XS)X^{1/2}\|_F \\ &\qquad\quad + \theta^2\|X^{-1/2}UVX^{1/2}\|_F \\ &\qquad \leq (1-\theta)\alpha\mu + \omega\mu\theta + \delta\mu\theta^2 \\ &\qquad \leq (1-\theta)\mu\beta \quad \text{for} \quad 0 \leq \theta \leq \hat{\theta} \\ &\qquad = \beta\mu(\theta). \end{aligned}$$

Hence, we have $X(\theta) \succ 0$ and $S(\theta) \succ 0 \; \forall\, \theta \in [0, \hat{\theta}]$. Otherwise, there exists a $\theta' \in [0, \hat{\theta}]$ such that $X(\theta')S(\theta')$ is singular, which means

$$\lambda_{\min}(X(\theta')S(\theta') - \mu(\theta')I) \leq -\mu(\theta'). \tag{3.9}$$

On the other hand, (3.1) with $M = X^{-1/2}$ and $E = X(\theta')S(\theta') - \mu(\theta')I$ implies that

$$\begin{aligned}\lambda_{\min}(X(\theta')S(\theta') - \mu(\theta')I) &\geq \lambda_{\min}(H_{X^{-1/2}}(X(\theta')S(\theta') - \mu(\theta')I)) \\ &\geq -\|H_{X^{-1/2}}(X(\theta')S(\theta') - \mu(\theta')I)\|_F \\ &\geq -\beta\mu(\theta') > -\mu(\theta'),\end{aligned}$$

which contradicts (3.9). Using (3.2) with $G = X(\theta), J = S(\theta)$, and $M = X^{-1/2}$, we have

$$d(X(\theta), S(\theta)) \leq \|H_{X^{-1/2}}(X(\theta)S(\theta) - \mu(\theta)I)\|_F \leq \beta\mu(\theta) \quad \text{for} \quad \theta \in [0, \hat{\theta}].$$

Therefore, $(X(\theta), S(\theta)) \in \mathcal{N}(\beta)$ for $0 \leq \theta \leq \hat{\theta}$. The result follows from the definition of $\overline{\theta}$. □

4. Superlinear convergence under strict complementarity and nondegeneracy.

In this section we will investigate the asymptotic behavior of the predictor-corrector algorithm. Throughout the paper we assume that the SDP problem has a strictly complementary solution (X^*, y^*, S^*) of (2.3), i.e., $X^* + S^* \succ 0$. We will also assume the following nondegeneracy condition introduced by Kojima, Shida, and Shindoh [4, 5]. First, let us define an affine space $\mathcal{G}_0$ by

$$\mathcal{G}_0 = \Bigg\{(U, V) \in \mathcal{S}^n \times \mathcal{S}^n : A_i \bullet U = 0,\ i = 1, \cdots, m, \\ \sum_{i=1}^m w_i A_i + V = 0,\ w_i \in \mathbb{R}^m\Bigg\}.$$

Nondegeneracy Assumption. If $X^*V + US^* = 0$ and $(U, V) \in \mathcal{G}_0$, then $(U, V) = 0$.

For simplicity, throughout the paper we will use $\{(\overline{X}^k, \overline{S}^k)\} = \{(X^k(\overline{\theta}_k), S^k(\overline{\theta}_k))\}$ to denote the predicted pairs of the predictor-corrector algorithm. As remarked in section 5 of Kojima, Shida, and Shindoh [5], under the strict complementarity assumption, the above nondegeneracy condition is equivalent to the combination of primal and dual nondegeneracy conditions given by Alizadeh, Haeberly, and Overton [2]. Under these assumptions, the solution (X^*, S^*) is unique. Therefore the iteration sequence $\{(X^k,\ S^k)\}$ converges to (X^*, S^*) and so does the sequence of predicted pairs $\{(\overline{X}^k,\ \overline{S}^k)\}$.

Lemma 4.1 (see Kojima, Shida, and Shindoh [5, Lemma 5.3]). *Assume that*

$$H_I(US^* + X^*V) = 0 \quad \text{and} \quad (U, V) \in \mathcal{G}_0.$$

Then $(U, V) = (0, 0)$.

Let R be a nonsingular matrix and

$$\tilde{A}_i = (R^{-1})^T A_i R^{-1},\ i = 1, \ldots, m, \qquad \tilde{C} = (R^{-1})^T C R^{-1}, \tilde{b} = b.$$

It is easily seen that the R-scaled SDP

$$\tilde{A}_i \bullet X = \tilde{b}_i,\ i = 1, \ldots, m, \tag{4.1a}$$

$$\sum_{i=1}^m y_i \tilde{A}_i + S = \tilde{C}, \tag{4.1b}$$

$$XS = 0, \quad X \succeq 0, \quad S \succeq 0, \tag{4.1c}$$

also satisfies the strict complementarity and nondegeneracy conditions. Its unique solution is $(RX^*R^T,\ y^*, (R^{-1})^T S^* R^{-1})$.

Using Lemma 4.1 and considering the new SDP (4.1), we can easily obtain the following lemma.

LEMMA 4.2. *Assume that for some nonsingular matrix* R,

$$H_R(US^* + X^*V) = 0 \quad \textit{and} \ (U, V) \in \mathcal{G}_0.$$

Then $(U, V) = (0, 0)$.

For a strict complementarity solution (X^*, S^*), there exists an orthogonal matrix $Q = (q_1, \ldots, q_n)$ whose columns $q_1, \ldots, q_n$ are common eigenvectors of X^* and S^*, and define

$$\mathbb{B} = \{i : q_i^T X^* q_i > 0\}, \qquad \mathbb{N} = \{i : q_i^T S^* q_i > 0\}.$$

It is easily seen that $\mathbb{B} \cup \mathbb{N} = \{1, 2, \ldots, n\}$. For simplicity, let us assume that

$$Q^T X^* Q = \begin{pmatrix} \Lambda_B & 0 \\ 0 & 0 \end{pmatrix}, \quad Q^T S^* Q = \begin{pmatrix} 0 & 0 \\ 0 & \Lambda_N \end{pmatrix},$$

where Λ_B and Λ_N are diagonal matrices. Here and in what follows, if we write a matrix M in the block form

$$M = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix},$$

then we assume that the dimensions of M_{11} and M_{22} are $|\mathbb{B}| \times |\mathbb{B}|$ and $|\mathbb{N}| \times |\mathbb{N}|$, respectively.

Using the fact that

$$\mu_{k+1} = \mu_k(\overline{\theta}) = (1 - \overline{\theta})\mu_k,$$

as in Lemma 4.4 of Potra and Sheng [13], we can write

$$Q^T (\overline{X}^k)^{1/2} Q = \begin{pmatrix} O(1) & O(\sqrt{\mu_{k+1}}) \\ O(\sqrt{\mu_{k+1}}) & O(\sqrt{\mu_{k+1}}) \end{pmatrix}, \quad Q^T (\overline{X}^k)^{-1/2} Q = \begin{pmatrix} O(1) & O(1) \\ O(1) & O(1/\sqrt{\mu_{k+1}}) \end{pmatrix},$$

$$Q^T (\overline{S}^k)^{1/2} Q = \begin{pmatrix} O(\sqrt{\mu_{k+1}}) & O(\sqrt{\mu_{k+1}}) \\ O(\sqrt{\mu_{k+1}}) & O(1) \end{pmatrix}, \quad Q^T (\overline{S}^k)^{-1/2} Q = \begin{pmatrix} O(1/\sqrt{\mu_{k+1}}) & O(1) \\ O(1) & O(1) \end{pmatrix}.$$

Thus, we have

$$\|\overline{X}^k \overline{S}^k + \overline{S}^k \overline{X}^k\|_F = O(\sqrt{\mu_{k+1}}). \tag{4.2}$$

Under strict complementarity and nondegeneracy assumptions, superlinear convergence with Q-order 1.5 is proved in [15, Theorem 4.3] if the KSH/HRVW/M search direction and the AHO direction are used in the predictor and corrector steps, respectively. Using a similar argument, we can extend the result to the case in which $\|\overline{P}^k\|_F \|(\overline{P}^k)^{-1}\|_F$ is bounded. We define $\mathrm{cond}_F(B) \equiv \|B\|_F \|B^{-1}\|_F$ as the condition number of a matrix B. First, we need the following technical result.

LEMMA 4.3. *If* $\mathrm{cond}_F(\overline{P}^k) = O(1)$, *then*

$$\|(\overline{U}^k, \overline{V}^k)\|_F = O(\sqrt{\mu_{k+1}}). \tag{4.3}$$

Proof. Let $R^k = \overline{P}^k / \|\overline{P}^k\|_F$. Then $\|R^k\|_F = 1$ and $\|(R^k)^{-1}\|_F = \mathrm{cond}_F(\overline{P}^k) = O(1)$. At the corrector step of the algorithm, we have

$$H_{R^k}\left(\overline{U}^k\overline{S}^k + \overline{X}^k\overline{V}^k\right) = H_{R^k}\left(\mu_{k+1}I - \overline{X}^k\overline{S}^k\right). \tag{4.4}$$

Suppose (4.3) is not true, i.e., the sequence $\{(\overline{U}^k, \overline{V}^k)/\sqrt{\mu_{k+1}}\}$ is unbounded. Then we can choose a subsequence such that

$$\frac{(\overline{U}^k, \overline{V}^k)}{\sqrt{\mu_{k+1}}} \to \infty,$$

$$R^k \to R^*, \quad (R^k)^{-1} \to (R^*)^{-1},$$

and

$$\frac{(\overline{U}^k, \overline{V}^k)}{\|(\overline{U}^k, \overline{V}^k)\|_F} \to (U', V') \neq 0.$$

Obviously, $(U',\ V') \in \mathcal{G}_0$. The fact that the matrices A_i, $i = 1, \ldots, m$, are linearly independent, together with $(U',\ V') \in \mathcal{G}_0$, implies that $(U', V') \neq 0$. Dividing both sides of (4.4) by $\|(\overline{U}^k, \overline{V}^k)\|_F$ and letting $k \to \infty$ along a subsequence, together with (4.2), we obtain

$$H_{R^*}(U'S^* + X^*V') = 0,$$

which contradicts Lemma 4.2. □

Let us define a linear manifold

$$\mathcal{M} \equiv \Bigg\{(X', y', S') \in \mathcal{S}^n \times \mathbb{R}^m \times \mathcal{S}^n : \begin{array}{l} A_i \bullet X' = b_i, \quad i = 1, \ldots, m, \\ \displaystyle\sum_{i=1}^m y_i' A_i + S' = C, \\ q_i^T X' q_j = 0 \text{ if } i \text{ or } j \in \mathbb{N}, \\ q_i^T S' q_j = 0 \text{ if } i \text{ or } j \in \mathbb{B} \end{array}\Bigg\}.$$

The following quantity plays an important role in our analysis:

$$\eta_k = \eta_k(\Gamma) = \frac{1}{\mu_k}\|(X^k)^{-1/2}(X^k - \breve{X}^k)(S^k - \breve{S}^k)(X^k)^{1/2}\|_F, \tag{4.5}$$

where $(\breve{X}^k, \breve{y}^k, \breve{S}^k)$ is the solution of the following minimization problem:

$$\min\{\|(X^k)^{-1/2}(X^k - X')(S^k - S')(X^k)^{1/2}\|_F : (X', y', S') \in \mathcal{M},\ \|(X', S')\|_F \leq \Gamma\}, \tag{4.6}$$

and Γ is a constant such that $\|(X^k, S^k)\|_F \leq \Gamma\ \forall k$. Note that every accumulation point of (X^k, y^k, S^k) belongs to the feasible set of the above minimization problem and the feasible set is bounded. Therefore $(\breve{X}^k, \breve{S}^k)$ exists for each k.

THEOREM 4.4. *Under the strict complementarity and nondegeneracy assumptions, if* $\operatorname{cond}_F(\overline{P}^k) = O(1)$, *then the algorithm is superlinearly convergent with Q-order at least* 1.5.

Proof. The proof is similar to those of [14, Theorem 4.3] and [13, Theorem 4.7]. For the sake of completeness, we include a sketch of its proof. At the predictor step, we have

$$(X^k,\ S^k) = (\overline{X}^{k-1},\ \overline{S}^{k-1}) + (\overline{U}^{k-1},\ \overline{V}^{k-1}).$$

Thus,

$$H_{\overline{P}^{k-1}}(X^k S^k) = \mu_k I + H_{\overline{P}^{k-1}}(\overline{U}^{k-1}\overline{V}^{k-1}).$$

Then, by Lemma 4.3, we obtain

$$\|H_{\overline{P}^{k-1}}(X^k S^k)\|_F = O(\mu_k).$$

Note that

$$\begin{aligned} \|H_{\overline{P}^{k-1}}(X^k S^k)\|_F^2 &= \frac{1}{2}\|\overline{P}^{k-1} X^k S^k (\overline{P}^{k-1})^{-1}\|_F^2 + \frac{1}{2}\|(X^k)^{1/2} S^k (X^k)^{1/2}\|_F^2 \\ &\geq \frac{1}{2}\|\overline{P}^{k-1} X^k S^k (\overline{P}^{k-1})^{-1}\|_F^2. \end{aligned} \tag{4.7}$$

Therefore,

$$\|X^k S^k\|_F = \|(\overline{P}^{k-1})^{-1}\overline{P}^{k-1} X^k S^k (\overline{P}^{k-1})^{-1}\overline{P}^{k-1}\|_F \leq O(1)\|H_{\overline{P}^{k-1}}(X^k S^k)\|_F = O(\mu_k). \tag{4.8}$$

Define

$$\phi_k = \max\{\|X^k S^k\|_F/\sqrt{\mu_k}, \sqrt{\mu_k}\}.$$

In view of (4.8), we deduce that $\phi_k = O(\sqrt{\mu_k})$. From the proofs of [14, Theorems 4.9 and 6.1], we see that $\eta_k = O(\phi_k) = O(\sqrt{\mu_k})$. For simplicity, let us omit the index k. It is easily seen that $(U + X - \breve{X}, w + y - \breve{y}, V + S - \breve{S})$ satisfies (3.4) with

$$K = (X - \breve{X})(S - \breve{S}). \tag{4.9}$$

Here we have used the relation $\breve{X}\breve{S} = \breve{S}\breve{X} = 0$. The matrix

$$\Delta = X^{-1/2}(X - \breve{X})(S - \breve{S})X^{1/2} \tag{4.10}$$

clearly satisfies the equation

$$\|\Delta\|_F = \|X^{-1/2} K X^{1/2}\|_F = \eta\mu. \tag{4.11}$$

Denoting

$$\Delta_x = X^{-1/2}(U + X - \breve{X})X^{-1/2}, \qquad \Delta_s = X^{1/2}(V + S - \breve{S})X^{1/2},$$

and applying (i) of Lemma 3.1, we obtain

$$\mu\|\Delta_x\|_F \leq \gamma_1\|\Delta\|_F = \gamma_1\eta\mu,$$

which implies

$$\|\Delta_x\|_F \le \gamma_1 \eta. \tag{4.12}$$

Similarly,

$$\|\Delta_s\|_F \le \gamma_1 \|\Delta\|_F = \gamma_1 \eta\mu. \tag{4.13}$$

Using (4.11)–(4.13) and following the proof of [13, Theorem 4.7], we have

$$\begin{aligned}
&\|X^{-1/2}(X-\breve{X})X^{-1/2}\|_F = O(1)\ , \qquad \|S^{-1/2}(S-\breve{S})S^{-1/2}\|_F = O(1)\ ,\\
&\|X^{-1/2}UVX^{1/2}\|_F \le \|\Delta_x\|_F\|\Delta_s\|_F + \|X^{-1/2}(X-\breve{X})X^{-1/2}\|_F\|\Delta_s\|_F\\
&\qquad\qquad + \|X^{1/2}S^{1/2}\|^2\|\Delta_x\|_F\|S^{-1/2}(S-\breve{S})S^{-1/2}\|_F + \|\Delta\|_F\\
&\qquad\qquad = O(\eta\mu).
\end{aligned}$$

Hence, $\delta_k = O(\sqrt{\mu_k})$. Applying (ii) of Lemma 3.1, we obtain

$$\|X^{-1/2}[X(V+S-\breve{S}) + (U+X-\breve{X})S - K]X^{1/2}\|_F \le \gamma_2\|\Delta\|_F = \gamma_2\eta\mu\ .$$

Noting that

$$X(V+S-\breve{S}) + (U+X-\breve{X})S - K = XV + US + XS,$$

we deduce

$$\omega = \frac{1}{\mu}\|X^{-1/2}(XV+US+XS)X^{1/2}\|_F \le \gamma_2\eta.$$

Hence, $\omega_k = O(\sqrt{\mu_k})$. From Lemma 3.2, it follows that

$$\begin{aligned}
1-\overline{\theta} &\le 1 - \frac{2}{\sqrt{\left(\frac{\omega}{\beta-\alpha}+1\right)^2 + \frac{4\delta}{\beta-\alpha}} + \frac{\omega}{\beta-\alpha} + 1}\\
&\le \frac{3\omega}{\beta-\alpha} + \frac{\omega^2}{(\beta-\alpha)^2} + \frac{4\delta}{\beta-\alpha} = O(\sqrt{\mu})
\end{aligned}$$

Therefore, $\mu_{k+1} = (1-\overline{\theta}_k)\mu_k = O(\mu_k^{1.5})$. □

The above result says that the superlinear convergence of the predictor-corrector algorithm is independent of the choice of the scaling matrix P^k in the predictor step of the algorithm, while the scaling matrices used in the corrector step need to be "well-conditioned" for superlinear convergence. Clearly, the family of scaling matrices admissible in the corrector step for superlinear convergence includes the identity matrix defining the AHO as a special case. By imposing the same assumption on the scaling matrices used in the predictor step and a new strategy for the step size, we can improve the order of convergence stated in Theorem 4.4.

In order to achieve quadratic convergence we need to slightly modify the choice of the step size. Instead of $\overline{\theta}_k$ given by (2.9), we will use

$$\overline{\theta}_k = \max\left\{\tilde{\theta} \in [0, \max\{.99, 1-\mu_k^2\}] : (X^k(\theta), S^k(\theta)) \in \mathcal{N}(\beta)\ \ \forall\, \theta \in [0,\tilde{\theta}]\right\}. \tag{4.14}$$

The predictor-corrector algorithm with this new strategy will be called the modified predictor-corrector algorithm. It is easily seen that the modified predictor-corrector algorithm still has polynomial complexity. In what follows we will show that it is also quadratically convergent.

THEOREM 4.5. *Under the hypothesis of Theorem* 4.4, *if* $\operatorname{cond}_F(P^k) = O(1)$, *then the modified predictor-corrector algorithm is quadratically convergent.*

Proof. From the proof of Theorem 4.4 (cf. (4.8)), we have

$$X^k S^k = O(\mu_k). \tag{4.15}$$

Using (4.15) and an argument similar to that employed in the proof of Lemma 4.3 we get

$$U^k = O(\mu_k), \quad V^k = O(\mu_k). \tag{4.16}$$

Then we can write

$$\begin{aligned} H_{P^k}(\overline{X}^k\overline{S}^k) &= H_{P^k}([X^k + \overline{\theta}_k U^k][S^k + \overline{\theta}_k V^k]) \\ &= (1-\overline{\theta}_k) H_{P^k}(X^k S^k) + \overline{\theta}_k^2 H_{P^k}(U^k V^k) \\ &= (1-\overline{\theta}_k) O(\mu_k) + O(\mu_k^2) = O(\omega_k \mu_k), \end{aligned}$$

where $\omega_k = \max\{\mu_k,\ 1-\overline{\theta}_k\}$. As in (4.7)–(4.8), we can prove that

$$\|\overline{X}^k\overline{S}^k\|_F = O(\omega_k \mu_k), \tag{4.17}$$

which further implies

$$\|\overline{X}^k\overline{S}^k\|_F / \|(\overline{U}^k, \overline{V}^k)\|_F \le O(\omega_k\mu_k/\|(\overline{U}^k,\overline{V}^k)\|_F). \tag{4.18}$$

Based on (4.18) and the fact that

$$\mu_{k+1}/\|(\overline{U}^k,\overline{V}^k)\|_F = (1-\overline{\theta}_k)\mu_k/\|(\overline{U}^k,\overline{V}^k)\|_F \le \omega_k\mu_k/\|(\overline{U}^k,\overline{V}^k)\|_F,$$

a similar argument employed in the proof of Lemma 4.3 can be used to deduce

$$\overline{U}^k = O(\omega_k\mu_k) \quad \text{and} \quad \overline{V}^k = O(\omega_k\mu_k). \tag{4.19}$$

Observing that

$$H_{\overline{P}^k}(X^{k+1}S^{k+1}) - \mu_{k+1} I = H_{\overline{P}^k}(\overline{U}^k\overline{V}^k) = O(\omega_k^2\mu_k^2),$$

we have

$$\|H_{\overline{P}^k}(X^{k+1}S^{k+1}) - \mu_{k+1} I\|_F/\mu_{k+1} = \frac{O(\omega_k^2\mu_k^2)}{\mu_{k+1}} = \frac{O(\omega_k^2\mu_k)}{1-\overline{\theta}_k}.$$

Since $\operatorname{cond}_F(P^k) \le C_1$ and $\operatorname{cond}_F(\overline{P}^k) \le C_1$ for some constant C_1, we can write

$$\begin{aligned} \|H_{P^k}(X^kS^k) - \mu_k I\|_F/\mu_k &\le C_1\|X^kS^k - \mu_k I\|_F/\mu_k \\ &\le C_1^2 \|\overline{P}^{k-1}X^kS^k(\overline{P}^{k-1})^{-1} - \mu_k I\|_F/\mu_k \\ &\le C_2\mu_{k-1}\max\left\{\frac{\mu_{k-1}^2}{1-\overline{\theta}_{k-1}}, 1-\overline{\theta}_{k-1}\right\}, \end{aligned} \tag{4.20}$$

where C_2 is a positive constant. Without loss of generality, we may assume

$$\mu_{k-1} \le \min\{.1, \alpha/C_2\} \qquad \text{for} \;\; k \ge K,$$

which, together with (4.14) and (4.20), implies that

$$\overline{\theta}_{k-1} \le 1 - \mu_{k-1}^2 \qquad \text{for} \;\; k \ge K$$

and

$$\overline{\alpha}_k \equiv \|H_{P^k}(X^k S^k) - \mu_k I\|_F / \mu_k \le C_2 \mu_{k-1} \le \alpha \qquad \text{for} \;\; k \ge K.$$

Let

$$\theta'_k = \frac{2}{\sqrt{1 + 4\overline{\delta}_k/(\beta - \overline{\alpha}_k)} + 1}, \;\; \overline{\delta}_k = \|P^k U^k V^k (P^k)^{-1}\|_F / \mu_k.$$

Evidently, $\overline{\delta}_k = O(\mu_k)$. Then $\forall\, \theta \in [0, \theta'_k]$, it follows from (3.2) that

$$\begin{aligned} \|(X^k(\theta))^{1/2} S^k(\theta) (X^k(\theta))^{1/2} - \mu_k(\theta) I\|_F &\le \|H_{P^k}(X^k(\theta) S^k(\theta)) - \mu_k(\theta) I\|_F \\ &\le [(1-\theta)\overline{\alpha}_k + \theta^2 \overline{\delta}_k]\mu_k \le \beta(1-\theta)\mu_k. \end{aligned}$$

This means

$$\overline{\theta}_k \ge \theta'_k \qquad \text{for} \;\; k \ge K.$$

Therefore

$$1 - \overline{\theta}_k \le 1 - \theta'_k = O(\overline{\delta}_k) = O(\mu_k)$$

and $\mu_{k+1} = (1 - \overline{\theta}_k)\mu_k = O(\mu_k^2)$. □

5. Remarks. In this paper we consider only the feasible version of the predictor-corrector method to keep the presentation simple. However, the analysis used here can be easily extended to the infeasible predictor-corrector algorithms based on the unified direction proposed by Monteiro and Zhang. Under the strict complementarity and nondegeneracy assumptions we have established the superlinear convergence with Q-order 1.5 of the "pure" predictor-corrector algorithm if the scaling matrices for the corrector step satisfy $\mathrm{cond}_F(\overline{P}^k) = O(1)$. Whether superlinear convergence can be obtained under a weaker condition is an interesting topic for future research. Finally, we mention that quadratic convergence is established for the predictor-corrector algorithm with a slight modification of the step size selection. It would be interesting to find out whether quadratic convergence can be proved for the "original" predictor-corrector algorithm.

REFERENCES

[1] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability, and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.

[2] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.

[3] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[4] M. Kojima, M. Shida, and S. Shindoh, *Local convergence of predictor-corrector infeasible-interior-point method for SDPs and SDLCPs*, Math. Programming, 80 (1998), pp. 129–160.

[5] M. Kojima, M. Shida, and S. Shindoh, *A predictor-corrector interior-point algorithm for the semidefinite linear complementarity problem using the Alizadeh–Haeberly–Overton search direction*, SIAM J. Optim., 9 (1999), pp. 444–465.

[6] M. Kojima, S. Shindoh, and S. Hara, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.

[7] Z.-Q. Luo, J. F. Sturm, and S. Zhang, *Superlinear convergence of a symmetric primal-dual path-following algorithm for semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 59–81.

[8] R. D. C. Monteiro, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.

[9] R. D. C. Monteiro, *Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions*, SIAM J. Optim., 8 (1998), pp. 797–812.

[10] R. D. C. Monteiro and S. J. Wright, *Local convergence of interior-point algorithms for degenerate monotone LCP*, Comput. Optim. Appl., 3 (1994), pp. 131–155.

[11] R. D. C. Monteiro and Y. Zhang, *A unified analysis for a class of path-following primal-dual interior-point algorithms for semidefinite programming*, Math. Programming, 81 (1998), pp. 281–299.

[12] Y. E. Nesterov and M. J. Todd, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

[13] F. A. Potra and R. Sheng, *A superlinearly convergent primal-dual infeasible-interior-point algorithm for semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 1007–1028.

[14] F. A. Potra and R. Sheng, *Superlinear convergence of interior-point algorithms for semidefinite programming*, J. Optim. Theory Appl., 99 (1998), pp. 103–119.

[15] F. A. Potra and R. Sheng, *Superlinear convergence of a predictor-corrector method for semidefinite programming without shrinking central path neighborhood*, Rep. Comput. Math. 91, Department of Mathematics, The University of Iowa, Iowa City, IA, 1996.

[16] R. Sheng, F. A. Potra, and J. Ji, *On a general class of interior-point algorithms for semidefinite programming with polynomial complexity and superlinear convergence*, Rep. Comput. Math. 89, Department of Mathematics, The University of Iowa, Iowa City, IA, 1996.

[17] Y. Zhang, *On extending primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 211–230

PENALTY AND BARRIER METHODS: A UNIFIED FRAMEWORK*

A. AUSLENDER[†]

Abstract. It is established that many optimization problems may be formulated in terms of minimizing a function $x \to f_0(x) + H_\infty(f_1(x), f_2(x), \dots, f_m(x)) + L_\infty(Ax - b)$, where the f_i are closed functions defined on $\mathbb{R}^N$, and where H_∞ and L_∞ are the recession functions of closed, proper, convex functions H and L. A is a linear transformation from $\mathbb{R}^N$ to a finite dimensional vector space Y with $b \in Y$. A generic algorithm, based on the properties of recession functions, is proposed. This algorithm not only encompasses almost all penalty and barrier methods in nonlinear programming and in semidefinite programming, but also generates new types of methods. Primal and dual convergence theorems are given.

Key words. convex and nonlinear programming, semidefinite programming, penalty and barrier methods

AMS subject classifications. 90C05, 90C25, 90C31, 65F15, 90C48

PII. S1052623497324825

1. Introduction.

In [3], Ben Tal and Teboulle remarked that there are many optimization problems that may be formulated as follows:

$$(P^{B,T}) \qquad \text{minimize } \phi(x) := V_\infty(g_1(x), \dots, g_p(x)) \quad \text{over } \mathbb{R}^N,$$

where V_∞ is the recession or asymptotic function of V (see the definition on p. 66 of [15]). The functions g_j and V defined on $\mathbb{R}^N$ and $\mathbb{R}^P$ are closed and proper and V is assumed to be convex. For the definition of *closed and proper*, we refer to Rockafellar's book [15]. Furthermore, all notation and definitions in this paper are standard and can be found in [15].

In [3], some additional assumptions are required on V and g_j, in particular,

$$(1.1) \qquad 0 \in \text{ dom } V := \{y : V(y) < +\infty\}, \qquad rV\left(\frac{y}{r}\right) \geq V_\infty(y) \quad \forall y, \ \forall r > 0.$$

Indeed, in many cases the V in question is finite. The following examples [3], [4] are of particular interest:

(i) ℓ_1-norm approximation problems,

$$V(y) = \sum_{i=1}^{p} \sqrt{1 + y_i^2}, \quad V_\infty(y) = \sum_{i=1}^{p} |y_i|,$$

(ii) discrete minmax problems,

$$V(y) = \log\left(\sum_{i=1}^{p} e^{y_i}\right), \quad V_\infty(y) = \max_{i=1,2,\dots,p} y_i,$$

Received by the editors July 23, 1997; accepted for publication (in revised form) December 7, 1998; published electronically November 29, 1999. This work was supported by CEE grant CI1 CT 92-0046.

http://www.siam.org/journals/siopt/10-1/32482.html

[†]Laboratoire d'Econométrie, Ecole Polytechnique I, 1 rue Descartes, Paris 75005 (auslen@poly.polytechnique.fr).

(iii) nonsmooth problems,

$$V(y) = \sum_{i=1}^{N} \sqrt{1 + \sum_{j=1}^{r} d_{ij}(y_i^j)^2}, \quad V_\infty(y) = \sum_{i=1}^{N} \sqrt{\sum_{j=1}^{r} d_{ij}(y_i^j)^2}$$

with $d_{ij} > 0$.

The ingenious idea proposed by Ben Tal and Teboulle [3] is the following: since $V(0)$ is finite, Theorem 8.5 of [15] implies that

$$V_\infty(y) = \lim_{r \to 0^+} rV\left(\frac{y}{r}\right) \quad \forall y. \tag{1.2}$$

Then, in order to approach $(P^{B,T})$, Ben Tal and Teboulle consider the problem

$$(P_r^{B,T}) \qquad \alpha_r = \inf_x \phi_r(x) := rV\Big(\frac{g_1(x)}{r}, \dots, \frac{g_p(x)}{r}\Big).$$

When V is C^1, the new problem is smooth and the approach is very natural. In fact, the assumption that $0 \in \text{dom}V$ is somewhat restrictive. Recently it was shown by Seeger [16] that (1.2) holds also for polyhedral convex functions. In addition, (1.2) holds generally for each $y \in$ domV (Corollary 8.5.2 of [15]). In fact, for our purpose, it suffices for (1.2) to be valid on the relative interior of $\text{dom}V_\infty$. Furthermore, the other part of assumption (1.1) used only in [3] to obtain convergence results is also restrictive, and the main contribution of this paper consists of the observation that there is a very wide class of optimization problems that may be formulated as follows:

$$(P) \qquad \alpha = \inf\{\phi(x) \mid x \in \mathbb{R}^N\}$$

with
(1.3)

$$\phi(x) = f_0(x) + H_\infty(f_1(x), f_2(x), \dots, f_m(x)) + L_\infty(Ax - b) \text{ if } x \in \bigcap_{i=1}^{n} \text{dom}f_i, \ +\infty \text{ else}$$

under an essential assumption concerning the structure of (P) (denoted by (A_0)). In order to introduce this assumption, let

$$H_r(y) = rH\left(\frac{y}{r}\right) \quad \text{and} \quad L_r(z) = rL\left(\frac{z}{r}\right), \tag{1.4}$$

and denote by riC the relative interior of C. Then (A_0) is stated as follows.

(A_0) $H : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a closed, proper, convex function that is isotonic; i.e.,

$$y_i \le z_i \quad \forall i = 1, 2, \dots, m \Longrightarrow H(y) \le H(z), \tag{1.5}$$

L is a closed, proper, convex function defined on a finite dimensional space Y, and L and H satisfy

$$\text{(i)} \lim_{r\to 0+} H_r(y) = H_\infty(y), \ \lim_{r\to 0+} L_r(z) = L_\infty(z) \ \forall (y, z) \in \ \text{ridom}H_\infty \times \ \text{ridom}L_\infty. \tag{1.6}$$

$$\text{(ii)}\text{ridom}H_\infty \subset \ \text{ridom}H_r \ \text{ and } \ \text{ridom}L_\infty \subset \ \text{ridom}L_r \ \ \forall r > 0. \tag{1.7}$$

(iii) for each $(x, y) \in \mathrm{dom} H_\infty \times \mathrm{dom} L_\infty$, each sequence $\{(x_n, y_n)\} \subset \mathrm{ridom} H_\infty \times \mathrm{ridom} L_\infty$ converging to (x, y) we have

$$\lim_{n\to\infty} H_\infty(x_n) = H_\infty(x), \quad \lim_{n\to\infty} L_\infty(y_n) \ = L_\infty(y). \tag{1.8}$$

(iv) The constancy space of H_∞ (see the definition on p. 69 of [15]) is reduced to zero.

Of course, as will be seen shortly, (A_0) is satisfied in all of the examples given in this paper.

The data of the problem, the f_i, A, and b, are to satisfy the following conditions.

(A_1) (i) A is a linear map from $\mathbb{R}^N$ to Y and b is a vector in Y.

(ii) The functions $f_i : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\},\ i = 0, 1, \ldots, m$, are closed and proper and satisfy

$$(f_i)_\infty(d) > -\infty \ \ \forall d. \tag{1.9}$$

(The definition of the recession function $(f_i)_\infty$ in the nonconvex case is the same as in the convex case [15, p. 66].)

(iii) There exists $x_0 \in \mathrm{dom} f_0$ such that $F(x_0) \in \mathrm{dom} H_\infty$, $Ax_0 - b \in \mathrm{dom} L_\infty$, where

$$F(x) := (f_1(x), f_2(x), \ldots, f_m(x)). \tag{1.10}$$

When $(0, 0) \notin \mathrm{dom} H \times \mathrm{dom} L$, we suppose in addition that

$$F(x_0) \in \ \mathrm{ridom} H_\infty, \quad Ax_0 - b \in \ \mathrm{ridom} L_\infty. \tag{1.11}$$

Note that (1.9) is always satisfied when f_i is convex, so that (A_1) appears as a "minimal" condition on f_i.

Given (1.6), it is natural to approximate (P) by the problem

$$(P_r) \qquad \alpha_r = \inf\{\phi_r(x) | x \in \mathbb{R}^N\},$$

where ϕ_r is defined by

$$\phi_r(x) := f_0(x) + H_r(F(x)) + L_r(Ax - b) \text{ if } x \in \bigcap_{i=1}^{m} \mathrm{dom} f_i, +\infty \text{ otherwise.} \tag{1.12}$$

The isotonicity assumption on H is essential. It implies that H_∞ is isotonic. Then it can be seen easily that given (A_0) and (A_1), the functions ϕ and ϕ_r are closed and proper (this will be proved in Lemma 2.1). Furthermore, if the functions $f_i, i = 1 \ldots m$, are convex, then isotonicity of H ensures that $H_r(F(.))$ and $H_\infty(F(.))$ are also convex. Thus, if f_0 is also convex, then ϕ and ϕ_r are closed, proper, and convex and (P) and (P_r) are convex problems.

Obviously, $(P^{B,T})$ is a particular case of (P), and the theoretical results given by Ben Tal and Teboulle in [3] may be improved. There are many other areas that are encompassed within this framework and not by the one proposed by Ben Tal and Teboulle, in particular, penalty and barrier methods for standard nonlinear constrained problems and barrier methods for semidefinite programming. Indeed, suppose first that we wish to solve the classical constrained optimization problem:

$$(P^m) \qquad \alpha = \inf\{f_0(x) \mid x \in C\}$$

with

$$C = \{x : f_i(x) \leq 0 \quad \forall i = 1, 2, \ldots, m\}.$$

Write for each set D

$$\delta(y \mid D) = 0 \quad \text{if } y \in D, \text{ and } +\infty \text{ otherwise.}$$

Then (P^m) is equivalent to minimizing the function $f_0 + \delta(F(\cdot) \mid \mathbb{R}^m_-)$ on $\mathbb{R}^N$.

A fundamental and original remark is now the following: the indicator function $\delta(\cdot \mid \mathbb{R}^m_-)$ can be considered as the recession function of a wide class of functions.

Indeed, denote by $\mathcal{G}$ the class of functions $\theta : \mathbb{R} \to \mathbb{R} \cup \{+\infty\}$ closed, proper, convex, and nondecreasing with $\text{dom}\theta =]-\infty, \eta[$ and

$$0 \leq \eta \leq +\infty, \quad \theta_\infty(-1) = 0, \quad \theta_\infty(1) = +\infty.$$

For $\theta \in \mathcal{G}$, define H as follows:

$$H(y) = \sum_{i=1}^{m} \theta(y_i). \tag{1.13}$$

Then it can be seen easily that H satisfies the assumption (A_0) and

$$H_\infty(.) = \delta(.|\mathbb{R}^m_-). \tag{1.14}$$

But note that (1.1) is not satisfied, and so this problem cannot be treated with the approach of Ben Tal and Teboulle.

The approximation (P) by (P_r) with H given by (1.13) and $\theta \in \mathcal{G}$ was introduced in this case by Auslender, Cominetti, and Haddou [2]. These authors give a systematic way to generate functions $\theta \in \mathcal{G}$. Then (P_r) is nothing but a penalty or barrier method. Particular cases of interest are

$\theta_1(u) = \exp\,(u)$ (exponential penalty [18]),

$\theta_2(u) = -\log(1-u)$ for $u < 1$, and $= +\infty$ otherwise (modified barrier [13]),

$$\theta_3(u) = \begin{cases} u + \frac{1}{2}u^2 \text{ if } u \geq -\frac{1}{2}, \\ -\frac{1}{4}\log(-2u) - \frac{3}{8} \text{ if } u \leq -\frac{1}{2} \end{cases} \quad \text{(quadratic logarithmic method [5]),}$$

$\theta_4(u) = \frac{u}{1-u}$ for $u < 1$, and $+\infty$ otherwise (hyperbolic modified barrier method [9], [14]),

$\theta_5(u) = -\log(-u)$ for $u < 0$ and $+\infty$ otherwise (logarithmic barrier [10]),

$\theta_6(u) = -\frac{1}{u}$ for $u < 0$, and $+\infty$ otherwise (inverse barrier method [7])

$$\theta_7(u) = \begin{cases} -\log(-u) \text{ if } \delta \leq u < 0 & \text{(truncated logarithmic barrier [6]),} \\ -a - \frac{b}{u^2} - \frac{c}{u} \text{ if } u \leq \delta & \text{and} + \infty \text{ if } u \geq 0, \end{cases}$$

where $\delta < 0$ and the parameters a, b, and $c > 0$ are chosen so that θ_7 is twice differentiable.

In [2], the authors analyzed the existence of primal and dual paths generated by these penalty and barrier methods as well as their convergence to primal and dual optimal sets. Our results will include the above and more, since the nonconvex case will also be considered in this paper, and this is not the case considered in [2]. Furthermore, contrary to [2], we can treat nonseparable functions H, for example,

$$H(y) = -\log\left(-\sum_{i=1}^{m} y_i\right) - \sum_{i=1}^{m} \log(-y_i) \quad \text{if } y < 0, \text{ and } +\infty \text{ otherwise.} \tag{1.15}$$

A surprising application is to semidefinite programming. This will be developed in section 4, but it is easy to outline the approach for positive definite programming. This problem consists of minimizing a linear function of a variable $x \in \mathbb{R}^N$ subject to a linear matrix inequality:

$$(PDP) \qquad \text{minimize} \quad c^t x$$

$$(1.16) \qquad \text{subject to} \quad B(x) \le 0,$$

where $B(x) := (B_0 + \sum_{i=1}^N x_i B_i)$.

The problem data are c and the $N+1$ symmetric $p \times p$ matrices B_i. The inequality sign in (1.16) means that $B(x)$ is negative semidefinite, i.e., that $z^t B(x) z \le 0$ for all $z \in \mathbb{R}^P$.

Here $H := 0, f_0(x) = c^t x$, and Y is the space of $p \times p$ symmetric matrices. For $A, B \in Y$, we use the scalar product $\mathrm{tr}(AB)$ (tr denotes the trace). Let K be the set of negative semidefinite matrices in Y. It is a closed, pointed, convex cone, and its interior is the set of negative definite matrices. Then (PDP) consists of minimizing the function $x \to c^t x + \delta(B(x) \mid K)$ on $\mathbb{R}^N$.

It is easy to verify that $\delta(\cdot \mid K)$ may be considered the recession function of L defined by

$$(1.17) \qquad L(A) = -\mathrm{logdet}\ (-A) \quad \text{if } A \in \ \mathrm{int}\ K, \quad \text{and} + \infty \text{ otherwise,}$$

and that L satisfies assumption (A_0), but there are many other functions that have the same property. This will be analyzed in section 4. The function L defined by (1.17) is the barrier function introduced by Nesterov and Nemirovskii in [14] used for solving problems of this type. With this function (P_r) becomes an interior C^∞ method (see, for example, formulas (37) and (38) in [19], giving the gradient and the Hessian of ϕ_r). Again we note that $0 \notin \mathrm{dom}\ L$, and as a consequence the classical barrier method cannot be handled by the treatment given by Ben Tal and Teboulle [3].

The structure of the paper is now simple. In section 2 we prove under a coercivity condition the existence of a path of optimal solutions $\{x_r\}_{r>0}$ of problem (P_r) and its convergence when r goes to 0 toward the optimal set of the original problem (P). This section concerns the nonconvex case.

In section 3, we consider the convex case (the functions f_i are supposed to be convex), and we study the existence and the convergence of an associated dual minimizing path. In section 4, the theory is applied to semidefinite programming.

Finally, in section 5, other types of approximation that are, again, related to properties of recession functions, are proposed, and in section 6 a conclusion is given.

2. Primal results.

2.1. Preliminaries. In order to prove convergence, let us recall some important properties of recession functions. Recall first that for a set $Q \subset \mathbb{R}^N$, its asymptotic or recession cone is denoted by Q_∞ or 0^+Q and is defined by

$$Q_\infty = \left\{ y \mid \exists t_k \to +\infty, x_k \in Q \text{ with } y = \lim_{k\to\infty} x_k/t_k \right\}.$$

Now, let $f, g : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ be closed, proper functions. By definition, the recession function of f, usually denoted by f_∞ or $f0^+$, is defined by $\mathrm{epi} f_\infty = (\mathrm{epi} f)_\infty$, where epi$f = \{(x,r) : f(x) \leq r\}$. As a straightforward consequence, for each vector y we get

$$f_\infty(y) = \inf\left\{\liminf_{k\to\infty} \frac{f(t_k x_k)}{t_k} : t_k \to +\infty, x_k \to y\right\}, \tag{2.1}$$

where $\{t_k\}$ and $\{x_k\}$ are sequences in $\mathbb{R}$ and $\mathbb{R}^N$, respectively. For closed convex functions we also have for $\lambda > 0$

$$f_\infty(y) = \lim_{\lambda\to+\infty} \frac{f(x+\lambda y) - f(x)}{\lambda} \quad \forall x \in \text{dom} f \quad \text{with } \frac{f(x+\lambda y) - f(x)}{\lambda} \leq f_\infty(y). \tag{2.2}$$

f_∞ is closed, positively homogeneous with $f_\infty(0) = 0$ or $f_\infty(0) = -\infty$, and, in addition, proper convex when f is convex. Furthermore,

$$(f+g)_\infty(y) \geq f_\infty(y) + g_\infty(y) \quad \forall y$$

with equality when f and g are in addition convex.

In addition, if we want to minimize f over $\mathbb{R}^N$, the optimal set is a nonempty compact set if

$$f_\infty(y) > 0 \quad \forall y \neq 0, \tag{2.3}$$

and this condition becomes necessary when f is convex.

Finally, since H_∞ is convex and isotonic and satisfies (A_0), (iv), let us remark that by Theorem 8.6 of [15] we have for each $a \in \mathbb{R}^m$:

$$\lim_{\lambda\to+\infty} H_\infty(a_1, \ldots, a_{i-1}, a_i + \lambda, a_{i+1}, \ldots, a_m) = +\infty. \tag{2.4}$$

2.2. Convergence results.

We now want to solve the problem (P) defined in the introduction, and we suppose for the remainder of this paper that the assumptions (A_0) and (A_1) given in the introduction hold.

LEMMA 2.1. *Suppose that (A_0) and (A_1) hold. Then, for $r > 0$, the functions ϕ and ϕ_r are closed and proper. Furthermore, if $(0,0) \in \text{dom}H \times \text{dom}L$, then for each $x \in \text{dom}\phi$, x belongs to $\text{dom}\phi_r$.*

Proof. (i) Let us prove that ϕ is closed; with the same arguments we can prove that ϕ_r is also closed. Let x be arbitrary and let $\{x_n\}$ be a sequence converging to x. We have to prove that $\lim_{n\to\infty}\inf\, \phi(x_n) \geq \phi(x)$. As a consequence we have only to consider the case where $x_n \in \text{dom}\ \phi$. Let $\epsilon > 0$; since the functions f_i and L_∞ are closed, then for n sufficiently large we have

$$f_i(x_n) \geq f_i(x) - \epsilon \quad \forall i = 0, 1 \ldots m \quad \text{and} \quad L_\infty(Ax_n - b) \geq L_\infty(Ax - b) - \epsilon.$$

Now, since H_∞ is isotonic, it follows that for n sufficiently large,

$$\phi(x_n) \geq f_0(x) + H_\infty(f_1(x) - \epsilon, \ldots, f_m(x) - \epsilon) + L_\infty(Ax - b) - 2\epsilon.$$

Passing to the limit when $n \to \infty$ we obtain

$$\liminf_{n\to\infty} \phi(x_n) \geq f_0(x) + H_\infty(f_1(x) - \epsilon, \ldots, f_m(x) - \epsilon) + L_\infty(Ax - b) - 2\epsilon.$$

Now let $\epsilon \to 0^+$; since H_∞ is closed we get

$$\phi(x) \leq \liminf_{n\to\infty}\ \phi(x_n),$$

and it follows that the ϕ is closed.

(ii) Since $f_0, L_\infty, H_\infty, L, H$ are proper, it follows that ϕ and ϕ_r never assume the value $-\infty$. From (A_1) (iii), $\phi(x_0)$ is finite. If (0,0) $\in$ domH $\times$ domL it follows from (2.2) that $\phi_r(x_0)$ is also finite. In the other case, since ridomH_∞ and ridomL_∞ are cones, it follows that $\frac{F(x_0)}{r} \in$ ridomH_∞ and $\frac{Ax_0-b}{r} \in$ ridomL_∞, and then from (1.7), it follows that $\phi_r(x_0)$ is finite so that ϕ and ϕ_r are proper.

(iii) Now suppose that (0,0) $\in$ domH $\times$ domL and let $x \in$ domϕ. Then $F(x) \in$ dom H_∞, $Ax - b \in$ domL_∞, $f_0(x)$ is finite, and from (2.2) it follows that $\phi_r(x)$ is finite. □

PROPOSITION 2.2. *Write* $F_\infty(d) = ((f_1)_\infty(d), \dots, (f_m)_\infty(d))$ *and*

$$\widetilde{\phi}_\infty(d) = (f_0)_\infty(d) + H_\infty(F_\infty(d)) + L_\infty(Ad) \text{ if } d \in \bigcap_{i=1}^{m} \text{dom}(f_i)_\infty, +\infty \text{ otherwise.}$$

Then, if (A_0) *and* (A_1) *hold, we have*

$$\phi_\infty(d) \geq \widetilde{\phi}_\infty(d) \quad \forall d \tag{2.5}$$

with equality if in addition the functions f_i *are convex.*

Proof. Let $a_i < (f_i)_\infty(d)$ for $i = 0, 1, \dots, m$ and $d_n \to d, t_n \to +\infty$ with

$$\phi_\infty(d) = \liminf_{n\to+\infty} \frac{\phi(t_n d_n)}{t_n}.$$

Then, using formula (2.1), for n sufficiently large we have

$$\forall i \; f_i(t_n d_n) \geq a_i t_n \Leftrightarrow F(t_n d_n) \geq a t_n \quad \text{with } a = (a_1, a_2, \dots, a_m). \tag{2.6}$$

Furthermore, since recession functions are positively homogeneous, it follows that

$$\frac{\phi(t_n d_n)}{t_n} = \frac{f_0(t_n d_n)}{t_n} + H_\infty\left(\frac{F(t_n d_n)}{t_n}\right) + L_\infty\left(Ad_n - \frac{b}{t_n}\right).$$

Since H_∞ is isotonic we get from (2.6) that

$$\frac{\phi(t_n d_n)}{t_n} \geq a_0 + H_\infty(a) + L_\infty\left(Ad_n - \frac{b}{t_n}\right).$$

Now let $a_i \to (f_i)_\infty(d)$. Then, since H_∞ and L_∞ are closed, using formula (2.1) and passing to the limit in the above formula, we obtain

$$\phi_\infty(d) \geq (f_0)_\infty(d) + H_\infty(F_\infty(d)) + L_\infty(Ad) \quad \text{if } d \in \bigcap_{i=1}^{m} \text{dom}(f_i)_\infty,$$

and (2.5) holds for such a direction. Otherwise, the result remains valid by using formula (2.4).

Now suppose that the functions f_i, $i = 0, 1, \dots, m$, are convex; since H_∞ and L_∞ are sublinear and since H_∞ is isotonic it follows that

$$\frac{\phi(x + \lambda d) - \phi(x)}{\lambda} \leq (f_0)_\infty(d) + H_\infty(F_\infty(d)) + L_\infty(Ad) \quad \text{if } d \in \bigcap_{i=1}^{m} \text{dom}(f_i)_\infty.$$

Passing to the limit when $\lambda \to +\infty$ we get $\phi_\infty(d) \le \widetilde{\phi}_\infty(d)$. □

We denote by S (resp., S_r) the optimal set of (P) (resp., (P_r)) and suppose for *the remainder* of this paper that

$$\widetilde{\phi}_\infty(d) > 0 \quad \forall d \neq 0. \tag{2.7}$$

Then from formula (2.5) it follows that S is nonempty and compact. Conversely, when the functions f_i are convex, (2.7) holds when S is nonempty and compact.

We now define two kinds of assumptions that complement condition (A_1)(iii), both ensuring convergence:

(H_1) (0,0) ∈ domH × domL.

(H_2) There exists x_0 such that

$$\text{(i)} \quad x_0 \in \mathrm{dom} f_0, \quad F(x_0) \in \mathrm{ridom} H_\infty, \quad Ax_0 - b \in \mathrm{ridom} L_\infty, \tag{2.8}$$

(ii) for each $x \in \mathrm{dom}\phi$, there exists a sequence $\{u_n\}$ converging to x satisfying (2.8) (i.e., $u_n \in \mathrm{dom} f_0$, $F(u_n) \in \mathrm{ridom} H_\infty$, $Au_n - b \in \mathrm{ridom} L_\infty$) and such that $f_0(u_n) \to f_0(x), F(u_n) \to F(x)$.

Remark 2.1. If we consider problem (P^m), when H is defined by (1.13) with $\theta \in \mathcal{G}$ and $\eta > 0$, then (H_1) holds. This is the case for θ_1, θ_2, θ_3, θ_4. □

Remark 2.2. Assumption (H_2)(i) is a regularity condition, which coincides with assumption (A_1)(iii) when (0,0) ∉ domH × domL. For the constrained classical optimization (P^m), it is exactly Slater's condition. This condition was also used in [3] for the problem $P^{(B,T)}$ in order to obtain duality results. For problem (PDP) it coincides with the strict feasibility condition.

Remark 2.3. Let us now give two important examples for which (H_2)(ii) holds.

(i) Consider the problem (P^m); suppose that the functions f_i, $i = 0, 1 \ldots m$, are not only closed but continuous and that the set $\{x : f_i(x) < 0 \ \forall i = 1 \ldots m\}$ is the interior of $C := \{x : f_i(x) \le 0 \ \forall i = 1 \ldots m\}$. (This last condition holds in particular when the functions f_i are C^1 and the usual Mangasarian–Fromovitz condition holds for each $x \in C$.) Then, obviously, $(H2)$(ii) holds.

(ii) Now suppose that the f_i are convex (not necessarily continuous) and that (H_2)(i) is satisfied with $x_0 \in \bigcap_{i=0}^m$ ridom f_i; then condition (H_2)(ii) will be satisfied if we suppose in addition that

$$y \in \ \mathrm{ridom} H_\infty, \quad y' \le y \Longrightarrow y' \in \ \mathrm{ridom} H_\infty. \tag{2.9}$$

(This last condition holds for all the examples related to problem (P^m) given in the introduction since $\mathrm{ridom} H_\infty = \mathbb{R}^m_{--}$.)

Indeed, let $x \in \mathrm{dom}\phi$; then $Ax - b \in \mathrm{dom} L_\infty$, $F(x) \in \mathrm{dom} H_\infty$.

Set $u_n = \frac{1}{n}x_0 + (1 - \frac{1}{n})x$. Since the functions f_i are convex, $F(u_n) \le \frac{1}{n}F(x_0) + (1 - \frac{1}{n})F(x)$. By assumption (H_2) (i) $F(x_0) \in \mathrm{ridom} H_\infty$. This implies that $[\frac{1}{n}F(x_0) + (1 - \frac{1}{n})F(x)]$ belongs to $\mathrm{ridom} H_\infty$, and then using (2.9), it follows that $F(u_n) \in$ ridom H_∞. Furthermore $(Au_n - b) \in$ ridom L_∞, $u_n \in \bigcap_{i=0}^m$ ridom f_i. Finally, using theorem 7.5 of [15], it follows that $f_0(u_n) \to f_0(x)$, $F(u_n) \to F(x)$, and assumption (H_2)(ii) is satisfied.

THEOREM 2.3. *Suppose that* (A_0), (A_1), *and formula* (2.7) *hold and that* (H_1) *or* (H_2) *is satisfied. Then, for each* $r > 0$, *the optimal set* S_r *is nonempty and compact and every selection* $x_r \in S_r$ *stays bounded with all its limit points in* S. *Furthermore,* $\alpha = \lim_{r \to 0^+} \alpha_r$.

Proof. 1. Let us prove first that S_r is nonempty and compact. Let $a_i < (f_i)_\infty(d)$ for $i = 0, 1, \ldots, m$ and $d_n \to d, d \neq 0$, $t_n \to +\infty$ with

$$(\phi_r)_\infty(d) = \lim_{n\to+\infty} \frac{\phi_r(t_n d_n)}{t_n}.$$

Then, for n sufficiently large, we have (2.6) again, and since H is isotonic we deduce that

$$\frac{\phi_r(t_n d_n)}{t_n} \geq a_0 + \frac{r}{t_n}\; H\left(\frac{a t_n}{r}\right) + \frac{r}{t_n}\; L\left(\frac{t_n}{r}\left(Ad_n - \frac{b}{t_n}\right)\right).$$

Then by (2.1), if we take the limit when $t_n \to +\infty$ we get

$$(\phi_r)_\infty(d) \geq a_0 + H_\infty(a) + L_\infty(Ad).$$

Let $a_i \to (f_i)_\infty(d)$; since H_∞ is closed we get

$$(\phi_r)_\infty(d) \geq (f_0)_\infty(d) + H_\infty(F_\infty(d)) + L_\infty(Ad) > 0 \quad \text{if } d \in \bigcap_{i=1}^{m} \operatorname{dom}(f_i)_\infty, +\infty \text{ else,}$$

and then we can deduce that S_r is a nonempty compact set.

2. $\{x_r\}_{r>0}$ is bounded. Consider an optimal path $x_r \in S_r$ and let $x_0 \in \operatorname{dom}\phi$, satisfying in addition (2.8) when (H_2) holds. Then $x_0 \in \operatorname{dom}\, \phi_r$ for each $r > 0$ and (2.10)

$$f_0(x_r) + rH\left(\frac{F(x_r)}{r}\right) + rL\left(\frac{A(x_r - b)}{r}\right) \leq f_0(x_0) + rH\left(\frac{F(x_0)}{r}\right) + rL\left(\frac{A(x_0 - b)}{r}\right).$$

Suppose that $\{x_r\}$ is not bounded and choose $r_k \to 0^+$ with

$$\lim_{k\to+\infty} \|x_{r_k}\| = +\infty, \qquad \lim_{k\to+\infty} \frac{x_{r_k}}{\|x_{r_k}\|} = d \neq 0.$$

Let $a_i < (f_i)_\infty(d)$ and take k_0 such that $f_i(x_{r_k}) \geq a_i \|x_{r_k}\|$ for all $k \geq k_0$ and $i = 0, 1, \ldots, m$. Since H is isotonic we deduce from (2.10) that

$$\begin{aligned} a_0 + \frac{r_k}{\|x_{r_k}\|}\; H\left(a_i \frac{\|x_{r_k}\|}{r_k}\right) + \frac{r_k}{\|x_{r_k}\|}\; L\left(\frac{\|x_{r_k}\|}{r_k}\left(\frac{Ax_{r_k}}{\|x_{r_k}\|} - \frac{b}{\|x_{r_k}\|}\right)\right) \\ \leq \frac{1}{\|x_{r_k}\|}\left[f_0(x_0) + r_k\; H\left(\frac{F(x_0)}{r_k}\right) + r_k\; L\left(\frac{Ax_0 - b}{r_k}\right)\right]. \end{aligned}$$

Then, passing to the limit, it follows from (2.1) and (1.6) that

$$a_0 + H_\infty(a) + L_\infty(Ad) \leq 0.$$

Let $a_i \to (f_i)_\infty(d)$; then, by formula (2.4), $d \in \bigcap_{i=1}^m \operatorname{dom}(f_i)_\infty$, and since H_∞ is closed we get

$$(f_0)_\infty(d) + H_\infty(F_\infty(d)) + L_\infty(Ad) \leq 0, \quad d \neq 0,$$

which contradicts (2.7).

3. Accumulation points. Let $\overline{x} = \lim_{k\to+\infty} x_{r_k}$ be an accumulation point of the sequence $\{x_r\}_{r>0}$ and take $a_i < f_i(\overline{x})$ for $i = 0, 1, \ldots, m$. Let $x_0 \in \operatorname{dom}\phi$ if (H_1)

holds, and let x_0 satisfy (2.8) otherwise. Since the functions f_i are closed, we have $f_i(x_{r_k}) > a_i$ for k large, and since H is isotonic we deduce from (2.10) that

$$a_0 + r_k\ H\left(\frac{a}{r_k}\right) + r_k\ L\left(\frac{Ax_{r_k} - b}{r_k}\right) \le f_0(x_0) + r_k\ H\left(\frac{F(x_0)}{r_k}\right) + r_k\ L\left(\frac{Ax_0 - b}{r_k}\right).$$

Passing to the limit it follows that

$$a_0 + H_\infty(a) + L_\infty(A\overline{x} - b) \le f_0(x_0) + H_\infty(F(x_0)) + L_\infty(Ax_0 - b),$$

and then, when $a_i \to f_i(\overline{x})$, we deduce that

$$\phi(\overline{x}) \le \phi(x_0), \tag{2.11}$$

and the theorem is proved when (H_1) holds. If (H_2) holds, then for each $x \in \mathrm{dom}\phi$, there exists a sequence $\{u_n\}$ converging to x that satisfies (2.8) and such that $f_0(u_n) \to f_0(x), F(u_n) \to F(x)$. From (2.11) it follows that $\phi(\overline{x}) \le \phi(u_n)$, and then, since (1.8) holds, passing to the limit we obtain that $\phi(\overline{x}) \le \phi(x)$ so that $\overline{x} \in S$.

4. Optimal value limit. Taking the same arguments as above, we deduce that

$$\phi(\overline{x}) \le \liminf_{r\to\infty} \phi_r(x_r) \le \limsup_{r\to\infty} \phi_r(x_r) \le \phi(x^*) \quad \forall x^* \in S.$$

As a result we have $\alpha = \lim_{r\to 0^+} \alpha_r$. □

3. The convex case: Duality results. In this section we suppose that (A_0), (A_1), and formula (2.7) hold and in addition that the functions f_i, $i = 0, 1, \dots, m$, are convex. As a consequence, ϕ and ϕ_r are closed, proper, convex functions. We then associate a dual problem to (P) by considering the perturbation function

$$\psi(x, y, z) = f_0(x) + H_\infty(F(x) + y) + L_\infty(Ax - b + z) \text{ if } x \in E, \text{ and} + \infty \text{ otherwise},$$

with

$$E := \bigcap_{i=0}^{m} \mathrm{dom} f_i.$$

Then it can be seen easily that ψ is convex, proper, and closed, and thanks to duality theory [11, Chapter 7], the dual problem via this perturbation function is

$$(D) \qquad \beta = \inf\{\psi^*(0, \lambda, \mu) \mid \lambda \in \mathbb{R}^m, \mu \in Y\},$$

where ψ^* denotes the Fenchel conjugate of ψ, which may be computed as

$$\psi^*(0, \lambda, \mu) = \sup_{x\in E} \left\{ - f_0(x) + \sup_y \left[(\lambda, F(x) + y) - (\lambda, F(x)) - H_\infty(F(x) + y)\right] \right.$$
$$\left. + \sup_z \left[(\mu, Ax - b + z) - (\mu, Ax - b) - L_\infty(Ax - b + z)\right] \right\}.$$

Since $H_\infty(\cdot) = \delta^*(\cdot \mid \overline{\mathrm{dom} H^*})$ it follows that

$$\psi^*(0, \lambda, \mu) = \sup_{x\in E} \{-f_0(x) - (\lambda, F(x)) - (\mu, Ax - b) + \delta(\lambda \mid \overline{\mathrm{dom} H^*}) + \delta(\mu \mid \overline{\mathrm{dom} L^*})\},$$

and then we have

$$(3.1)\qquad (D)\quad \beta=\inf\{p(\lambda,\mu):(\lambda,\mu)\in\mathbb{R}^m\times Y\}$$

with

$$p(\lambda,\mu)=\begin{cases}-\inf\limits_{x\in E}\{f_0(x)+(\lambda,F(x))+(\mu,Ax-b)\} & \text{for } (\lambda,\mu)\in\overline{\mathrm{dom}H^*}\times\overline{\mathrm{dom}L^*},\\ +\infty \quad \text{otherwise.}\end{cases}$$

Let us remark that since S is nonempty and compact, duality theory ensures that $\alpha+\beta=0$.

Similarly, we associate a dual problem with (P_r) by considering the perturbation function

$$\psi_r(x,y,z)=f_0(x)+r\left[H\left(\frac{F(x)+y}{r}\right)+L\left(\frac{Ax+y}{r}\right)\right] \text{ if } x\in E, \text{ and} +\infty \text{ otherwise.}$$

Then it can be seen easily that ψ_r is convex, closed, and proper, and the dual problem is

$$(3.2)\qquad (D_r)\quad \beta_r=\inf\{\psi_r^*(0,\lambda,\mu)\mid(\lambda,\mu)\in\mathbb{R}^m\times Y\},$$

where

$$\begin{aligned}\psi_r^*\big(0,\lambda,\mu\big)=\sup_{x\in E}\Bigg\{&\Big[-f_0(x)-\big(\lambda,F(x)\big)-\big(\mu,Ax-b\big)\Big]\\ &+r\sup_y\left[\left(\lambda,\frac{F(x)+y}{r}\right)-H\left(\frac{F(x)+y}{r}\right)\right]\\ &+r\sup_z\left[\left(\mu,\frac{Ax-b+z}{r}\right)-L\left(\frac{Ax-b+z}{r}\right)\right]\Bigg\}\\ =-\inf_{x\in E}&\Big[f_0(x)+(\lambda,F(x))+(\mu,Ax-b)\Big]+r\Big[H^*(\lambda)+L^*(\mu)\Big].\end{aligned}$$

As a consequence, the infimum in (3.2) can be taken over $\overline{\mathrm{dom}H^*}\times\overline{\mathrm{dom}L^*}$, and then (D_r) can be written as

$$(D_r)\qquad \beta_r=\inf\{t^r(\lambda,\mu)\mid(\lambda,\mu)\in\mathbb{R}^m\times Y\}$$

with

$$(3.3)\qquad t^r(\lambda,\mu)=p(\lambda,\mu)+r(H+L)^*(\lambda,\mu).$$

Again, if S_r is nonempty and compact, duality theory ensures that $\alpha_r+\beta_r=0$, and if (H_1) or (H_2) holds, as a consequence of Theorem 2.3 we get

$$\beta=\lim_{r\to0^+}\beta_r.$$

Let T (resp., T_r) be the optimal set of solutions of (D) (resp., D_r). Without additional assumptions we cannot ensure that T and T_r are nonempty, and we suppose that

(H_3) $\mathrm{intdom}H_\infty\times$ $\mathrm{intdom}L_\infty$ is nonempty,

(H_4) $0 \in \overline{\mathrm{domH}} \times \overline{\mathrm{dom}L}$.

Remark 3.1. (H_3) and (H_4) are satisfied in all the examples given in this paper.

PROPOSITION 3.1. *Suppose that (A_0), (A_1), formula (2.7), (H_2)(i), (H_3), and (H_4) hold and that the functions f_i are convex. Suppose also that (H_1) or (H_2)(ii) is satisfied. Then for $r > 0$, T and T_r are nonempty compact sets and every selection $(\lambda_r, \mu_r) \in T_r$ stays bounded when $r \to 0^+$ with all its limit points in T.*

Proof. 1. Let x_0 satisfy (2.8). Since $\mathrm{ridom}H_\infty\times$ $\mathrm{ridom}L_\infty = \mathrm{intdom}H_\infty\times$ $\mathrm{intdom}L_\infty$, it follows that the function $(y, z) \to \psi(x_0, y, z)$ is continuous at $(0, 0)$, and then from Theorem 7.6.1 of [11], this implies that T is a nonempty compact set. This is equivalent to saying that

$$p_\infty(\lambda, \mu) > 0 \quad \forall(\lambda, \mu) \neq 0. \tag{3.4}$$

Furthermore, since

$$t^r_\infty(\lambda, \mu) = p_\infty(\lambda, \mu) + r((H + L)^*)_\infty(\lambda, \mu), \tag{3.5}$$

$$((H + L)^*)_\infty = \delta^*(\cdot \mid \overline{\mathrm{dom}H} \times \overline{\mathrm{dom}L}), \tag{3.6}$$

using (H_4) we deduce that

$$t^r_\infty(\lambda, \mu) \geq p_\infty(\lambda, \mu) > 0 \quad \forall(\lambda, \mu) \neq 0, \tag{3.7}$$

and it follows that T_r is a nonempty compact set.

2. Set $u = (\lambda, \mu)$, $K = (H + L)^*$. Let us prove that the sequence $\{u_r\}_{r>0}$ is bounded when $r \to 0^+$. If $\{u_r\}_{r>0}$ were not bounded, we would find $r_k \to 0^+$ such that

$$\|u_{r_k}\| \to +\infty, \qquad \frac{u_{r_k}}{\|u_{r_k}\|} \to \overline{u} \neq 0.$$

Since $\beta_r \to \beta$, for $\varepsilon > 0$ we have for k sufficiently large

$$\frac{p(u_{r_k})}{\|u_{r_k}\|} + r_k \frac{K(u_{r_k})}{\|u_{r_k}\|} \leq \frac{\beta + \varepsilon}{\|u_{r_k}\|}, \quad \frac{K(u_{r_k})}{\|u_{r_k}\|} \geq K_\infty(\overline{u}) - \varepsilon,$$

and then taking the limit we obtain

$$p_\infty(\overline{u}) \leq 0, \qquad \overline{u} \neq 0,$$

which contradicts (3.4).

3. Let $\varepsilon > 0$. Then, for r sufficiently small, we have

$$p(u_r) + rK(u_r) \leq \beta + \varepsilon.$$

Then, if $\overline{u}$ is a limit point of $\{u_r\}$, since K and p are lower semicontinuous, passing to the limit we deduce that $p(\overline{u}) \leq \beta + \varepsilon$. Then, with $\varepsilon \to 0^+$, it follows that $\overline{u} \in T$. □

If we consider formula (3.3), we see that the method (D_r) can be interpreted as a result of a "viscosity" or "Tikhonov" regularization method. This method has been considerably studied and we shall use Proposition 2.5 of [2] to prove the convergence of the sequence $\{\lambda_r, \mu_r\}$ to a single point.

COROLLARY 3.1. *Suppose that the assumptions of Proposition* 3.1 *are satisfied. Assume also that* argmin $(p) \cap \operatorname{dom}(H+L)^*$ *is nonempty and that* $(H+L)^*$ *is strictly convex on its domain. Then* (D_r) *has a unique optimal solution* (λ_r, μ_r) *which converges when* $r \to 0^+$ *to the unique point* $(\overline{\lambda}, \overline{\mu}) \in \operatorname{argmin}(p)$*, which minimizes on this set the function* $(H+L)^*$.

Proof. In order to use Proposition 2.5 of [2], which gives the result, we have only to prove that

(i) p has bounded level sets,

(ii) $(H+L)^*_\infty$ is nonnegative.

But (i) is a consequence of assumptions (H_2)(i) and (H_3), and (ii) follows from assumption (H_4). □

4. Applications for semidefinite programming. Let Y be the space of (p,p) real symmetric matrices endowed with the inner product $(U,V) = \operatorname{tr} UV$ ($\operatorname{tr} U$ denotes the trace of U), and let K be the subset of negative semidefinite matrices. Recall that the inequality sign in $X \leq 0$ for a matrix X means that X is semidefinite negative. Consider now the following (PDP) optimization problem:

$$(PDP) \qquad \alpha = \inf c^t x \quad \text{subject to } B(x) \leq 0$$

with $B(x) = (B_0 + \sum_{i=1}^m x_i B_i)$. The problem data are the vector $c \in \mathbb{R}^m$ and the $(m+1)$ symmetric matrices $B_0, B_1, \dots, B_m$.

In this section we consider the set $\mathcal{F}$ of functions $f : \mathbb{R}^P \to \mathbb{R} \cup \{+\infty\}$ which are closed, convex, proper, and symmetric, i.e., $f(\lambda) = f(P\lambda)$ for any permutation matrix P. For each $X \in Y$ we denote by $\lambda(X) = (\lambda_1(X), \lambda_2(X), \dots, \lambda_p(X))$ the vector of eigenvalues of X in nondecreasing order. Then each function $f \in \mathcal{F}$ induces a matrix function $f_Y : Y \to \mathbb{R} \cup \{+\infty\}$ defined by

$$f_Y(X) = f(\lambda(X)) \quad \forall X \in Y. \tag{4.1}$$

It has been proved by Lewis [12] that f_Y is a closed, proper, convex function defined on Y.

Furthermore, it has been proved recently by Seeger [17] that

$$(f_Y)_\infty = (f_\infty)_Y. \tag{4.2}$$

Now we remark that problem (PDP) can be written as follows:

$$(PDP) \qquad \alpha = \inf c^t x + \delta(Ax - b \mid K)$$

with $B(x) = (Ax - b)$, where $Ax := \sum_{i=1}^m x_i B_i$ and $b := -B_0$.

Then let $\ell(y) = \sum_{i=1}^m \theta(y_i)$ and suppose that $\theta : \mathbb{R} \to \mathbb{R} \cup \{+\infty\}$ is a function which belongs the class $\mathcal{G}$ defined in the introduction. We have seen that

$$\ell_\infty(y) = \delta(y \mid \mathbb{R}^P_-).$$

Furthermore, $\delta(\cdot \mid \mathbb{R}^P_-) \in \mathcal{F}$ and $\delta(X \mid K) = \delta(\lambda(X) \mid \mathbb{R}^P_-)$. Then, since $l \in \mathcal{F}$, from formula (4.2) it follows that

$$\delta(\cdot \mid K) = L_\infty \quad \text{with } L = \ell_Y.$$

Moreover, $\operatorname{intdom} L_\infty = \operatorname{int} K \subset \operatorname{intdom} L$, and assumption (A_0) is satisfied. Now we suppose strict feasibility for (PDP); i.e., there exists x_0 such that

$$Ax_0 - b < 0 \quad (\Leftrightarrow Ax_0 - b \in \operatorname{int} K). \tag{4.3}$$

Then assumptions (A_1) and (H_2) are satisfied. Now let us approximate (PDP) by

$$(PDP)_r \qquad \alpha_r = \inf_x \; c^t x + r\; L\left(\frac{Ax-b}{r}\right).$$

If we suppose in addition that the optimal set of (PDP) is nonempty and compact, then Theorem 2.3 holds.

The following examples are of particular interest:

$\theta_1 \to L_1(D) = \text{tr}(\exp D)$,
$\theta_2 \to L_2(D) = -\log(\det\,(I-D))$ for $D < I$, $+\infty$ otherwise,
$\theta_4 \to L_4(D) = \text{tr}((I-D)^{-1}D)$ for $D < I$, $+\infty$ otherwise,
$\theta_5 \to L_5(D) = -\log(\det(-D))$ for $D < 0$, $+\infty$ otherwise,
$\theta_6 \to L_6(D) = \text{tr}(-D^{-1})$ for $D < 0$, $+\infty$ otherwise.

We note that for these examples L is C^∞ on the interior of its domain and the gradient and the Hessian can be easily obtained (see, for example, [19]).

Let us now consider the dual (D) of (PDP). Since $\overline{\text{dom} L^*} = -K$ it follows from formula (3.1) in a straightforward way that (D) can be written as

$$(D) \qquad \beta = \inf\, -\,\text{tr}B_0 Z$$
$$\text{subject to} \quad -\text{tr}B_i Z = c_i, \quad i = 1, 2, \ldots, m,$$
$$Z \geq 0.$$

In the same way it follows from formula (3.3) that the dual problem (D_r) associated with $(PDP)_r$ is

$$(D_r) \qquad \beta_r = \inf -\,\text{tr}B_0 Z + rL^*(Z)$$

$$\text{subject to} \quad -\text{tr}B_i Z = c_i, \quad i = 1, 2, \ldots, m.$$

Since (H_3) and (H_4) are satisfied, Proposition 3.1 holds, and for $r > 0$ the optimal set T and T_r of (D) and (D_r) are nonempty compact sets and every selection $Z_r \in T_r$ stays bounded when $r \to 0^+$ with all its limit points in T.

Let us end this section by giving two examples of such a situation. It has been proved by Lewis [12] that for each $f \in \mathcal{F}$ its conjugate $f^* \in \mathcal{F}$ and

$$(f_Y)^* = (f^*)_Y.$$

Then, using this formula, it follows that

$$L_5^*(D) = -n - \;\text{logdet} D \quad \text{for } D > 0, +\infty \text{ otherwise},$$

$$L_6^*(D) = -2\;\text{tr}(D^{\frac{1}{2}}) \qquad \text{for } D \geq 0, +\infty \text{ otherwise}.$$

5. Other related approximations.

Let us first remark that, thanks to formula (2.1), (1.6) is also satisfied for $y \notin \text{dom} H_\infty$ and $z \notin \text{dom} L_\infty$, so that $\phi_r(x) \to \phi(x)$ when $r \to 0^+$ for each x such that $F(x) \notin \text{br}\;(\text{dom} H_\infty)$ and $Ax - b \notin \text{br}\;(\text{dom} L_\infty)$. (Here br denotes the relative boundary.) This explains the success of the approximation. Then we can also note that assumption (A_0)(i) can be satisfied by functions other than the basic functions H and L which generate H_∞ and L_∞. More precisely, for the remainder of this section we suppose that $\widetilde{H}$ and $\widetilde{L}$ are convex functions *finite* everywhere and that

$$(5.1) \qquad \widetilde{H}_\infty = H_\infty \text{ on } \;\text{dom} H_\infty, \quad \widetilde{L}_\infty = L_\infty \text{ on } \;\text{dom} L_\infty.$$

Then we have, for $y \in \mathrm{dom} H_\infty$, $z \in \mathrm{dom} L_\infty$,

$$\lim_{r \to 0^+} r\widetilde{H}\left(\frac{y}{r}\right) = H_\infty(y), \quad \lim_{r \to 0^+} r\widetilde{L}\left(\frac{z}{r}\right) = L_\infty(z). \tag{5.2}$$

Unfortunately, this will not hold in general for $y \notin \mathrm{dom} H_\infty$ and $z \notin \mathrm{dom} L_\infty$. We shall suppose that

$$0 < \widetilde{H}_\infty(y) < +\infty \quad \text{for } y \notin \mathrm{dom} H_\infty, \quad 0 < \widetilde{L}_\infty(z) < +\infty \quad \text{for } z \notin \mathrm{dom} L_\infty. \tag{5.3}$$

Then let $\alpha : \mathbb{R}_+ \to \mathbb{R}$ be a function such that

$$\alpha(r) > 0 \ \forall r > 0, \quad \lim_{r \to 0^+} \alpha(r) = 0, \quad \liminf_{r \to 0^+} \frac{\alpha(r)}{r} = +\infty. \tag{5.4}$$

From (5.3) and (5.4) it follows that for each $y \notin \mathrm{dom} H_\infty$ and $z \notin \mathrm{dom} L_\infty$ we have

$$\lim_{r \to 0^+} \alpha(r)\widetilde{H}\left(\frac{y}{r}\right) = H_\infty(y), \quad \lim_{r \to 0^+} \alpha(r)\widetilde{L}\left(\frac{z}{r}\right) = L_\infty(z). \tag{5.5}$$

The same can happen on $\mathrm{dom} H_\infty \times \mathrm{dom} L_\infty$ if H_∞ and L_∞ are equal to zero on their domains, and then it is natural to replace ϕ_r in formula (1.12) by the following definition:
(5.6)

$$\phi_r(x) = f_0(x) + \alpha(r)\left[\widetilde{H}\left(\frac{F(x)}{r}\right) + \widetilde{L}\left(\frac{Ax-b}{r}\right)\right] \text{ if } x \in \bigcap_{i=1}^{m} \mathrm{dom}\, f_i \text{ and } +\infty \text{ otherwise.}$$

This idea has its origin in [2]. Indeed, if we consider problem (P^m), we have to set $L = \tilde{L} = 0$ and $H_\infty(.) = \delta(.|\mathbb{R}^m_-)$. Denote by $\mathcal{G}^*$ the class of functions $\theta : \mathbb{R} \to \mathbb{R}_+$, convex, strictly increasing such that

$$\lim_{u \to -\infty} \theta(u) = 0, \quad 0 < \theta_\infty(1) < +\infty,$$

and for $\theta \in \mathcal{G}^*$ define

$$\tilde{H}(y) = \sum_{i=1}^{m} \theta(y_i). \tag{5.7}$$

From the properties of θ it follows that $\tilde{H}$ is finite and convex and satisfies relations (5.1) and (5.3). Furthermore, $\tilde{H}$ is positive and isotonic; such properties will be needed in what follows.

Then, when minimizing ϕ_r on $\mathbb{R}^N$, ϕ_r being defined by (5.6) with the conditions (5.4), we recognize in this case the algorithm, proposed and analyzed by Auslender, Cominetti, and Haddou [2]. In [8], Chen and Mangasarian provided a systematic way to generate functions $\theta \in \mathcal{G}^*$. Particular cases of interest are

$$\theta_8(u) = \log(1 + e^u), \quad \theta_9(u) = \frac{u + \sqrt{u^2 + 4}}{2}.$$

But the interest of this framework is that it can be used for other purposes. Indeed, let $\alpha > 0, \beta > 0$, and U be a closed, convex, pointed cone in Y. For y, z in Y, such a cone induces an order relation $y \leq z$ if $y - z \in U$, that is, a partial order.

To simplify, we shall consider here the particular case where

(5.8) $H_\infty = \alpha\delta(\cdot \mid \mathbb{R}^m_-),\ L_\infty = \beta\delta(\ \mid U)$ with $\alpha = +1$ or 0, $\beta = +1$ or $0, \alpha + \beta \neq 0$.

If we set

$$C = \{x : \alpha F(x) \in \mathbb{R}^m_-,\ \beta(Ax - b) \leq 0\},$$

then obviously our optimization problem consists of minimizing f_0 on C.

We shall suppose also that (A_1) holds, that the functions f_i, $i = 0, 1 \ldots m$, are convex, and that $\tilde{H}$ and $\tilde{L}$ are positive and isotonic (i.e., $\tilde{L}$ is isotonic if $y \leq z \Longrightarrow \tilde{L}$ $(y) \leq \tilde{L}(z)$). From these assumptions it follows obviously that ϕ and ϕ_r are closed, proper, convex functions. We suppose in addition that the optimal set of (P) is nonempty and compact. Under the above assumptions, since

$$C_\infty = \{x : \alpha F_\infty(x) \leq 0,\ \beta Ax \leq 0\}, \tag{5.9}$$

this is equivalent to saying that

$$x \neq 0, \quad \alpha F_\infty(x) \leq 0,\ \beta Ax \leq 0 \Rightarrow (f_0)_\infty(x) > 0. \tag{5.10}$$

This formulation can be applied to the (PDP) problem $(\alpha = 0, \beta = 1)$ with $U = K$.

In this case, as a consequence of formula (4.2), we can associate to $\widetilde{L}$ with θ_8 and θ_9 the functions,

$$\widetilde{L}_8(D) = \log(\det(I + \exp D)), \quad \widetilde{L}_9(D) = \operatorname{tr}\left(\frac{D + \sqrt{D^2 + 4I}}{2}\right).$$

From the properties of the class $\mathcal{G}^*$, it can be seen easily that $\tilde{L}_8$, $\tilde{L}_9$ are finite, convex functions, positive and isotonic, satisfying relations (5.1) and (5.3). These examples illustrate the importance of the notion of recession function for establishing new algorithms in optimization theory.

Theorem 5.1. *Let us consider problem (P) with H_∞ and L_∞ defined by* (5.8), *where in this formula U is a closed, convex, pointed cone in Y, and suppose that in problem (P_r), ϕ_r is defined by* (5.6), *with $\alpha(.)$ satisfying* (5.4). *Suppose that (A_1) holds, that the functions f_i, $i = 0, 1 \ldots m$, are convex, and that $\tilde{H}$ and $\tilde{L}$ are convex, finite functions, positive and isotonic, satisfying relations* (5.1) *and* (5.3). *Suppose also that the optimal set S of (P) is nonempty and compact. Then for r sufficiently small, the optimal set S_r is nonempty and compact. Furthermore, each sequence $\{x_r\}_{r \to 0^+}$ with $x_r \in S_r$ is bounded, and all its limit points belong to the optimal set of (P).*

Proof. For the sake of simplicity we suppose that $\alpha = \beta = 1$.

1. Let us prove first that S_r is nonempty and compact for r sufficiently small. Let $a_i < (f_i)_\infty(d)$ for $i = 0, 1, \ldots, m$ and $d_n \to d, t_n \to +\infty$ with $(\phi_r)_\infty(d) = \lim_{n\to\infty} \phi_r(d_n t_n)/t_n$. Then for n sufficiently large, we again have (2.6), and since $\widetilde{H}$ is isotonic, we deduce that

$$\frac{\phi_r(t_n d_n)}{t_n} \geq a_0 + \frac{\alpha(r)}{r}\ \left[\frac{r}{t_n}\widetilde{H}\left(\frac{a t_n}{r}\right) + \frac{r}{t_n}\widetilde{L}\left(\frac{t_n}{r}\left(Ad_n - \frac{b}{t_n}\right)\right)\right].$$

Then by (2.1), if we take the limit when $t_n \to +\infty$, we get

$$(\phi_r)_\infty(d) \geq a_0 + \left[\widetilde{H}_\infty(a) + \widetilde{L}_\infty(Ad)\right]\ \frac{\alpha(r)}{r}.$$

Let $a_i \to (f_i)_\infty(d)$ by formula (2.4), and since $\widetilde{H}_\infty$ is closed it follows that

$$(\phi_r)_\infty(d) = +\infty \quad \text{if } d \notin \bigcap_{i=1}^{m} (\text{dom} f_i)_\infty$$

and that

$$(\phi_r)_\infty(d) \geq (f_0)_\infty(d) + \left[\widetilde{H}_\infty(F_\infty(d)) + \widetilde{L}_\infty(Ad)\right] \frac{\alpha(r)}{r} \quad \text{if } d \in \bigcap_{i=1}^{m} (\text{dom} f_i)_\infty.$$

Suppose now that it is not true that for r sufficiently small S_r is a nonempty compact set. Then, since ϕ_r is convex, there exists a sequence (r_n, d_n) with

$$r_n \to 0^+, \quad d_n \to d, \quad \|d_n\| = 1, \quad (\phi_{r_n})_\infty(d_n) \leq 0,$$

and it follows that

$$(f_0)_\infty(d_n) + \left[\widetilde{H}_\infty(F_\infty(d_n) + \widetilde{L}_\infty(Ad_n)\right] \frac{\alpha(r_n)}{r_n} \leq 0. \tag{5.11}$$

Then it can be seen easily that the sequence $\{F_\infty(d_n)\}$ is bounded, and without loss of generality we can suppose that it converges.

Since $\widetilde{H}_\infty$, $(f_i)_\infty$, $\widetilde{L}_\infty$ are closed and since $\widetilde{H}_\infty$ is isotonic we deduce that

$$\widetilde{H}_\infty(F_\infty(d)) + \widetilde{L}_\infty(Ad) \leq \liminf_{n\to\infty} \left[\widetilde{H}_\infty(F_\infty(d_n)) + \widetilde{L}_\infty(Ad_n)\right]. \tag{5.12}$$

Then, since $(f_0)_\infty$ is closed and proper, it follows by (5.3) and (5.4) that $F_\infty(d) \leq 0, Ad \leq 0$. Without loss of generality we can suppose that $\frac{\alpha(r_n)}{r_n} \geq 1$ for n sufficiently large, and then from (5.11) and (5.1) it follows that

$$\phi_\infty(d) \leq 0, \quad d \neq 0,$$

which contradicts relation (2.7).

2. Let us prove now that each sequence $x_n \in S_{r_n}$ with $r_n \to 0^+$ is bounded. In the opposite case there would exist a sequence $x_n \in S_{r_n}$ such that

$$\frac{x_n}{\|x_n\|} \to \overline{x}, \quad \|x_n\| \to +\infty \quad \text{if } n \to +\infty.$$

(i) By definition, we have

$$\begin{aligned} & f_0(x_n) + \alpha(r_n) \left[\widetilde{H}\left(\frac{F(x_n)}{r_n}\right) + \widetilde{L}\left(\frac{Ax_n - b)}{r_n}\right)\right] \\ & \qquad \leq f_0(x) + \alpha(r_n) \left[\widetilde{H}\left(\frac{F(x)}{r_n}\right) + \widetilde{L}\left(\frac{Ax - b}{r_n}\right)\right] \quad \forall x \in C. \end{aligned} \tag{5.13}$$

Then, by (5.8), since $\widetilde{H}$ and $\widetilde{L}$ are isotonic, it follows that

(5.14)

$$f_0(x_n) + \alpha(r_n) \left[\widetilde{H}\left(\frac{F(x_n)}{r_n}\right) + \widetilde{L}\left(\frac{Ax_n - b)}{r_n}\right)\right] \leq f_0(x) + \alpha(r_n) \left[\widetilde{H}(0) + \widetilde{L}(0)\right] \forall x \in C.$$

Since $\widetilde{H}$ and $\widetilde{L}$ are positive, we deduce that

$$\frac{f_0(x_n)}{\|x_n\|} \leq \frac{1}{\|x_n\|} \left[f_0(x) + \alpha(r_n)\ (\widetilde{H}(0) \cap \widetilde{L}(0))\right] \quad \forall x \in \text{dom}\phi,$$

and passing to the limit, we get

$$(f_0)_\infty(\overline{x}) \leq 0. \tag{5.15}$$

(ii) Let us prove now that $\overline{x} \in C_\infty$. By (5.9) and (5.10) this will imply a contradiction with (5.15). Suppose the contrary. Then $F_\infty(\overline{x}) \notin \mathbb{R}^m_-$ or $A(\overline{x}) \notin U$ or both. Suppose, for example, that $F_\infty(\overline{x}) \notin \mathbb{R}^m_-$. Then, thanks to formula (2.1), it follows that there exists $y \notin \mathbb{R}^m_-$ such that for n sufficiently large

$$\frac{F(x_n)}{\|x_n\|} \geq y.$$

Since $\widetilde{L} \geq 0$ and $\widetilde{H}$ is isotonic, then by (5.14) we have

$$\frac{f_0(x_n)}{\|x_n\|} + \frac{\alpha(r_n)}{r_n} \; \frac{\widetilde{H}\big(\frac{y\|x_n\|}{r_n}\big)}{\frac{\|x_n\|}{r_n}} \leq \frac{f_0(x) + \alpha(r_n)\; \big[\widetilde{H}(0) + \widetilde{L}(0)\big]}{\|x_n\|} \quad \forall x \in C. \tag{5.16}$$

Furthermore, we have

$$0 < \widetilde{H}_\infty(y) \leq \liminf_{n\to\infty} \frac{\widetilde{H}\big(\frac{y\|x_n\|}{r_n}\big)}{\frac{\|x_n\|}{r_n}} \quad \text{for } y \notin \mathbb{R}^m_-.$$

Then, since for some $\varepsilon > 0$ we have for n sufficiently large that

$$\frac{f_0(x_n)}{\|x_n\|} \geq f_\infty(\overline{x}) - \varepsilon,$$

we get a contradiction with (5.16) and (5.4).

Now suppose that $A(\overline{x}) \notin U$; then, since $\widetilde{H}$ is positive, by (5.14) we have

$$\frac{f_0(x_n)}{\|x_n\|} + \frac{\alpha(r_n)}{r_n} \; \frac{\widetilde{L}\Big(A\Big[\Big(\frac{x_n}{\|x_n\|}\Big) - \frac{b}{\|x_n\|}\Big]\frac{\|x_n\|}{r_n}\Big)}{\frac{\|x_n\|}{r_n}} \leq \frac{f_0(x) + \alpha(r_n)\big[\widetilde{H}(0) + \widetilde{L}(0)\big]}{\|x_n\|}$$
$$\forall x \in C.$$

Since

$$0 < \widetilde{L}_\infty(A\overline{x}) \leq \liminf_{n\to\infty} \; \widetilde{L}\Big(A\Big[\frac{x_n}{\|x_n\|} - \frac{b}{\|x_n\|}\Big] \; \frac{\|x_n\|}{r_n}\Big)\Big/\frac{\|x_n\|}{r_n}$$

for $A(\overline{x}) \notin U$, as above, this gives a contradiction with (5.4).

3. Finally, let us prove that every limit point of a sequence $\{x_n\}$ with $x_n \in S_{r_n}$ and $r_n \to 0^+$ is an optimal solution. Let $\overline{x}$ be such a point; without loss of generality we can suppose that $\overline{x} = \lim_{n\to\infty} x_n$. From (5.13), again (5.14) holds, and since $\widetilde{H}$ and $\widetilde{L}$ are positive it follows that

$$f_0(\overline{x}) \leq f_0(x^*) \quad \text{for } x^* \in S.$$

Then, to prove that $\overline{x} \in C$, we proceed as above in part 2(ii). □

Finally, using the same notation as in section 3, we can associate the dual problem

$$\beta_r = \inf\{t^r(\lambda,\mu) \mid (\lambda,\mu) \mid (\lambda,\mu) \in \mathbb{R}^m \times Y\} \tag{D_r}$$

with

$$t^r(\lambda, \mu) = p(\lambda, \mu) + \alpha(r)\ (\tilde{H} + \tilde{L})^* \left(\frac{(\lambda, \mu) r}{\alpha(r)} \right),$$

and we can easily prove (taking the same kinds of arguments as in the proof of Proposition 2.4 of [2]) that if Slater's condition holds, i.e., if

$$'' \exists x_0 \in \mathrm{dom} f_0 \quad \text{such that } F(x_0) < 0, \quad A(x_0) \in \mathrm{int} U'',$$

then for each $r > 0$ sufficiently small, the optimal set T_r is a nonempty compact set. Furthermore, as in section 3, if $r_n \to 0^+$, $\lambda_n \in T_{r_n}$, then the sequence $\{\lambda_n\}$ is bounded and has its limit points in T.

6. Conclusion. This paper presents a unified approach for barrier and penalty methods in optimization problems that generalizes the results obtained in Auslender, Cominetti, and Haddou [2] and in Ben Tal and Teboulle [3].

Moreover, for standard problems (P^m) described by a finite number of inequalities:

(i) primal convergence is obtained for the nonconvex case in contrast with [2], where the data were supposed to be convex functions;

(ii) the barrier function H is not necessarily separable (compare with [2]), leading to new schemes like those proposed in formula (1.15);

(iii) the results extend and cover those given by Ben Tal and Teboulle [3]. Indeed, in [3], there are no results concerning the existence of approximate solutions of (P_r) and no results concerning the boundedness of the primal and dual paths and the convergence to optimal solutions. Furthermore, the assumptions given in [3] are not satisfied for barrier functions.

In addition, this generalization has been applied to semidefinite programming and it has generated in sections 4 and 5 new methods for solving (PDP), for which no previous convergence proofs were available. This is the case in particular for the functions L_1 (D), L_2 (D), L_4 (D), L_6 (D), $\tilde{L}_8$ (D), $\tilde{L}_9$ (D).

Finally, this generalization can certainly be applied to other similar types of problems, in particular, to conical programming with the Lorentz cone, an area that covers many applications (see, e.g., [14]). Other questions are open, such as the important question of convergence of the whole sequence $\{x_r\}$ to a single point. In [2], concerning (P^m) problems, a systematic study was done for the case of linear programming and sufficient conditions were established ensuring convergence to a single point for most of the methods.

For the special problem of minimizing the function $f = \max\{f_i \mid i = 1, \dots, m\}$, convergence to a single point with the particular exponential approximation was also established in [1] for a wide class of functions called analytical functions, a class that contains affine functionals.

In contrast, for (PDP) problems, no systematic study is available concerning the convergence to a single point for general classes of barrier and penalty methods.

REFERENCES

[1] F. Alvarez, *Métodos Continuos, en Optimización Paramétrica: El Método de Newton y Aplicaciones a la Optimización Estructural*, Technical report, Memoria Universidad de Chile, Santiago, 1996.

[2] A. Auslender, R. Cominetti, and M. Haddou, *Asymptotic analysis for penalty and barrier methods in convex and linear programming*, Math. Oper. Res., 22 (1997), pp. 43–63.

[3] A. Ben Tal and M. Teboulle, *A smoothing technique for nondifferentiable optimization problems*, in Optimization. Fifth French German Conference, Lecture Notes in Math. 1405, Springer-Verlag, New York, 1989, pp. 1–11.

[4] A. Ben Tal, M. Teboulle, and W. H. Hang, *A least squares-based method for a class of nonsmooth minimization problems with applications in plasticity*, Appl. Math. Optim., 24 (1991), pp. 273–288.

[5] A. Ben Tal and M. Zibulevski, *Penalty/barrier multiplier methods for convex programming problems*, SIAM J. Optim., 7 (1997), pp. 347–366.

[6] A. Ben Tal and G. Roth, *A truncated log-barrier algorithm for large scale convex programming and minmax problems: Implementation and computational results*, to appear.

[7] C. W. Caroll, *The created response surface technique for optimizing restrained systems*, Oper. Res., 9 (1961). pp. 169–184.

[8] C. Chen and O. L. Mangasarian, *A class of smoothing functions for nonlinear and mixed complementarity problems*, Comput. Optim. Appl., 5 (1996), pp. 97–138.

[9] A. Fiacco and G. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley, New York, 1968.

[10] K.R. Frisch, *The Logarithmic Potential Method of Convex Programming*, Memorandum, University Institute of Economics, Oslo, 1955.

[11] J.P. Laurent, *Approximation et Optimisation*, Hermann, Paris, 1972.

[12] A.S. Lewis, *Convex analysis on the Hermitian matrices*, SIAM J. Optim., 6 (1996), pp. 164–177.

[13] R. Polyak, *Modified barrier functions: Theory and methods*, Math. Programming, 54 (1992), pp. 177–222.

[14] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[15] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[16] A. Seeger, *Smoothing a polyhedral convex function via cumulant transformation and homogenization*, Ann. Polon. Math., 1997, to appear.

[17] A. Seeger, *Convex analysis of spectrally defined matrix functions*, SIAM J. Optim., 7 (1997), pp. 679–696.

[18] P. Tseng and D.P. Bertsekas, *On the convergence of the exponential multiplier method for convex programming*, Math. Programming, 60 (1993), pp. 1–19.

[19] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1995), pp. 49–95.

SIAM J. Optim.
Vol. 10, No. 1, pp. 231–251

CUT SIZE STATISTICS OF GRAPH BISECTION HEURISTICS[*]

G. R. SCHREIBER[†] AND O. C. MARTIN[‡]

Abstract. We investigate the statistical properties of cut sizes generated by heuristic algorithms which solve the graph bisection problem approximately. On an ensemble of sparse random graphs, we find empirically that the distribution of the cut sizes found by "local" algorithms becomes peaked as the number of vertices in the graphs becomes large. Evidence is given that this distribution tends toward a Gaussian whose mean and variance scales linearly with the number of vertices of the graphs. Given the distribution of cut sizes associated with each heuristic, we provide a ranking procedure that takes into account both the quality of the solutions and the speed of the algorithms. This procedure is demonstrated for a selection of local graph bisection heuristics.

Key words. graph partitioning, heuristics, self-averaging, ranking

AMS subject classifications. 90C27, 82B44, 82B30

PII. S1052623497321523

1. Introduction.

Algorithms for tackling combinatorial optimization problems [27] may be divided into two classes. Exact algorithms, such as exhaustive search, branch-and-bound, or branch-and-cut, form the first class; they determine (exactly) the optimum of the cost function that is to be minimized. However, for *NP*-hard problems, they require large computation resources, and in particular, large computation times. The second class consists of "heuristic" algorithms; these are not guaranteed to find the optimal (lowest cost) solution, nor even a solution very close to the optimum, but in practice they find good approximate solutions very fast. For problems in science, one's main interest is in the optimal solution, so an exact algorithm is required. However, for many engineering applications, the heuristic approach may be preferable. There are several reasons for this: (i) The computational resources are simply insufficient to solve the instances of interest by exact methods. (ii) The cost function one wants to minimize is computationally very demanding, and limited resources force one to use an approximate cost function instead. This is the rule rather than the exception with very complex systems such as VLSI. If the true cost function cannot be used, there is little point in finding the true optimum for the wrong problem. (iii) Heuristic algorithms typically generate numerous "good enough" solutions, thus providing information about the statistical properties of low-cost solutions. This information can in turn be used to generate better heuristics or to find new criteria for guiding the branching in exact algorithms such as branch-and-bound.

For almost any combinatorial optimization problem, it is very easy to devise heuristic algorithms that perform quite well; this is probably why so many such algorithms have been proposed to date. Usually they fall into just a few families, the most popular of which are local search, simulated annealing, tabu search, and evolutionary

[*] Received by the editors May 13, 1997; accepted for publication (in revised form) June 4, 1998; published electronically November 29, 1999.
http://www.siam.org/journals/siopt/10-1/32152.html

[†] Division de Physique Théorique, Institut de Physique Nucléaire, Université Paris-Sud, F-91406 Orsay Cedex, France. Current address: Service de Physique Théorique, Orme des Merisiers, C.E.A.-Saclay, F-91191 Gif-sur-Yvette Cedex, France (georg_schreiber@batch-pc.es). The work of this author was supported by individual EC grant ERBCHBICT941665.

[‡] Division de Physique Théorique, Institut de Physique Nucléaire, Université Paris-Sud, F-91406 Orsay Cedex, France (martino@ipno.in2p3.fr). The work of this author was supported by the Institut Universitaire de France.

computation. Practitioners are frequently confronted with the problem of choosing which method to use. Thus they would like to rank these algorithms and determine which one is best for their "instance" (the set of parameters that completely specify the cost function). A difficulty then arises because most heuristic algorithms are stochastic, so that they can give many different solutions for a single instance. In general, the distributions of solution costs generated by the different heuristics overlap, so that the winning algorithm varies from one trial to another. Furthermore, it is necessary to balance the quality of the solutions found against the time necessary to find them, since in practice heuristics run at very different speeds. The final goal of this paper is to do just this kind of balancing: in section 8 we shall introduce a generally applicable ranking method that is based on the possibility of performing multiple runs from random starts for each algorithm until an allotted amount of computer time is exhausted. Our ranking method then determines whether it is better to have a fast heuristic that gives rather poor solutions or a slower heuristic that can give better solutions.

Establishing a ranking on a *single* instance may be what is needed for a real-world problem, but it is not a useful prediction tool. It is preferable to consider the effectiveness of a heuristic when it is applied to a *family* of instances. Since a detailed knowledge of the distribution of costs is necessary for our ranking procedure, the major part of this paper is an in-depth study of the *statistics* of costs found by several classes of heuristics. The *NP*-hard [9] combinatorial optimization problem chosen for our study is the graph bisection problem, hereafter simply called the graph partitioning (or graph "bisection") problem (GPP). This choice is justified by the wide range of practical applications of the GPP. These include host scheduling [3], memory paging and program segmentation [17], load balancing [21], and numerous aspects of VLSI design such as logic partitioning [12] and placement [6, 19]. Because of these applications, the GPP has been used as a testing ground for many heuristics. For our work, a selection had to be made; in view of the previous studies by Johnson et al. [13], Lang and Rao [20], and Berry and Goldberg [4], we have restricted our study to iterative improvement heuristics based on local search and to simulated annealing. Having made a choice of optimization problem and algorithms, it remains to define the class of instances for the testbeds. Ideally, this family of instances should reflect the structure of the actual instances of interest to the practitioner. Since we do not have a particular application in mind, we shall follow the studies of [13, 20, 4] and consider an ensemble of sparse random graphs.From our numerical study, we have found that all of the heuristics tested share the following properties when the random graphs become large: (i) each algorithm can be characterized by a fixed percentage excess above the optimum cost; (ii) the partitions generated have a *distribution* of costs which becomes peaked, both within a given graph and across all graphs; (iii) these distributions tend toward Gaussians. Because of these properties, our ranking of heuristics on large graphs is largely determined by the mean and variance of the costs found, and thus a constant speed-up factor has only a very small effect on the ranking. We expect this property to hold for most problems and heuristics of practical interest, leading to a very robust ranking.

The paper is organized as follows. In section 2 we define the GPP as well as the ensemble of random graphs used for our testbed. Section 3 derives properties of random partitions and shows that the distribution of cut sizes has a relative width that goes to zero as the instance size grows. In section 4 we argue why this property should hold also for the distribution of costs found by *heuristic* algorithms based on local iterative processes. In section 5 we discuss the heuristic algorithms we have included in our tests. Section 6 gives the mean and standard deviation of the costs found as a function of graph size; the distribution for the costs is indeed found to be peaked. This leads to a first ranking which, however, does not take into account computation times. To implement our speed-dependent ranking, we must determine the *distribution* of cut sizes found by the different algorithms. This is the subject of section 7, where evidence is given that the distribution on any typical graph tends toward a Gaussian in the limit of large graphs. In section 8 we present our ranking method, which takes into account both the quality of the solutions and the speed of the heuristics. In section 9, finally, we discuss the results and conclude.

2. Minimum cuts. The GPP can be defined as follows. Consider a graph $G = (V, E)$ which consists of a set of N vertices $V = \{v_1, v_2, \ldots, v_N\}$ and a set of (nonoriented) edges connecting pairs of vertices. It is convenient to introduce the matrix E_{ij}, called the connectivity matrix, given by

$$E_{ij} = \begin{cases} 1 & \text{if } v_i \text{ is connected to } v_j, \\ 0 & \text{otherwise.} \end{cases}$$

Since the edges are nonoriented, $E_{ij} = E_{ji}$. (Some of what will be discussed applies to weighted graphs; then E_{ij} will represent the weight of the ij edge.) A partition of G is given by dividing the vertices of G into two disjoint subsets V_1 and V_2 such that $V = V_1 \cup V_2$. The number of edges connecting V_1 to V_2 is called the cut of the partition and will be denoted by $\mathcal{C}$. It is given by

$$\mathcal{C}[V_1, V_2] = \sum_{i \in V_1, j \in V_2} E_{ij}. \tag{2.1}$$

The GPP (or “min-cut” problem) consists of finding the partition (V_1, V_2) for which the cost (2.1) is the minimum subject to given constraints on the sizes of V_1 and V_2. The GPP is *NP*-hard [9]. In the standard formulation to which we shall restrict ourselves in this work, V_1 and V_2 have equal sizes.

For our study, it is necessary to fix an ensemble of graphs for the testbed. We have chosen $G(N, p)$ the ensemble of random graphs of N vertices where each edge is present with the probability p. The choice of $G(N, p)$ is justified by its tractable mathematical properties and by the fact that many workers [13, 20, 4] have used graphs in this ensemble to test heuristics. The problem of finding the properties of the minimum cut size when the graphs belong to such an ensemble is sometimes called the *stochastic* GPP. Let us review some of the known results for this problem; this will serve to motivate our conjectures for the behavior of cuts obtained from *heuristics.* For each graph G_i, call $\mathcal{C}_0$ its minimum cut size. Taking G_i from the ensemble $G(N, p)$, $\mathcal{C}_0$ is a random variable. Following derivations now standard in a number of other stochastic combinatorial optimization problems (COPs), it is possible to show, using Azuma’s inequality [1], that the distribution of $\mathcal{C}_0$ becomes peaked as $N \to \infty$. This means that as N becomes large, $(\mathcal{C}_0 - \langle \mathcal{C}_0 \rangle)/\langle \mathcal{C}_0 \rangle$, the relative fluctuations about the mean tend to zero. This property, often referred to as “self-averaging,” is typical of processes to which many terms contribute. For certain stochastic COPs, it is possible to show further that the mean minimum cost satisfies a power scaling law in N, so that $\mathcal{C}_0/N^\gamma$ converges in probability to a limiting value as $N \to \infty$. In the case of the stochastic GPP, there is no proof that such a property holds. Nevertheless, it is believed that such a scaling holds: within the $G(N, p)$ ensemble at p fixed, calculations show that $\mathcal{C}_0/N^2 \to p/4$ with probability 1 as $N \to \infty$ [8]. As will be shown in the next section, this is also the limiting behavior of random cuts, and so the ensemble at p fixed is not a challenging one for heuristics. The reason for this "uninteresting" scaling is the high number of edges connecting to any vertex. Thus we consider in this work the ensemble $G(N,p)$, $p = \alpha/(N-1)$, with α fixed; α is the mean connectivity (number of neighbors of a vertex) of the graphs. These graphs are sparse, in contrast to the dense graphs obtained by taking p to be independent of N. Consider the optimal partition. At a typical vertex in V_1, some finite fraction of its edges will connect to vertices in V_2. With each vertex contributing an $O(1)$ amount to the cut size, $\mathcal{C}_0$ is expected to grow linearly with N. Since $\mathcal{C}_0/N$ is known to be peaked at large N, it is natural to conjecture the stronger property that $\mathcal{C}_0/N$ tends toward a constant with probability 1 as $N \to \infty$. A major motivation for this work is our expectation that an identical scaling law should hold if we replace $\mathcal{C}_0$ by the cost found by a heuristic algorithm, though the limiting constant depends on the heuristic. To motivate such a property, the next section analyzes the cut sizes of random partitions; then, in section 4, we consider the "statistical physics" of the GPP so as to interpolate between the case of minimum cuts and that of random cuts.

3. Cuts of random partitions.

Here we show explicitly that a large N scaling law holds for the cut sizes of random partitions, and that asymptotically these random cuts have a Gaussian distribution with a relative variance proportional to $1/N$.

Consider any graph in $G(N,p)$. One can always write the cut size of a random partition as $\mathcal{C} = X + Y$, where X is the mean (random) cut size for the graph under consideration, and $\lceil Y \rceil = 0$. ($\lceil\ \rceil$ is the average over the random partitions.) Averaging explicitly over all balanced partitions of the fixed graph, we find $X = \sum E_{ij} N/[2(N-1)]$. The interpretation of this formula is very simple: any edge of weight E_{ij} has a probability $N/[2(N-1)]$ of being cut.

In the ensemble $G(N,p)$ of random graphs, it is easy to calculate the first few moments of X. In particular, we find $\langle X \rangle = pN^2/4$ and $\langle (X - \langle X \rangle)^2 \rangle = p(1-p)N^3/[8(N-1)]$. ($\langle\ \rangle$ denotes the average over the ensemble $G(N,p)$.) We also see that X is the sum of $M = N(N-1)/2$ *independent* random variables; this implies that the kth cumulant (connected moment) of the distribution of X satisfies

$$\left\langle X^k \right\rangle_c = (N-1)^2 \left[\frac{N}{2(N-1)}\right]^{k+1} \left\langle E_{ij}^k \right\rangle_c . \tag{3.1}$$

At large N, we then have $\langle X^k \rangle_c \sim N^2$ in the constant p ensemble, and $\langle X^k \rangle_c \sim \alpha N$ in the $p \sim \alpha/N$ ensemble.

The random variable Y is more subtle, as it is the sum of M *correlated* variables. Nevertheless, for any graph, it is possible to compute the moments of Y, and we have done this explicitly for the second and third moments. (The expressions are too long to be given here.) If we average Y^2 both over random partitions and over $G(N,p)$, we obtain

$$\langle \lceil Y^2 \rceil \rangle = \frac{p(1-p)}{8} N^2 (N-2)/(N-1). \tag{3.2}$$

The calculations get significantly more complicated for the higher moments. In order to keep to simple expressions, we limit ourselves to the ensemble with $p = \alpha/(N-1)$. Then we find

$$\langle \lceil Y^2 \rceil \rangle = \frac{\alpha}{8} N + O(1), \qquad \langle \lceil Y^3 \rceil \rangle = -\frac{\alpha}{8} + O\left(\frac{1}{N}\right). \tag{3.3}$$

Furthermore, the graph-to-graph fluctuations of $\lceil Y^2 \rceil$ become negligible in relative magnitude, so that the ratio of a typical variance to the mean variance goes to 1 at large N. This, however, is not true for the higher moments; for instance, we find that the typical value of $\lceil Y^3 \rceil$ grows as $N^{1/2}$, but taking in addition the mean over graphs leads to N-independent behavior. Finally, one can show that $\lceil Y^k \rceil_c / \lceil Y^2 \rceil^{k/2} \to 0$ with probability 1. This shows that as $N \to \infty$, Y has a Gaussian distribution of zero mean, and of variance growing linearly with N, whose coefficient is graph independent.

Coming back to $\mathcal{C} = X + Y$, the cut size of a random partition, we find that the normalized correlation coefficients between powers of X and Y tend to zero at large N, and thus X and Y become independent random variables in that limit. This, along with the results previously derived, shows that at large N, $\mathcal{C}$ itself has a Gaussian distribution. From these results, we deduce the large N behavior,

$$\frac{\langle \lceil (\mathcal{C} - \langle \lceil \mathcal{C} \rceil \rangle)^2 \rceil \rangle}{\langle \lceil \mathcal{C} \rceil \rangle^2} \sim \frac{4}{\alpha N} \ , \tag{3.4}$$

so that *relative* deviations from the mean go to zero. Thus the distribution of $\mathcal{C}$ becomes peaked, and $\mathcal{C}/N \to \alpha/4$ with probability 1 as $N \to \infty$. The convergence of the distribution of $\mathcal{C}/N$ to a "delta" function is referred to as the self-averaging of $\mathcal{C}$.

The scaling of the variances can be summarized at large N by writing

$$c \equiv \frac{\mathcal{C}}{N} \sim \langle \lceil c \rceil \rangle + \frac{\sigma_X^*}{\sqrt{N}} x + \frac{\sigma_Y^*}{\sqrt{N}} y, \tag{3.5}$$

where x and y are independent Gaussian random variables of zero mean and unit variance; $\sigma_X^* = \sqrt{\alpha/8}$ is the standard deviation (rescaled by $1/\sqrt{N}$) of X, and $\sigma_Y^* = \sqrt{\alpha/8}$, that of Y. Thus σ_Y^* describes the fluctuations of the cut sizes within a graph, and σ_X^* describes the fluctuations of the mean cut size from graph to graph.

We have used these analytical results to test the validity of our computer programs. The first two moments of X allowed us to test our generation of random graphs in $G(N,p)$. Similarly, a check on our random number generator was obtained by verifying on several graphs that the second moment of Y found by the numerics was in agreement with our formulae. Finally, we also checked that random cut sizes have a limiting Gaussian distribution, with a third moment that scales to zero at large N. (For this check, we performed random partitions on $100,000$ graphs for $N = 100, 500, 1000$, and 2000.)

4. Statistical physics of the GPP. We saw that cut sizes of random partitions in $G(N,p)$ have a self-averaging property; we conjectured that this property also holds for the minimum cut. It is possible to interpolate between these two kinds of partitions (random and min-cut) by following the formalism of statistical physics. For any given graph, consider the "Boltzmann" probability distribution p_B, defined for an arbitrary partition P of cut size $\mathcal{C}(P)$:

$$p_B(P) = \frac{e^{-\mathcal{C}(P)/T}}{Z} \ . \tag{4.1}$$

Z is chosen so that p_B is normalized (a probability distribution) and T is an arbitrary positive parameter called the temperature. When $T \to \infty$, we recover the ensemble of random partitions where all partitions are equally probable, whereas when $T \to 0$, the ensemble reduces to the partitions of minimum cut size. For intermediate values of the temperature, the partitions are weighted according to an exponential of their cut size. In this "Boltzmann" ensemble, one can define the moments of the cut sizes just as was done in the case of random partitions. In most statistical physics problems, it is possible to show that the quantity in the exponential of (4.1) (here, the cut size) is self-averaging. For *random* graphs, however, the proofs are inapplicable; nevertheless, other evidence indicates that the cut size is self-averaging at any temperature [26]. This self-averaging can be understood qualitatively at low temperature as follows. The number $\mathcal{N}(\mathcal{C})$ of partitions of cut size $\mathcal{C}$ is a sharply increasing function of $\mathcal{C}$, whereas the Boltzmann factor is a sharply decreasing function of $\mathcal{C}$. Note that the probability distribution $P(\mathcal{C})$ of $\mathcal{C}$ is given by the product of these two functions. Using naive but standard statistical physics arguments for $\mathcal{N}(\mathcal{C})$, one finds that $P(\mathcal{C})$ has a peak at $\mathcal{C}^*(T)$ that grows linearly with N and that the width of the distribution is $O(\sqrt{N})$, which gives the self-averaging property for $\mathcal{C}$. In addition, this kind of argument says that $P(\mathcal{C})$ becomes Gaussian at large N, a result that is usually correct in statistical physics systems.

A number of statistical physics results have been obtained for the GPP in the ensemble of dense random graphs, that is, for $G(N, p)$ at p fixed. In particular, highly technical calculations [26, 8] indicate that the cut sizes are self-averaging at all temperatures, that is, as $N \to \infty$, relative fluctuations within a fixed graph become negligible, as do those from graph to graph. The mean cut size is given by

$$\langle \lceil \mathcal{C} \rceil \rangle = \frac{pN^2}{4} - U(T)\sqrt{p(1-p)}N^{3/2} + O(N) \tag{4.2}$$

as $N \to \infty$. (If the mean over graphs is not performed, the formula remains valid for "almost all" sequences of graphs with $N \to \infty$.) In this equation, $U(T)$ is a function of temperature only; there is no dependence on p as long as p is independent of N. The limit $T \to 0$ gives the expected (and typical) value of the minimum cut, with $U(T = 0) = 0.3816$. Although there is no proof yet that these calculations are exact, there is general agreement in the statistical physics community that the results are correct.

The case of sparse random graphs ($p \sim 1/N$) has also been studied within the statistical physics approach [2, 5]. So far, however, the problem has proven to be intractable, and there is no plausible solution in sight. Nevertheless, it is expected that the cut sizes are self-averaging at any temperature and that the mean of the distribution scales linearly with N at large N.

The property of self-averaging seems quite generic. The reason it should hold in these systems is that the cut size of a partition is the sum of a large number of random variables that are not *too* correlated. It is very plausible that the cut size is self-averaging whenever partitions are generated by an iterative process involving just a few vertices at a time. All local search methods and modifications thereof, such as simulated annealing, fall into this category. Thus our claim is that any heuristic algorithm that generates partitions iteratively according to local (in vertex space) criteria will lead to cut sizes that are self-averaging. Thus the distribution of cut sizes found by any such heuristic should become peaked as $N \to \infty$. Furthermore, in this limit, the distribution should converge toward a Gaussian in the way given by the central limit theorem. We will see in the sections to follow that this is indeed borne out empirically for all of the heuristics that we have investigated.

The arguments we have presented are not specific to the GPP, so we expect them to apply to most stochastic COPs having many variables in their cost function. Surprisingly, there has been very little research on this topic. In the context of the

"NK" model with binary variables, a study by Kauffman and Levin [16] found that the costs of *local minima* became peaked toward the value of a *random* cost as N grew. (This peculiar property is due to the structure of the energy landscape in that model.) However, concerning the behavior of *heuristic* solutions, research has almost exclusively focused on the case of the Euclidean traveling salesman problem, where points are laid out on the plane. Most practitioners in that field know that local search heuristics give rise to costs whose relative variance decreases as the number of points increases. Furthermore, it was observed by Johnson and McGeoch [14], among others, that the costs tend toward a fixed percentage excess above the optimum. Our purpose here is to show *how* this convergence occurs, albeit in a different combinatorial optimization problem, and to provide a theoretical framework for understanding where this behavior comes from. Also, we pay special attention to the distinction between fluctuations within an instance and from one instance to another. We believe our findings are quite general and in particular that the ensemble of instances considered need not be based on points in a physical space.

5. Algorithms used in the testbed. In view of the previous arguments, we have restricted ourselves to local heuristics. Without trying to be either complete or representative, we have studied the statistics of cut sizes for three types of local search and four versions of simulated annealing algorithms. In this section we sketch the workings of these heuristics. In sections 6 and 7, we show that the same self-averaging properties hold for all these algorithms in spite of their significant differences. There is thus no reason to believe that our claims are affected by the details of such algorithms; rather, the properties are most likely generic to dynamics that are local.

5.1. Kernighan–Lin algorithm. In simple local search, one performs elementary transformations to a feasible solution of the COP as long as they decrease the cost, a procedure sometimes called λ-opting [22]. A more sophisticated version consists in using "variable depth" search: one builds a sequence of p elementary transformations, usually according to a greedy criterion. p is not set ahead of time and depends on the sequence of costs found. The elementary transformations are not imposed to decrease the cost, but the sequence of length p must do so if it is to be applied to the current solution. Such a procedure was first proposed by Kernighan and Lin [18]—in fact in the framework of the GPP. Hereafter we will refer to their algorithm as "KL." The elementary transformation they use is the exchange of a pair of vertices: one vertex in V_1 being exchanged for one in V_2. A sequence of such exchanges is built up in a greedy and tabu fashion by performing a "sweep" of all the vertices: at each step of the sweep, one finds the best (largest cost gain) pair to exchange among those vertices that have not yet been moved in the sweep (tabu condition). The sweep has length $N/2$. When the sweep is finished, one finds the position p along the sequence of exchanges generated where the cut size is minimum. If this minimum leads to an improved partition, the transformation of p exchanges is performed on the partition and another sweep is initiated; otherwise, the search is stopped and the partition is "KL-opt"; that is, it is a local minimum under KL.

The KL algorithm is deterministic, although it is possible to introduce stochasticity to break degeneracies in selecting the best pair to exchange. Its computational complexity is not easy to estimate because the number of sweeps is not known in advance. (This is a generic difficulty in estimating the speed of iterative improvement heuristics.) However, in practice, one finds that KL finishes in a "small" number of sweeps. Thus the computational complexity is estimated to be a few times that of performing the last sweep, known as the check-out sweep. For our study, we have used our own implementation of KL [24], which uses heaps to find the best pair to exchange at each step. For sparse graphs, this leads to $\mathcal{O}(N \ln N)$ operations per sweep. A nearly identical KL is provided in the Chaco software package, which gives sensibly identical results. A faster implementation of the algorithm has been given by Fiduccia and Mattheyses [7] whenever the use of a radix sort is possible; then the time for each sweep is $\mathcal{O}(N)$.

In terms of quality of solutions found, KL is quite good. What is surprising is that, though Kernighan and Lin proposed their method over 20 years ago, KL remains relatively unchallenged, at least as a general purpose method applicable to any kind of graph, regardless of its structure. Of course, for special kinds of graphs, such as meshes, other heuristics (e.g., spectral bisection) perform better [4, 11, 13, 15].

5.2. A multilevel KL algorithm: CHACO. The Chaco software package includes a number of heuristics for partitioning graphs. (For information about this package, see *The Chaco User's Guide* [10].) For our purposes, we have used only its "multilevel" generalization of KL, hereafter referred to simply as CHACO. The CHACO algorithm is based on a coarse graining or "compactification" of the graph to be partitioned. At each level, vertices are paired using a matching algorithm, and paired vertices are then considered as the vertices of the next higher level of compactification. Because of this process, it is necessary to have weighted edges; the weights are also propagated to the higher level. The compactification is repeated until a sufficiently small graph is obtained to which spectral bisection is applied to get a first partition. Then this partition is used as the starting partition in KL for the graph at the level below it. This process is recursive until one obtains a KL-opt partition of the original graph. (Note that this construction is deterministic and does not require an initial "random" partition.) Such a multilevel strategy has been very successful for unstructured two- and three-dimensional meshes [11, 15], both in terms of solution quality (much better than for KL alone) and in terms of speed (much faster than KL because of the hierarchical nature). However, the usefulness of CHACO on random graphs is not a priori obvious in terms of either speed or quality of solutions.

5.3. Simulated annealing algorithms. We have chosen as a third comparative algorithm simulated annealing (SA). SA is based on a set of elementary moves, just like local search, but now moves that increase the cost are accepted with (low) probability. Because of this, it is sometimes appropriate to consider SA as a noisy local search method. SA is really a family of algorithms. To include some of the different bells and whistles proposed for this algorithm, we have considered four variations, which we will now describe. (i) The SA as first introduced by Kirkpatrick, Gelatt, and Vecchi [19] (referred to as FSA) has the initial and final temperatures fixed ahead of time by the user and a predetermined number of trial moves performed at each temperature. (ii) Kirkpatrick, Gelatt, and Vecchi [19] also proposed to determine the initial and final temperatures of the schedule dynamically. They set the initial temperature at the beginning of the run using the criterion that about 80% of the trial moves are accepted at that temperature. Similarly, they stop the cooling if the energy does not decrease for five cooling steps. We will refer to this method as KSA. (iii) Johnson et al. [13] improved the speed of this algorithm by allowing an early exit to the next temperature of the schedule; the condition they proposed for exiting is having accepted a minimum number of moves. Also, they modified the termination criterion to having an acceptance rate less than a threshold value. We will refer to this version as JSA. All three of these SA methods use an exponential cooling schedule with a cooling factor of 0.95. (iv) The last SA variation consists of using an *adaptive schedule* whereby the next temperature value is determined on the fly according to the energy fluctuations at the current temperature. We have chosen for this variation the implementation of van Laarhoven and Aarts [28, 29], which we call ASA. To obtain good results with this SA, one would have to spend a long time in the "freezing" phase of the cooling. Since this would increase the computation times significantly, we have chosen not to use a fine-tuned adaptive schedule but one that provides a cooling factor of the same magnitude as in the other SA algorithms presented. This allows us to have similar computation times for all the SA algorithms investigated.

In SA, one can use the same elementary moves as in local search, that is, for the GPP pair exchanges. However, once a low-cost partition is obtained, it will take a long time (or a lot of luck) to find further good exchanges. Finding a good *pair* is best done by finding the first vertex to transfer and then the second, that is, by using a *sequential* process. This suggests relaxing the constraint of having balanced partitions and replacing it by a penalty function that keeps the sizes of V_1 and V_2 nearly equal (small *off-balance*). We have followed a slightly different approach, where each move destroying the balance must be followed by a move restoring the balance. Then the Markov chain explores the partitions which are balanced and those with off-balance of ± 1. It is easy to see that this method is equivalent to having the cost of all the other partitions equal infinity; at fixed temperature and for long chains, one generates partitions with cut sizes given by the Boltzmann factor, within the constraint for the off-balance. Indeed, the succession of accept/reject decisions makes the global probability distribution Boltzmannian in this enlarged space, so that we guarantee the same convergence properties as in the standard case.

Some remarks concerning our implementations are in order. First, at fixed temperature, we perform a certain number of "sweeps." In each sweep, every vertex is sequentially considered as a candidate for changing sides of the partition; if the move were to violate our limit on the off-balance, the move would be rejected (in fact, it simply would not be considered). A sweep thus requires $O(N)$ operations. Our sweeps use random *permutations* rather than a fixed or random ordering of the vertices. The use of random permutations should, according to certain authors [13, 28, 29], result in an enhancement of the quality of the solutions found. Second, the maximum number of sweeps at any temperature is set to $\alpha\lambda$, with $\lambda = 10$ for all of our implementations. For FSA and KSA, this is in fact the (actual) number of sweeps, so that their computational complexity is $O(\alpha\lambda N)$ times the number of temperature steps used. The cases of JSA and ASA are more difficult to evaluate. In practice we find that JSA is faster than KSA, but not by more than a constant factor. ASA (adaptive SA), on the other hand, spends more time at intermediate temperatures as N increases; empirically, we have found an $O(N^{3/2})$ complexity.

In terms of quality, we are aware of no systematic study on sparse random graphs. In a previous SA work on the GPP, Van Laarhoven and Aarts used an adaptive decrement rule [28, 29] and claimed a gain of about 13% over simpler nonadaptive algorithms. They also compared their results to those from the algorithm used by Johnson et al. [13] for the GPP; Johnson et al. claimed an enhancement of about 5% for JSA over the KL algorithm. The small gain found by Johnson et al. [13] is, according to van Laarhoven and Aarts [28, 29], due to the use of a nonadaptive choice of the temperature decrement rule. However, we have found for sparse random graphs that the different variants of SA are nearly indistinguishable in terms of quality of solutions. This may be due to our not using a penalty term or to the different nature of the graphs used in the present study.

5.4. Chained local optimization. The chained local optimization (CLO) strategy is a synthesis of local search and of SA [25]. The essential idea is to have SA sample not all solutions, but only locally optimal solutions. This strategy is guaranteed to be at least as good as local search and has been successfully applied to the traveling salesman problem [23] and to the partitioning of unstructured meshes [24].

In this work, we use KL as the local search engine. Given any initial KL-opt partition P_i, the simplest implementation of CLO will: (i) apply a perturbation or "kick" to modify significantly the partition (in practice, this means exchanging *clusters* of vertices); (ii) run KL on the modified partition so as to reach a new KL-opt partition P_f; (iii) apply the accept/reject procedure for going from the initial partition (P_i) to the final one (P_f). This defines the analogue of one move of an SA algorithm, except that many modifications to the partition have occurred in this single step. The temperature may be modified according to a schedule if desired, but for simplicity, we have set the temperature to zero in all of our runs.

As was discussed in the context of SA, it is inefficient to exchange vertices or clusters simultaneously; it is better to do it sequentially. Our present CLO algorithm thus proceeds as follows. Given P_i, an initial balanced KL-opt partition, choose a (connected) cluster of p vertices in V_1 (or V_2), and move them into V_2 (respectively, V_1). KL-optimize this partition to generate an intermediate (off-balance) partition. Now choose a cluster of p vertices in V_2 (V_1) and move them into V_1 (V_2); KL-optimize this modified partition to generate P_f, the final (and *balanced*) partition. This whole procedure is our "simulated annealing" step, and we apply the accept/reject criterion for going from P_i to P_f.

When running CLO on irregular meshes [24], it was possible to perform large kicks, exchanging many vertices at once. Unfortunately, for sparse random graphs, we find that the acceptance when doing so becomes low. We have thus used "small" kicks, creating clusters of sizes varying randomly between 3 and 13. Given such small kicks, KL usually terminates in just two sweeps, and the speed of CLO per kick is about half that of KL.

Now consider the limit of large N. Using the analogy with SA, if a fixed (N-independent) number of small kicks are used, it can be expected that CLO will perform no better than KL itself. We have thus chosen to use a number of kicks which scales linearly in N, namely, λN with $\lambda = 0.1$. This choice of course influences the quality of the solutions generated, a larger value of λ giving a priori better results. The computational complexity of this algorithm is then of order $N^2 \log(N)$.

6. Self-averaging of the cut sizes. In the rest of this paper, we study the statistical properties of the cut sizes generated by the algorithms described in section 5 when applied to random initial partitions. The ensemble of graphs used is that of random graphs with mean connectivity $\alpha = p(N-1) = 5$ (see section 2). This value was chosen because at much larger connectivities, the ratio between the best and worst cut sizes approaches 1, and at lower connectivities, algorithms taking explicit advantage of disconnected parts of the graph will outperform general purpose heuristics. In order to minimize effects associated with our finite sample of graphs in the ensemble, we have benchmarked all the algorithms on the *same* graphs. The number of graphs used during the production runs was $10,000$, with values of N ranging between 50 and 200; however, because the CHACO algorithm was fast, we have also performed runs on $100,000$ graphs for that heuristic.

The purpose of this section is to give numerical evidence that the distribution of cut sizes becomes peaked in the limit of large graphs, for each of the heuristics considered. (Further properties of the distribution will be given in section 7.) We find that each algorithm generates cut sizes for which both the mean and variance scale linearly in N. From this behavior, it is clear that the distribution of cut sizes becomes peaked at large N, that is, that the cut sizes are self-averaging. Also, assuming (cf. section 2) that the minimum (i.e., optimum) cut size scales linearly with N at large N, we then see that each heuristic algorithm leads to a fixed percentage excess above the true optimum. (Note that the worst cut size also has a linear scaling in N.) This percentage excess provides a first ranking of the algorithms, which, however, does not take into account the speed of execution.

If $\mathcal{C}(i,m)$ is the cut obtained by a heuristic for the graph G_i and an initial partition m, define the mean cut per vertex $\langle\lceil c\rceil\rangle$ by

$$\langle\lceil c\rceil\rangle \equiv \left\langle \left\lceil \frac{\mathcal{C}(i,m)}{N} \right\rceil \right\rangle, \tag{6.1}$$

where the averages are over initial partitions and over the ensemble of graphs studied (cf. section 3 for the notation). We compute these ensemble averages numerically using the standard estimator (hereafter, overlines refer to numerical averages):

$$\overline{c} \equiv \frac{\sum_i \sum_m \mathcal{C}(i,m)}{N \sum_i \sum_m 1} \approx \langle\lceil c\rceil\rangle. \tag{6.2}$$

The approximation is due to a statistical error e associated with fluctuations of $\mathcal{C}(i,m)$ with both m and i. It is not difficult to see that for our problem, one does not need to perform an average over m; using any finite number R of partitions for each graph G_i provides an unbiased estimator of $\langle\lceil c\rceil\rangle$. Furthermore, the statistical error e is not very sensitive to R, making it numerically inefficient to take a large value for R. Because of this, we have performed the numerical averages with $R = 1$, and this leads to a simple expression for e, the statistical error on $\overline{c}$:

$$e^2 = \frac{\langle\lceil (c - \langle\lceil c\rceil\rangle)^2\rceil\rangle}{\sum_i 1} \approx \frac{(\overline{c^2} - \overline{c}^2)}{\sum_i 1}. \tag{6.3}$$

Figure 6.1 shows the dependence of $\overline{c}$ on $1/N$. (The error bars are too small to be visible. Also, in order to avoid cluttering the figure, we have included among the SA algorithms only KSA; the other implementations of simulated annealing give nearly identical results.)

For all algorithms, the figure suggests that there is a limiting large N value for $\overline{c}$ and that the convergence to this limit is linear in $1/N$. We have thus fitted the data to a linear function:

$$\frac{\overline{\mathcal{C}}}{N} \equiv \overline{c} \approx A + \frac{B}{N}.$$

The values of the A and B coefficients obtained from the fits are given in Table 6.1, and the χ^2 values show that the fits are good.

An identical analysis can be performed on the *variance* of the cuts found by the different algorithms. Figure 6.2 shows the dependence on N for the rescaled quantity $N(\overline{c^2} - \overline{c}^2)$. The scaling in N is apparent, just as it was for $\overline{c}$.

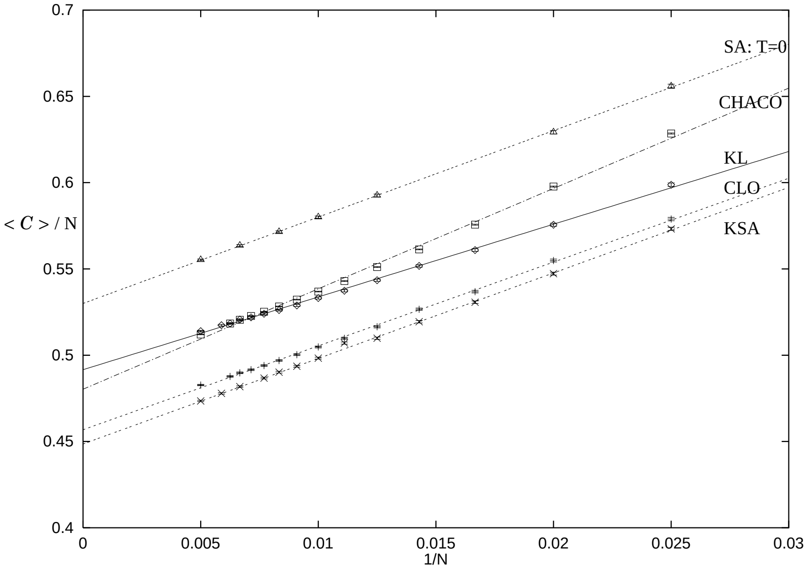


Fig. 6.1. *Scaled mean cut sizes for the different algorithms.*

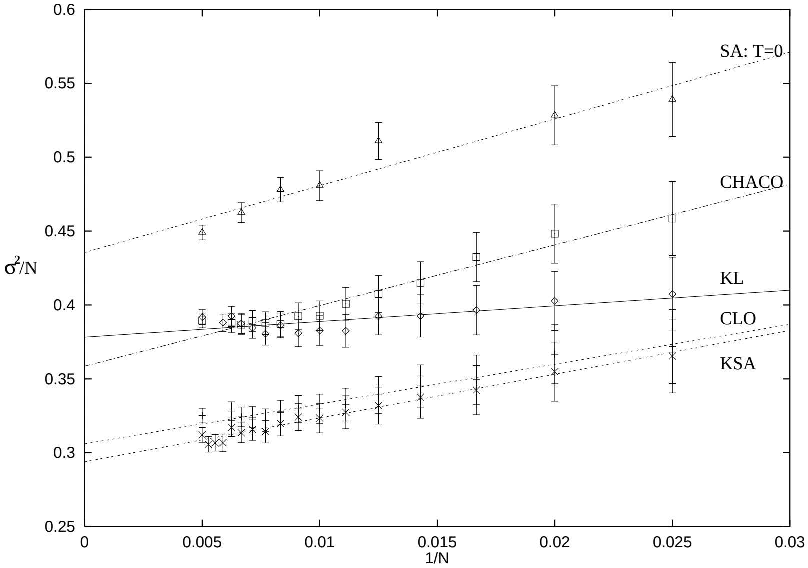


Fig. 6.2. *Scaled variance of the cut sizes for the different algorithms.*

In summary, our data lead us to conclude that the mean and variance of $\mathcal{C}$ scale linearly with N at large N. Then the relative width of the distribution of $\mathcal{C}$ is proportional to $1/\sqrt{N}$, showing that the distribution for the cut sizes becomes peaked for all the algorithms investigated. (One can also say that the distribution of $\mathcal{C}(i,m)/N$ tends toward a delta function as $N \to \infty$, which is what we mean by self-averaging.) Since the fluctuations of $\mathcal{C}(i,m)$ include both graph-to-graph fluctuations and fluctuations within a graph, we can conclude that the relative fluctuations within a fixed typical

TABLE 6.1

Estimates for the large N value and slope of the mean cut size per vertex and percentage excess relative to the KSA heuristic.

Algorithm	A	B	% excess
KSA	0.4485	4.95	0.00
FSA	0.4489	4.92	0.08
ASA	0.4499	4.96	0.32
JSA	0.4513	4.88	0.63
CLO	0.4568	4.85	1.8
CHACO	0.4802	5.81	7.1
KL	0.4916	4.21	9.6
SA $T = 0$	0.5302	4.79	18.2

graph necessarily also go to zero. (Note: Although for our runs we use $R = 1$, our observable $(\overline{c^2} - \overline{c}^2)$ is an unbiased estimator for $\langle \lceil (c - \lceil \langle c \rceil \rangle)^2 \rceil \rangle$ which includes both types of fluctuations.) Thus, in the large N limit, each algorithm will give a fixed percentage excess above the minimum for almost all graphs and almost all random initial partitions.

A speed-independent ranking. Since each algorithm is characterized by a percentage excess, we can introduce a ranking of the different heuristics according to their excess in the large N limit. (Of course, this ranking does not take into account the speed of the algorithms!) For our graphs and our implementation of the different heuristics, the winners are in the class of SA. The best is KSA; using this as the reference rather than the true min-cut size (which is unknown), JSA has an excess of 0.63%, ASA an excess of 0.32%, and FSA an excess of 0.08%. The next best heuristic is the CLO algorithm, followed by CHACO, and finally KL. (The results for the excesses are given in Table 6.1.) We have also included for general interest the excess obtained by a zero temperature "simulated annealing": 18.21%; note that it gives much worse results than KL, while true SA gives much better results than KL.

As a comment, let us remark that the relative solution quality of the algorithms is determined to higher precision than the absolute quality. Simply put, the cut sizes we obtain for the different algorithms are correlated because they are performed on the same graphs, so that the statistical error on $\langle \lceil c_{CLO} - c_{KL} \rceil \rangle$, for instance, is 3.2 times smaller than the statistical error on $\langle \lceil c_{CLO} \rceil \rangle$ alone. This is why it is possible to give reliable values for the excesses of the different SA algorithms even though their solution quality is very similar. Nevertheless, the ranking for the SA algorithms is not without ambiguity. The FSA algorithm is, for larger N, within the statistical error of the KSA algorithm, and hence we have no strong evidence that one is better than the other.

The other algorithms are easily ranked. KL and CHACO are 9.6% and 7.1% worse than KSA, but CLO is only 1.8% worse. The comparison with KL is qualitatively (though not quantitatively) similar to that given by Johnson et al. [13] and by van Laarhoven and Aarts [29]. Both claimed a gain of the SA algorithm over the KL algorithm of about 5% and 13%, respectively. The differences with our results have several origins. First, we have performed an average over an ensemble of graphs. Second, our graphs have slightly different characteristics than the ones they used. Third, we have not introduced a penalty term in our implementation of SA; this probably affects the quality of the solutions found.

7. Distribution of cut sizes. In this section we deepen our statistical study of $\mathcal{C}$. As shown in the previous section, the distribution of $\mathcal{C}/N$ tends toward a delta

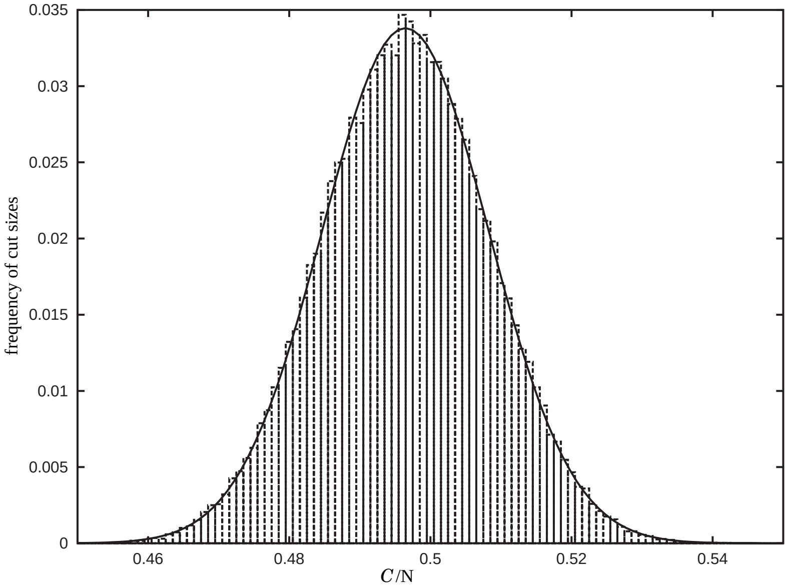


FIG. 7.1. *Histogram of KL cut sizes for one $N = 1000$ graph with overlaid Gaussian.*

function; it is natural to ask *how* this limit is reached and to understand the nature of intra- and intergraph fluctuations. It is convenient to use the framework introduced in section 3, but with *random* partitions replaced by the partitions found by applying one of our heuristics to a random start. For each graph G_i, and each initial partition m, we define

$$\mathcal{C}(i,m) = X(i) + Y(i,m),$$

where $\lceil Y(i,m)\rceil = 0$, so that $X(i)$ is the average cut size found on graph G_i and $Y(i,m)$ gives the fluctuation of the cut size about its mean for that graph. For each of our heuristics, our study indicates that for a large random graph G_i, Y has a nearly Gaussian distribution, and that the width of this distribution is essentially independent of i. We study this distribution at large N and show that its width is self-averaging and its relative asymmetry goes to zero. Finally, we have evidence that X and Y become independent variables at large N. These properties will lead to a fast and robust ranking of the heuristics in section 8.

Figure 7.1 shows the distribution of cut sizes found by KL on one $N = 1000$ graph chosen at random from $G(N,p)$ with $p = \alpha/(N-1)$. Superposed is a Gaussian with the same mean and variance. The figure gives good evidence that the distribution of Y for that graph is very close to a Gaussian. Then an obvious question is whether the distribution of Y is similar across different graphs. For each of our heuristics, we find that the answer is yes, as indicated by the following study of the moments of Y. (Note that for the CHACO algorithm, the default parameter setting generates the initial starting partition deterministically by application of the coarse graining strategy; then a spectral method is applied. Since there is no "random" initial partition, there are no fluctuations in the cut size as a function of m and so little in this section applies to CHACO with these parameter settings.)

To quantify how $\sigma_Y^2(i) \equiv \lceil Y^2(i,m)\rceil$ varies from graph to graph, we measured its

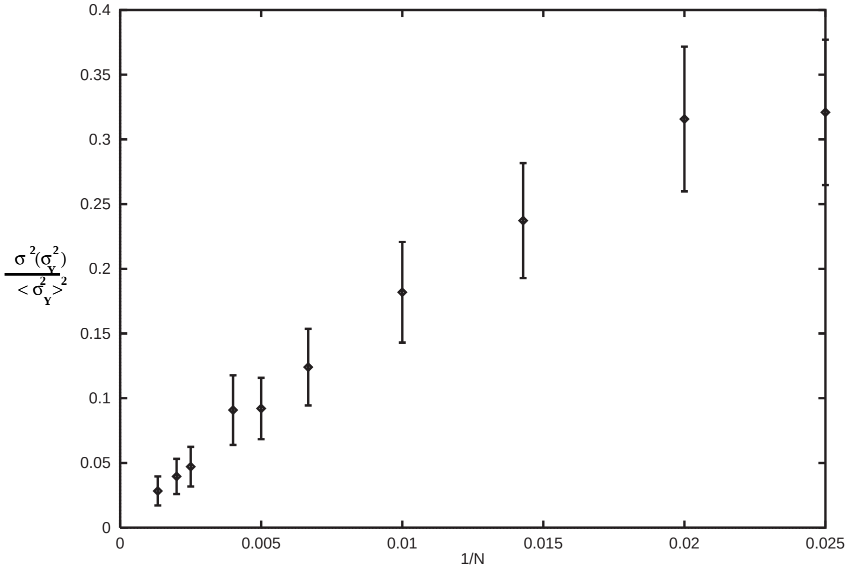


FIG. 7.2. *Relative variance of the intragraph cut size variance* σ_Y^2.

mean and variance over i. First, we measured the ensemble averages $\langle\sigma_Y^2(i)\rangle/N$. For each heuristic, the data extrapolates to a limiting value as N becomes large. Comparing with the results for the mean cut size, we find that the algorithms which lead to the best cut sizes also have the smallest widths for the Y distribution. Second, we studied the *variance* of $\sigma_Y^2(i)$, that is, $\sigma^2\left(\sigma_Y^2(i)\right)$. This study requires high statistics, and so was performed to high accuracy only for KL, the fastest of our algorithms; however, the other algorithms show qualitatively the same behavior. Figure 7.2 displays for KL the $1/N$ dependence of the relative variance of $\sigma_Y^2(i)$, that is, the intergraph variance of $\sigma_Y^2(i)$ divided by the square of its mean. As can be seen from the figure, the ratio goes to zero at large N, showing that $\sigma_Y^2(i)$ is self-averaging. Simply put, this means that the width (over m) of the Y distribution has relative fluctuations from graph to graph that disappear as $N \to \infty$. (Our lower statistics data for the other heuristics are consistent with this conclusion.)

Following the statistical physics analogy given in section 4, there is reason to believe that the distribution of Y tends toward a Gaussian as in the case of random partitions. To test this conjecture, we have measured the asymmetry of the distribution of Y on numerous graphs for KL. First, we find that the typical asymmetry is small and that the mean of the third moment of Y satisfies

$$\langle\lceil Y^3(i,m)\rceil\rangle/\langle\sigma_Y^2(i)\rangle^{3/2} \to 0$$

as $N \to \infty$. Second, we have checked that the average of the squared asymmetry is also small, that is, that

$$\langle\lceil Y^3(i,m)\rceil^2\rangle/\langle\sigma_Y^2(i)\rangle^3 \to 0.$$

These properties give strong evidence that the distribution of Y for any graph tends toward a Gaussian of zero mean and of variance AN as $N \to \infty$, where A depends on the heuristic but not on the actual graph.

The distribution of $X(i)$ can be studied similarly. The previous section gave its mean as a function of N and also showed that it is self-averaging. It is of interest to quantify the decrease with N of its relative variance. We have found that the distribution of X is roughly compatible with a Gaussian distribution of width proportional to $\sqrt{N}$ for each of the algorithms. (Unfortunately, a quantitative test of this requires very high statistics.) However, the distribution of $X(i)$ is not essential for our ranking procedure as will be clear in the next section, so we have not studied it in greater depth.

Finally, to completely specify the statistics of $\mathcal{C}(i,m)$, it is necessary to describe the correlations between $X(i)$ and $Y(i,m)$. We have found numerically that these variables are nearly uncorrelated, with, in particular, the correlation between $X(i)$ and $\sigma_Y^2(i)$ tending toward zero as $N \to \infty$. Assuming that this holds and that X has a Gaussian distribution, then the distribution of $\mathcal{C}(i,m)$ is also Gaussian. Our measurement of the asymmetry (jointly over i and m) of $\mathcal{C}(i,m)$ is compatible with this property at large N. (The total variance is then given by the sum of the variances of X and Y.) This can be summarized mathematically by introducing two Gaussian random variables x and y of zero mean and unit variance and modeling the rescaled cut size as the following sum:

$$c(i,m) \sim \langle \lceil c \rceil \rangle + \frac{\sigma_X^*}{\sqrt{N}}\, x(i) + \frac{\sigma_Y^*}{\sqrt{N}}\, y(i,m).$$

This equation is then the exact analogue of what was derived for the cut sizes of random partitions (see (3.5)).

8. A speed-dependent ranking of heuristics. In this section we come back to the initial motivation for this work, namely, the necessity of comparing heuristics of very different speeds. The possibility of doing so is very relevant, as for most COPs local search is quite fast and SA notoriously slow. Any meaningful ranking must determine whether it is better to have a fast heuristic that gives rather poor solutions or a slower heuristic that gives better solutions. We now show how to introduce such a ranking when considering first just one graph, and then how to generalize to an ensemble of graphs. Finally, we illustrate what this ranking gives in the case of the heuristics in our testbed when applied to sparse random graphs.

The case of one graph. Consider a single graph G on which one is to provide a ranking of a number of heuristics that give various cut sizes and run at different speeds. To take into account both the speed of the algorithms and the quality of the solutions they generate, we fix the amount of computation time allotted per algorithm. Call this time τ (measured, for instance, in CPU seconds on a given machine). Each heuristic then generates (nonoptimal) solutions during that time using multiple random initial starts. Suppose that the speed of the algorithm of interest is such that k independent starts can be performed in the allotted time τ. (We shall assume that the execution time is insensitive to the random initial start, as this is the case in practice with our heuristics. Knowledge of the speed of the algorithm then gives the value of k that can be used.) For each start, there is an output or "best-found" cost. The output at the end of the k starts is the best of these k costs, hereafter called "best-of-k." The different algorithms are then ranked on the basis of the *ensemble mean* of their best-of-k (the value of k depending on τ and on the algorithm). This ensemble average is the average over the random numbers used both for the random initial starts and for running the algorithms (if any). This establishes a ranking for a particular graph and for a given amount of computation time τ.

It is inefficient to perform the average just mentioned in a "direct" way, that is, by extracting values of best-of-k over many multiple runs; it is far better to compute the average starting with the *distribution* of the "best-found" cut sizes associated with single random starts. Call $P(\mathcal{C})$ the probability of finding a best-found cut size of value $\mathcal{C}$, and $Q(\mathcal{C})$ the associated cumulative distribution, that is, the probability of finding a cut size (strictly) smaller than $\mathcal{C}$. Since the cut sizes are integer valued, we then have $P(\mathcal{C}) = Q(\mathcal{C}+1) - Q(\mathcal{C})$. Introducing the analogous probabilities $\tilde{P}_k$ and $\tilde{Q}_k$ for the best-of-k values, one has

$$1 - \tilde{Q}_k(\mathcal{C}) = (1 - Q(\mathcal{C}))^k.$$

The distribution for best-of-k can thus be generated from that of best-found, and then $\mathcal{C}^*$, the mean of best-of-k, is easily extracted. (This construction explains why we studied the distribution of single cut sizes in section 7.) Note also that it is possible to extract $\mathcal{C}^*$ for a whole range of τ values with essentially no extra work since τ affects only k and the determination of the mean of best-of-k represents a negligible amount of work once the distribution of best-found is known.

The quantity $\mathcal{C}^*$ is in effect a quantitative measure of the effectiveness of the algorithm. Of course, $\mathcal{C}^*$ depends on the amount of computation resources allotted, that is, τ. As τ increases, k increases (in jumps of unity), and $\mathcal{C}^*$ decreases. The broader the distribution of best-found, the faster the decrease of $\mathcal{C}^*$ and the more useful it is to perform multiple runs.

To establish the ranking, simply order the algorithms according to their $\mathcal{C}^*$. In general, this ranking may depend on τ, and clearly it is sensitive to the lower tail of the distribution of best-found. Let us illustrate this by considering, for instance, two heuristics H_1 and H_2 having two overlapping distributions for best-found, with averages satisfying $\lceil \mathcal{C}_{H_1} \rceil < \lceil \mathcal{C}_{H_2} \rceil$. In the mean, H_1 seems better than H_2, but if H_2 is significantly faster, and if the tail of its distribution extends well into the domain of $\mathcal{C}_{H_1}$, then one can have $\mathcal{C}^*_{H_2} < \mathcal{C}^*_{H_1}$. H_2 may then be the more effective algorithm, assuming, of course, that τ is large enough so that indeed H_2 can be run multiple times. Some general properties may be derived assuming, for instance, that $\mathcal{C}_{H_1}$ and $\mathcal{C}_{H_2}$ are described by the same distribution but are shifted with respect to one another. Then, if the tail of the distribution falls off as an exponential or faster, H_2 will *not* become more effective than H_1 as $\tau \to \infty$.

Ranking on an ensemble of graphs. The extension of this ranking to an ensemble of graphs is straightforward. Assume that $\mathcal{C}^*$ is known for each graph G and for each heuristic. $\mathcal{C}^*$ is a (real number) measure of the effectiveness of the heuristic on that graph, given an amount of computation time τ. We can then generalize this measure from one graph to an ensemble of graphs by considering $\langle \mathcal{C}^* \rangle$, the mean of $\mathcal{C}^*$ over the relevant ensemble. The final ranking is then simply given by the ordering of the algorithms according to their mean effectiveness.

Our expectation is that in a relatively homogeneous ensemble, the effectiveness (and thus the ranking) will be nearly the same for essentially all sufficiently large graphs and so the average behavior is also the typical behavior. We can expect this to happen whenever the distribution of cut sizes associated with the different heuristics does not overlap too much and has the same pattern regardless of the graph. This is what occurs in the case of our ensemble of random graphs: indeed, we saw that each algorithm leads to a fixed percentage excess cost at large N and that the distribution of costs is peaked. Then two algorithms have nonoverlapping distributions as $N \to \infty$ (unless they give rise to the same percentage excess). It is then clear that at large N,

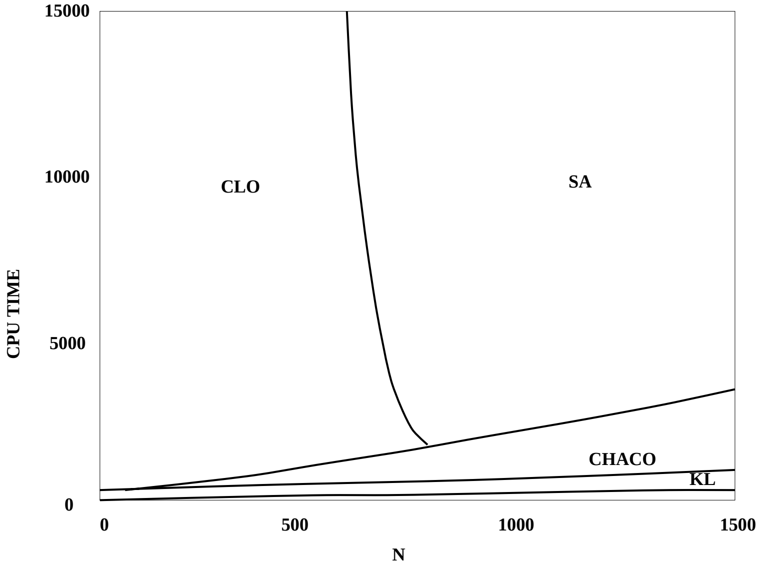


FIG. 8.1. *Ranking diagram.*

the mean ranking is the same as the typical ranking. It is also clear that increasing the amount of computer resources (τ and thus k) or speeding up an algorithm while keeping the quality of its solutions the same does very little to improve its ranking.

Illustration. For each value of N and τ, we can follow the procedure just given to obtain $\mathcal{C}^*$ for the different heuristics of interest for any given graph G and repeat this for many graphs in $G(N,p)$. There are, however, a number of possible speed-ups in our case because of the statistical properties derived in the previous sections. First, although in principle the best-of-k construction has to be repeated for each graph, the results of section 7 provide a short cut. Since the distribution for best-found is (to high accuracy) Gaussian, it is possible to map the mean of best-found to that of best-of-k once and for all: the mapping is just a shift by a k-dependent number of standard deviations. Second, noting that at fixed N, the variance of this Gaussian as well as the speed of the algorithm is essentially constant from graph to graph, we can calculate $\langle \mathcal{C}^* \rangle$ (the average over graphs) in terms of: (i) the CPU time necessary to find one best-found; (ii) the mean cut size, $\langle X(i) \rangle$; (iii) the variance of the intragraph cut sizes, $\lceil Y^2(i,m) \rceil$, which is graph independent at large N. These quantities were measured for a number of values of N, and then fits were performed to interpolate to arbitrary values of N. From these fits, it is possible to compute analytically the values of $\langle \mathcal{C}^* \rangle$ for any values of N and τ, and in particular the "winning" algorithm (the first in our ranking). From this, define regions in (N, τ) space where a given heuristic is the winner, leading to a "diagram," as in Figure 8.1.

In our construction of this diagram, we have included JSA in our ranking but not FSA, KSA, or ASA. This is because, for our choice of parameters, all of the SA algorithms tested give solutions very similar in quality, but JSA is slightly faster. Although the *effectiveness* of all these SA algorithms are nearly identical, their ranking depends on N and τ because of the discrete jumps in k. (Whenever one algorithm increases its k before the others, it may change its ranking.) In the diagram of Figure 8.1, we have labeled the different regions according to the associated "winner," and have indicated the boundaries separating them. (Again, because of the discrete nature of k, we have smoothed these curves.) The labeling "SA" in fact corresponds to JSA. The CPU time is expressed in multiples of CPU cycles. To give these units a machine-independent and less technical meaning, it is enough to say that the lower boundary of the CHACO region corresponds to the time CHACO needs to run once.

From this diagram, we see that at large N, given enough CPU time, the best algorithm is SA, simply because its mean excess cost is lower than that of the other algorithms. In this limit, the distributions for the cut sizes overlap very little, so the ranking is relatively insensitive to the algorithm's speed: using multiple random starts does *very* little to improve the quality of the solutions found as fluctuations about the mean become negligible. At smaller values of N, the fluctuations arising from different random starts are not negligible, so faster algorithms can outperform SA by using the best of k runs. If we compare KL, CHACO, and CLO, we see that CLO is a bit slower but leads to substantially better solutions, and so is the winner if the amount of CPU time is enough for it to run. The other algorithms are competitive only if neither CLO nor SA can terminate a run. This explains why the KL region is nearly invisible, squeezed under the CHACO region, itself below the CLO and SA regions. (Note: (i) On our random graphs, CHACO is *slower* than KL; (ii) the initial partition is set deterministically within the default settings of CHACO, so that its best-found and best-of-k values are identical.)

9. Discussion and conclusions. We have studied the *statistics* of cut sizes generated by graph partitioning heuristics, both within a given graph and over an ensemble of graphs. Motivated by a statistical physics analogy and by what happens for random partitions (section 3), we obtained strong numerical evidence that the cut sizes generated on sparse random graphs are self-averaging, that is, that their distribution becomes peaked as the number of vertices N becomes large. (Quantitatively, this simply means that the *relative* fluctuations about the mean tend to zero as $N \to \infty$.) For the mean cut size, we found a linear dependence on N, indicating that each heuristic leads to a fixed percentage excess cut size above the true minimum. We expect analogous properties to hold for all local heuristics applied to any combinatorial optimization problem in which each variable is coupled to just a few others.

We also investigated how the distribution of cut sizes approaches its limiting large N behavior and gave evidence that on typical graphs the distribution of cut sizes generated becomes Gaussian as $N \to \infty$. In that limit, each heuristic is then characterized by a mean cut size (over all graphs) and a variance describing the fluctuations in the cut sizes on any typical graph. This variance seems to scale linearly with N in the large N limit and to be self-averaging also.

The principal motivation for this work was to introduce a method to rank heuristics while taking into account both the quality of the solutions found and the speed of the algorithms. Knowledge of the distribution of cut sizes allows one to establish a meaningful ranking of the heuristics by assuming that the algorithms may be applied to k different random starts, with the best of the k runs giving the final cost. Although this ranking can be done by brute force, we have used the properties just described to demonstrate it on the heuristics in our testbed. At "large" values of N ($N > 700$), the winner is almost always SA. In fact, at large N, the distributions associated with the algorithms we have tested do not overlap significantly, so that the use of multiple runs to explore the tail of the distributions is not effective. For smaller values of N, the faster algorithms are more competitive, and we find that the winner is CLO except when the allotted time is too short to run CLO even once. Since the graph-to-graph fluctuations in the variance of the cut sizes found are small, this ranking "in the mean" is also in almost all cases the ranking on individual graphs; it is thus very robust.

A number of questions remain open. How can one characterize the distribution of $X(i)$, the mean cut size on graph i? To what extent do similar properties hold for heuristics that are manifestly not local? Can the information found help generate better heuristics? Concerning this last question, it is worth pointing out that although SA is a general purpose method, it outperforms the other heuristics that were specifically developed for the GPP. This suggests that some improvements in these methods might be obtainable by suitable modifications.

Acknowledgment. We are indebted to Bruce Hendrickson and Robert Leland for providing us with their software package, Chaco 2.0. We also thank S. W. Otto and N. Sourlas for stimulating discussions. Furthermore, the first author would like to express his gratitude to Professor J. M. Gómez Gómez for his generous hospitality at the Department of Theoretical Physics of the Universidad Complutense de Madrid, where part of this work was accomplished.

REFERENCES

[1] X. Azuma, *Weighted sums of certain dependent random variables*, Tôhoku Math. J., 19 (1967), pp. 357–367.

[2] J. R. Banavar, D. Sherrington, and N. Sourlas, *Graph bipartitioning and statistical mechanics*, J. Phys. A, 20 (1987), pp. L1–L8.

[3] M. Berger and S. Bokhari, *A partitioning strategy for non-uniform problems on multiprocessors*, IEEE Trans. Comput., C-36 (1987), p. 570–580.

[4] J. W. Berry and M. K. Goldberg, *Path optimization for graph partitioning problems*, Discrete Appl. Math., 90 (1999), pp. 27–50.

[5] C. de Dominicis and Y. Goldschmidt, *Replica symmetry breaking in finite connectivity systems: A large connectivity expansion at finite and zero temperature*, J. Phys. A, 22 (1989), pp. L775–L781.

[6] A. Dunlop and B. Kernighan, *A procedure for placement of standard-cell VLSI circuits*, IEEE Trans. Computer-Aided Design, CAD-4, 1 (1985), p. 92–98.

[7] C. Fiduccia and R. Mattheyses, *A linear-time heuristic for improving network partitions*, in Proc. 19th Design Automation Workshop, Las Vegas, IEEE, 1982, p. 175–181.

[8] Y. Fu and P. Anderson, *Application of statistical mechanics to NP-complete problems in combinatorial optimization*, J. Phys. A, 19 (1986), p. 1605–1620.

[9] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, New York, 1979.

[10] B. Hendrickson and R. Leland, *The Chaco user's guide: Version* 2.0, Tech. Report SAND94–2692, Sandia National Labs., Albuquerque, NM, June 1995.

[11] B. Hendrickson and R. Leland, *A multilevel algorithm for partitioning graphs*, in Proc. Supercomputing '95, ACM, New York, 1995.

[12] F. Johannes, *Partitioning of VLSI circuits and systems*, in Proc. 33rd Design Automation Conference, Las Vegas, NV, June 1996.

[13] D. Johnson, C. Aragon, L. McGeoch, and C. Schevon, *Optimization by simulated annealing: An experimental evaluation, Part* I *(graph partitioning)*, Oper. Res., 37 (1989), pp. 865–892.

[14] D. Johnson and L. McGeoch, *The traveling salesman problem: A case study in local optimization*, in Local Search in Combinatorial Optimization, E. Aarts and J. Lenstra, eds., John Wiley, New York, 1996.

[15] G. Karypis and V. Kumar, *A fast and high quality multilevel scheme for partitioning irregular graphs*, SIAM J. Sci. Comput., 20 (1999), pp. 359–392.

[16] S. Kauffman and S. Levin, *Towards a general theory of adaptive walks on rugged landscapes*, J. Theoret. Biol., 128 (1987), pp. 11–45.

[17] B. Kernighan, *Some Graph Partitioning Problems Related to Program Segmentation*, Ph.D. Thesis, Princeton University, Princeton, NJ, 1969.

[18] B. Kernighan and S. Lin, *An efficient heuristic procedure for partitioning graphs*, Bell System Tech. J., 49 (1970), pp. 291–307.

[19] S. Kirkpatrick, C. Gelatt, and M. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.

[20] K. Lang and S. Rao, *Finding near-optimal cuts: An empirical evaluation*, in Proc. Fourth Annual ACM-SIAM Symposium on Discrete Algorithms, SIAM, Philadephia, 1993.

[21] R. Leland and B. Hendrickson, *An empirical study of static load balancing algorithms*, in Scalable High Performance Computing Conference, IEEE Computer Society Press, Los Alamitos, CA, 1994, pp. 682–685.

[22] S. Lin, *Computer solutions of the traveling salesman problem*, Bell System Tech. J., 44 (1965), pp. 2245–2269.

[23] O. Martin, S. W. Otto, and E. W. Felten, *Large-step Markov chains for the TSP incorporating local search heuristics*, Oper. Res. Lett., 11 (1992), pp. 219–224.

[24] O. C. Martin and S. W. Otto, *Partitioning of unstructured meshes for load balancing*, Concurrency: Practice and Experience, 7 (1995), pp. 303–314.

[25] O. C. Martin and S. W. Otto, *Combining simulated annealing with local search heuristics*, Ann. Oper. Res., 63 (1996), pp. 57–75.

[26] M. Mezard, G. Parisi, and M. A. Virasoro, eds., *Spin Glass Theory and Beyond*, World Sci. Lecture Notes Phys. 9, World Scientific, Singapore, 1987.

[27] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[28] P. J. M. van Laarhoven and E. H. L. Aarts, *Statistical cooling: A general approach to combinatorial optimization problems*, Philips J. Res., 40 (1985), pp. 193–226.

[29] P. J. M. van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*, Mathematics and Its Applications, D. Reidel, Dordrecht, the Netherlands, 1987, pp. 39–98.

SIAM J. Optim.
Vol. 10, No. 1, pp. 252–268

BEST APPROXIMATION FROM THE INTERSECTION OF A CLOSED CONVEX SET AND A POLYHEDRON IN HILBERT SPACE, WEAK SLATER CONDITIONS, AND THE STRONG CONICAL HULL INTERSECTION PROPERTY[*]

FRANK DEUTSCH[†], WU LI[‡], AND JOSEPH D. WARD[§]

Abstract. Let X be a (real) Hilbert space, C be a closed convex subset, and $H_i = \{x \in X \mid \langle x, h_i \rangle \leq b_i\}$ $(i = 1, 2, \ldots, m)$ be a finite collection of half-spaces. Under the assumption that $K := C \cap (\cap_1^m H_i)$ is not empty, the problem of characterizing the best approximation from K to any $x \in X$ is considered. The "strong conical hull intersection property" (strong CHIP), which was introduced by us in 1997, is shown to be both necessary and sufficient for the following "perturbation property" to hold: for each $x \in X$, an element $x_0 \in K$ satisfies $x_0 = P_K(x)$ if and only if $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$ for some scalars $\lambda_i \geq 0$ with $\lambda_i[\langle x_0, h_i \rangle - b_i] = 0$ for each i. Here $P_D(z)$ denotes the unique best approximation from D to z. In other words, determining the best approximation from the set K to any point is equivalent to the (generally easier) problem of determining the best approximation from the set C to a perturbation of that point. Moreover, even when the strong CHIP does not hold, the perturbation property still holds, except now C must be replaced by a certain convex extremal subset of C. We also show that the strong CHIP is weaker than any of the weak Slater conditions that one can naturally impose on the sets in question. These results generalize the main results of our 1997 paper [*J. Approx. Theory*, 90, pp. 385–444] and hence those of several other papers as well.

Key words. best constrained approximation, shape-preserving interpolation, strong conical hull intersection property, strong CHIP, cones, dual cones, normal cones, duality, weak Slater conditions, Hilbert space

AMS subject classifications. Primary, 41A65; Secondary, 41A29

PII. S1052623498337273

1. Introduction.

During the last several years, there has been much interest centered on the following constrained approximation problem: in a Hilbert space X, find the best approximation $P_K(x)$ to any $x \in X$ from the set

$$K := C \cap A^{-1}(b) = C \cap \{x \in X \mid Ax = b\},$$

where C is a closed convex subset of X, A is a bounded linear operator from X into a finite-dimensional Hilbert space Y, and $b \in Y$. One of the main reasons for this interest is that this problem contains as a special case the "shape-preserving interpolation" problem that arises in curve and surface fitting (see [3]). The (sometimes implicit) goal of several papers (see, e.g., [5, 6, 11, 12, 15, 16]) was to determine the pairs of sets $\{C, A^{-1}(b)\}$ for which the following "perturbation property" holds: for every $x \in X$, there exists $y \in Y$ so that

$$P_K(x) = P_C(x + A^*y).$$

[*] Received by the editors April 13, 1998; accepted for publication (in revised form) October 9, 1998; published electronically November 29, 1999.

http://www.siam.org/journals/siopt/10-1/33727.html

[†] Department of Mathematics, The Pennsylvania State University, University Park, PA 16802 (deutsch@math.psu.edu).

[‡] Department of Mathematics, Old Dominion University, Norfolk, VA 23529 (wuli@math.odu.edu).

[§] Department of Mathematics, Texas A&M University, College Station, TX 77843 (jward@math.tamu.edu). The research of this author was supported by NSF grant DMS-9505460 and AFOSR grant F49620-95-1-0194.

In other words, when can determining the best approximation to x from K be replaced by the problem of determining the best approximation to a perturbation $x + A^*y$ of x from the set C? In the applications, it is generally easier to compute best approximations from C than from the whole intersection K. It was seen in [11] that this holds if and only if the sets $\{C, A^{-1}(b)\}$ satisfy the "strong conical hull intersection property" (strong CHIP). In particular, the above perturbation property was seen to hold whenever b was in the relative interior of $A(C)$ (see [6], [11]).

The main thrust of this paper is to treat the more general problem of giving an intrinsic characterization of those pairs of sets $\big\{C, \{x \mid Ax \leq b\}\big\}$ for which an analogous perturbation property holds. Again, the strong CHIP turns out to be the characterizing property. In fact, if A is defined on X by $Ax := (\langle x, h_1, \rangle, \langle x, h_2\rangle, \ldots, \langle x, h_m\rangle)$ for some given $h_i \in X \setminus \{0\}, b \in \mathbb{R}^m$, and $K = C \cap \{x \in X \mid Ax \leq b\}$, then the strong CHIP is a necessary and sufficient condition to ensure the following "perturbation property": for each $x \in X$, an element $x_0 \in K$ satisfies $x_0 = P_K(x)$ if and only if $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$ for some scalars $\lambda_i \geq 0$ with $\lambda_i[\langle x_0, h_i\rangle - b_i] = 0$ for all i (see Corollary 3.3). The merits of such a characterization theorem are mainly that it is generally easier to compute best approximants from C than from K. Also, in many applications, computing $P_K(x)$ is intrinsically an infinite-dimensional problem. However, the computation of $P_C(x - \sum_1^m \lambda_i h_i)$ involves only a *finite* number of parameters λ_i.

The present results certainly generalize previous results since polyhedral sets of the form $A^{-1}(b)$ can be expressed as the intersection of half-spaces. Despite the similarity of the characterization theorem with past results, there are striking differences. Most notable, perhaps, is that unlike the "flat" case, the present characterization theorem holds in certain cases when $b \notin A(C)$! Furthermore, a key technical result (Theorem 2.7) gives sufficient conditions for $D + \text{con}F$ to be closed, where D is a closed convex cone and F is a finite subset of the Hilbert space. The corresponding result with $\text{con}F$ replaced by a finite-dimensional subspace (see [11, Theorem 3.11]) was much more transparent.

The strong CHIP is defined in section 2, and we also include some useful equivalent reformulations. The main theoretical results are stated in sections 3 and 4. In section 3, the strong CHIP is seen to be equivalent to the aforementioned perturbation property. We also show connections between strong CHIP and the Karush–Kuhn–Tucker and Lagrange multiplier conditions. In section 4, we show that even when the strong CHIP is not present, it still *is* possible to use the perturbation technique, except now one must replace C by a certain convex *extremal subset* of C! In section 5, we show the connection between "weak Slater" conditions and the strong CHIP. In section 6, we show how to recover the results of [11] for the "equality case" (i.e., $K = C \cap A^{-1}(b)$) from the inequality case described here. We also give some examples and applications of the theory described.

The notion of the strong CHIP, introduced by us in [11], has taken on an increasingly important role in optimization theory. For example, in [10] and more recently in [18], it was shown that the strong CHIP is an essential property when establishing duality relations between certain convex optimization problems. In [2], the close relationship between the strong CHIP, "bounded linear regularity," "Jameson's property (G)," and error bounds in convex optimization was studied. Finally, [9] showed that the strong CHIP is *the* fundamental property in general convex optimization theory when the constraint set is an intersection of finitely many closed convex sets.

We conclude the introduction by describing some notation used and stating a useful theorem which characterizes best approximations that will be used many times in this paper. Recall that a subset K of the Hilbert space X is *convex* (resp., a *convex cone*) provided that

$$\lambda K + (1-\lambda)K \subset K \quad \text{for all} \quad 0 \leq \lambda \leq 1$$

(resp., $\rho K \subset K$ and $K + K \subset K$ for all $\rho \geq 0$). For any nonempty subset S of X, the *convex hull* (resp., *conical hull*, *linear hull*) of S, denoted $\operatorname{co}(S)$ (resp., $\operatorname{con}(S)$, $\operatorname{span}(S)$), is the intersection of all convex sets (resp., convex cones, linear subspaces) which contain S. If S is nonempty, the *dual cone* (resp., *orthogonal complement*) of S is the set

$$S^\circ := \{x \in X \mid \langle x, y\rangle \leq 0 \text{ for all } y \in S\}$$

(resp., $S^\perp := \{x \in X \mid \langle x, y\rangle = 0 \text{ for all } y \in S\}$). Note that S° (resp., $S^\perp$) is a nonempty closed convex cone (resp., closed linear subspace). The *closure* (resp., *interior, relative interior, boundary*) of any set S is denoted by $\overline{S}$ (resp., $\operatorname{int} S$, $\operatorname{ri} S$, $\operatorname{bd} S$).

It is well known that if K is a closed convex subset of the Hilbert space X, every $x \in X$ has a unique best approximation $P_K(x)$ in K to x. That is, $P_K(x) \in K$ and

$$\|x - P_K(x)\| = \inf\{\|x - y\| \mid y \in K\}.$$

The following result is also well known.

THEOREM 1.1 (characterization of best approximations). *Let K be a closed convex subset of the Hilbert space X, $x \in X$, and $x_0 \in K$. Then $x_0 = P_K(x)$ if and only if $x - x_0 \in (K - x_0)^\circ$.*

If A is a bounded linear operator from X to Y, then A^*, $\mathcal{R}(A)$, and $\mathcal{N}(A)$ denote its adjoint map, range, and null space, respectively. All other undefined terminology and notation is standard.

2. A sufficient condition for strong CHIP. Throughout this section, C is a closed convex subset of a Hilbert space X, $h_j \in X \setminus \{0\}$, $b_j \in \mathbb{R}$, $H_j := \{x \in X \mid \langle h_j, x\rangle \leq b_j\}$ is a (closed) half-space $(j = 1, 2, \ldots, m)$, $K := C \cap (\cap_1^m H_j) \neq \emptyset$, and A is the linear mapping $A : X \to \mathbb{R}^m$ defined by

$$Ax := (\langle h_1, x\rangle, \ldots, \langle h_m, x\rangle).$$

In this notation, we can write K in the form

$$K = C \cap \{x \in X \mid Ax \leq b\}.$$

That is, K is the solution set of the following constrained linear inequalities:

$$\langle h_1, x\rangle \leq b_1, \quad \langle h_2, x\rangle \leq b_2, \ldots, \langle h_m, x\rangle \leq b_m, \quad x \in C. \tag{2.1}$$

For any index subset J of $\{1, 2, \ldots, m\}$, we use $A_J x$ (resp., b_J) to denote the vector obtained by deleting the components of Ax (resp., b) whose indices are not in J. For example, if $J = \{3, 7\}$, then $A_J x = (\langle h_3, x\rangle, \langle h_7, x\rangle)$ (resp., $b_J = (b_3, b_7)$). We first give some known results about the strong CHIP and then establish a new sufficient condition for guaranteeing when $\{C, H_1, H_2, \ldots, H_m\}$ has the strong CHIP.

Definition 2.1. *Let $\{C_0, \ldots, C_m\}$ be a collection of closed convex sets and let S be a nonempty subset of $\cap_0^m C_j$. Then $\{C_0, \ldots, C_m\}$ has the* strong CHIP relative to S *if, for every $x \in S$,*

$$\left(\bigcap_0^m C_j - x\right)^\circ = \sum_0^m (C_j - x)^\circ. \tag{2.2}$$

In addition, we say $\{C_0, C_1, \ldots, C_m\}$ has the strong CHIP if it has the strong CHIP relative to the whole intersection $\cap_0^m C_i$, that is, if (2.2) *holds for every $x \in \cap_0^m C_i$.*

Remarks. 1. The notion of the strong CHIP (relative to the whole intersection $\cap_0^m C_i$) was first introduced in [11]. This, in turn, was a strengthening of CHIP introduced in [5].

2. The motivation for the name "CHIP" is provided in the equivalence of statements 1 and 4 in Lemma 2.2 below. Namely, the closed conical operation commutes with intersections.

The *indicator function* of a set S in X, denoted I_S, is defined by $I_S(x) = 0$ if $x \in S$ and by $I_S(x) = \infty$ if $x \notin S$. The *subdifferential* of a function $f : X \to \mathbb{R} \cup \{\infty\}$, denoted ∂f, is the set-valued mapping defined on the domain of f by

$$\partial f(x) := \{z \in X \mid f(x) + \langle z, y - x\rangle \le f(y) \text{ for every } y \in X\}.$$

It is well known and easy to verify that $\partial I_S(x) = (S - x)^\circ$ for every $x \in S$. This set is also called the *normal cone* to S at x.

Lemma 2.2. *For given $S \subset \cap_0^m C_j$, the following statements are equivalent:*

1. *$\{C_0, \ldots, C_m\}$ has the strong CHIP with respect to S.*

2. *For each $x \in S$,*

$$\left[\bigcap_0^m C_j - x\right]^\circ \subset \sum_0^m (C_j - x)^\circ.$$

3. *For each $x \in S$,*

$$\partial\left(\sum_0^m I_{C_j}\right)(x) = \sum_0^m \partial I_{C_j}(x).$$

4. *For each $x \in S$,*

$$\overline{\text{con}}\left(\bigcap_0^m C_j - x\right) = \bigcap_0^m \overline{\text{con}}\,(C_j - x)$$

and

$$\sum_0^m (C_j - x)^\circ \text{ is closed.} \tag{2.3}$$

5. *For each $x \in S$,*

$$\overline{\text{con}}\left(\bigcap_0^m C_j - x\right) \supset \bigcap_0^m \overline{\text{con}}\,(C_j - x)$$

and (2.3) *holds.*

Proof. Since the above lemma is true if S is a singleton (cf. Lemma 2.4 in [11]), it is also true for any subset S of C. □

In general, if some type of weak Slater condition is satisfied, then the strong CHIP holds as shown by the following proposition.

PROPOSITION 2.3. *Suppose that either*

1. $C \cap \left[\operatorname{int} \left(\bigcap_1^m C_j \right) \right] \neq \emptyset$, *or*
2. $(\operatorname{int} C) \cap \left(\bigcap_1^m C_j \right) \neq \emptyset$ *and* $C_1, \dots, C_m$ *are polyhedral sets.*

Then $\{C, C_1, \dots, C_m\}$ *has the strong CHIP. Moreover, if* $\dim X < \infty$*, then* $\operatorname{int} C$ *may be replaced by* $\operatorname{ri} C$ *in part* 2.

Proof. If either part 1 or part 2 holds, then [1, Corollary 2.5, p. 113] implies that

$$\partial\left(I_C + I_{\bigcap_1^m C_i} \right) = \partial I_C + \partial I_{\bigcap_1^m C_i}.$$

When statement 1 (resp., 2) holds, then [1, Corollary 2.5, p. 113] (resp., [17, Corollary 23.8.1, p. 223]) implies that $\partial I_{\cap_1^m C_i} = \sum_1^m \partial I_{C_i}$. Thus, in either case, we have

$$\partial\left(I_C + \sum_1^m I_{C_i} \right) = \partial(I_C + I_{\cap_1^m C_i}) = \partial I_C + \sum_1^m \partial I_{C_i}.$$

By Lemma 2.2, $\{C, C_1, \dots, C_m\}$ has the strong CHIP. The last sentence in the proposition follows by [17, Corollary 23.8.1, p. 223]. □

Remark. The above proposition implies that $\{C_0, C_1, \dots, C_m\}$ has the strong CHIP if

$$\operatorname{int}(\cap_0^m C_i) \neq \emptyset,$$

and this was proved in [10].

The following theorem was implicitly proved in [11]. In fact, it follows from Lemma 3.10, Theorem 3.11, and Lemmas 3.8 and 3.1 of [11] (cf. the proof of Theorem 3.12 in [11]).

THEOREM 2.4. *If* $b \in \operatorname{ri} A(C)$*, then* $\{C, \operatorname{bd} H_1, \operatorname{bd} H_2, \dots, \operatorname{bd} H_m\}$ *has the strong CHIP. Moreover,* $(C - x)^\circ \cap \operatorname{span}\{h_1, \dots, h_m\}$ *is a subspace for every* $x \in C \cap (\bigcap_1^m \operatorname{bd} H_j)$.

In general, we cannot even expect $b \in A(C)$ for (2.1). Therefore, we have to modify the condition $b \in \operatorname{ri} A(C)$ to obtain a sufficient condition for $\{C, H_1, \dots, H_m\}$ to have the strong CHIP.

To introduce the new condition for $\{C, H_1, \dots, H_m\}$ to have the strong CHIP, we need the following notations. For $x \in K$, let the set of *active indices* for x be defined by

$$I(x) := \{ j \in \{1, 2, \dots, m\} \mid x \in \operatorname{bd} H_j \} = \{ j \in \{1, 2, \dots, m\} \mid \langle x, h_j \rangle = b_j \},$$

and let $\bar{I}$ be the smallest active index set $I(x)$ among x in K; that is,

$$\bar{I} := \bigcap_{x \in K} I(x).$$

Note that $I\left(\frac{1}{2}(x+y)\right) = I(x) \cap I(y)$ for $x, y \in K$. Therefore, there exists an element $\bar{x} \in K$ such that $I(\bar{x}) = \bar{I}$; that is,

$$A_{\bar{I}}\bar{x} = b_{\bar{I}} \quad \text{and} \quad (A\bar{x})_i < b_i \quad \text{for } i \notin \bar{I}. \tag{2.4}$$

(One can start with $J_1 = I(x_1)$ for any $x_1 \in K$. If $J_1 \neq \bar{I}$, then there is an $x \in K$ such that $J_1 \cap I(x) \neq J_1$. Let $x_2 = 0.5(x_1 + x)$. Then $J_2 := I(x_2) = J_1 \cap I(x)$ is a proper subset of J_1. This reduction procedure can be done only finitely many times so one will get $\bar{I}$ after a finite number of reductions.)

Note that if $\bar{I}$ is empty, then $A\bar{x} < b$ and

$$\bar{x} \in C \cap \left(\operatorname{int} \bigcap_{1}^{m} H_j\right).$$

By Proposition 2.3, $\{C, H_1, \ldots, H_m\}$ has the strong CHIP. Therefore, for the remainder of this section, we will assume

$$\bar{I} \text{ is not empty.} \tag{2.5}$$

It follows that $b_{\bar{I}} \in A_{\bar{I}}(C)$. It is interesting that the condition $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$, whose importance will be seen in Theorem 2.8, can be described *without* using the common active index set $\bar{I}$ and is a weak Slater condition for (2.1)! See section 5 for details.

Next we will establish our main result in the section. Namely, if $b_{\bar{I}}$ is in the relative interior of $A_{\bar{I}}(C)$, then $\{C, H_1, \ldots, H_m\}$ has the strong CHIP. The proof of this result is based on the following two identities:

$$(K - x)^\circ = \left((C - x) \cap \left[\bigcap_{i \in \bar{I}} (H_i - x)\right]\right)^\circ + \sum_{i \notin \bar{I}} (H_i - x)^\circ \quad \text{for } x \in K \tag{2.6}$$

and

$$\left((C - x) \cap \left[\bigcap_{i \in \bar{I}} (H_i - x)\right]\right)^\circ = (C - x)^\circ + \sum_{i \in \bar{I}} (H_i - x)^\circ \quad \text{for } x \in K. \tag{2.7}$$

The first identity follows from Proposition 2.3, and the proof of the second identity needs the following three lemmas.

Lemma 2.5. *Suppose that $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$. Then for every $x \in K$,*

$$\overline{\operatorname{con}}\,(C - x) \cap \left(\bigcap_{i \in \bar{I}} \overline{\operatorname{con}}\,(H_i - x)\right) = \overline{\operatorname{con}}\left[(C - x) \cap \left(\bigcap_{i \in \bar{I}} (H_i - x)\right)\right]. \tag{2.8}$$

Proof. It is obvious that

$$\overline{\operatorname{con}}\left[(C - x) \cap \left(\bigcap_{i \in \bar{I}} (H_i - x)\right)\right] \subset \overline{\operatorname{con}}\,(C - x) \cap \left(\bigcap_{i \in \bar{I}} \overline{\operatorname{con}}\,(H_i - x)\right).$$

On the other hand, let $u \in \overline{\operatorname{con}}\,(C - x) \cap \left(\bigcap_{i \in \bar{I}} \overline{\operatorname{con}}\,(H_i - x)\right)$. By the definition, there exist elements $x_k \in C$ and scalars $\alpha_k \geq 0$ such that $\alpha_k(x_k - x)$ converges to u. Since

$$u \in \bigcap_{i \in \bar{I}} \overline{\operatorname{con}}\,(H_i - x) = \bigcap_{i \in \bar{I}} \operatorname{con}(H_i - x) = \operatorname{con}\left(\bigcap_{i \in \bar{I}} H_i - x\right),$$

there exist $\hat{x} \in \bigcap_{i\in\bar{I}} H_i$ and $\rho \geq 0$ such that $u = \rho(\hat{x} - x)$. Since $\bar{I}$ is the common active index set for elements in K, $A_{\bar{I}}x = b_{\bar{I}}$. Therefore,

$$A_{\bar{I}}u = \rho A_{\bar{I}}(\hat{x} - x) = \rho[A_{\bar{I}}(\hat{x}) - b_{\bar{I}}] \leq 0. \tag{2.9}$$

Note that

$$A_{\bar{I}}u = \lim_{k\to\infty} \alpha_k A_{\bar{I}}(x_k - x) = \lim_{k\to\infty} \alpha_k (A_{\bar{I}}x_k - b_{\bar{I}}) \in \operatorname{span}[A_{\bar{I}}(C) - b_{\bar{I}}].$$

Since $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$, there exist a scalar $\epsilon > 0$ and an element $x^\epsilon \in C$ such that $b_{\bar{I}} + \epsilon A_{\bar{I}}u = A_{\bar{I}}x^\epsilon$. By (2.9), we have

$$A_{\bar{I}}x^\epsilon \leq b_{\bar{I}}. \tag{2.10}$$

Choose $\bar{x}$ in K such that (2.4) holds and choose $\delta > 0$ (with $\delta \leq 1$) small enough so that

$$(A[(1-\delta)\bar{x} + \delta x^\epsilon])_i < b_i \quad \text{for } i \notin \bar{I}. \tag{2.11}$$

Then it follows from (2.10) and (2.11) that

$$A[(1-\delta)\bar{x} + \delta x^\epsilon] \leq b.$$

Since $(1-\delta)\bar{x} + \delta x^\epsilon \in C$, the above inequality implies

$$A_{\bar{I}}[(1-\delta)\bar{x} + \delta x^\epsilon] = b_{\bar{I}}$$

or

$$(1-\delta)b_{\bar{I}} + \delta(b_{\bar{I}} + \epsilon A_{\bar{I}}u) = b_{\bar{I}}.$$

Thus, $A_{\bar{I}}u = 0$ and $u \in \bigcap_{i\in\bar{I}} \operatorname{bd}(H_i - x)$. As a consequence,

$$\begin{aligned} u \in \overline{\operatorname{con}}\,(C - x) \cap \left(\bigcap_{i\in\bar{I}} \operatorname{bd}(H_i - x)\right) &= \overline{\operatorname{con}}\left[(C - x) \cap \left(\bigcap_{i\in\bar{I}} \operatorname{bd}(H_i - x)\right)\right] \\ \subset \overline{\operatorname{con}}\left[(C - x) \cap \left(\bigcap_{i\in\bar{I}} (H_i - x)\right)\right], \end{aligned}$$

where the equality follows from Lemma 3.8 in [11]. This proves (2.8). □

Remark. The Robinson–Ursescu theorem cited in [11] should have been in a Banach space setting (instead of in normed linear spaces, as stated in Theorem 3.7 of [11]), and in the proof of Lemma 3.8 in [11], W should have been defined as the *closure* of $\operatorname{span}(C - x)$ instead of $\operatorname{span}(C - x)$. (We are indebted to Heinz Bauschke for pointing this out.)

LEMMA 2.6. *Let F be a finite subset of X consisting of N elements, and let $z \in \operatorname{con}(F) \setminus \{0\}$. Then any $f \in \operatorname{con}(F)$ can be written as*

$$f = \rho z + f',$$

where $\rho \geq 0$, $f' \in \operatorname{con}(F')$, and F' is a subset of F consisting of at most $N - 1$ elements.

Proof. Letting $F = \{f_1, \ldots, f_N\}$, we see that $z = \sum_{j=1}^{N} \gamma_j f_j$ for some $\gamma_j \geq 0$. Since $z \neq 0$, we may assume some $\gamma_j > 0$. Let $f \in \operatorname{con}(F)$. Then $f = \sum_{j=1}^{N} \lambda_j f_j$ for some $\lambda_j \geq 0$.

Case 1. If $\lambda_j = 0$ for some j, the conclusion of the lemma clearly holds with $\rho = 0$.

Case 2. If $\lambda_j > 0$ for all j, set $\rho := \min\left\{\frac{\lambda_j}{\gamma_j} \mid \gamma_j > 0\right\}$. Then

$$f - \rho z = \sum_{j=1}^{N} (\lambda_j - \rho\gamma_j) f_j.$$

It is easily checked that the coefficients of f_j are all nonnegative with at least one coefficient equal to 0. Setting $f' = \sum_{j=1}^{N} (\lambda_j - \rho\gamma_j) f_j$ verifies the lemma. □

THEOREM 2.7. *If D is a closed convex cone in X, Y is a finite-dimensional subspace of X such that $D \cap Y$ is a subspace, and F is a finite subset of Y, then $D + \operatorname{con}(F)$ is closed.*

Proof. Let $x_n \in D + \operatorname{con}(F)$ and $x_n \to x$. It suffices to show that $x \in D + \operatorname{con}(F)$. We can write $x_n = d_n + f_n$, where $d_n \in D$ and $f_n \in \operatorname{con}(F)$. We use that well-known fact that a finitely generated cone, hence $\operatorname{con}(F)$, is closed (see [13, p. 130]). We consider two cases.

Case 1. $D \cap \operatorname{con}(F) = \{0\}$.

If $\{f_n\}$ has no bounded subsequence, then $\|f_n\| \to \infty$. By passing to a subsequence, we may assume that the bounded sequence $\{f_n/\|f_n\|\}$ in the finite-dimensional space $\operatorname{con}(F)$ converges: $f_n/\|f_n\| \to f \in \operatorname{con}(F)$, and thus $\|f\| = 1$. Then

$$\frac{d_n}{\|f_n\|} = \frac{x_n}{\|f_n\|} - \frac{f_n}{\|f_n\|} \to 0 - f \in -\operatorname{con}(F) \subset Y.$$

But $d_n/\|f_n\| \in D$ for every n and thus $-f \in D \cap Y$. Since $D \cap Y$ is a subspace, $f \in D \cap Y$ and hence $f \in D \cap \operatorname{con}(F) = \{0\}$, which contradicts $\|f\| = 1$.

Thus we may assume that $\{f_n\}$ has a bounded subsequence. By passing to a further subsequence if necessary, we may assume that $f_n \to f \in \operatorname{con}(F)$. Then $d_n = x_n - f_n \to x - f$. Since D is closed, $d := x - f \in D$ and $x = d + f \in D + \operatorname{con}(F)$.

Case 2. $D \cap \operatorname{con}(F) \neq \{0\}$.

Choose any $d \in D \cap \operatorname{con}(F) \setminus \{0\}$ and use Lemma 2.6 to obtain that $f_n = \rho_n d + \tilde{f}_n$, where $\rho_n \geq 0$, $\tilde{f}_n \in \operatorname{con}(F_n)$, and each $F_n \subset F$ contains at most $N - 1$ elements of F, where N is the cardinality of F. Then we see that

$$x_n = d_n + f_n = \tilde{d}_n + \tilde{f}_n, \quad \text{where} \quad \tilde{d}_n := d_n + \rho_n d \in D.$$

Further, by passing to a subsequence, we can assume that all the sets F_n are the same, say, $F_n = F_1$, where F_1 contains at most $N - 1$ elements of F. After $k \leq N$ repeated applications of this procedure, we either end up with a representation of (a subsequence of) x_n in the form $x_n = d_n' + f_n'$, where $d_n' \in D$ and $f_n' \in \operatorname{con}(F')$, where F' is a subset of F and $D \cap \operatorname{con}(F') = \{0\}$, or we end up (after N steps) with $x_n = d_n' \in D$ for every n. In the former case, we deduce by case 1 that $x \in D + \operatorname{con}(F)$. In the latter case, we see that $x \in D \subset D + \operatorname{con}(F)$. □

Remarks. 1. In the particular case when X is finite-dimensional, Theorem 2.7 is known (see, e.g., [17, Theorem 20.3, p. 183]).

2. Let $Y = \operatorname{span}\{f_1, f_2, \ldots, f_N\}$ and $F = \{f_1, f_2, \ldots, f_N, -f_1, -f_2, \ldots, -f_N\}$. Then Theorem 2.7 implies that $D + Y$ is closed if $D \cap Y$ is a subspace. This particular result was proved in [11, Theorem 3.11].

Theorem 2.8. *If $b_{\bar I} \in \operatorname{ri} A_{\bar I}(C)$, then $\{C, H_1, \dots, H_m\}$ has the strong CHIP.*

Proof. Fix any $x \in K$. Since $\left[C \cap \left(\bigcap_{i\in\bar I} H_i\right)\right] \cap \left[\bigcap_{i\notin\bar I} \operatorname{int} H_i\right] \neq \emptyset$ (see (2.4)), Proposition 2.3 implies that (2.6) holds. Applying Theorem 2.4 to $\{b_{\bar I}, A_{\bar I}\}$, we get that $(C-x)^\circ \cap \operatorname{span}\{h_i \mid i \in \bar I\}$ is a subspace. From Theorem 2.7 we obtain that

$$(C-x)^\circ + \sum_{i\in\bar I} (H_i - x)^\circ = (C-x)^\circ + \operatorname{con}\{h_i \mid i \in \bar I\}$$

is closed. This, along with Lemma 2.5, implies (2.7) (cf. the equivalence of statements 1 and 4 of Lemma 2.2). Therefore, by (2.6) and (2.7), we have

$$\begin{aligned}(K-x)^\circ &= \left((C-x) \cap \left[\bigcap_{i\in\bar I}(H_i - x)\right]\right)^\circ + \sum_{i\notin\bar I}(H_i-x)^\circ \\ &= (C-x)^\circ + \sum_{i\in\bar I}(H_i-x)^\circ + \sum_{i\notin\bar I}(H_i-x)^\circ \quad \text{for } x \in K.\end{aligned}$$

That is, $\{C, H_1, \dots, H_m\}$ has the strong CHIP. □

3. Reformulations of the best approximation problem. The following lemma isolates a local condition that is not dependent on strong CHIP but still allows the computation of $P_K(x)$ via a perturbation technique.

Lemma 3.1. *Suppose that the element $x_0 := P_C(x - \sum_1^m \lambda_i h_i)$ is in K for some $\lambda_i \geq 0$ with $\lambda_i = 0$ for each $i \notin I(x_0)$. Then $x_0 = P_K(x)$.*

Proof. We have that $\lambda_i = 0$ for all $i \notin I(x_0)$, so $x_0 = P_C(x - \sum_{i\in I(x_0)} \lambda_i h_i)$ and Theorem 1.1 implies that $x - \sum_{i\in I(x_0)} \lambda_i h_i - x_0 \in (C - x_0)^\circ$. Hence

$$\begin{aligned}x - x_0 \in (C-x_0)^\circ + \sum_{i\in I(x_0)} \lambda_i h_i &\subset (C-x_0)^\circ + \sum_{i\in I(x_0)} \operatorname{con}(h_i) \\ &= (C-x_0)^\circ + \sum_1^m (H_i - x_0)^\circ \subset (K-x_0)^\circ.\end{aligned}$$

Since $x_0 \in K$, Theorem 1.1 implies that $x_0 = P_K(x)$. □

In terms of perturbations of best approximations, we can give an alternate characterization of the strong CHIP. For a vector $z = (z_1, z_2, \dots, z_m) \in \mathbb{R}^m$, we use z_+ to denote the vector whose ith component is $\max\{z_i, 0\}$.

Theorem 3.2. *Let $x_0 \in K$ and $\alpha > 0$. Then the following four statements are equivalent:*

1. *$\{C, H_1, \dots, H_m\}$ has the strong CHIP at x_0.*
2. *For every $x \in X$ with $P_K(x) = x_0$,*

$$P_K(x) = P_C\left(x - \sum_1^m \lambda_i h_i\right) \tag{3.1}$$

for some scalars $\lambda_i \geq 0$ with $\lambda_i = 0$ for all $i \notin I(x_0)$.

3. *For every $x \in X$ with $P_K(x) = x_0$,*

$$\begin{aligned}\left\langle P_C\left(x - \sum_1^m \lambda_i h_i\right), h_j\right\rangle &= b_j \quad \textit{for all} \quad j \in I(x_0), \\ \left\langle P_C\left(x - \sum_1^m \lambda_i h_i\right), h_j\right\rangle &< b_j \quad \textit{for all} \quad j \notin I(x_0)\end{aligned} \tag{3.2}$$

for some scalars $\lambda_i \geq 0$ *with* $\lambda_i = 0$ *for all* $i \notin I(x_0)$*.*

4. *For every* $x \in X$ *with* $P_K(x) = x_0$*,* (3.1) *holds with* $\lambda = (\lambda_1, \ldots, \lambda_m)$ *being a solution of the following nonlinear equation:*

$$\lambda = \left(\lambda + \alpha \left[AP_C\left(x - \sum_1^m \lambda_i h_i\right) - b\right]\right)_+ . \tag{3.3}$$

Moreover, for any set of scalars $\lambda_i \geq 0$ *with* $\lambda_i = 0$ *for all* $i \notin I(x_0)$*,* (3.1) *holds if and only if* (3.2) *holds.*

Proof. Let m scalars λ_i satisfy $\lambda_i \geq 0$ for all i and $\lambda_i = 0$ for all $i \notin I(x_0)$. If (3.1) holds, then it is clear that

$$P_C\left(x - \sum_1^m \lambda_i h_i\right) \in \left(\bigcap_{j \in I(x_0)} \operatorname{bd} H_j\right) \cap \left(\bigcap_{i \notin I(x_0)} \operatorname{int} H_i\right), \tag{3.4}$$

and hence (3.2) holds. Conversely, if (3.2) holds, then (3.4) holds. Setting $x_0 := P_C(x - \sum_1^m \lambda_i h_i)$, we see from (3.4) that $x_0 \in K$. Thus, by Lemma 3.1, $x_0 = P_K(x)$, and (3.1) holds. This proves the equivalence of statements 2 and 3 as well as the last statement of the theorem.

To see that statement 1 implies statement 2, let $\{C, H_1, \ldots, H_m\}$ have the strong CHIP at x_0 and let $x \in X$ satisfy $P_K(x) = x_0$. Then Theorem 1.1 and Lemma 2.2 imply that

$$x - x_0 \in (K - x_0)^\circ \subset (C - x_0)^\circ + \operatorname{con}\{h_j \mid j \in I(x_0)\},$$

so there exist scalars $\lambda_j \geq 0$ for all $j \in I(x_0)$ such that $x - x_0 \in (C - x_0)^\circ + \sum_{j \in I(x_0)} \lambda_j h_j$ or $x - \sum_{j \in I(x_0)} \lambda_j h_j - x_0 \in (C - x_0)^\circ$. By Theorem 1.1, $x_0 = P_C(x - \sum_{j \in I(x_0)} \lambda_j h_j)$. This proves statement 2.

Now assume statement 2 holds and we wish to prove statement 1. Choose any $z \in (K - x_0)^\circ$ and set $x := z + x_0$. Note that $x - x_0 = z \in (K - x_0)^\circ$. By Theorem 1.1, $P_K(x) = x_0$. By statement 2, there exists scalars $\lambda_i \geq 0$ with $\lambda_i = 0$ for all $i \notin I(x_0)$ such that $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$. By Theorem 1.1,

$$z = x - x_0 = \left(x - \sum_1^m \lambda_i h_i - x_0\right) + \sum_1^m \lambda_i h_i \in (C - x_0)^\circ + \operatorname{con}\{h_j \mid j \in I(x_0)\}.$$

Since z was an arbitrary point in $(K - x_0)^\circ$, this shows that

$$(K - x_0)^\circ \subset (C - x_0)^\circ + \operatorname{con}\{h_j \mid j \in I(x_0)\} = (C - x_0)^\circ + \sum_1^m (H_i - x_0)^\circ.$$

It follows by Lemma 2.2 that statement 1 holds.

For $z = (z_1, z_2, \ldots, z_m)$ and $\lambda = (\lambda_1, \lambda_2, \ldots, \lambda_m)$, it is easy to verify that $\lambda = (\lambda + \alpha z)_+$ if and only if

$$z_i \leq 0, \quad \lambda_i \geq 0, \quad \text{and} \quad \lambda_i z_i = 0 \qquad \text{for } 1 \leq i \leq m.$$

Therefore, letting $z = AP_C(x - \sum_1^m \lambda_j h_j) - b$, we get the equivalence of (3.2) and (3.3). This proves the equivalence of statements 3 and 4. □

The next theorem shows that the strong CHIP is the *precise* condition that allows one to replace the problem of determining best approximations to any $x \in X$ from K to that of determining the best approximation to a certain perturbation of x from C.

COROLLARY 3.3. *The following statements are equivalent:*

1. *$\{C, H_1, \dots, H_m\}$ has the strong CHIP.*

2. *For each $x \in X \setminus K$ and an element $x_0 \in K$, $x_0 = P_K(x)$ if and only if $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$ for some scalars $\lambda_i \geq 0$ with $\lambda_i[\langle x_0, h_i\rangle - b_i] = 0$ for all i.*

3. *For each $x \in X \setminus K$, $P_K(x) = P_C(x - \sum_1^m \lambda_i h_i)$, with $\lambda = (\lambda_1, \dots, \lambda_m)$ being a solution of* (3.3).

Proof. $1 \Rightarrow 2$. Suppose $\{C, H_1, \dots, H_m\}$ has the strong CHIP, $x \in X \setminus K$, and $x_0 \in K$. If $x_0 = P_K(x)$, then Theorem 3.2 implies that $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$ for some $\lambda_i \geq 0$ with $\lambda_i = 0$ for all $i \notin I(x_0)$. Hence $\lambda_i[\langle x_0, h_i\rangle - b_i] = 0$ for every i. Conversely, if $x_0 = P_C(x - \sum_1^m \lambda_i h_i)$ for some $\lambda_i \geq 0$ with $\lambda_i[\langle x_0, h_i\rangle - b_i] = 0$ for all i, then $\lambda_i = 0$ for all $i \notin I(x_0)$ and by Lemma 3.1, $x_0 = P_K(x)$. This proves part 2.

$2 \Rightarrow 1$. Assume part 2 holds, which implies Theorem 3.2, part 2 for each $x_0 \in K$. By Theorem 3.2, $\{C, H_1, \dots, H_m\}$ has the strong CHIP at every point in K.

The equivalence of parts 1 and 3 also follows from Theorem 3.2. □

THEOREM 3.4. *If $b_{\bar I} \in \operatorname{ri} A_{\bar I} C$, then, for each $x \in X \setminus K$,*

$$P_K(x) = P_C\left(x - \sum_1^m \lambda_i h_i\right),$$

where $\lambda = (\lambda_1, \dots, \lambda_m)$ is a solution of (3.3).

Proof. This follows immediately from Theorem 2.8 and Corollary 3.3. □

It is worth noting that the nonnegative scalars λ_i which work in the above results are precisely the "Lagrange multipliers" obtained by a *formal* application of the Karush–Kuhn–Tucker or Lagrange multiplier conditions applied to the convex programming problem of minimizing the function

$$f(y) := \frac{1}{2}\|y - x\|^2, \quad y \in X,$$

over the set of all $y \in C$ with $f_i(y) := \langle y, h_i\rangle - b_i \leq 0$ for $i = 1, 2, \dots, m$ (see [20, Theorem 47.E, p. 394]). However, without some kind of Slater or weak Slater condition holding in this situation (e.g., as in [20, p. 394]), such a formal application cannot be rigorously justified. As we will see in section 5, the strong CHIP is implied by all these "weak Slater"–type conditions.

4. Reformulations without strong CHIP. In Corollary 3.3, we saw that *if the sets $\{C, H_1, \dots, H_m\}$ had the strong CHIP*, then it was possible to determine $P_K(x)$ by determining $P_C(x - y)$ for an appropriate y in the conical hull of the h_j's. In this section, we shall show that the same conclusion holds *without* the assumption of the strong CHIP provided we replace C by C_b, a certain prescribed extremal subset of C! In fact, if $\{C, H_1, \dots, H_m\}$ does not satisfy the strong CHIP, then we replace C by a subset C_b such that $b_{\bar I} \in \operatorname{ri} A_{\bar I}(C_b)$, which implies

$$(K - x)^\circ = (C_b - x)^\circ + \sum_1^m (H_i - x)^\circ \quad \text{for } x \in K.$$

Since $\{C, H_1, \dots, H_m\}$ has the strong CHIP if $\bar I$ is empty, we may assume in this section that $\bar I$ is *not* empty.

Definition 4.1. *Let C_b be the smallest closed, convex, extremal subset of C such that*

$$C_b \supset C \cap \left(\bigcap_{j \in \bar{I}} \operatorname{bd} H_j \right).$$

More precisely,

$$C_b := \bigcap \left\{ E \mid E \text{ closed convex extremal in } C, \text{ and } E \supset C \cap \left(\bigcap_{j \in \bar{I}} \operatorname{bd} H_j \right) \right\}.$$

Note that

$$C_b \cap \left(\bigcap_{j \in \bar{I}} \operatorname{bd} H_j \right) = C \cap \left(\bigcap_{j \in \bar{I}} \operatorname{bd} H_j \right).$$

Next let F_b denote the smallest relatively closed convex extremal subset of $A_{\bar{I}}(C)$ which contains $b_{\bar{I}}$, and set

$$C_{F_b} := C \cap A_{\bar{I}}^{-1}(F_b).$$

Then the following two lemmas about $b_{\bar{I}}$, C_b, F_b, and C_{F_b} were given in [11, Proposition 4.3 and Lemma 4.4]. (We should note that the "relatively closed" part of the definition of F_b was not included in [11]. However, it is essential for the proof of Proposition 4.3(2) of [11]. We are indebted to Heinz Bauschke for pointing out this omission.)

Lemma 4.2. *The following statements hold:*

1. $C_{F_b} = C_b$.
2. $A_{\bar{I}}(C_b) = F_b$.
3. $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C_b)$.

Lemma 4.3. $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ *if and only if* $C = C_b$.

The next result shows that the perturbation method *always* works provided we replace C by C_b.

Theorem 4.4. *Fix any $\alpha > 0$. For every $x \in X \setminus K$,*

$$P_K(x) = P_{C_b}\left(x - \sum_1^m \lambda_i h_i \right),$$

where $\lambda = (\lambda_1, \ldots, \lambda_m)$ is a solution of the following nonlinear equation:

$$\lambda = \left(\lambda + \alpha \left[A P_{C_b} \left(x - \sum_1^m \lambda_i h_i \right) - b \right] \right)_+ .$$

Proof. By Lemma 4.2, $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C_b)$. The theorem then follows from Theorem 3.4 (with C replaced by C_b). □

5. Weak Slater conditions.

From the previous two sections, it is clear that $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ is crucial to get various reformulations of the best approximation problem. In this section, we show that $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ (when $\bar{I} \neq \emptyset$) is the natural "weak Slater" condition for the following constrained linear inequalities:

$$\langle h_1, x \rangle \leq b_1, \quad \langle h_2, x \rangle \leq b_2, \ldots, \langle h_m, x \rangle \leq b_m, \quad x \in C. \tag{5.1}$$

(For a discussion of related Slater-type constraint qualifications, see [13, Chapter VII, section 2].) One Slater-type constraint qualification condition for (5.1) is that there exists $\bar{x} \in X$ such that

$$\langle h_1, \bar{x}\rangle < b_1, \quad \langle h_2, \bar{x}\rangle < b_2, \ldots, \langle h_m, \bar{x}\rangle < b_m, \quad \bar{x} \in C, \tag{5.2}$$

which could be called a *constrained strong Slater* condition for (5.1). The topological reformulation of (5.2) is

$$\bar{x} \in C \cap \left(\bigcap_j \operatorname{int} H_j \right) \neq \emptyset. \tag{5.3}$$

Another Slater-type constraint qualification for (5.1) is

$$\operatorname{int} C \cap \left(\bigcap_j H_j \right) \neq \emptyset. \tag{5.4}$$

When C is defined by nonlinear inequalities, (5.4) is the so-called *weak Slater condition* for (5.1). However, a *topological weak Slater condition* is actually weaker than (5.4) and can be defined as follows:

$$\operatorname{ri} C \cap \left(\bigcap_j H_j \right) \neq \emptyset. \tag{5.5}$$

Note that either (5.3) or (5.4) implies the strong CHIP of $\{C, H_1, \ldots, H_m\}$ and, when X is finite-dimensional, (5.5) also implies the strong CHIP of $\{C, H_1, \ldots, H_m\}$ (cf. Proposition 2.3).

Obviously, (5.4) is stronger than (5.5). When X is a finite-dimensional space, one can easily verify that (5.3) also implies (5.5). In fact, we can choose $\hat{x} \in \operatorname{ri} C$ $(\neq \emptyset)$. If (5.3) holds, then for $\theta > 0$ small enough and $x_\theta := \theta\hat{x} + (1-\theta)\bar{x}$, we have

$$x_\theta \in \operatorname{ri} C \cap \left(\bigcap_j \operatorname{int} H_j \right) \neq \emptyset. \tag{5.6}$$

Therefore, (5.5) is the weakest among the Slater-type conditions mentioned above and is sufficient for the strong CHIP of $\{C, H_1, \ldots, H_m\}$ when X is finite-dimensional. However, if X is infinite-dimensional, $\operatorname{ri} C$ might be empty. (For example, let $X = L_2[0,1]$ denote the Hilbert space of all square-integrable functions on $[0,1]$ with inner product $\langle x, y\rangle = \int_0^1 x(t)y(t)\,dt$ and $C = \{x \in X \mid x \geq 0\}$. Then aff $C = X$, so that $\operatorname{ri} C = \operatorname{int} C = \emptyset$.) Thus (5.5) makes no sense in this case.

It is perhaps also worth mentioning here that Borwein and Lewis [4] have defined and studied the “quasi-relative interior” of a convex set which is more general than the relative interior, at least in infinite-dimensional spaces (see Example 3.11(i) in [4]). In the example of the last paragraph, the quasi-relative interior of C (denoted by qri(C)) is not empty. It would be interesting to know whether $\operatorname{qri}(C) \cap (\cap_1^m H_j) \neq \emptyset$ is also a sufficient condition for the strong CHIP of $\{C, H_1, \ldots, H_m\}$.

It turns out that $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ (when $\bar{I} \neq \emptyset$) is equivalent to (5.5) when $\operatorname{ri} C \neq \emptyset$. Moreover, each of the Slater-type constraint qualifications (5.3)–(5.5) always implies

$b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ (when $\bar{I} \neq \emptyset$). *Therefore, $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ (when $\bar{I} \neq \emptyset$) is weaker than the Slater-type conditions mentioned above and is sufficient for the strong CHIP of $\{C, H_1, \ldots, H_m\}$ no matter whether X is infinite-dimensional or finite-dimensional.*

Fact 5.1. If X and Y are Hilbert spaces, C is a closed and convex subset of X, and $B : X \to Y$ is a bounded linear operator with finite-dimensional range, then $x \in \operatorname{ri} C$ implies $Bx \in \operatorname{ri} B(C)$.

To verify this fact, first note that if X is finite-dimensional, then this is a consequence of known results (see, e.g., [17, Theorem 6.6, p. 48]). In general, if $x \in \operatorname{ri} C$, then in particular, for each $y \in C$ there exists $\mu > 1$ such that $\mu x + (1-\mu)y \in C$. Hence $\mu Bx + (1-\mu)By \in B(C)$. Since $B(C)$ is finite-dimensional, it follows (see [17, Theorem 6.4, p. 47] applied to $B(C)$ instead of C) that $Bx \in \operatorname{ri} B(C)$, and Fact 5.1 is proved. In contrast to the case when both X and Y are finite-dimensional, it is false in general that $B(\operatorname{ri} C) = \operatorname{ri} B(C)$. Indeed, as we saw four paragraphs earlier, $\operatorname{ri} C$ may even be empty!

THEOREM 5.1. *Let $H := \bigcap_j H_j$ and $M = \mathcal{N}(A)$, the null space of A. Consider the following statements:*

1. $\operatorname{ri} C \cap H \neq \emptyset$.
2. $\operatorname{ri} P_{M^\perp}(C) \cap H \neq \emptyset$.
3. $\operatorname{ri} A(C) \cap (b - \mathbb{R}^m_+) \neq \emptyset$.
4. *There exists $\bar{x} \in C \cap H$ such that $A\bar{x} \in \operatorname{ri} A(C)$.*
5. $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$ *when* $\bar{I} \neq \emptyset$.
6. $C \cap \operatorname{int} H \neq \emptyset$.

Then $1 \Rightarrow 2 \Leftrightarrow 3 \Leftrightarrow 4 \Leftrightarrow 5 \Leftarrow 6$, and each of the above conditions implies that $\{C, H\}$ has the strong CHIP. Moreover, if $\operatorname{ri} C \neq \emptyset$, then the first five statements are equivalent.

Proof. Since $M = \mathcal{N}(A)$ is the null space of A, then $M^\perp = R(A^*)$ is the range of A^*; hence $M^\perp$ is finite-dimensional.

$1 \Rightarrow 2$. Let $\bar{x} \in \operatorname{ri} C \cap H$. Since $P_{M^\perp}$ is a linear mapping with finite-dimensional range, Fact 5.1 implies that $P_{M^\perp}(\bar{x}) \in \operatorname{ri} P_{M^\perp}(C)$. Since $\bar{x} \in H$, the definition of M implies that $P_{M^\perp}(\bar{x}) = \bar{x} - P_M(\bar{x}) \in H$. This proves $P_{M^\perp}(\bar{x}) \in \operatorname{ri} P_{M^\perp}(C) \cap H$ and statement 2 holds.

$2 \Leftrightarrow 3$. If $x \in M^\perp$ and $Ax = 0$, then $x \in M \cap M^\perp = \{0\}$. Thus, the linear mapping A from $M^\perp$ to $\mathbb{R}^m$ is one-to-one. If a linear mapping L from a finite-dimensional space Y to a finite-dimensional space Z is one-to-one and $y \in Y$, then $y \in S \Leftrightarrow L(y) \in L(S)$ and $y \in \operatorname{ri} S \Leftrightarrow L(y) \in \operatorname{ri} L(S)$. Using this fact, we get

$$\bar{x} \in \operatorname{ri} P_{M^\perp}(C) \cap H \Leftrightarrow A\bar{x} \in \operatorname{ri} A\left[P_{M^\perp}(C)\right] \cap A(H). \tag{5.7}$$

However, for $x \in C$, we have $x = P_M(x) + P_{M^\perp}(x)$ and $Ax = A[P_M(x)] + A[P_{M^\perp}(x)] = A[P_{M^\perp}(x)]$. Thus, $A(C) = A[P_{M^\perp}(C)]$. It is easy to see that $A\bar{x} \in A(H)$ if and only if $A\bar{x} \in b - \mathbb{R}^m_+$. Thus, (5.7) implies the equivalence of statements 2 and 3.

$3 \Leftrightarrow 4$. This is trivially true.

$4 \Rightarrow 5$. Let $\bar{x} \in C \cap H$ be such that $A\bar{x} \in \operatorname{ri} A(C)$. If $\bar{I} = \emptyset$, then there is nothing to prove; otherwise, $A_{\bar{I}}\bar{x} \in \operatorname{ri} A_{\bar{I}}(C)$. Since $\bar{x} \in K = C \cap H$ and $\bar{I}$ is the common active index set for elements in K, we have $A_{\bar{I}}\bar{x} = b_{\bar{I}}$ and, as a consequence, $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$.

$5 \Rightarrow 4$ and $5 \Rightarrow 1$ (when $\operatorname{ri} C \neq \emptyset$). The following fact will be used repeatedly in this proof. (See [17, Theorem 6.1, p. 45] for the proof when the space is finite-dimensional. But the same proof works in the infinite-dimensional case.)

> If S is a closed convex set in a normed linear space and $\bar{x} \in \operatorname{ri} S$,
> then $\theta x + (1-\theta)\bar{x} \in \operatorname{ri} S$ for any $x \in S$ and $0 < \theta < 1$.

Let $\bar{x} \in K$ be such that

$$A_{\bar{I}}\bar{x} = b_{\bar{I}} \quad \text{and} \quad (A\bar{x})_i < b_i \quad \text{for } i \notin \bar{I}. \tag{5.8}$$

Such a point exists by (2.4). Choose

$$\hat{x} \in \operatorname{ri} C \quad \text{if} \quad \operatorname{ri} C \neq \emptyset, \tag{5.9}$$

or choose

$$\hat{x} \in C \text{ with } A\hat{x} \in \operatorname{ri} A(C) \quad \text{if} \quad \operatorname{ri} C = \emptyset. \tag{5.10}$$

By Fact 5.1, (5.9) implies (5.10). We consider two cases.

Case 1. $\bar{I} = \emptyset$.

Then $\bar{x} \in \operatorname{int} H$ and $(A\bar{x})_i < b_i$ for all i. Let $x_\theta := \theta\hat{x} + (1-\theta)\bar{x}$ for $0 < \theta < 1$. Then for $\theta > 0$ small enough, we have $Ax_\theta \leq b$. Thus, $x_\theta \in C \cap H$ and $Ax_\theta \in \operatorname{ri} A(C)$. This proves statement 4. If, in addition, $\operatorname{ri} C \neq \emptyset$, then $x_\theta \in \operatorname{ri} C$ by (5.9). Thus, $x_\theta \in \operatorname{ri} C \cap H$ and statement 1 holds.

Case 2. $\bar{I} \neq \emptyset$ and $b_{\bar{I}} \in \operatorname{ri} A_{\bar{I}}(C)$.

Then for $\hat{x} \in C$, there exists ϵ such that $0 < \epsilon < 1$ and $b_{\bar{I}} - \epsilon(A_{\bar{I}}\hat{x} - b_{\bar{I}}) \in A_{\bar{I}}(C)$. So there is $z \in C$ such that $A_{\bar{I}}z = b_{\bar{I}} - \epsilon(A_{\bar{I}}\hat{x} - b_{\bar{I}})$, or equivalently (cf. (5.8)),

$$A_{\bar{I}}(z - \bar{x}) = -\epsilon A_{\bar{I}}(\hat{x} - \bar{x}). \tag{5.11}$$

Let $0 < \theta < 1/2$. Then

$$x_\theta := \bar{x} + \theta[\epsilon(\hat{x} - \bar{x}) + z - \bar{x}] = (1 - \theta - \epsilon\theta)\bar{x} + \theta\epsilon\hat{x} + \theta z \in C. \tag{5.12}$$

It follows from (5.11) and (5.8) that

$$A_{\bar{I}}x_\theta = A_{\bar{I}}\bar{x} = b_{\bar{I}}. \tag{5.13}$$

By (5.8), if $\theta > 0$ is small enough, we have

$$(Ax_\theta)_j < b_j \quad \text{for } j \notin \bar{I}. \tag{5.14}$$

Thus, if $\theta > 0$ is small enough, we have $x_\theta \in C \cap H$. However, since $A\hat{x} \in \operatorname{ri} A(C)$, we get

$$Ax_\theta = (1 - \theta - \epsilon\theta)A\bar{x} + \theta\epsilon A\hat{x} + \theta Az \in \operatorname{ri} A(C).$$

This proves 5 ⇒ 4. If, in addition, $\operatorname{ri} C \neq \emptyset$, then it follows from (5.12) and (5.9) that $x_\theta \in \operatorname{ri} C$. This proves 5 ⇒ 1.

6 ⇒ 5. This holds vacuously, since in this case $\bar{I} = \emptyset$.

Finally, if any of statements 1–6 is satisfied, then statement 5 is true. By Theorem 2.8, $\{C, H_1, \ldots, H_m\}$ has the strong CHIP. By remark 2 following Proposition 2.3, $\{C, H\}$ has the strong CHIP. □

Remarks. It seems that the weak Slater conditions given in Theorem 5.1 depend on the representation of H or $\{x \in X \mid Ax \leq b\}$. However, Theorem 5.1, part 2 indicates that parts 2–5 in Theorem 5.1 are *intrinsic* conditions for $\{C, \bigcap_j H_j\}$.

6. Applications. We first show that the problem considered in [11] is a special case of the one considered in this paper and how the main results of [11] can be deduced from this fact.

In [11], the set-up was this: Let C be a closed convex subset of the Hilbert space X, $h_i \in X\backslash\{0\} \quad (i = 1, 2, \ldots, m)$, and $b = (b_1, b_2, \ldots, b_m) \in \mathbb{R}^m$. Define $A : X \to \mathbb{R}^m$ by

$$Ax := (\langle x, h_1\rangle, \langle x, h_2\rangle, \ldots, \langle x, h_m\rangle), \quad x \in X,$$

and set $K = C \cap A^{-1}(b)$. In other words,

$$K = C \cap \{x \in X \mid \langle x, h_i\rangle = b_i \quad (i = 1, 2, \ldots, m)\}$$

and K is the intersection of C with m hyperplanes. In [11], the main interest was in *characterizing* best approximations from K. Note that if we define $h_{i+m} = -h_i$ and $b_{i+m} = -b_i$ for $i = 1, 2, \ldots, m$, and if we define $2m$ half-spaces by

$$H_i = \{x \in X \mid \langle x, h_i\rangle \leq b_i\} \quad (i = 1, 2, \ldots, 2m),$$

then $\cap_1^{2m} H_i = A^{-1}(b)$, and we may rewrite K in the form

$$K = C \cap \left(\bigcap_1^{2m} H_i\right).$$

Moreover, owing to the fact that any finite collection of half-spaces has the strong CHIP [11], we see that $\{C, H_1, \ldots, H_{2m}\}$ has the strong CHIP if and only if $\{C, \cap_1^{2m} H_i\}$ has the strong CHIP if and only if $\{C, A^{-1}(b)\}$ has the strong CHIP. Using these facts, it is easy to deduce the following consequence of Corollary 3.3.

Corollary 6.1 (see [11, Theorem 3.2]). *The following statements are equivalent:*

1. $\{C, A^{-1}(b)\}$ *has the strong CHIP.*
2. *For each* $x \in X$,

$$P_K(x) = P_C\left(x - \sum_1^m \alpha_i h_i\right) \quad \big(= P_C(x - A^*\alpha)\big) \tag{6.1}$$

for some scalars $\alpha_i \in \mathbb{R}$.

Moreover, for any scalars α_i *such that* $P_C(x - \sum_1^m \alpha_i h_i) \in K$, (6.1) *must hold.*

In this setting, for every $x_0 \in K$, we have $\langle x_0, h_i\rangle = b_i \quad (i = 1, \ldots, 2m)$. That is, $I(x_0) = \{1, 2, \ldots, 2m\}$ and hence $\bar{I} = \{1, 2, \ldots, 2m\}$. It follows that Definition 4.1 reduces to: let C_b be the smallest closed convex extremal subset of C such that

$$C_b \supset C \cap \left(\bigcap_1^{2m} \operatorname{bd} H_i\right) = K.$$

That is, C_b is defined exactly as in [11]. Also, we can deduce the following consequence of Theorem 4.4 and Lemma 4.3.

Corollary 6.2 (see [11, Theorem 4.5 and Lemma 4.4]). *For each* $x \in X$,

$$P_K(x) = P_{C_b}\left(x - \sum_1^m \alpha_i h_i\right) \quad \big(= P_{C_b}(x - A^*\alpha)\big) \tag{6.2}$$

for some scalars $\alpha_i \in \mathbb{R}$. *Moreover,* $C = C_b$ *if and only if* $b \in \operatorname{ri} A(C)$.

REFERENCES

[1] V. Barbu and Th. Precupanu, *Convexity and Optimization in Banach Spaces*, Sijthoff & Noordhoff, the Netherlands, 1978.

[2] H. H. Bauschke, J. M. Borwein, and W. Li, *Strong conical hull intersection property, bounded linear regularity, Jameson's property (G), and error bounds in convex optimization*, Math. Programming, to appear.

[3] C. de Boor, *On "best" interpolation*, J. Approx. Theory, 16 (1976), pp. 28–48.

[4] J. M. Borwein and A. S. Lewis, *Partially finite convex programming Part* I: *Quasi relative interiors and duality theory*, Math. Programming, 57 (1992), pp. 15–48.

[5] C. K. Chui, F. Deutsch, and J. D. Ward, *Constrained best approximation in Hilbert space*, Constr. Approx., 6 (1990), pp. 35–64.

[6] C. K. Chui, F. Deutsch, and J. D. Ward, *Constrained best approximation in Hilbert space* II*,* J. Approx. Theory, 71 (1992), pp. 231–238.

[7] F. Deutsch, *Interpolation from a convex subset of Hilbert space: A survey of some recent results*, in Approximation Theory, Wavelets and Applications, S. P. Singh, ed., Kluwer Academic Publishers, Dordrecht, the Netherlands, 1995, pp. 95–105.

[8] F. Deutsch, *Dykstra's cyclic projections algorithm: The rate of convergence*, in Approximation Theory, Wavelets and Applications, S. P. Singh, ed., Kluwer Academic Publishers, Dordrecht, the Netherlands, 1995, pp. 87–94.

[9] F. Deutsch, *The role of the strong conical hull intersection property in convex optimization and approximation*, in Approximation Theory IX, Vol. I: Theoretical Aspects, C. K. Chui and L. L. Schumaker, eds., Vanderbilt University Press, Nashville, TN, 1998, pp. 105–112.

[10] F. Deutsch, W. Li, and J. Swetits, *Fenchel duality and the strong conical hull intersection property*, J. Optim. Theory Appl., 102 (1999), pp. 681–695.

[11] F. Deutsch, W. Li, and J. D. Ward, *A dual approach to constrained interpolation from a convex subset of Hilbert space*, J. Approx. Theory, 90 (1997), pp. 385–444.

[12] F. Deutsch, V. Ubhaya, J. Ward, and Y. Xu, *Constrained best approximation in Hilbert space* III: *Applications to n-convex functions*, Constr. Approx., 12 (1996), pp. 361–384.

[13] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* I, Springer-Verlag, New York, 1993.

[14] W. Li, P. M. Pardalos, and C. G. Han, *Gauss-Seidel method for least-distance problems*, Optim. Theory Appl., 75 (1992), pp. 487–500.

[15] C. A. Micchelli, P. W. Smith, J. Swetits, and J. D. Ward, *Constrained L_p-approximation*, Constr. Approx., 1 (1985), pp. 93–102.

[16] C. A. Micchelli and F. I. Utreras, *Smoothing and interpolation in a convex subset of a Hilbert space*, SIAM J. Sci. Statist. Comput., 9 (1988), pp. 728–746.

[17] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[18] I. Singer, *Duality for optimization and best approximation over finite intersections*, Numer. Funct. Anal. Optim., 19 (1998), pp. 903–915.

[19] P. W. Smith and H. Wolkowicz, *A nonlinear equation for linear programming*, Math. Programming, 34 (1986), pp. 235–238.

[20] E. Zeidler, *Nonlinear Functional Analysis and Its Applications* III: *Variational Methods and Optimization*, Springer-Verlag, New York, 1985.

[21] K. Zhao, *Best interpolation with convex constraints*, J. Approx. Theory, 73 (1993), pp. 119–135.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 269–288

FORTIFIED-DESCENT SIMPLICIAL SEARCH METHOD: A GENERAL APPROACH[*]

PAUL TSENG[†]

Abstract. We propose a new simplex-based direct search method for unconstrained minimization of a real-valued function f of n variables. As in other methods of this kind, the intent is to iteratively improve an n-dimensional simplex through certain reflection/expansion/contraction steps. The method has three novel features. First, a user-chosen integer $\bar{m}_k$ specifies the number of "good" vertices to be retained in constructing the initial trial simplices—reflected, then either expanded or contracted—at iteration k. Second, a trial simplex is accepted only when it satisfies the criteria of *fortified descent*, which are stronger than the criterion of strict descent used in most direct search methods. Third, the number of additional function evaluations needed to check a trial reflected/expanded simplex for fortified descent can be controlled. If one of the initial trial simplices satisfies the fortified-descent criteria, it is accepted as the new simplex; otherwise, the simplex is shrunk a fraction of the way toward a best vertex and the process is restarted, etc., until either a trial simplex is accepted or the simplex effectively has shrunk to a single point.

We prove several theoretical properties of the new method. If f is continuously differentiable, bounded below, and uniformly continuous on its lower level set and we choose $\bar{m}_k$ with the same value at all iterations k, then *every* cluster point of the generated sequence of iterates is a stationary point. The same conclusion holds if the function is continuously differentiable, bounded below, and we choose $\bar{m}_k = 1$ at all iterations k.

Key words. unconstrained minimization, direct search, Nelder–Mead method, multidirectional search method

AMS subject classifications. 49M30, 49M37, 90C26, 90C30

PII. S1052623495282857

1. Introduction.

Consider the unconstrained minimization problem

$$\min_{x \in \Re^n} f(x),$$

where f is a continuous function from $\Re^n$ to $\Re$. An interesting class of methods for solving this problem is that of direct search methods, which update the iterate based on only a few function evaluations along linearly independent directions. In contrast to gradient methods, these methods do not use the function values to explicitly construct an approximation to the gradient nor do they necessarily move iterates along gradient directions. As is noted in [25], the direct search methods can be classified into two subclasses: those that modify the search directions at the end of each iteration, as exemplified by the methods of Box [2], Nelder and Mead [15], Powell [17], Rosenbrock [18], and Zangwill [31], and those that use a fixed set of search directions at all iterations, as exemplified by the methods of Box and Wilson [1], Hooke and Jeeves [10], Spendley, Hext, and Himsworth [23], and Dennis and Torczon [6, 25]. Further studies of these and related methods are presented in [3, 4, 7, 8, 24, 26, 29, 30].

1.1. Motivation for the new method.

Three previously proposed direct search methods are based on the intriguing idea of simplicial search, in which an n-dimensional

[*] Received by the editors March 10, 1995; accepted for publication (in revised form) February 8, 1999; published electronically November 29, 1999. This work was supported by National Science Foundation grant CCR-9311621.

http://www.siam.org/journals/siopt/10-1/28285.html

[†] Department of Mathematics, University of Washington, Seattle, WA 98195 (tseng@math.washington.edu).

simplex (represented by its vertices) is iteratively "improved" through certain reflection/expansion/contraction steps, possibly interspersed with nonimproving shrink (i.e., restart with a smaller simplex) steps. The notion of improvement varies in these methods, although in all cases it involves a criterion of *strict descent*, i.e., whether the function values at certain vertices have strictly decreased.

The first method, proposed by Spendley, Hext, and Himsworth in 1962 [23], tries to improve the worst vertex (with highest f-value) by isometrically reflecting it with respect to the centroid of the n best vertices or else reflecting the second-worst vertex with respect to the centroid of the n other vertices. If neither yields an improvement, it is suggested to shrink the simplex a fraction of the way toward the best vertex and restart the process. In this method, the set of interior angles of the simplex remains constant.

The second method, proposed by Nelder and Mead in 1965 [15] as an improvement on the method of Spendley, Hext, and Himsworth, is likely the most popular direct search method. This method tries to improve the worst vertex by reflecting it with respect to the centroid of the n best vertices, and it allows for nonisometric reflections, corresponding to expansion and "outside" contraction, as well as "inside" contraction of the simplex. If none of these steps yields an improvement, it shrinks the simplex a fraction of the way toward the best vertex and restarts the process. In this method, the simplex can assume arbitrary shapes, and some of its interior angles can become arbitrarily small.

The third method, proposed by Torczon [24, 25] (also see [6]) and called the multidirectional search (MDS) method, tries to find a new simplex with an improved best vertex by reflecting and then possibly expanding or contracting the n worst vertices with respect to the best vertex. If none of these steps yields an improvement, it shrinks the simplex a fraction of the way toward the best vertex and restarts the process. As in the method of Spendley, Hext, and Himsworth, the set of interior angles of the simplex remains constant.

Motivated by these methods, we propose a simplicial search method with three novel features:

- Flexibility in the number of vertices to be retained when constructing trial simplices. When a trial reflected/expanded/contracted simplex is constructed, the number m of "good" vertices to be retained from the current simplex can be chosen flexibly and dynamically. This number m can be any integer between 1 and n, except when the current simplex was produced by a nonimproving step, in which case m would be required to be below a certain *consistency index* between 1 and n. This index depends on the current simplex and the most recent simplex produced by an improving step.
- New criteria, called *fortified descent*, for accepting a trial simplex. Fortified-descent criteria are analogous to standard "sufficient descent" conditions in gradient-based methods for unconstrained optimization. In our experience, using fortified descent rather than strict descent does not significantly alter the practical behavior of the method. However, fortified descent is essential for our convergence proofs and cannot be replaced by strict descent.
- Flexibility in the number of additional function evaluations required to check for fortified descent. The number of additional function evaluations needed to check a trial reflected/expanded simplex for fortified descent can be controlled within the method. Detailed discussions of this feature and the associated trade-offs are given in section 2.

In section 3, we prove various convergence properties of the new method. For convenience of analyis, we count a sequence of nonimproving steps followed by an improving step as a *single* iteration. Let $\bar{m}_k$ denote the number of "good" vertices that are retained when constructing the *initial* trial simplices at iteration k. This number can be chosen freely between 1 and n. Note that if $\bar{m}_k > 1$, the number of vertices retained in a nonimproving step of iteration k may be less than $\bar{m}_k$. Also, there may be an arbitrary number of nonimproving steps during a given iteration. We prove the following: If f is continuously differentiable, bounded below, and uniformly continuous on its lower level set, and if we choose $\bar{m}_k$ to have the same value at all iterations k, then every cluster point of the generated sequence of iterates is a stationary point (see Corollary 3.3). If we choose $\bar{m}_k = 1$ at all iterations k, the same conclusion holds under milder assumptions on f, namely, that f is continuously differentiable and bounded below.

1.2. Related work. A recent overview of simplicial direct search methods was given by Wright [28]. Here we mention only a selection of related research.

Yu [29] proved that, if f is continuously differentiable and has a bounded lower level set, then at least one cluster point of the iterates generated by a modification of the method of Spendley, Hext, and Himsworth [23] is a stationary point of f. The modification includes replacing strict descent by a stronger criterion of sufficient descent (in the order of the diameter squared of the simplex) for accepting a reflection.

Despite the popularity of the Nelder–Mead method, very few papers have studied its theoretical properties. In a 1985 Ph.D. thesis, Woods [27] proved that, if f is strictly convex and coercive, a modified Nelder–Mead method generates simplices having certain limiting properties. (The modifications include a "relative decrease" criterion, stronger than strict descent and designed for when f is nonnegative-valued, for accepting a reflection and a contraction acceptance criterion different from that used by Nelder and Mead.) Woods also gave a pictorial example of a nonconvex differentiable function of two variables ($n = 2$), for which every iteration of the modified Nelder–Mead method entails a shrink of the simplex toward a nonminimizer of f.

There has been recent interest in theoretical properties of the *original* Nelder–Mead method. McKinnon [14] presented a family of strictly convex, coercive functions f of two variables ($n = 2$) with different degrees of smoothness, for which the simplices generated by the Nelder–Mead method, with particular choices for the initial simplex, contract to a nonstationary point. Thus, the Nelder–Mead method can fail to generate a stationary point in dimension two or higher, even when f is "nice." Lagarias et al. [12] proved that, for any strictly convex, coercive function f (not necessarily differentiable) of one variable ($n = 1$), the simplices generated by the Nelder–Mead method converge to the unique global minimizer of f. In addition, [12] proves that, for $n = 2$, the diameter of the simplices converges to zero so the function values at the vertices converge to the same value.

For the MDS method, Torczon [24, 25] proved that if f is continuously differentiable and has a bounded lower level set, then at least one cluster point of the generated sequence of iterates is a stationary point of f (also see [26] for extensions to pattern search methods).

Various papers about direct search methods have appeared in the Russian literature; see [11] for a recent survey. Dambrauskas [5] proposed an extension of the method of Spendley, Hext, and Himsworth in which the simplex may also contract toward its centroid. Rykov [19, 20, 21, 22] proposed modifications of the methods of Spendley, Hext, and Himsworth and of Nelder and Mead that allow, as in the method proposed here, reflection, expansion, and contraction of the simplex with respect to its m best vertices with m depending on the simplex. However, Rykov's methods differ from ours in several ways. In Rykov's method, vertices are reflected in specific manners: a subset of them (or centroid of the subset) is reflected through the centroid of the m best vertices, and the remaining vertices are moved in parallel with this subset (or centroid of the subset); m is chosen at each iteration by maximizing a certain function of m (six such functions were proposed); each reflection is determined by a criterion of sufficient descent similar to that of Yu (i.e., descent in the order of the diameter squared of the simplex) and requires either 1 or $n+1-m$ additional function evaluations (see reflections 1–5 in [21, 22]). In our method, vertices are reflected only in the general sense that the rays emanating from the reflected vertices toward the m best vertices should contain, in their convex hull, the rays emanating from a weighted centroid of the m best vertices toward the to-be-reflected vertices (see (2.5)); m is chosen freely between 1 and a certain index depending on the current simplex and the most recent simplex produced by an improving step; each reflection is determined by criteria of fortified descent and requires anywhere from 1 to $n-m+1$ additional function evaluations. Also, our convergence results require only that f be continuously differentiable and bounded below (for some results, we further require f to be uniformly continuous on a lower level set or to have a bounded lower level set), whereas the convergence results of Rykov further require f to be convex and to have a bounded lower level set and its gradient to be Lipschitz continuous.

1.3. Notation. Throughout, $\Re^n$ denotes the vector space of real n-tuples $x = (x_1, \ldots, x_n)$, viewed as column vectors and referred to as "points" or "n-vectors." We denote by $\|x\|$ the 2-norm of x. For any set S (of points/vectors) in $\Re^n$, we denote by $|S|$, $\mathrm{conv}(S)$, and $\mathrm{diam}(S)$ the cardinality, the convex hull, and the diameter, respectively, of S. In particular,

$$\mathrm{diam}(S) = \max_{s \in S, s' \in S} \|s - s'\|.$$

For any set $S = \{s_1, \ldots, s_{n+1}\}$ of $n+1$ points in $\Re^n$, we denote

$$\mathrm{von}(S) = \big|\det\,[\, s_2 - s_1 \quad \cdots \quad s_{n+1} - s_1 \,]\,\big| / \mathrm{diam}(S)^n.$$

We note that $\mathrm{von}(S)/n!$ is the volume of the (normalized) unit-diameter simplex with vertices $(s_i - s_1)/\mathrm{diam}(S)$, $i = 1, \ldots, n+1$ (see [9]). Thus, $\mathrm{von}(S) = 0$ if and only if the simplex with vertex set S has an interior angle equal to zero or, equivalently, the edges emanating from each vertex of this simplex are linearly dependent. For any sets S and S' in $\Re^n$ and any number $c > 0$, we denote $S - S' = \{s - s' : s \in S, s' \in S'\}$, $S \backslash S' = \{s \in S : s \notin S'\}$, and $cS = \{cs : s \in S\}$.

For any finite set S in $\Re^n$ of cardinality p, we denote by $F(S)$ the p-vector comprising the f-value of the elements of S permuted into increasing order, i.e.,

$$F(S) = \begin{bmatrix} F_1(S) \\ \cdots \\ F_p(S) \end{bmatrix}, \quad \text{where} \quad F_1(S) \le \cdots \le F_p(S),$$

and $F_i(S)$ denotes the ith smallest element of $f(s)$, $s \in S$. Denote

$$f_{\min}(S) = F_1(S), \qquad f_{\max}(S) = F_p(S).$$

For any two p-vectors c and d, we define their *consistency index* by

$$l(c, d) = \max\ i \in \{0, 1, \ldots, p\} \ \text{ such that } \ c_j \le d_j \text{ for } 1 \le j \le i.$$

Thus $l(c,d) = p$ means $c \le d$ while $l(c,d) = q < p$ means $c_j \le d_j$ for $1 \le j \le q$ and $c_{q+1} > d_{q+1}$.

Finally, let Φ denote the following class of functions:

$$\phi \in \Phi \quad \Longleftrightarrow \quad \phi : [0,\infty) \mapsto [0,\infty), \quad \phi \text{ is continuous}, \quad \lim_{t\to 0} \phi(t)/t = 0.$$

2. Description of new method. In this section we formally describe our simplicial search method and discuss its relation to other simplex-based direct search methods. At each iteration k, the method generates a new simplex, with vertex set S^{k+1}, from the current simplex, with vertex set S^k, by constructing trial reflected/expanded/contracted simplices, checking these for fortified descent, and, if needed, shrinking the current simplex a fraction of the way toward a best vertex. The interior angles of the trial simplices are further required to be bounded away from zero.

FORTIFIED-DESCENT SIMPLICIAL SEARCH (FDSS) METHOD. *Choose a set S^0 of $n+1$ points in $\Re^n$ satisfying* $\mathrm{von}(S^0) > 0$. *Choose constants $\theta_{\mathrm{r}} \in (0,1)$, $\tau_{\mathrm{r}} \in [1, 1/\theta_{\mathrm{r}})$, $\nu \in (0, \mathrm{von}(S^0)]$, $\theta_{\mathrm{bad}} \in (0,1]$, $\gamma_{\mathrm{s}} \in (0,1)$, and $\gamma_{\mathrm{e}} > 1$. Choose two functions $\alpha \in \Phi$ and $\beta \in \Phi$, with α also satisfying*

$$\inf_{t\ge c} \alpha(t) > 0 \quad \forall c > 0. \tag{2.1}$$

For $k = 0, 1, \ldots,$ generate S^{k+1} and $(x^k, \Delta_k, \bar{m}_k, m_k)$ from S^k by the following iteration k:

Step 0. *Let $S = S^k$ and go to Step* 1.

Step 1. *Construct a reflection of S: Let $\Delta = \mathrm{diam}(S)$. Choose an integer m satisfying*

$$1 \le m \le \min\{n, l(F(S), F(S^k))\}.$$

Partition S into two disjoint subsets S_{good} and S_{bad} such that $|S_{\mathrm{good}}| = m$ (so $|S_{\mathrm{bad}}| = n+1-m$) and

$$f_{\max}(S_{\mathrm{good}}) \le f_{\min}(S_{\mathrm{bad}}). \tag{2.2}$$

Choose a nonempty subset S_0 of S_{good} such that

$$f_{\max}(S_0) \le f_{\min}(S_{\mathrm{good}} \backslash S_0), \tag{2.3}$$

and scalars μ_s, $s \in S_0$, with $\mu_s \ge \theta_{\mathrm{r}}/|S_0|$ and $\sum_{s\in S_0} \mu_s = 1$. Define a weighted centroid $\hat{x}$ and its interpolated function value $\hat{f}$ by

$$\hat{x} = \sum_{s\in S_0} \mu_s s, \qquad \hat{f} = \sum_{s\in S_0} \mu_s f(s). \tag{2.4}$$

Choose a set S_{r} of $n+1-m$ points in $\Re^n$ satisfying

$$S_{\mathrm{bad}} - \hat{x} \subseteq \tau_{\mathrm{r}} \ \mathrm{conv}(S_{\mathrm{good}} - S_{\mathrm{r}}) \tag{2.5}$$

(i.e., $S_{\mathrm{good}} \cup S_{\mathrm{r}}$ is a "reflection" of S with respect to S_{good}) and

$$\Delta \le \mathrm{diam}(S_{\mathrm{good}} \cup S_{\mathrm{r}}) \le \gamma_{\mathrm{e}}\Delta \quad \textit{and} \quad \mathrm{von}(S_{\mathrm{good}} \cup S_{\mathrm{r}}) \ge \nu.$$

If $S = S^k$ (i.e., Step 1 is entered at iteration k for the first time), let $x^k = \hat{x}$ and $\bar{m}_k = m$. Go to Step 2.

Step 2. *Check whether reflected simplex satisfies fortified descent: Choose a nonempty subset $\Sigma_{\rm bad}$ of $S_{\rm bad}$ satisfying*

$$f_{\max}(\Sigma_{\rm bad}) - \hat{f} \geq \theta_{\rm bad}\left(f_{\max}(S) - \hat{f}\right) \tag{2.6}$$

and a nonempty subset $\Sigma_{\rm r}$ of $S_{\rm r}$ satisfying

$$\Sigma_{\rm bad} - \hat{x} \subseteq \tau_{\rm r}\ \mathrm{conv}(S_{\rm good} - \Sigma_{\rm r}). \tag{2.7}$$

(Since S_r satisfies (2.5), $\Sigma_{\rm r}$ may be chosen to be $S_{\rm r}$.) If the fortified-descent criteria

$$f_{\min}(\Sigma_{\rm r}) \leq f_{\max}(S_{\rm good}) - \alpha(\Delta) \quad \textit{and} \tag{2.8}$$

$$f_{\min}(\Sigma_{\rm r}) \leq f_{\max}(S_{\rm good}) - \theta_{\rm r}\left(f_{\max}(\Sigma_{\rm bad}) - \hat{f}\right) + \beta(\Delta) \tag{2.9}$$

are satisfied, then go to Step 3; else go to either Step 4 or Step 5. (The decision is user specified.)

Step 3. *Attempt an expansion and accept either the reflected or the expanded simplex: Choose a set $S_{\rm e}$ of $n+1-m$ points in $\Re^n$ satisfying*

$$\mathrm{diam}(S_{\rm good} \cup S_{\rm r}) \leq \mathrm{diam}(S_{\rm good} \cup S_{\rm e}) \leq \gamma_{\rm e}\Delta \quad \textit{and} \quad \mathrm{von}(S_{\rm good} \cup S_{\rm e}) \geq \nu.$$

($S_{\rm e}$ may be chosen to be $S_{\rm r}$, if expansion is not desired.) Choose a nonempty subset $\Sigma_{\rm e}$ of $S_{\rm e}$. If

$$f_{\min}(\Sigma_{\rm e}) \ \leq \ f_{\min}(\Sigma_{\rm r}), \tag{2.10}$$

then let $S^{k+1} = S_{\rm good} \cup S_{\rm e}$ (accept the expanded simplex); else let $S^{k+1} = S_{\rm good} \cup S_{\rm r}$ (accept the reflected simplex). In either case, let $\Delta_k = \Delta, m_k = m$, and terminate iteration k.

Step 4. *Attempt to find a contracted simplex satisfying fortified descent: Choose a set $S_{\rm c}$ of $n+1-m$ points in $\Re^n$ satisfying*

$$\mathrm{diam}(S_{\rm good} \cup S_{\rm c}) \leq \Delta \quad \textit{and} \quad \mathrm{von}(S_{\rm good} \cup S_{\rm c}) \geq \nu.$$

If $S_{\rm good} \cup S_{\rm c}$ satisfies the following consistency and fortified descent criteria relative to S^k,

$$F_i(S_{\rm good} \cup S_{\rm c}) \leq F_i(S^k), \quad i = 1, \ldots, m+1, \quad \textit{and} \tag{2.11}$$

$$\sum_{i=1}^{m+1} F_i(S_{\rm good} \cup S_{\rm c}) \leq \sum_{i=1}^{m+1} F_i(S^k) - \alpha(\Delta), \tag{2.12}$$

then let $S^{k+1} = S_{\rm good} \cup S_{\rm c}$ (accept the contracted simplex), $\Delta_k = \Delta$, $m_k = m+1$, and terminate iteration k. Otherwise, go to Step 5.

Step 5. *Shrink simplex toward a best vertex and check fortified descent: Choose an $s_{\rm best} \in \arg\min_{s \in S} f(s)$ and let S' be $S' = s_{\rm best} + \gamma_{\rm s}(S - s_{\rm best})$. If*

$$f_{\min}(S') \leq f_{\min}(S^k) - \alpha(\Delta), \tag{2.13}$$

then let $S^{k+1} = S'$ (accept the shrunken simplex), $\Delta_k = \Delta$, $m_k = 1$, and terminate iteration k. Otherwise, let $S = S'$, and return to Step 1 (accept a nonimproving *shrink and restart the process). If Step* 1 *is returned to an infinite number of times so we never terminate iteration k, output the point to which s_{best} converges and quit the method. (The point s_{best} converges because each time Step* 1 *is returned,* diam(S) *is decreased by a factor of γ_{s} and the new S is contained in the convex hull of the previous S.)*

Thus, each iteration k of the FDSS method either (i) performs a finite number of nonimproving shrinks followed by an improving reflection/expansion/contraction/shrink or (ii) performs an infinite number of nonimproving shrinks, in which case the method outputs the limit point. Below we discuss in more detail the various features of the method.

1. Choosing m. At each iteration k, when we enter Step 1 for the first time (from Step 0), we have $S = S^k$ so that $l(F(S), F(S^k)) = n + 1$, implying we can choose m to be any integer between 1 and n. This m is denoted by $\bar{m}_k$. If we subsequently return to Step 1 from Step 5, then $l(F(S), F(S^k))$ could possibly be less than n, and hence m cannot be chosen as freely. If f is convex or, more generally, quasi-convex (see [13]) in the sense that

$$f(x + \gamma(y - x)) \leq \max\{f(x), f(y)\} \quad \forall \gamma \in [0, 1],\ \forall x, y \in \Re^n, \tag{2.14}$$

then we have $l(F(S), F(S^k)) \geq n$ every time we return to Step 1 so that m can always be chosen freely. This is because when we shrink the simplex S in Step 5 toward the best vertex s_{best} to obtain $S' = s_{\text{best}} + \gamma_{\text{s}}(S - s_{\text{best}})$, we have $f(s_{\text{best}}) \leq f(s)$ for each $s \in S$ which, together with (2.14), implies

$$f(s_{\text{best}} + \gamma_{\text{s}}(s - s_{\text{best}})) \ \leq \ \max\{f(s_{\text{best}}), f(s)\} \ = \ f(s),$$

and so $F_i(S') \leq F_i(S)$, $i = 1, \ldots, n$. An induction argument yields that, each time we return to Step 1 from Step 5, we have $F_i(S) \leq F_i(S^k)$, $i = 1, \ldots, n$, and so $l(F(S), F(S^k)) \geq n$.

2. Choosing $\hat{x}$ and S_{r}. The set S_{r} is a reflection (in a general sense) of S_{bad} with respect to S_{good}. There are many choices for S_0 and the weights μ_s, which define $\hat{x}$ via (2.4) and S_{r}. For $\hat{x}$, a possible choice is

$$S_0 = S_{\text{good}}, \qquad \mu_s = \frac{1}{m}, \qquad \hat{x} = \frac{1}{m} \sum_{s \in S_{\text{good}}} s, \tag{2.15}$$

which makes $\hat{x}$ the centroid of S_{good}. For the reflected vertices S_{r}, a possible choice is

$$S_{\text{r}} = 2\hat{x} - S_{\text{bad}}. \tag{2.16}$$

When $m = n$, the choices (2.15) and (2.16) produce the standard reflected simplex from the methods of Spendley, Hext, and Himsworth and Nelder and Mead. When $m = 1$, these choices produce the reflected simplex from the MDS method. Furthermore, the resulting S_{r} automatically satisfies (2.5) with $\tau_{\text{r}} = 1$ and $\text{diam}(S_{\text{good}} \cup S_{\text{r}}) = \text{diam}(S)$. This is illustrated in Figure 1 in the case $n = 3$, for $m = 1, 2, 3$. For example, for $m = 3$, we have $S_{\text{good}} = \{a, b, c\}$, $S_{\text{bad}} = \{d\}$, $S_{\text{r}} = \{d'\}$, and, by isometry of reflection, $d - \hat{x} = \hat{x} - d'$. Since $\hat{x} \in \text{conv}(\{a, b, c\})$ so that $\hat{x} - d' \in \text{conv}(\{a - d', b - d', c - d'\})$, we see that

$$S_{\text{bad}} - \hat{x} = \{d - \hat{x}\} = \{\hat{x} - d'\} \subset \text{conv}(\{a - d', b - d', c - d'\}) = \text{conv}(S_{\text{good}} - S_{\text{r}}).$$

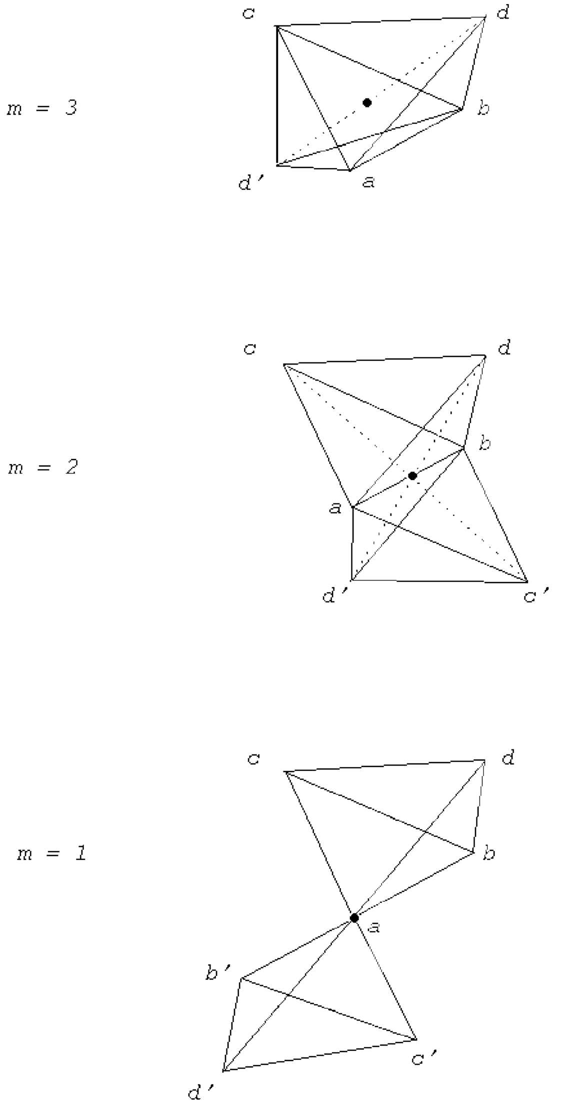


FIG. 1. *$S = \{a, b, c, d\}$ and $m = |S_{\text{good}}|$. For $m = 3$, we have $S_{\text{good}} = \{a, b, c\}$, $S_{\text{bad}} = \{d\}$, $S_{\text{r}} = \{d'\}$; for $m = 2$, we have $S_{\text{good}} = \{a, b\}$, $S_{\text{bad}} = \{c, d\}$, $S_{\text{r}} = \{c', d'\}$; for $m = 1$, we have $S_{\text{good}} = \{a\}$, $S_{\text{bad}} = \{b, c, d\}$, $S_{\text{r}} = \{b', c', d'\}$.*

Another possible choice for S_{r} is

$$S_{\text{r}} = \{2[s]_H^+ - s : s \in S_{\text{bad}}\}, \quad \text{with} \quad H = \text{aff}(S_{\text{good}}) + \text{aff}(S_{\text{bad}} - s_{\text{bad}}),$$

where s_{bad} is any element of S_{bad}, $[s]_H^+$ denotes the orthogonal projection of s onto H, and aff( ) denotes the affine hull. With this choice of S_{r}, $\text{diam}(S_{\text{good}} \cup S_{\text{r}}) = \text{diam}(S)$, but (2.5) is not necessarily satisfied.

3. Choosing Σ_{bad}. There are many choices for Σ_{bad} (the subset of S_{bad} used in Step 2 to check fortified descent). Possible choices are the worst vertex in S_{bad} or S_{bad} itself. Choosing Σ_{bad} with a small cardinality may be advantageous in that Σ_{r}, defined by (2.7), may also have a small cardinality. Since $|\Sigma_{\text{r}}|$ additional function

evaluations are needed to check the fortified-descent criteria in Step 2, a small $|\Sigma_r|$ leads to a more economical acceptance test at a given iteration. For example, if $m = 1$, so that S_{bad} contains the n worst vertices, and if $\hat{x}$ and S_{r} are chosen by (2.15) and (2.16), then, by choosing

$$\Sigma_{\text{bad}} = \arg \max_{s \in S_{\text{bad}}} f(s), \qquad \Sigma_{\text{r}} = 2\hat{x} - \Sigma_{\text{bad}},$$

only one additional function evaluation is needed to test fortified descent for the reflected simplex. Note, however, that checking fortified descent for the reflected simplex with a smaller number of function evaluations tends to make the reflected simplex less likely to be accepted. A similar trade-off occurs when choosing Σ_{e}, the analogous set for the expanded simplex. This trade-off is evidenced in our numerical experience.

4. Relation to other methods. With appropriate algorithmic choices, the FDSS method can be made to generate trial simplices analogous to those produced by other simplex-based direct search methods. For example, when $m = \min\{n, l(F(S), F(S^k))\}$, the FDSS method may be viewed as similar in spirit to the methods of Spendley, Hext, and Himsworth and Nelder and Mead. When $m = 1$, the FDSS method may be viewed as related to the MDS method. On the other hand, the acceptance criteria for a new simplex used in the FDSS method (based on fortified descent) are different and inherently more stringent than strict descent used in these other methods. The FDSS method further differs from the Nelder–Mead method in that it maintains the interior angles of the simplex to be bounded away from zero.

5. Finite termination of FDSS method. As described, the FDSS method generates an infinite sequence of simplices. In practice, suitable termination criteria are needed. Previous suggestions for termination criteria (see, e.g., [24, 28]) are based on small diameter for the simplex and/or small differences in the vertex function values. Here we consider alternative criteria. For an n-dimensional simplex in $\Re^n$ with vertex set $S = \{s_1, \ldots, s_{n+1}\}$, let

$$g(S) = \begin{bmatrix} (s_2 - s_1)^T \\ \vdots \\ (s_{n+1} - s_1)^T \end{bmatrix}^{-1} \begin{bmatrix} f(s_2) - f(s_1) \\ \vdots \\ f(s_{n+1}) - f(s_1) \end{bmatrix},$$

where superscript T denotes transpose. Given a tolerance ϵ, one can terminate the FDSS method whenever S in Step 1 satisfies

$$\operatorname{diam}(S) \leq \epsilon \quad \text{and} \quad \|g(S)\| \leq \epsilon.$$

When f is continuously differentiable, $\operatorname{von}(S)$ is bounded away from zero, and $\operatorname{diam}(S)$ tends to zero, we have that $g(S)$ approaches $\nabla f(s_1)$. To avoid solving an $n \times n$ linear system, one can use the following alternative criteria:

$$\operatorname{diam}(S) \leq \epsilon \quad \text{and} \quad \|\tilde{g}(S)\| \leq \epsilon, \tag{2.17}$$

where

$$\tilde{g}(S) = \begin{bmatrix} (f(s_2) - f(s_1))/\|s_2 - s_1\| \\ \vdots \\ (f(s_{n+1}) - f(s_1))/\|s_{n+1} - s_1\| \end{bmatrix}. \tag{2.18}$$

As long as $\mathrm{von}(S)$ is bounded away from zero, $\|\tilde{g}(S)\|$ differs from $\|g(S)\|$ by only a constant factor. Criteria (2.17), which have the nice feature of yielding an approximate stationary point, were used in our computational tests (see section 4).

6. The quantity $\hat{f}$ in (2.4) is an approximation to $f(\hat{x})$, and may be replaced throughout the FDSS method by $f(\hat{x})$ without affecting the theoretical convergence. However, this change significantly increases the total number of function evaluations in practice. Also, τ_{r} need not be constant, provided it is bounded away from $1/\theta_{\mathrm{r}}$.

7. The quantity $\mathrm{von}(S)$ may be replaced throughout the FDSS method by any nonnegative continuous function of S (with S viewed as a point in $\Re^{(n+1)n}$) that is zero if and only if the normalized simplex with vertex set $(S-s)/\mathrm{diam}(S)$, where $s \in S$, has zero volume.

8. Significance of fortified descent. In the FDSS method, fortified-descent criteria appear in (2.8) and (2.9) for the reflected simplex, in (2.10) for the expanded simplex, in (2.11) and (2.12) for the contracted simplex, and in (2.13) for the shrunken simplex. By requiring an improving simplex to satisfy one of these sets of conditions, we will be able to prove that the simplex diameter converges to zero and that at least one cluster point of $\{x^k\}$ is a stationary point of f.

From a numerical standpoint, there is very little difference between fortified-descent criteria and the (less stringent) strict descent required in some other direct search methods. In particular, we can choose the function α, which appears in (2.8), (2.12), and (2.13), to be small everywhere (e.g., $\alpha(t) = 10^{-5}\min\{t^2, 1\}$); we can choose β, which appears in (2.9), to have fast growth away from zero (e.g., $\beta(t) = 10^5 t^2$); and we can choose θ_{r}, which appears in (2.9), to have a value near zero. From a theoretical standpoint, however, there is often a large difference between fortified descent and strict (i.e., simple) descent. Almost all the complications in the proof of Lagarias et al. [12] arise because arbitrarily small improvements can be accepted.

The fortified-descent condition (2.10) for the expanded simplex can be replaced by the less stringent conditions

$$f_{\min}(\Sigma_{\mathrm{e}}) \le f_{\max}(S_{\mathrm{good}}) - \alpha(\Delta) \quad \text{and} \tag{2.19}$$

$$f_{\min}(\Sigma_{\mathrm{e}}) \le f_{\max}(S_{\mathrm{good}}) - \theta_{\mathrm{r}}(f_{\max}(\Sigma_{\mathrm{bad}}) - \hat{f}) + \beta(\Delta), \tag{2.20}$$

where α, β, and θ_{r} may differ from their counterparts for the reflected simplex, i.e., (2.8)–(2.9). The convergence results of Theorem 3.2 will still apply with only minor modifications in the proofs. In practice, using (2.19) and (2.20) rather than (2.10) typically yields faster convergence of the simplices.

3. Convergence analysis of new method.

In this section we analyze the convergence properties of the FDSS method. In particular, we show that, under mild assumptions on f, there is at least one cluster point of $\{x^k\}$ that is a stationary point of f and, if we choose $\bar{m}_k$ with the same value at all iterations k, then every cluster point of $\{x^k\}$ is a stationary point of f. First, we need the following lemma showing that, under appropriate assumptions, $\{\mathrm{diam}(S^k)\} \to 0$. The proof of this is based on showing that $\{F(S^k)\}$ is a sufficiently "lexicographically decreasing" sequence.

LEMMA 3.1. *Assume that the FDSS method does not quit at some iteration k and let $\{(S^k, x^k, \Delta_k, \bar{m}_k, m_k)\}_{k=0,1,\ldots}$ be the generated sequence. Then, $f_{\min}(S^{k+1}) \le f_{\min}(S^k)$ and $\mathrm{von}(S^k) \ge \nu$ for all k, and the following hold:*

(a) *For all $k = 0, 1, \ldots$,*

$$F_i(S^{k+1}) \le F_i(S^k),\ i = 1, \ldots, m_k, \quad \sum_{i=1}^{m_k} (F_i(S^{k+1}) - F_i(S^k)) \le -\alpha(\Delta_k). \tag{3.1}$$

(b) *If* $m_k = 1$ *for all* k *or if* f *is uniformly continuous on* $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$*, then either* (i) $\{f_{\min}(S^k)\} \to -\infty$ *or* (ii) $\{\Delta_k\} \to 0$ *and* $\{\mathrm{diam}(S^k)\} \to 0$.

Proof. That $f_{\min}(S^{k+1}) \le f_{\min}(S^k)$ for all k follows from the fact that, in reflecting or contracting or shrinking a simplex S, one of the best vertices of S (i.e., an element of $\arg\min_{s\in S} f(s)$) is held fixed. That $\mathrm{von}(S^k) \ge \nu$ for all k follows from the observation that, in reflecting or contracting S with respect to S_{good}, the new S is always chosen to satisfy $\mathrm{von}(S) \ge \nu$ while, in shrinking S toward s_{best}, $\mathrm{von}(S)$ is unchanged.

(a) Fix any $k \in \{0, 1, \ldots\}$. Consider the last time we pass through Steps 1–2 when generating S^{k+1} and (x^k, Δ_k, m_k) from S^k at iteration k. We have that either (i) (2.8) holds and $S^{k+1} = S_{\mathrm{good}} \cup S_{\mathrm{r}}$ and $m_k = m$ or (ii) (2.8) and (2.10) hold and $S^{k+1} = S_{\mathrm{good}} \cup S_{\mathrm{e}}$ and $m_k = m$ or (iii) (2.11) and (2.12) hold and $S^{k+1} = S_{\mathrm{good}} \cup S_{\mathrm{c}}$ and $m_k = m + 1$ or (iv) (2.13) holds and $S^{k+1} = S'$ and $m_k = 1$. In case (i), we have from (2.8) and $\Sigma_{\mathrm{r}} \subseteq S_{\mathrm{r}}$ that

$$F_1(S_{\mathrm{r}}) = f_{\min}(S_{\mathrm{r}}) \le f_{\min}(\Sigma_{\mathrm{r}}) \le f_{\max}(S_{\mathrm{good}}) - \alpha(\Delta) = F_m(S_{\mathrm{good}}) - \alpha(\Delta),$$

and from (2.2) and $|S_{\mathrm{good}}| = m \le l(F(S), F(S^k))$ that $F_i(S_{\mathrm{good}}) = F_i(S) \le F_i(S^k)$ for $i = 1, \ldots, m$, so $S^{k+1} = S_{\mathrm{good}} \cup S_{\mathrm{r}}$ yields $F_i(S^{k+1}) \le F_i(S^k)$ for $i = 1, \ldots, m$, and

$$\sum_{i=1}^{m} (F_i(S^{k+1}) - F_i(S^k)) \le \sum_{i=1}^{m-1} (F_i(S_{\mathrm{good}}) - F_i(S^k)) + (F_1(S_{\mathrm{r}}) - F_m(S^k)) \le -\alpha(\Delta),$$

where the first inequality also uses the observation that the sum of the first m components of $F(S^{k+1})$ is less than or equal to the sum of any m components of $F(S^{k+1})$. This, together with $(\Delta, m) = (\Delta_k, m_k)$, shows that (3.1) holds in case (i). A similar argument shows that (3.1) holds in case (ii). In cases (iii) and (iv), (3.1) holds trivially.

(b) If $m_k = 1$ for all iterations k, then we have from the choice of x^k and part (a) that

$$f_{\min}(S^{k+1}) \le f_{\min}(S^k) - \alpha(\Delta_k)$$

for all k so, by (2.1), either $\{f_{\min}(S^k)\} \to -\infty$ or $\{\Delta_k\} \to 0$. Instead, suppose f is uniformly continuous on $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ and we will argue by contradiction that either $\{f_{\min}(S^k)\} \to -\infty$ or $\{\Delta_k\} \to 0$. Suppose $\{f_{\min}(S^k)\} \not\to -\infty$ (which, since $\{f_{\min}(S^k)\}$ is nonincreasing, implies that $\{f_{\min}(S^k)\}$ converges) and yet $\{\Delta_k\} \not\to 0$. For each $i \in \{1, \ldots, n+1\}$, let

$$K_i = \left\{k \in \{0, 1, \ldots\} : F_i(S^{k+1}) \le F_i(S^k) - \alpha(\Delta_k)/m_k\right\}.$$

Since (3.1) holds and $m_k \le n + 1$ for all k (so, for each $k \in \{0, 1, \ldots\}$, there exists at least one $i \in \{1, \ldots, n+1\}$ such that $k \in K_i$), we have $\bigcup_{i=1}^{n+1} K_i = \{0, 1, \ldots\}$ and so

$$\bar{i} = \min\{i \in \{1, \ldots, n+1\} : |K_i| = \infty, \{\Delta_k\}_{k\in K_i} \not\to 0\}$$

is well defined. Since $\{\Delta_k\}_{k\in K_{\bar{i}}} \not\to 0$, there exist $c > 0$ and subsequence K of $K_{\bar{i}}$ such that $\Delta_k \ge c$ for all $k \in K$. This implies

$$f_{\min}(S^{k+1}) \le F_{\bar{i}}(S^{k+1}) \le F_{\bar{i}}(S^k) - \alpha(\Delta_k)/m_k \le F_{\bar{i}}(S^k) - \left(\inf_{t\ge c} \alpha(t)\right)/(n+1)$$

for all $k \in K$. For each $k \in \{1, 2, \ldots\}$, let r_k be the largest $t \in \{1, 2, \ldots, k\}$ satisfying $F_{\bar{i}}(S^t) > F_{\bar{i}}(S^{t-1})$ (with $r_k = 0$ if no such t exists). Since $\{f_{\min}(S^k)\}$ converges and, by (2.1), the infimum above is a positive constant, we have that $r_k \to \infty$ as $k \to \infty$ (otherwise, $\{F_{\bar{i}}(S^k)\}$ would have a monotonically decreasing tail, and the above relation would imply $\{F_{\bar{i}}(S^k)\} \to -\infty$ and so $\{f_{\min}(S^k)\} \to -\infty$). Also, we have trivially that $F_{\bar{i}}(S^k) \le F_{\bar{i}}(S^{k-1}) \le \cdots \le F_{\bar{i}}(S^{r_k})$, so the above relation implies

$$f_{\min}(S^{k+1}) \le F_{\bar{i}}(S^{r_k}) - \left(\inf_{t \ge c} \alpha(t)\right) / (n+1) \quad \forall k \in K. \tag{3.2}$$

Lastly, we have trivially that $F_{\bar{i}}(S^{r_k}) > F_{\bar{i}}(S^{r_k - 1})$ for all k sufficiently large so that $r_k > 0$, in which case, since (3.1) holds at all iterations k (so, in particular, at iteration $r_k - 1$), there must exist an $i < \bar{i}$ such that $r_k - 1 \in K_i$. By further passing into a subsequence if necessary, we can assume that $r_k > 0$ and it is the same i for all $k \in K$, implying $\{r_k - 1 : k \in K\} \subseteq K_i$. Then, $|K_i| = \infty$ (since $r_k \to \infty$ as $k \to \infty$) and so, by the choice of $\bar{i}$, we have $\{\Delta_{r_k - 1}\}_{k \in K} \to 0$. Since S^{k+1} is obtained by reflecting/expanding (by a factor of at most γ_{e}) or contracting/shrinking a simplex S with $\mathrm{diam}(S) = \Delta_k$, we also have that

$$\mathrm{diam}(S^{k+1}) \le \gamma_{\mathrm{e}} \Delta_k, \quad k = 0, 1, \ldots, \tag{3.3}$$

so together we have $\{\mathrm{diam}(S^{r_k})\}_{k \in K} \to 0$. Then, since $f_{\min}(S^{r_k}) \le f_{\min}(S^0)$ for all $k \in K$, the uniform continuity of f on $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ implies $\{F_{\bar{i}}(S^{r_k}) - f_{\min}(S^{r_k})\}_{k \in K} \to 0$, which, together with $\{f_{\min}(S^{k+1}) - f_{\min}(S^{r_k})\} \to 0$ (since $\{f_{\min}(S^k)\}$ converges and $r_k \to \infty$ as $k \to \infty$), contradicts (3.2). Thus, $\{\Delta_k\} \to 0$ and, by (3.3), $\{\mathrm{diam}(S^k)\} \to 0$. □

By using Lemma 3.1, we prove our main convergence result below, showing that if f is continuously differentiable and bounded below (and some other mild conditions hold) and $\{\Delta_k\} \to 0$, then at least one cluster point of $\{x^k\}$ is a stationary point of f and, if we choose $\bar{m}_k$ with the same value at every iteration k, then every cluster point of $\{x^k\}$ is a stationary point of f. (It can be seen with examples that f being differentiable and bounded below and $\{\Delta_k\} \to 0$ are necessary for convergence to a stationary point of f, so our sufficient conditions for convergence are in some sense close to being necessary.) The proof is based on showing, using $\{\Delta_k\} \to 0$ and the fortified-descent criteria, that, along any subsequence of $\{x^k\}$ where a contraction/shrinking step is taken at each iteration k in the subsequence, any cluster point is a stationary point of f. To show that every cluster point of $\{x^k\}$ is a stationary point when $\bar{m}_k$ is constant, we show that whenever S^k is in a neighborhood of a nonstationary point and $\mathrm{diam}(S^k)$ is sufficiently small, the sum $\sum_{i=1}^{m} F_i(S^k)$ decreases by an amount in the order of $\mathrm{diam}(S^k)$. Using this fact, we argue that if one of the cluster points is nonstationary, then the above sum cannot converge, and we thereby obtain a contradiction.

Theorem 3.2. *Assume that* $\inf_{x \in \Re^n} f(x) > -\infty$ *and* f *is continuously differentiable on* $\Re^n$. *Let* $\{(S^k, x^k, \Delta_k, \bar{m}_k, m_k)\}_{k=0,1,\ldots}$ *be generated by the FDSS method. If at some iteration* k, *the method quits (because Step* 1 *is returned to an infinite number of times), the output point is a stationary point of* f. *If the method does not quit at some iteration and* $\{\mathrm{diam}(S^k)\} \to 0$, *then the following hold:*

(a) *If* $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ *is bounded, then at least one cluster point of* $\{x^k\}$ *is a stationary point of* f.

(b) *If we choose, at all iterations* k *beyond some number,* $\bar{m}_k$ *to be a constant between* 1 *and* n, *then every cluster point of* $\{x^k\}$ *is a stationary point of* f.

Proof. If at some iteration k, the FDSS method quits because Step 1 is returned to an infinite number of times, then, since each time as we return to Step 1 we shrink S toward s_{best} (from Step 5) by a factor of γ_{s}, we have that $\mathrm{diam}(S) \to 0$. Then, by using an argument analogous to the proof of the claim below, we obtain that the point to which s_{best} converges is a stationary point. For brevity, we omit the argument. Thus, in what follows, we assume that the FDSS method does not quit at some iteration and $\{\sigma_k\} \to 0$, where for brevity we let $\sigma_k = \mathrm{diam}(S^k)$ for all k.

We claim that, for any subsequence K of $\{0, 1, \dots\}$ such that a contraction/shrinking step is taken at iteration k for all $k \in K$ and $\{x^k\}_{k \in K} \to$ some x^∞, we have that x^∞ is a stationary point of f. To show this, fix any such subsequence K and, for each $k \in K$, let $S_0^k, S_{\mathrm{good}}^k, S_{\mathrm{bad}}^k, S_{\mathrm{r}}^k$, Σ_{bad}^k, Σ_{r}^k denote the $S_0, S_{\mathrm{good}}, S_{\mathrm{bad}}, S_{\mathrm{r}}$, Σ_{bad}, Σ_{r} used when Steps 1–2 are first entered during iteration k. By passing into a subsequence if necessary, we will assume that $|S_0^k|$, $|S_{\mathrm{good}}^k|$, $|\Sigma_{\mathrm{bad}}^k|$, $|\Sigma_{\mathrm{r}}^k|$ are the same for all $k \in K$, so that

$$S_0^k = \{s_i^k\}_{i \in I_0}, \quad S_{\mathrm{good}}^k = \{s_i^k\}_{i \in I_1}, \quad S_{\mathrm{bad}}^k = \{s_j^k\}_{j \in I_2}, \quad S_{\mathrm{r}}^k = \{t_j^k\}_{j \in I_2}, \tag{3.4}$$

$$\Sigma_{\mathrm{bad}}^k = \{s_j^k\}_{j \in J_2}, \quad \Sigma_{\mathrm{r}}^k = \{t_j^k\}_{j \in J_3} \tag{3.5}$$

for some partition (I_1, I_2) of $N = \{1, \dots, n+1\}$, some nonempty $I_0 \subseteq I_1$, $J_2 \subseteq I_2$, $J_3 \subseteq I_2$, and some sets of points $\{s_i^k\}_{i \in N}$ and $\{t_j^k\}_{j \in I_2}$ in $\Re^n$ $(k \in K)$. Since $\max_{i \in I_0, j \notin I_0} \|s_j^k - s_i^k\| \le \mathrm{diam}(S_{\mathrm{good}}^k \cup S_{\mathrm{bad}}^k) = \sigma_k$ and $\max_{i \in I_1, j \in I_2} \|t_j^k - s_i^k\| \le \mathrm{diam}(S_{\mathrm{good}}^k \cup S_{\mathrm{r}}^k) \le \gamma_{\mathrm{e}} \sigma_k$ for all $k \in K$, by further passing into a subsequence if necessary, we will assume that

$$\{(s_j^k - s_i^k)/\sigma_k\}_{k \in K} \to d_{ij}, \; i \in I_0, j \in N, \quad \{(t_j^k - s_i^k)/\sigma_k\}_{k \in K} \to e_{ij}, \; i \in I_1, j \in I_2 \tag{3.6}$$

for some sets of n-vectors $\{d_{ij}\}_{i \in I_0, j \in N}$ and $\{e_{ij}\}_{i \in I_1, j \in I_2}$. Furthermore, we will assume that $\max_{j \in J_2} f(s_j^k)$ is attained by the same index $\bar{j} \in J_2$ for all $k \in K$. For each $k \in K$, since (2.4) holds with $(S_0, \hat{x}) = (S_0^k, x^k)$, we have (also using (3.4))

$$x^k = \sum_{i \in I_0} \mu_i^k s_i^k$$

for some set of scalars $\{\mu_i^k\}_{i \in I_0}$ exceeding $\theta_{\mathrm{r}}/|I_0|$ and summing to 1, so that

$$s_j^k - x^k = \sum_{i \in I_0} \mu_i^k (s_j^k - s_i^k), \quad j \notin I_0.$$

Similarly, for each $k \in K$, since (2.7) holds with $(S_{\mathrm{good}}, \Sigma_{\mathrm{bad}}, \Sigma_{\mathrm{r}}, \hat{x}) = (S_{\mathrm{good}}^k, \Sigma_{\mathrm{bad}}^k, \Sigma_{\mathrm{r}}^k, x^k)$, we have (also using (3.4) and (3.5))

$$s_{\bar{j}}^k - x^k \in \tau_{\mathrm{r}} \; \mathrm{conv}\{s_i^k - t_j^k\}_{i \in I_1, j \in J_3}.$$

By further passing into a subsequence if necessary, we will assume that, for each $i \in I_0$, $\{\mu_i^k\}_{k \in K}$ converges to some positive scalar μ_i. Then, dividing each side of the above two relations by σ_k and using $\{\sigma_k\}_{k \in K} \to 0$ and (3.6) yields in the limit (as $k \to \infty$, $k \in K$)

$$d_j = \sum_{i \in I_0} \mu_i d_{ij}, \quad j \notin I_0, \tag{3.7}$$

$$-d_{\bar{j}} = \tau_{\mathrm{r}} \sum_{i \in I_1, j \in J_3} \lambda_{ij} e_{ij}, \tag{3.8}$$

where $\{\lambda_{ij}\}_{i\in I_1, j\in J_3}$ is a set of nonnegative numbers summing to 1 and $\{d_j\}_{j\in I_2}$ is a set of n-vectors satisfying

$$\{(s_j^k - x^k)/\sigma_k\}_{k\in K} \to d_j, \quad j \notin I_0.$$

For each $k \in K$, since (2.2) and (2.3) hold with $(S_0, S_{\rm good}, S_{\rm bad}) = (S_0^k, S_{\rm good}^k, S_{\rm bad}^k)$, we have (also using (3.4))

$$f(s_j^k) - f(s_i^k) \geq 0, \quad i \in I_0,\ j \notin I_0,$$

so dividing both sides by σ_k and using $\{\sigma_k\}_{k\in K} \to 0$ and (3.6) yield in the limit (as $k \to \infty$, $k \in K$)

$$\nabla f(x^\infty)^T d_{ij} \geq 0, \quad i \in I_0, j \notin I_0. \tag{3.9}$$

Since a contraction or shrinking step is taken at iteration k for all $k \in K$, by further passing into a subsequence if necessary, we can assume that either (a) (2.8) does not hold with $(S_{\rm good}, \Sigma_{\rm r}, \Delta) = (S_{\rm good}^k, \Sigma_{\rm r}^k, \sigma_k)$ for all $k \in K$ or (b) (2.9) does not hold with $(S_{\rm good}, \Sigma_{\rm bad}, \Sigma_{\rm r}, \Delta) = (S_{\rm good}^k, \Sigma_{\rm bad}^k, \Sigma_{\rm r}^k, \sigma_k)$ and $\hat f = \sum_{i\in I_0} \mu_i^k f(s_i^k)$ for all $k \in K$. In case (a), we have (also using (3.4) and (3.5))

$$f(t_j^k) - f(s_i^k) \geq -\alpha(\sigma_k), \quad i \in I_1,\ j \in J_3,$$

so dividing both sides by σ_k and using $\{\sigma_k\}_{k\in K} \to 0$, $\alpha \in \Phi$, and (3.6) yield in the limit (as $k \to \infty$, $k \in K$) $\nabla f(x^\infty)^T e_{ij} \geq 0$ for $i \in I_1,\ j \in J_3$, so, by (3.8) and $\lambda_{ij} \geq 0$ for all $i \in I_1, j \in J_3$,

$$-\nabla f(x^\infty)^T d_{\bar j} = \tau_{\rm r} \sum_{i\in I_1, j\in J_3} \lambda_{ij} \nabla f(x^\infty)^T e_{ij} \geq 0. \tag{3.10}$$

In case (b), we have (also using (3.4))

$$f(t_j^k) - f(s_i^k) \geq \theta_{\rm r} \left(\sum_{i\in I_0} \mu_i^k f(s_i^k) - f(s_{\bar j}^k) \right) + \beta(\sigma_k), \quad i \in I_1,\ j \in J_3,$$

so dividing both sides by σ_k and using $\{\sigma_k\}_{k\in K} \to 0$, $\beta \in \Phi$, and (3.6) yield in the limit (as $k \to \infty$, $k \in K$) that

$$\nabla f(x^\infty)^T e_{ij} \geq -\theta_{\rm r} \sum_{i\in I_0} \mu_i \nabla f(x^\infty)^T d_{i\bar j} = -\theta_{\rm r} \nabla f(x^\infty)^T d_{\bar j}, \quad i \in I_1,\ j \in J_3,$$

where the equality follows from (3.7). This, together with (3.8) and $\lambda_{ij} \geq 0$ for all $i \in I_1, j \in J_3$, and $\sum_{i\in I_1, j\in J_3} \lambda_{ij} = 1$, yields

$$-\nabla f(x^\infty)^T d_{\bar j} = \tau_{\rm r} \sum_{i\in I_1, j\in J_3} \lambda_{ij} \nabla f(x^\infty)^T e_{ij} \geq -\tau_{\rm r}\theta_{\rm r} \cdot \nabla f(x^\infty)^T d_{\bar j},$$

so, by $\theta_{\rm r} \in (0, 1/\tau_{\rm r})$, (3.10) holds. Thus, in either case, (3.10) holds. Since by (3.7) and (3.9) we also have

$$\nabla f(x^\infty)^T d_{\bar j} = \sum_{i\in I_0} \mu_i \nabla f(x^\infty)^T d_{i\bar j} \geq 0,$$

this yields $\nabla f(x^\infty)^T d_{\bar{j}} = 0$. For each $k \in K$, since (2.6) holds with $(S, \Sigma_{\rm bad}) = (S^k, \Sigma^k_{\rm bad})$ and $\hat{f} = \sum_{i \in I_0} \mu_i^k f(s_i^k)$ (also using (3.4) and (3.5)),

$$\sum_{i \in I_0} \mu_i^k f(s_i^k) - f(s_{\bar{j}}^k) \leq \theta_{\rm bad} \left(\sum_{i \in I_0} \mu_i^k f(s_i^k) - f(s_j^k) \right), \quad j \notin I_0.$$

Dividing both sides by σ_k and using $\{\sigma_k\}_{k \in K} \to 0$ and (3.6) yield in the limit (as $k \to \infty$, $k \in K$)

$$- \sum_{i \in I_0} \mu_i \nabla f(x^\infty)^T d_{i\bar{j}} \leq -\theta_{\rm bad} \sum_{i \in I_0} \mu_i \nabla f(x^\infty)^T d_{ij}, \quad j \notin I_0.$$

By (3.7), the left-hand side of this inequality equals $-\nabla f(x^\infty)^T d_{\bar{j}}$, which was just shown to equal zero, so (3.9) and the fact $\mu_i > 0$ for all $i \in I_0$ imply

$$\nabla f(x^\infty)^T d_{ij} = 0, \quad i \in I_0, \ j \notin I_0. \tag{3.11}$$

We show below that the elements of $\{d_{ij}\}_{i \in I_0, j \notin I_0}$ span $\Re^n$, which together with (3.11) would imply $\nabla f(x^\infty) = 0$, thus proving the claim. Fix any $\bar{i} \in I_0$. By (3.4) and Lemma 3.1, we have $\left| \det \left[(s_i^k - s_{\bar{i}}^k)/\sigma_k \right]_{i \in N \setminus \{\bar{i}\}} \right| = \mathrm{von}(S^k) \geq \nu$ for all k, so (3.6) and the continuity of det[ ] yield in the limit (as $k \to \infty$, $k \in K$) $\left| \det \left[d_{\bar{i}i} \right]_{i \in N \setminus \{\bar{i}\}} \right| \geq \nu$. Thus the elements of $\{d_{\bar{i}i}\}_{i \in N}$ span $\Re^n$. Since, by (3.6), $d_{\bar{i}i} = d_{\bar{i}j} - d_{ij}$ for all $i \in I_0$ and $j \notin I_0$, and so these elements may be expressed as linear combinations of the elements of $\{d_{ij}\}_{i \in I_0, j \notin I_0}$, the latter must also span $\Re^n$.

(a) Suppose $\{x \in \Re^n : f(x) \leq f_{\min}(S^0)\}$ is bounded. Since $f_{\min}(S^k) \leq f_{\min}(S^0)$ for each k so that at least one element of S^k is in this set, it follows from $\{\sigma_k\} \to 0$ that $\{x^k\}$ approaches this set and hence is bounded. Since $\{\sigma_k\} \to 0$ and $\sigma_{k+1} < \sigma_k$ only if at least one contraction or shrinking step is taken at iteration k, there must exist a subsequence K of $\{0, 1, \ldots\}$ such that a contraction or shrinking step is taken at iteration k for all $k \in K$. By the above claim, any cluster point x^∞ of $\{x^k\}_{k \in K}$ is a stationary point of f.

(b) Suppose that we choose, at all iterations k beyond some number $\hat{k}$, $\bar{m}_k = \bar{m}$ with $\bar{m} \in \{1, \ldots, n\}$. Let x^∞ be any cluster point of $\{x^k\}$. Suppose x^∞ is not a stationary point of f and we will arrive at a contradiction. Since f is continuously differentiable, this implies that there exists a $\delta > 0$ such that $B(x^\infty, \delta) = \{x \in \Re^n : \|x - x^\infty\| \leq \delta\}$ contains no stationary point of f. Let

$$K = \{k \in \{0, 1, \ldots\} : x^k \in B(x^\infty, \delta)\}.$$

There must exist a $\tilde{k} \geq \hat{k}$ such that no contraction or shrinking step is taken at iteration k for all $k \in K$ with $k \geq \tilde{k}$ (otherwise there would exist a subsequence K' of K such that a contraction or shrinking step is taken at iteration k for all $k \in K'$ so, by the preceding claim, any cluster point of $\{x^k\}_{k \in K'}$, which would lie in $B(x^\infty, \delta)$, would be a stationary point of f, contradicting our choice of δ). Then, at each iteration $k \in K$ with $k \geq \tilde{k}$, since no contraction or shrinking step is taken so that Step 1 is entered only once, we have $m_k = \bar{m}$.

For each $k \in K$, let $\hat{f}^k$ denote the $\hat{f}$ computed when Step 1 is first entered during iteration k. If there exists a subsequence K' of K satisfying

$$\lim_{k \to \infty, k \in K'} \left(\hat{f}^k - f_{\max}(S^k) \right) / \sigma_k \geq 0, \tag{3.12}$$

then, by passing into a subsequence if necessary, we can assume from the boundedness of $\{x^k\}_{k\in K}$ that $\{x^k\}_{k\in K'} \to$ some $\bar{x} \in B(x^\infty, \delta)$ and from (2.4) that

$$S^k = \{s_1^k, \dots, s_{n+1}^k\}, \qquad x^k = \sum_{i\in I_0} \mu_i^k s_i^k, \qquad \hat{f}^k = \sum_{i\in I_0} \mu_i^k f(s_i^k)$$

for all $k \in K'$, where $I_0 \subset N = \{1, \dots, n+1\}$ and, for each $k \in K'$, $\{s_i^k\}_{i\in N}$ is some set of points in $\Re^n$ and $\{\mu_i^k\}_{i\in I_0}$ is some set of scalars exceeding $\theta_{\rm r}/|I_0|$ and summing to 1. Also, we can assume that

$$\{(s_j^k - s_i^k)/\sigma_k\}_{k\in K'} \to d_{ij}, \quad i \in I_0,\ j \in N,$$

for some set of n-vectors $\{d_{ij}\}_{i\in I_0, j\in N}$ in $\Re^n$, and that, for each $i \in I_0$, $\{\mu_i^k\}_{k\in K'}$ converges to some positive scalar μ_i. Then, we would have from (3.12) (and passing into the limit as $k \to \infty$, $k \in K'$) that

$$-\sum_{i\in I_0} \mu_i \nabla f(\bar{x})^T d_{ij} \geq 0, \quad j \notin I_0,$$

and, as in the proof of (3.9), that $\nabla f(\bar{x})^T d_{ij} \geq 0$ for all $i \in I_0$, $j \notin I_0$. These together would imply

$$\nabla f(\bar{x})^T d_{ij} = 0, \quad i \in I_0,\ j \notin I_0,$$

and, since $\text{von}(S^k) \geq \nu$ for all k so that the elements of $\{d_{ij}\}_{i\in I_0, j\notin I_0}$ span $\Re^n$ as argued earlier, it would follow that $\nabla f(\bar{x}) = 0$, a contradiction of $B(x^\infty, \delta)$ containing no stationary point of f. Thus, there cannot exist a subsequence K' of K satisfying (3.12) or, equivalently, we must have

$$\lim_{t\to\infty} \sup_{k\in K, k\geq t} \left\{ \left(\hat{f}^k - f_{\max}(S^k) \right) / \sigma_k \right\} < 0. \tag{3.13}$$

On the other hand, at each iteration $k \in K$ with $k \geq \tilde{k}$, since no contraction or shrinking step is taken, then during the first pass through Steps 1–2 (so $|S_{\rm good}| = m = \bar{m}$ and $(S, \hat{x}, \Delta) = (S^k, x^k, \sigma_k)$), we have that (2.8)–(2.9) hold and either (i) $S^{k+1} = S_{\rm good} \cup S_{\rm r}$ or (ii) (2.10) holds and $S^{k+1} = S_{\rm good} \cup S_{\rm e}$. In case (i), we have from $\Sigma_{\rm r} \subseteq S_{\rm r}$ and (2.6) and (2.9) that

$$F_1(S_{\rm r}) = f_{\min}(S_{\rm r}) \leq f_{\min}(\Sigma_{\rm r}) \leq F_m(S_{\rm good}) + \theta_{\rm r}\theta_{\rm bad}(\hat{f} - f_{\max}(S)) + \beta(\Delta),$$

and from (2.2) and $S_{\rm good} \cup S_{\rm bad} = S = S^k$ that $F_i(S_{\rm good}) = F_i(S) = F_i(S^k)$ for $i = 1, \dots, m$, so the facts $m = \bar{m}$ and $S^{k+1} = S_{\rm good} \cup S_{\rm r}$ yield $F_i(S^{k+1}) \leq F_i(S^k)$ for $i = 1, \dots, \bar{m}$, and

$$\begin{aligned}\sum_{i=1}^{\bar{m}} (F_i(S^{k+1}) - F_i(S^k)) &\leq \sum_{i=1}^{\bar{m}-1} (F_i(S_{\rm good}) - F_i(S^k)) + (F_1(S_{\rm r}) - F_{\bar{m}}(S^k)) \\ &\leq \theta_{\rm r}\theta_{\rm bad}(\hat{f} - f_{\max}(S)) + \beta(\Delta) \\ &= \theta_{\rm r}\theta_{\rm bad}\left(\hat{f}^k - f_{\max}(S^k)\right) + \beta(\sigma_k),\end{aligned}$$

where the first inequality also uses the observation that the sum of the first $\bar{m}$ components of $F(S^{k+1})$ is less than or equal to the sum of any $\bar{m}$ components of $F(S^k)$. A similar argument shows that the above relation holds in case (ii) also. Since $\{\sigma_k\} \to 0$ and $\beta \in \Phi$ so that $\{\beta(\sigma_k)/\sigma_k\} \to 0$, then (3.13) implies that there exist $\bar{k} \geq \tilde{k}$ and a constant $c < 0$ such that the right-hand side of the above relation is bounded above by $c\sigma_k$ for all $k \in K$ with $k \geq \bar{k}$; i.e.,

$$\sum_{i=1}^{\bar{m}} (F_i(S^{k+1}) - F_i(S^k)) \leq c\sigma_k \quad \forall k \in K \text{ with } k \geq \bar{k}. \tag{3.14}$$

On the other hand, since $\{\sigma_k\} \to 0$ (so there is an infinite number of iterations in which a contraction or shrinking step is taken), then $\{1, 2, \ldots\}\backslash K$ is also an infinite subsequence, so $\{x^k\}$ enters and exits $B(x^\infty, \delta)$ an infinite number of times. Since x^∞ is a cluster point of $\{x^k\}$ and $\{\sigma_k\} \to 0$, this implies that x^k must cross between R_1 and R_3 through R_2 an infinite number of times, where we let

$$\begin{aligned}
R_1 &= \{x \in \Re^n : 2\delta/3 \leq \|x - x^\infty\| \leq \delta\}, \\
R_2 &= \{x \in \Re^n : \delta/3 < \|x - x^\infty\| < 2\delta/3\}, \\
R_3 &= B(x^\infty, \delta/3).
\end{aligned}$$

More precisely, there exist integers $k_1, l_1, k_2, l_2, \ldots$ such that $\bar{k} < k_1 < l_1 < k_2 < l_2 < \cdots$ and

$$x^{k_t} \in R_1, \qquad x^{k_t+1} \in R_2, \ \ldots, \ x^{l_t-1} \in R_2, \qquad x^{l_t} \in R_3$$

for $t = 1, 2, \ldots$. Then, we have $\{k_t, k_t + 1, \ldots, l_t\} \subset \{k \in K : k \geq \bar{k}\}$ for all t, so (3.14) yields

$$\begin{aligned}
\sum_{i=1}^{\bar{m}} (F_i(S^{l_t}) - F_i(S^{k_t})) &= \left[\sum_{i=1}^{\bar{m}} (F_i(S^{l_t}) - F_i(S^{l_t-1}))\right] + \cdots + \left[\sum_{i=1}^{\bar{m}} (F_i(S^{k_t+1}) - F_i(S^{k_t}))\right] \\
&\leq c\sigma_{l_t-1} + \cdots + c\sigma_{k_t} \\
&\leq c\|x^{l_t} - x^{l_t-1}\|/(\gamma_\mathrm{e} + 1) + \cdots + c\|x^{k_t+1} - x^{k_t}\|/(\gamma_\mathrm{e} + 1) \\
&\leq c\|x^{l_t} - x^{k_t}\|/(\gamma_\mathrm{e} + 1) \\
&\leq c\delta/(3(\gamma_\mathrm{e} + 1)),
\end{aligned}$$

where the second inequality follows from $c < 0$ and the observation $\|x^{k+1} - x^k\| \leq \sigma_{k+1} + \sigma_k \leq (\gamma_\mathrm{e} + 1)\sigma_k$ for all k; the third inequality follows from the triangle inequality; the last inequality follows from the observation that the Euclidean distance between a point in R_1 and a point in R_3 is at least $\delta/3$. Since the above inequality holds for all $t = 1, 2, \ldots$, we see that $\{\sum_{i=1}^{\bar{m}} F_i(S^k)\}_{k \in K}$ cannot converge. On the other hand, we have that $\{f_{\min}(S^k)\}$ converges (since it is monotonically decreasing and bounded below) and $\{\sigma_k\} \to 0$, so, by $x^k \in B(x^\infty, \delta)$ for all $k \in K$ and the uniform continuity of f on $B(x^\infty, \delta)$, we see that $\{F_i(S^k)\}_{k \in K}$ also converges (to the same limit as does $\{f_{\min}(S^k)\}$) for $i = 1, \ldots, \bar{m}$. This contradicts the nonconvergence of $\{\sum_{i=1}^{\bar{m}} F_i(S^k)\}_{k \in K}$. □

By combining Lemma 3.1 and Theorem 3.2, we immediately obtain the following convergence result for the FDSS method. Recall that $\bar{m}_k$ can be chosen freely between 1 and n, so, in particular, we can choose $\bar{m}_k$ to be any constant between 1 and n.

Corollary 3.3. *Assume that* $\inf_{x \in \Re^n} f(x) > -\infty$ *and* f *is continuously differentiable on* $\Re^n$. *Let* $\{(S^k, x^k, \Delta_k, \bar{m}_k, m_k)\}_{k=0,1,\ldots}$ *be generated by the FDSS method. Then the following hold:*

(a) *If* $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ *is bounded, then either the method quits at some iteration* k *with a stationary point of* f *or the method does not quit at some iteration and at least one cluster point of* $\{x^k\}$ *is a stationary point of* f.

(b) *If we choose, at all iterations* k *beyond some number,* $m = 1$ *each time we enter Step* 1*, then either the method quits at some iteration* k *with a stationary point of* f *or the method does not quit at some iteration and every cluster point of* $\{x^k\}$ *is a stationary point of* f. *The same conclusion holds if* f *is uniformly continuous on* $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ *and we choose, at all iterations* k *beyond some number,* $\bar{m}_k$ *to be a constant between* 1 *and* n.

As a consequence of Corollary 3.3, part (a), we have that if, in addition to the assumptions therein, it is assumed that f has a unique stationary point on $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$, then $\{x^k\}$ generated by the FDSS method converges to this stationary point (which in fact would be the global minimizer of f). We note that the assumption that f be uniformly continuous on $\{x \in \Re^n : f(x) \le f_{\min}(S^0)\}$ is fairly mild and is satisfied by many functions that are continuously differentiable and bounded below. An example of a function that is continuously differentiable and bounded below but does not satisfy this assumption is $f(x) = e^{-x} + \cos(x^2)$ with $S^0 = \{-1, 0\}$ (so $f_{\min}(S^0) = 2$). We have $f(\sqrt{2\pi k}) \to 1$ as $k \to \infty$ while $f(\sqrt{2\pi k} + \epsilon_k) \to -1$ as $k \to \infty$, where $\epsilon_k = \sqrt{\pi}/(2\sqrt{2k}) \to 0$ as $k \to \infty$.

4. Preliminary numerical experience. While the focus of our work is on the convergence analysis of the FDSS method, we also implemented and tested the method to gain an understanding of its empirical behavior. We report our preliminary experience below.

First we describe the implementation. We coded the FDSS method in Matlab with $\hat{x}$ and $S_{\rm r}$ chosen by (2.15) and (2.16) and with $S_{\rm e}$ and $S_{\rm c}$ chosen by

$$S_{\rm e} = 3\hat{x} - 2S_{\rm bad}, \qquad S_{\rm c} = \begin{cases} 1.5\hat{x} - .5S_{\rm bad} & \text{if } f_{\min}(\Sigma_{\rm r}) < f_{\min}(S_{\rm bad}), \\ .5\hat{x} + .5S_{\rm bad} & \text{otherwise,} \end{cases}$$

as motivated by the Nelder–Mead method. Also, for a given $\Sigma_{\rm bad}$, we chose

$$\Sigma_{\rm r} = 2\hat{x} - \Sigma_{\rm bad},$$

and, accordingly, $\tau_{\rm r} = 1$ and $\gamma_{\rm e} = 2$. Lastly, we chose $\theta_{\rm r} = .01$, $\nu = 10^{-5}$, $\gamma_{\rm s} = .5$, $\alpha(t) = 10^{-5}\min\{.5t^2, t\}$, $\beta(t) = 10^6 t^2$, and, whenever we had a choice of going to either Step 4 or Step 5 from Step 2, we always went to Step 4. (This still leaves us with the freedom to choose m, S_0, and $\Sigma_{\rm bad}$.) We run our Matlab code on four test functions: two functions of Powell [16, 17] ($n = 4$ and $n = 3$, respectively), a function of Rosenbrock [18] ($n = 2$), and a quadratic function of Zangwill [31] ($n = 3$). For each test function, the initial simplex was constructed by taking the starting point used in the above references and adding to this point the ith unit coordinate vectors in $\Re^n$ for $i = 1, \ldots, n$. Termination occurs when the current simplex $S = \{s_1, \ldots, s_{n+1}\}$ satisfies (2.17) and (2.18) with $\epsilon = 10^{-3}$. This yields $\nabla f(s_1) \approx 0$ upon termination.

Next we describe our numerical experience. We found that the best performance of the implemented method, as measured by the total number of function evaluations, was achieved by choosing m as high as possible, i.e., $m = \min\{n, l(F(S), F(S^k))\}$, and choosing the sets S_0 and $\Sigma_{\rm bad}$ as large as possible, i.e., $S_0 = S_{\rm good}$ and $\Sigma_{\rm bad} = S_{\rm bad}$. (Smaller $\Sigma_{\rm bad}$ reduces the number of function evaluations per check for reflection, but it increases the number of iterations and total number of function evaluations.) In other words, the implementation that most closely resembles the Nelder–Mead

TABLE 1

Performance of the Nelder–Mead method and a specific implementation of the FDSS method on four test functions.

	Nelder–Mead method		FDSS method	
Function	#func. eval.*	f-value**	#func. eval.*	f-value**
Powell1 $(n = 4)$	236	$4.3 \cdot 10^{-6}$	230	$1.1 \cdot 10^{-9}$
Powell2 $(n = 3)$	95	-3.0000	95	-3.0000
Rosenbrock $(n = 2)$	149	$1.1 \cdot 10^{-7}$	149	$1.1 \cdot 10^{-7}$
Zangwill $(n = 3)$	86	$3.1 \cdot 10^{-7}$	86	$3.1 \cdot 10^{-7}$

* This is the number of times that f was evaluated upon termination.

** This is the value of f at the best vertex upon termination.

method worked the best.[1] We also found the criterion (2.19)–(2.20) to yield better performance than (2.10). The resulting implementation is then effectively the Nelder–Mead method with two modifications: strict descent is replaced by fortified descent and the interior angles of the simplex are kept away from zero. Table 1 tabulates the performance of this implementation, as well as our Matlab implementation of the Nelder–Mead method (as interpreted from the original paper [15]) using the same initial simplex and termination criterion on the test functions. As can be seen from Table 1, the two methods have identical performance on the last three functions. On the first function, the FDSS method performed slightly better, apparently due to the interior angles of the simplex being kept away from zero.

As with the Nelder–Mead method, the above implementation of the FDSS method can suffer from poor performance even for moderately large n. In particular, the method also exhibited slow convergence on a quadratic example of Wright [28, section 7] in which $n = 32$ and $f(x) = x_1^2 + \cdots + x_{32}^2$, and the initial simplex was constructed by taking $(1, 2, \ldots, 32)^T$ and adding to this point the ith unit coordinate vector in $\Re^{32}$ for $i = 1, \ldots, 32$. This shows that alternative techniques, such as those described in [3, 24, 28], are needed to make simplicial search methods effective on higher dimensional problems.

Acknowledgment. I thank the SIOPT associate editor, Margaret Wright, for her editorial efforts and, in particular, for her detailed comments and suggestions on an earlier version of this paper.

REFERENCES

[1] G. E. P. BOX AND K. B. WILSON, *On the experimental attainment of optimum conditions,* J. Roy. Statist. Soc. Ser. B, XIII (1951), pp. 1–45.

[2] M. J. BOX, *A new method of constrained optimization and a comparison with other methods*, Comput. J., 8 (1965), pp. 42–52.

[3] A. G. BUCKLEY AND H. MA, *A Derivative-Free Algorithm for Parallel and Sequential Optimization*, Computer Science Department Report, University of Victoria, BC, Canada, October 1994.

[4] J. CÉA, *Optimisation: Théorie et algorithmes*, Dunod, Paris, 1971.

[5] A. P. DAMBRAUSKAS, *The simplex optimization method with variable step*, Engrg. Cybernet., 1 (1970), pp. 28–36.

[1] However, we caution that these results are only preliminary. For example, if all vertices of the current simplex, except for one, have very high f-value, then it might be worth choosing $m = 1$, i.e., reflect all vertices through the best vertex, rather than choosing m to be as high as possible.

[6] J. E. Dennis, Jr., and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448–474.

[7] J. E. Dennis, Jr., and D. J. Woods, *Optimization on microcomputers: The Nelder–Mead simplex algorithm*, in New Computing Environments: Microcomputers in Large-Scale Computing, A. Wouk, ed., SIAM, Philadelphia, PA, 1987, pp. 116–122.

[8] R. Fletcher, *Function minimization without evaluating derivatives—a review*, Comput. J., 8 (1965), pp. 33–41.

[9] P. Gritzmann and V. Klee, *On the complexity of some basic problems in computational convexity:* II. *Volume and mixed volumes*, in Polytopes: Abstract, Convex and Computational, T. Bisztriczky, P. McMullen, R. Schneider, and A. Ivic Weiss, eds., Kluwer, Boston, 1994, pp. 373–466.

[10] R. Hooke and T. A. Jeeves, *"Direct search" solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–229.

[11] A. G. Kuznetsov, *Nonlinear Optimization Toolbox*, Report OUEL 1936/92, Department of Engineering Science, University of Oxford, UK, 1992.

[12] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, *Convergence properties of the Nelder–Mead simplex method in low dimensions*, SIAM J. Optim., 9 (1999), pp. 112–147.

[13] O. L. Mangasarian, *Nonlinear Programming*, McGraw–Hill, New York, 1969.

[14] K. I. M. McKinnon, *Convergence of the Nelder–Mead simplex method to a nonstationary point*, SIAM J. Optim., 9 (1999), pp. 148–158.

[15] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[16] M. J. D. Powell, *An iterative method for finding stationary values of a function of several variables*, Comput. J., 5 (1962), p. 147.

[17] M. J. D. Powell, *An efficient method for finding the minimum of a function of several variables without calculating derivatives*, Comput. J., 7 (1964), pp. 155–162.

[18] H. H. Rosenbrock, *An automatic method for finding the greatest or least value of a function*, Comput. J., 3 (1960), pp. 175–184.

[19] A. S. Rykov, *Simplex direct search algorithms*, Automat. Remote Control, 41 (1980), pp. 784–793.

[20] A. S. Rykov, *Simplex methods of direct search*, Engrg. Cybernet., 18 (1980), pp. 12–18.

[21] A. S. Rykov, *Design principles of controlled direct-search methods*, Soviet Phys. Dokl., 27 (1982), pp. 794–796.

[22] A. S. Rykov, *Simplex algorithms for unconstrained optimization*, Problems Control Inform. Theory, 12 (1983), pp. 195–208.

[23] W. Spendley, G. R. Hext, and F. R. Himsworth, *Sequential application of simplex designs in optimisation and evolutionary operation*, Technometrica, 4 (1962), pp. 441–461.

[24] V. Torczon, *Multi-Directional Search: A Direct Search Algorithm for Parallel Machines*, Ph.D. thesis, Rice University, Houston, TX, 1989; available as Technical Report 90-7, Department of Mathematical Sciences, Rice University, Houston, TX.

[25] V. Torczon, *On the convergence of the multidirectional search algorithm*, SIAM J. Optim., 1 (1991), pp. 123–145.

[26] V. Torczon, *On the convergence of pattern search algorithms*, SIAM J. Optim., 7 (1997), pp. 1–25.

[27] D. J. Woods, *An Interactive Approach for Solving Multi-Objective Optimization Problems*, Ph.D. thesis, Rice University, Houston, TX, 1985; available as Technical Report 85-5, Department of Mathematical Sciences, Rice University, Houston, TX.

[28] M. H. Wright, *Direct search methods: Once scorned, now respectable*, in Numerical Analysis 1995 (Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis), Pitman Res. Notes Math. Ser. 344, D. F. Griffiths and G. A. Watson, eds., Longman, Harlow, UK, 1996, pp. 191–208.

[29] W.-C. Yu, *The convergent property of the simplex evolutionary techniques*, Sci. Sinica 1979, Special Issue I on Math., pp. 69–77 (in Chinese).

[30] W.-C. Yu, *Positive basis and a class of direct search techniques*, Sci. Sinica 1979, Special Issue I on Math., pp. 53–68 (in Chinese).

[31] W. I. Zangwill, *Minimizing a function without calculating derivatives*, Comput. J., 10 (1967), pp. 293–296.

SIAM J. OPTIM.
Vol. 10, No. 1, pp. 289–313

DECREASING FUNCTIONS WITH APPLICATIONS TO PENALIZATION*

A. M. RUBINOV†, B. M. GLOVER‡, AND X. Q. YANG§

Abstract. The theory of increasing positively homogeneous functions defined on the positive orthant is applied to the class of decreasing functions. A multiplicative version of the inf-convolution operation is studied for decreasing functions. Modified penalty functions for some constrained optimization problems are introduced that are in general nonlinear with respect to the objective function of the original problem. As the perturbation function of a constrained optimization problem is decreasing, the theory of decreasing functions is subsequently applied to the study of modified penalty functions, the zero duality gap property, and the exact penalization.

Key words. decreasing functions, IPH functions, multiplicative inf-convolution, modified penalty functions, exact penalization

AMS subject classifications. 90C30, 65K05

PII. S1052623497326095

1. Introduction. In this paper we study positive decreasing functions defined on the positive orthant $\mathbb{R}^n_{++}$ and their applications to nonlinear penalization formed by increasing positively homogeneous (IPH) functions of the first degree. There exists a natural isomorphism between the ordered space of all positive decreasing functions and the ordered space of all IPH functions. The theory of abstract convexity (see [9, 12, 18]) allows us to consider duality in various nonconvex situations. Recently [16] a duality theory based on abstract convexity with respect to the so-called min-type functions was developed for IPH functions. The isomorphism between IPH functions and decreasing functions allows us to apply this theory in the study of positive decreasing functions. This approach is developed in the first part of the paper. We also introduce and study a multiplicative analogue of the inf-convolution operation. This operation exhibits very nice properties in the class of decreasing functions similar to the usual "sum" convolution, which exhibits very nice properties in the class of convex functions.

The second part of the paper is devoted to the study of modified penalty functions for the single constraint problem:

$$f_o(x) \longrightarrow \inf \quad \text{subject to} \quad x \in X,\ f_1(x) \leq 0$$

with a real-valued positive objective function f_o and a real-valued constraint function f_1.

The perturbation function

$$\tilde{\beta}(y) = \inf\{f_o(x) :\ x \in X, f_i(x) \leq y_i,\ i = 1, \dots, m\}, \quad y = (y_1, \dots, y_m),$$

*Received by the editors August 15, 1997; accepted for publication (in revised form) December 14, 1998; published electronically November 29, 1999. This research was supported by grants from the Australian Research Council.

http://www.siam.org/journals/siopt/10-1/32609.html

†School of Information Technology and Mathematical Sciences, The University of Ballarat, Ballarat, VIC 3353, Australia (amr@ballarat.edu.au).

‡Department of Mathematics, Curtin University of Technology, Perth, WA 6845, Australia (b.glover@ballarat.edu.au).

§Department of Mathematics, The University of Western Australia, Nedlands, WA 6907, Australia (yangx@maths.uwa.edu.au).

has useful applications in the study of the nonlinear programming problem:

$$f_o(x) \longrightarrow \inf \quad \text{subject to} \quad x \in X,\ f_i(x) \le 0,\ i = 1, \dots, m$$

(see for example [3, 5, 6, 8, 10, 11, 14] and the references therein). The perturbation function is decreasing: $y^1 \ge y^2 \implies \beta(y^1) \le \beta(y^2)$. Consequently, the study of perturbation functions should be based on a theory of decreasing functions.

It is well known that all constraints f_i can be convoluted into a single constraint (see [19] for a detailed discussion). For example, we can use a convolution by maximum: $\max_i f_i(x) \le 0 \iff f_i(x) \le 0$ for all i. So we restrict ourselves to the problem with a single constraint.

The classical penalty function for an optimization problem with a single constraint is formed by means of the classical convolution function $p(y_1, y_2) = y_1 + y_2$. Sometimes it is more convenient to consider the convolution by an increasing function p with some additional properties. This approach has been developed by Rubinov, Glover, and Yang [17] and Andramonov [1]. In such a case the modified penalty function $\mathcal{L}_p^+(x, d)$ has the form

$$\mathcal{L}_p^+(x, d) = p(f_o(x), d \max(f_1(x), 0)). \tag{1.1}$$

Among the many issues that arise in connection with such a setting, we indicate the following:

1. how to obtain conditions ensuring the zero duality gap property;
2. how to obtain conditions ensuring the exact penalization;
3. how to find convolution functions p such that an exact penalty parameter for the function $\mathcal{L}_p^+$ is substantially smaller than that of the classical function;
4. how to find convolution functions p such that the penalty function $\mathcal{L}_p^+$ is smooth at the solution.

The corresponding questions for the classical function (except question 3) have been discussed, for example, in [2, 5, 7, 11, 14].

The first three questions are addressed in this paper. We consider the penalty function of the form (1.1) involving only an IPH function p with some natural properties.

It is demonstrated that for a large class of IPH functions, p, the zero duality gap property

$$\inf\{f_o(x) :\ x \in X,\ f_1(x) \le 0\} = \sup_{d>0} \inf_{x \in X} \mathcal{L}_p^+(x, d)$$

holds if and only if the perturbation function

$$\beta(y) = \inf\{f_o(x) :\ f_1(x) \le y\}, \qquad y \ge 0,$$

is lower semicontinuous at the origin.

Thus the zero duality gap property depends only on the problem itself but does not depend on an outer convolution function from a very large class of such functions. The proof of this fact is based on the theory of multiplicative inf-convolution developed in the first part of the paper. In contrast to this result, we show that the exact penalization essentially depends on an outer convolution function. In particular, it is proved that as a rule the penalization with respect to the function $p_{+\infty}(\alpha, y) = \max\{\alpha, y\}$ is not exact.

The convolution with respect to the family of IPH functions

$$p_k(\delta, y) = (\delta^k + y^k)^{\frac{1}{k}} \quad (0 < k < +\infty)$$

is considered. We study especially the exact penalization by $p_{\frac{1}{2}}$ and demonstrate that this penalization always can be accomplished with a smaller penalty parameter than that of the classical convolution function p_1. We also obtain an asymptotically sharp estimate of the ratio $\bar{d}_{\frac{1}{2}}/\bar{d}_1$, where $\bar{d}_k$ is the least exact penalization parameter with respect to p_k. This estimate allows us to draw the following conclusion: *If the constrained minimum of the objective function is not very far from the unconstrained minimum of this function, then the penalization by $p_{\frac{1}{2}}$ can be accomplished with a substantially smaller exact penalty parameter $\bar{d}_{\frac{1}{2}}$ than $\bar{d}_1$.* Thus for many problems the ill-conditionedness of the penalty function with a large penalty parameter can be overcome using $p_{\frac{1}{2}}$.

We prove that the class of IPH functions is sufficiently large to provide the exact penalization: an exact modified penalty function can be found for a given problem under some very mild assumptions.

The paper is structured as follows. The class of IPH functions is studied in section 2 and decreasing functions in section 3. The multiplicative inf-convolution of decreasing functions is introduced and studied in section 4. The modified penalty function is discussed in section 5. Section 6 is devoted to the perturbation function, and its links with the modified penalty function and exact penalization are investigated in section 7.

2. Increasing positively homogeneous functions. Let I be a finite set of indices. We shall use the following notation:

- $\mathbb{R}^I$ is the space of all vectors $(x_i)_{i\in I}$;
- x_i is the ith coordinate of a vector $x \in \mathbb{R}^I$;
- if $x, y \in \mathbb{R}^I$, then $x \geq y \iff x_i \geq y_i$ for all $i \in I$;
- if $x, y \in \mathbb{R}^I$, then $x \gg y \iff x_i > y_i$ for all $i \in I$;
- $\mathbb{R}^I_+ = \{x = (x_i) \in \mathbb{R}^I : x \geq 0\}$;
- $\mathbb{R}^I_{++} = \{x = (x_i) \in \mathbb{R}^I : x \gg 0\}$.

If I consists of n elements we will also use the notation $\mathbb{R}^n$, $\mathbb{R}^n_+$, and $\mathbb{R}^n_{++}$ instead of $\mathbb{R}^I$, $\mathbb{R}^I_+$, and $\mathbb{R}^I_{++}$, respectively.

A function p defined on either the cone $\mathbb{R}^I_{++}$ or the cone $\mathbb{R}^I_+$ and mapping into $\mathbb{R}_{+\infty} := \mathbb{R}_+ \cup \{+\infty\}$ is called an IPH function if p is increasing ($x \geq y \implies p(x) \geq p(y)$) and positively homogeneous of degree 1 ($p(\lambda x) = \lambda p(x)$ for $\lambda > 0$) and if there is a point $y \in \mathbb{R}^I_{++}$ such that $p(y) < +\infty$. If p is an IPH function defined on $\mathbb{R}^I_+$, then the restriction of p to $\mathbb{R}^I_{++}$ is also an IPH function. We shall denote the class of all IPH functions defined on $\mathbb{R}^I_{++}$ by IPH_{++}.

It is known (see [16]) that every function $p \in IPH_{++}$ is continuous on $\mathbb{R}^I_{++}$.

The simplest example of an IPH function is a function of the form $\ell(y) = \langle \ell, y \rangle$, where $\ell = (\ell_1, \dots, \ell_m) \in \mathbb{R}^I_{++}$ and

$$\langle \ell, y \rangle = \min_{i \in I} \ell_i y_i. \tag{2.1}$$

The main tool in the study of IPH functions will be the so-called support set. In particular, we will use support sets of IPH functions defined on $\mathbb{R}^I_{++}$.

Definition 2.1 (see [16]). *Let $p \in IPH_{++}$. The set*

$$\operatorname{supp}(p) = \{\ell \in \mathbb{R}^I_{++} : \langle \ell, y \rangle \leq p(y) \textit{ for all } y \in \mathbb{R}^I_{++}\} \tag{2.2}$$

is called the support set of the function p.

For IPH functions defined on the cone $\mathbb{R}^I_+$ it is possible to define a support set in different ways. One of them has been studied in [15]. However, it will be more convenient to use the following definition in this paper.

DEFINITION 2.2. *Let* $\bar{p}$ *be an IPH function defined on* $\mathbb{R}^I_+$ *and let* p *be the restriction of the function* $\bar{p}$ *to the cone* $\mathbb{R}^I_{++}$. *Then the support set of the function* p*, defined by* (2.2)*, is called the support set of the function* $\bar{p}$.

We shall study in this section only support sets for IPH functions defined on $\mathbb{R}^I_{++}$. The following result shows that each IPH function can be reconstructed from its support set.

THEOREM 2.1 (see [16]). *Let* $p \in IPH_{++}$. *Then* $p(y) = \max\{\langle \ell, y\rangle : \ell \in \operatorname{supp}(p)\}$ *for all* $y \in \mathbb{R}^I_{++}$.

A subset U of $\mathbb{R}^I_{++}$ is called *normal* if $\ell_1 \in U$, $\ell_2 \in \mathbb{R}^I_{++}$, and $\ell_1 \geq \ell_2$ imply $\ell_2 \in U$. A subset U of $\mathbb{R}^I_{++}$ is called closed if it is closed in the topological space $\mathbb{R}^I_{++}$.

THEOREM 2.2 (see [16]). *Let* $U \subset \mathbb{R}^I_{++}$. *Then* $U = \operatorname{supp}(p)$ *for some* $p \in IPH_{++}$ *if and only if* U *is normal and closed.*

For $a, x \in \mathbb{R}^I_{++}$, $U \subset \mathbb{R}^I_{++}$ we shall require the following notational convention:

$$a \cdot x = (a_i x_i)_{i\in I}, \qquad a \cdot U = \{a \cdot u : u \in U\}, \qquad \frac{a}{x} = \left(\frac{a_i}{x_i}\right)_{i \in I}. \tag{2.3}$$

If $x \in \mathbb{R}^I_{++}$, then

$$\frac{1}{x} \equiv x^{-1} = \left(\frac{1}{x_i}\right)_{i\in I}.$$

The following result provides an explicit description of the support set.

THEOREM 2.3 (see [16]). *Let* p *be an IPH function defined on the cone* $\mathbb{R}^I_{++}$. *Then*

$$\operatorname{supp}(p) = \{\ell \in \mathbb{R}^I_{++} : p(\ell^{-1}) \geq 1\}.$$

We now describe some properties of support sets. It follows immediately from the definition that

$$p_1 \leq p_2 \iff \operatorname{supp}(p_1) \subset \operatorname{supp}(p_2) \quad \text{for} \quad p_1, p_2 \in IPH_{++}.$$

PROPOSITION 2.1. *Let* $p(x) = \inf_{\alpha\in A} p_\alpha(x)$*, where* $(p_\alpha)_{\alpha\in A}$ *is a family of IPH functions. Then*

$$\operatorname{supp}(p) = \cap_{\alpha\in A} \operatorname{supp}(p_\alpha).$$

Proof. Since the sets $\operatorname{supp}(p_\alpha)$ are normal and closed it follows that their intersection is also normal and closed. Theorem 2.2 shows that there exists an IPH function $\tilde{p}$ such that $\operatorname{supp}(\tilde{p}) = \cap_{\alpha\in A} \operatorname{supp}(p_\alpha)$. Since $\tilde{p} \leq p_\alpha$ for all $\alpha \in A$ it follows that $\tilde{p} \leq \inf_{\alpha\in A} p_\alpha = p$. The function $p(x) = \inf_{\alpha\in A} p_\alpha(x)$ is IPH. Since $p \leq p_\alpha$ for all α, it follows that $\operatorname{supp}(p) \subset \cap_{\alpha\in A} \operatorname{supp}(p_\alpha) = \operatorname{supp}(\tilde{p})$, so $p \leq \tilde{p}$. □

Let $a = (a_i)_{i\in I} \in \mathbb{R}^I_{++}$ and $p \in IPH_{++}$. We will require in what follows the function p^a, where $p^a(y) = p(a \cdot y)$ and $a \cdot y$ is defined by (2.3).

PROPOSITION 2.2. $\operatorname{supp}(p^a) = a \cdot \operatorname{supp}(p) \equiv \{a \cdot \ell : \ell \in \operatorname{supp}(p)\}$.

Proof. Let $\ell \in \operatorname{supp}(p^a)$, $y = (y_i)_{i\in I} \in \mathbb{R}^I_{++}$, and $z = (z_i)_{i\in I} = a \cdot y$. Then

$$p(z) = p(a \cdot y) \geq \langle \ell, y \rangle = \min_{i\in I} \ell_i y_i = \min_{i\in I} \frac{\ell_i}{a_i} z_i.$$

Thus the vector $\ell' = l/a$ belongs to $\operatorname{supp}(p)$. Since $\ell = a \cdot \ell'$ it follows that $\ell \in a \cdot \operatorname{supp}(p)$. □

LEMMA 2.1. *If U is a normal set and $a \geq a' \gg 0$, then $a \cdot U \supset a' \cdot U$.*

Proof. Let $\ell \in a' \cdot U$. Then there exists $u \in U$ such that $\ell = a' \cdot u$. Thus $(\ell/a') \in U$. Since U is normal it follows that $(\ell/a) \in U$, so $\ell \in a \cdot U$. □

We now give some examples of IPH functions and support sets.

EXAMPLE 2.1. Let $p(y) = \max_{i\in I} a_i y_i$ with $a = (a_i)_{i\in I}$ and $a_i > 0$. Clearly $p \in IPH_{++}$. Applying Theorem 2.3 we can easily conclude that the support set $\operatorname{supp}(p)$ coincides with the following set V_a:

$$V_a = \cup_{i\in I}\{\ell = (\ell_1, \dots, \ell_m) \in \mathbb{R}^I_{++} : \ell_i \leq a_i\} = \left\{\ell \in \mathbb{R}^I_{++} : \min_{i\in I} \frac{\ell_i}{a_i} \leq 1\right\}. \tag{2.4}$$

Assume, more generally, that $a_i \geq 0$ for all $i \in I$. Let $I_a = \{i \in I : a_i > 0\}$. It is easy to check that

$$V_a = \left\{\ell \in \mathbb{R}^I_{++} : \min_{i\in I_a} \frac{\ell_i}{a_i} \leq 1\right\}.$$

EXAMPLE 2.2. Let $0 < k < +\infty$ and

$$p_k(x) = \left(\sum_{i\in I} x_i^k\right)^{\frac{1}{k}} \quad \text{for all} \quad x \in \mathbb{R}^I_{++}.$$

Clearly $p_k \in IPH_{++}$. Applying Theorem 2.3 we obtain the following:

$$\operatorname{supp}(p_k) = \left\{\ell \in \mathbb{R}^I_{++} : \sum_{i\in I} \frac{1}{{\ell_i}^k} \geq 1\right\}. \tag{2.5}$$

If $k_1 \geq k_2$, then $p_{k_1} \geq p_{k_2}$; thus $\operatorname{supp}(p_{k_1}) \supset \operatorname{supp}(p_{k_2})$. For the function $p_\infty(x) := \max_{i\in I} x_i$ we have $p_\infty(x) = \inf_{k>0} p_k(x)$. From Proposition 2.1 it follows that

$$\operatorname{supp}(p_\infty) = \bigcap_{k>0} \operatorname{supp}(p_k).$$

Thus, from (2.4) and (2.5),

$$\left\{\ell \in \mathbb{R}^I_{++} : \min_{i\in I} \ell_i \leq 1\right\} = \bigcap_{0<k<+\infty} \left\{\ell \in \mathbb{R}^I_{++} : \sum_{i\in I} \frac{1}{{\ell_i}^k} \geq 1\right\}.$$

Clearly this equality can also be verified directly.

3. Decreasing functions. Let $I = \{1, \dots, m\}$, $I' = \{0\} \cup I$ and let U be a normal closed subset of $\mathbb{R}^{I'}_{++}$. Consider the function

$$g_U(y) = \sup\{\alpha : \ (\alpha, y) \in U\}, \qquad y \in \mathbb{R}^I_{++}. \tag{3.1}$$

This function maps $\mathbb{R}^I_{++}$ into $\mathbb{R}_{+\infty} = \mathbb{R}_+ \cup \{+\infty\}$. Since U is closed it follows that $g_U(y) = \max\{\alpha : (\alpha, y) \in U\}$ whenever $g_U(y) < +\infty$. By normality of the set U we obtain

$$U = \{(\alpha, y) \in \mathbb{R}^{I'}_{++} : \alpha \leq g_U(y),\ y \in \operatorname{dom} g_U\}, \tag{3.2}$$

where $\operatorname{dom} f = \{y : f(y) < +\infty\}$. Let $\operatorname{hyp} g_U = \{(\alpha, y) \in \mathbb{R} \times \mathbb{R}^I_{++} : \alpha \leq g_U(y), y \in \operatorname{dom} g_U\}$ be the hypograph of the function g_U. Then $U = (\operatorname{hyp} g_U) \cap \mathbb{R}^{I'}_{++}$, so we can consider U as the *positive part* of the hypograph $\operatorname{hyp} g_U$.

PROPOSITION 3.1. *For a closed normal subset U of $\mathbb{R}^{I'}_{++}$ the function g_U is decreasing (that is, $y_1 \geq y_2 \implies g_U(y_1) \leq g_U(y_2)$) and upper semicontinuous.*

Proof. Let $y_1 \geq y_2$ and $(\alpha, y_1) \in U$. Since U is normal and $(\alpha, y_1) \geq (\alpha, y_2)$ it follows that $(\alpha, y_2) \in U$. Therefore, $g_U(y_1) \leq g_U(y_2)$. Thus g_U is decreasing. Let $y_k \to y$. First assume that there exists a sequence k_s such that $g_U(y_{k_s}) = +\infty$. Thus $(\alpha, y_{k_s}) \in U$ for all $\alpha > 0$ and therefore $(\alpha, y) \in U$ for all $\alpha > 0$. So $g_U(y) = +\infty \geq \limsup_k g_U(y_k)$. Assume now that $g_U(y_k) < +\infty$ for all k. Then $(g_U(y_k), y_k) \in U$. If $\lambda := \limsup_k g_U(y_k) < +\infty$, then $(\lambda, y) \in U$ and therefore $\lambda \leq g_U(y)$. If $\lambda = +\infty$ then it easily follows that $g_U(y) = +\infty$. Thus $\limsup_k g_U(y_k) \leq g_U(y)$ in both cases. □

PROPOSITION 3.2. *Let $g \geq 0$ be a decreasing and upper semicontinuous function and $U = \{(\alpha, y) : y \gg 0, 0 < \alpha \leq g(y), y \in \text{dom } g\}$. Then U is a normal closed set and $g = g_U$.*

Proof. We first show that U is normal. Let $(\alpha_1, y_1) \in U$, $\alpha_2 > 0$, $y_2 \gg 0$, and $(\alpha_1, y_1) \geq (\alpha_2, y_2)$. Since g is decreasing we have $\alpha_2 \leq \alpha_1 \leq g(y_1) \leq g(y_2)$. Thus $(\alpha_2, y_2) \in U$. Since g is upper semicontinuous it follows that U is closed. We also have

$$g_U(y) = \sup\{\alpha : (\alpha, y) \in U\} = \sup\{\alpha : \alpha \leq g(y)\} = g(y) \quad (y \in \operatorname{dom} g). \qquad \square$$

Recall that $I' = \{0\} \cup I$. Consider an IPH function p defined on the cone $\mathbb{R}^{I'}_{++}$. Let $U = \operatorname{supp}(p)$ be the support set of the function p; then the set U generates the function g_U by (3.1).

DEFINITION 3.1. *Let $p \in IPH_{++}$ and $U = \operatorname{supp}(p)$. Then the function g_U defined by* (3.1) *is called the associated function to p.*

We shall denote the associated function to p by h_p.

EXAMPLE 3.1. Let

$$p(\delta, y) = \max\{\alpha\delta, a_1 y_1, \dots, a_m y_m\}$$

with $\alpha > 0$, $a = (a_1, \dots, a_m) \in \mathbb{R}^I_+$. Then (see Example 2.1) $U = \operatorname{supp}(p)$ coincides with the set $V_{(\alpha,a)}$ defined as follows:

$$V_{(\alpha,a)} = \left\{ \ell \in \mathbb{R}^{I'}_{++} : \min\left(\frac{\ell_0}{\alpha}, \min_{i \in I_a} \frac{\ell_i}{a_i} \right) \leq 1 \right\},$$

where $I_a = \{i : a_i > 0\}$. If $y \in \mathbb{R}^I_{++}$ is a vector such that $y_i \leq a_i$ for some i, then $(\delta, y) \in U$ for all $\delta > 0$ so $h_p(y) = g_U(y) = +\infty$. Assume now that $y \gg a$. Then $(\delta, y) \in V_{(\alpha,a)}$ if and only if $\delta \leq a_o$, so $h_p(y) = a_o$. Thus

$$h_p(y) = \begin{cases} \alpha & \text{if} \quad y \gg a, \\ +\infty & \text{otherwise.} \end{cases} \tag{3.3}$$

Assume now that $\alpha = 0$. Again applying Example 2.1, it is easy to check that h_p is defined by (3.3) with $\alpha = 0$. Thus this function coincides with the indicator function δ_Z of the set $Z = \{y : y \gg a\}$.

Example 3.2. Let $0 < k < +\infty$ and $p_k(\delta, y) = (\delta^k + \sum_{i\in I} y_i^k)^{\frac{1}{k}}$. Let $U = \operatorname{supp}(p)$. Furthermore, let

$$u = \left(1 - \sum_{i\in I} \frac{1}{y_i^k}\right)^{\frac{1}{k}}.$$

Then (see Example 2.2) we have

$$U = \left\{(\alpha, y) \in \mathbb{R}^{I'}_{++} : \frac{1}{\alpha^k} + \sum_{i\in I} \frac{1}{y_i^k} \geq 1\right\}.$$

So

$$h_p(y) = \sup\left\{\alpha : \frac{1}{\alpha^k} \geq 1 - \sum_{i\in I} \frac{1}{y_i^k}\right\}$$

$$= \begin{cases} \dfrac{1}{u} & \text{if } 1 > \displaystyle\sum_{i\in I} \frac{1}{y_i^k}, \\ +\infty & \text{otherwise.} \end{cases}$$

Let p be an IPH function defined on $\mathbb{R}^{I'}_{++}$. We now show that

$$\sup_{y\gg 0} p(1, y) = \sup_{y\gg 0} h_p(y).$$

We need the following simple assertion.

Lemma 3.1. *Let $\psi(\lambda)$ be a continuous decreasing function defined on the segment $(0, +\infty)$ and let $\lim_{\lambda\to+0} \psi(\lambda) = \bar{M} > 0$. Let $\chi_b(\lambda) = \min\{\psi(\lambda), b\lambda\}$ for $b > 0$. Then*

1. *for all $b > 0$ the function χ_b attains its maximum at a unique point $\lambda_b > 0$;*
2. *λ_b is a solution of the equation $\psi(\lambda) = b\lambda$;*
3. *$\lambda_b \to 0$ as $b \to +\infty$;*
4. *$\chi_b(\lambda_b) = \psi(\lambda_b) = b\lambda_b \to \bar{M}$ as $b \to +0$.*

Proof. The proof is straightforward. □

Proposition 3.3. *Let p be an IPH function defined on $\mathbb{R}^{I'}_{++}$. Then*

$$\sup_{y\gg 0} p(1, y) = \sup_{y\gg 0} h_p(y).$$

Proof. First we shall verify that

$$p(1, y) = \sup_{z\gg 0} \min\{h_p(z), \langle z, y\rangle\} \quad \text{for all} \quad y \in \mathbb{R}^I_{++}. \tag{3.4}$$

Indeed it follows, from the definition of the associated function h_p, that $\operatorname{supp}(p) = \{(\delta, z) : \ z \gg 0, \ 0 < \delta \leq h_p(z)\}$. So for $y \gg 0$ we have

$$p(1, y) = \sup\{\langle(\delta, z), (1, y)\rangle : \ (\delta, z) \in \operatorname{supp}(p)\} = \sup_{z\gg 0, \delta\leq h_p(z)} \min(\delta, \langle z, y\rangle).$$

Thus (3.4) holds. It follows that for an arbitrary $y \gg 0$ and $\varepsilon > 0$ there exists a vector $z \gg 0$ such that

$$p(1,y) - \varepsilon \le \min(h_p(z), \langle z, y\rangle) \le h_p(z) \le \sup_{u \gg 0} h_p(u).$$

Thus

$$\sup_{y \gg 0} p(1,y) \le \sup_{u \gg 0} h_p(u).$$

We now verify the reverse inequality.

Fix a vector $z \gg 0$ and consider the ray $\{\lambda z : \lambda > 0\}$. Let $\psi(\lambda) \equiv \psi_z(\lambda) = h_p(\lambda z)$. The function ψ is decreasing and

$$\lim_{\lambda \to +0} \psi(\lambda) = \lim_{y \to 0} h_p(y) = \sup_{y \gg 0} h_p(y). \tag{3.5}$$

For $y \gg 0$ consider the function $\chi_b(\lambda) = \min\{\psi(\lambda), b_y \lambda\}$, where $b_y = \langle z, y\rangle$. Let λ_y be a solution of the equation $\psi(\lambda) = b_y\lambda$. Lemma 3.1 shows that $\max_{\lambda > 0} \min\{\psi(\lambda), b_y\lambda\}$ is attained at the point λ_y and equals $\psi(\lambda_y)$. It follows from (3.4) that

$$p(1,y) \ge \max_{\lambda > 0} \min(\psi(\lambda), b_y\lambda) = \psi(\lambda_y).$$

Applying (3.5) and Lemma 3.1 we have

$$\sup_{y \gg 0} p(1,y) \ge \sup_{y \gg 0} \psi(\lambda_y) = \lim_{\lambda \to +0} \psi(\lambda) = \lim_{\lambda \to +0} h_p(\lambda z) = \sup_{u \gg 0} h_p(u). \qquad \square$$

We now show that the associated function h_p can be expressed in terms of the initial function p.

PROPOSITION 3.4. *Assume that p is an IPH function defined on $\mathbb{R}^{I'}_{++}$. Let $z \in \mathbb{R}^I_{++}$. Then the following hold:*

1. *If $\lim_{\tau \to +0} p(\tau, z^{-1}) \ge 1$, then $h_p(z) = +\infty$.*
2. *If $\lim_{\tau \to +\infty} p(\tau, z^{-1}) < 1$, then $h_p(z) = 0$.*
3. *If $\lim_{\tau \to +0} p(\tau, z^{-1}) < 1 \le \lim_{\tau \to +\infty} p(\tau, z^{-1})$, then*

$$h_p(z) = \frac{1}{b(\frac{1}{z})},$$

where $b(z)$ is the smallest solution of the equation $p(b, z) = 1$.

Proof. It follows from Theorem 2.3 that

$$\operatorname{supp}(p) = \{(\alpha, y) \in \mathbb{R}^{I'}_{++} : \ p(\alpha^{-1}, y^{-1}) \ge 1\}.$$

Let $z \in \mathbb{R}^I_{++}$ and $y = z^{-1}$. Then

$$\begin{aligned} h_p(z) = h_p(y^{-1}) &= \sup\{\alpha : (\alpha, y^{-1}) \in \operatorname{supp}(p)\} \\ &= \sup\{\alpha : \ p(\alpha^{-1}, y) \ge 1\} \\ &= \sup\{\tau^{-1} : \ p(\tau, y) \ge 1\} \\ &= \frac{1}{\inf\{\tau : \ p(\tau, y) \ge 1\}}. \end{aligned}$$

Let $\psi_y(\tau) = p(\tau, y)$. Thus

$$h_p(z) = \frac{1}{\inf\{\tau : \psi_y(\tau) \ge 1\}}. \tag{3.6}$$

It follows, from the properties of the function p, that ψ_y is an increasing continuous function on $\mathbb{R}_{++}$. Let

$$\gamma_- = \lim_{\tau \to +0} \psi_y(\tau) \quad \text{and} \quad \gamma_+ = \lim_{\tau \to +\infty} \psi(\tau).$$

If $\gamma_- \geq 1$, then $\inf\{\tau : \psi_y(\tau) \geq 1\} = 0$; if $\gamma_+ < 1$, then the set $\{\tau : \ \psi_y(\tau) \geq 1\}$ is empty and so the infimum of this set is defined to be $+\infty$. If $\gamma_- < 1 \leq \gamma_+$, then $\inf\{\tau : \ \psi_y(\tau) \geq 1\}$ is equal to the smallest root of the equation $\psi_y(\tau) = 1$. The desired result follows from (3.6). □

Proposition 3.4 allows us to describe some properties of IPH functions in terms of associated functions.

Let p be an IPH function defined on the cone $\mathbb{R}^{I'}_+$. The support set of the function p coincides (see Definition 2.2) with the support set of its restriction to $\mathbb{R}^{I'}_{++}$. We will denote this restriction by the same letter p. The following propositions will be useful in what follows. We give a sketch of their proofs.

PROPOSITION 3.5. *Let p be a continuous IPH function defined on the cone $\mathbb{R}^{I'}_+$. Then $\lim_{\min_i z_i \to +\infty} h_p(z) = 1$ if and only if $p(1, 0, \ldots, 0) = 1$.*

Proof. Let $\lim_{\min_i z_i \to +\infty} h_p(z) = 1$. Since $0 < h_p(z) < +\infty$, it follows from Proposition 3.4 that $h_p(z) = (b(z^{-1}))^{-1}$, where $p(b(z^{-1}), z^{-1}) = 1$. Since p is continuous we can conclude that

$$p(1, 0) = \lim_{\min_i z_i \to +\infty} p(b(z^{-1}), z^{-1}) = 1. \tag{3.7}$$

Now assume that $p(1, 0) = 1$. Let $t(z) = \{\alpha : p(\alpha, z) = 1\}$. Since p is positively homogeneous it follows from (3.7) that $t(0) = \{1\}$. By continuity of p we have

$$(\alpha \to \alpha', \quad z \to 0, \quad \alpha \in t(z)) \implies \alpha' = 1.$$

Since $b(z^{-1}) \in t(z^{-1})$ it follows that $h_p(z) = (b(z^{-1}))^{-1} \to 1$ as $\min_i z_i \to +\infty$. □

Let h be a function defined on $\mathbb{R}^I_{++}$ and $L = \lim_{\|z\| \to +\infty} h(1/z)$. We can present this limit in the following form:

$$L = \lim_{\max_i z_i \to +\infty} h(1/z) = \lim_{\min_i y_i \to +\infty} h(y). \tag{3.8}$$

PROPOSITION 3.6. *Let p be an IPH function defined on $\mathbb{R}^{I'}_{++}$ with $I' = \{0\} \cup I$ and $\lim_{\|u\| \to +\infty} p(1, u) = +\infty$. Then $\lim_{\min_i z_i \to 0} h_p(z) = +\infty$.*

Proof. First we show that

$$(p(\alpha, y) = 1, \|y\| \to +\infty) \implies \alpha \to 0. \tag{3.9}$$

Indeed, if $\alpha \geq 1$ then $p(\alpha, y) \geq p(1, y) \to +\infty$ (as $\|y\| \to +\infty$), which is a contradiction. Thus $\alpha < 1$. Since p is IPH it follows from $p(\alpha, y) = 1$ that $p(1, (y/\alpha)) = 1/\alpha$. Since p is an increasing function we can conclude that $p(\alpha, y) = 1$ implies

$$\lim_{\|y\| \to +\infty} p\left(1, \frac{y}{\alpha}\right) = \lim_{\|y\| \to +\infty} \frac{1}{\alpha} = \frac{1}{\alpha'}$$

with $\alpha' \leq 1$. If $\alpha' > 0$, then $\|y/\alpha\| \to +\infty$ and therefore $p(1, y/\alpha) = +\infty$, which is again a contradiction. Thus $\alpha' = 0$. Since

$$h_p\left(\frac{1}{y}\right) = (b(y))^{-1},$$

where $p(b(y), y) = 1$, it follows from (3.9) and (3.8) that

$$\lim_{\min_i z_i \to 0} h_p(z) = \lim_{\|y\| \to +\infty} h_p\left(\frac{1}{y}\right) = \lim_{\|y\| \to \infty} (b(y))^{-1} = +\infty. \qquad \square$$

4. Multiplicative inf-convolution of decreasing functions.

DEFINITION 4.1. *Let h and l be functions defined on $\mathbb{R}^I_{++}$ and mapping into $(0, +\infty]$. The function*

$$(4.1) \qquad (h \diamond l)(z) = \inf_{y \gg 0} h(y) l\left(\frac{z}{y}\right), \qquad z \in \mathbb{R}^I_{++},$$

is called the multiplicative inf-convolution of the functions h and l.

Since

$$\inf_{y \gg 0} h(y) l\left(\frac{z}{y}\right) = \inf_{u \gg 0} h\left(\frac{z}{u}\right) l(u),$$

it follows that multiplicative inf-convolution is a commutative operation: $h \diamond l = l \diamond h$.

If l is a decreasing function, then, applying (4.1), it is easy to check that the multiplicative inf-convolution, $h \diamond l$, of l and an arbitrary positive function h is also decreasing. Assume now that l is an upper semicontinuous function. Then for an arbitrary function h the function $z \to h(y)l(z/y)$ is upper semicontinuous for all $y \gg 0$ and therefore $h \diamond l$ is also upper semicontinuous. In particular, the following assertion holds.

PROPOSITION 4.1. *If l is a positive decreasing upper semicontinuous function, then $h \diamond l$ is decreasing and upper semicontinuous for any arbitrary positive function h.*

EXAMPLE 4.1. Let α be a nonnegative number and $a = (a_1, \dots, a_m)$ be a positive vector. Let $p(\delta, y) = \max\{\alpha\delta, a_1 y_1, \dots, a_m y_m\}$. Then (see Example 3.1) $h_p(y) = \alpha$ if $y \gg a$ and $h_p(y) = +\infty$ otherwise. Let l be a continuous decreasing function defined on $\mathbb{R}^I_{++}$. We have

$$(l \diamond h_p)(z) = \inf_{y \gg 0} l(y) h_p\left(\frac{z}{y}\right) = \inf_{u \gg 0} l\left(\frac{z}{u}\right) h_p(u) = \inf_{u \gg a} l\left(\frac{z}{u}\right) \alpha.$$

Since l is continuous and decreasing we conclude that $\inf_{u > a} l(\frac{z}{u}) = l(\frac{z}{a})$. Thus

$$(l \diamond h_p)(z) = \alpha l\left(\frac{z}{a}\right).$$

In particular, if $\alpha = 1$ and $a = (1, \dots, 1)$, then $l \diamond h_p = l$ for all continuous decreasing functions l.

We now describe the *positive part of the hypograph* of the multiplicative inf-convolution of decreasing functions. It is convenient to describe this in terms of the support set of the corresponding IPH function.

PROPOSITION 4.2. *Let l be a finite decreasing positive function defined on $\mathbb{R}^I_{++}$ and p be an IPH function defined on $\mathbb{R}^{I'}_{++}$ with $I' = \{0\} \cup I$. Let h_p be the associated function for p. Then*

$$(4.2) \qquad \bigcap_{y \gg 0} (l(y), y) \cdot \operatorname{supp}(p) = \{(\delta, z) : 0 < \delta \le (l \diamond h_p)(z),\ z \gg 0\},$$

where the product $a \cdot U$ is defined by (2.3).

Proof. Let us prove that for all $y \gg 0$:

$$(l(y), y) \cdot \operatorname{supp}(p) = \left\{ (\delta, z) : \ \delta \le l(y) h_p \left(\frac{z}{y} \right) \right\}. \tag{4.3}$$

Indeed, since $l(y) > 0$ it follows from the definition of the associated function that

$$\begin{aligned} (l(y), y) \cdot \operatorname{supp}(p) &= \{ (l(y) \cdot \gamma, y \cdot u) : (\gamma, u) \in \operatorname{supp}(p) \} \\ &= \{ (l(y) \cdot \gamma, y \cdot u) : \ \gamma \le h_p(u) \} \\ &= \left\{ (\delta, z) : \ \frac{\delta}{l(y)} \le h_p \left(\frac{z}{y} \right) \right\} \\ &= \left\{ (\delta, z) : \ \delta \le l(y) h_p \left(\frac{z}{y} \right) \right\}. \end{aligned}$$

Let V be the set on the left-hand side in (4.2). Then

$$\begin{aligned} (\delta, z) \in V &\iff (\delta, z) \in (l(y), y) \cdot \operatorname{supp}(p) \quad (\forall y \gg 0) \\ &\iff \delta \le l(y) h_p \left(\frac{z}{y} \right) \quad (\forall y \gg 0) \\ &\iff \delta \le \inf_{y \gg 0} l(y) h_p \left(\frac{z}{y} \right). \end{aligned}$$

So $V = \{ (\delta, z) : \delta \le (l \diamond h_p)(z), \ z \gg 0 \}$. □

Remark 4.1. Let U and V be closed normal subsets of $\mathbb{R}^{I'}_{++}$ and $g_U = l$. The set $\{ (l(y), y) : y \gg 0 \}$ represents the *upper boundary* of the normal set U. We can consider the set defined by (4.2) with $V = \operatorname{supp}(p)$ as a "product" of the sets U and V. Since the set V can be considered as the positive part of the hypograph $\operatorname{hyp} h_p$ and, similarly, U can be considered the positive part of $\operatorname{hyp} l$, it follows that the positive part of the hypograph of the multiplicative inf-convolution of h_p and l coincides with the "product" of the positive parts of the hypographs $\operatorname{hyp} h_p$ and $\operatorname{hyp} l$.

Let $h = l \diamond h_p$ be a multiplicative inf-convolution, where l and h_p are as in Proposition 4.2. It follows from Proposition 4.1 that h is a decreasing and upper semicontinuous function and therefore there exists an IPH function r such that $h = h_r$.

Proposition 4.3. *Let l be a decreasing positive function defined on $\mathbb{R}^I_{++}$ and let $p, \ r : \ \mathbb{R}^{I'}_{++} \to \mathbb{R}_{+\infty}$ be IPH functions. Then $h_r = l \diamond h_p$ if and only if*

$$\operatorname{supp}(r) = \bigcap_{y \gg 0} (l(y), y) \cdot \operatorname{supp}(p).$$

Proof. This follows directly from Proposition 4.2. □

Lemma 4.1. *Let h be a finite decreasing function defined on $\mathbb{R}^I_{++}$. Then the following limits exist:*

$$\lim_{z \to 0} h(z) = \sup_{z \gg 0} h(z), \qquad \lim_{\min_i z_i \to +\infty} h(z) = \inf_{z \gg 0} h(z).$$

Proof. The proof is straightforward. □

We now present the main result of this section.

THEOREM 4.1. *Let l and h be decreasing functions defined on $\mathbb{R}^I_{++}$ such that*

1. $0 < \gamma := \lim_{\min_i z_i \to +\infty} l(z), M := \lim_{y \to 0} l(y) < +\infty$*;*
2. $\mathrm{dom} h = \{y : \ h(y) < +\infty\} \neq \emptyset$ *and* $H := \lim_{\min_i z_i \to +\infty} h(z) > 0$*;*
3. $\liminf_{\min z_i \to 0} h(z) > \frac{M}{\gamma} H$*.*

Then

$$\lim_{z \to 0} (h \diamond l)(z) = \lim_{z \to 0} l(z) \times \lim_{\min_i z_i \to +\infty} h(z) = MH. \tag{4.4}$$

Proof. First we show that $(h \diamond l)(z) \le MH$ for all $z \gg 0$. Let $\mathbf{e} = (1, \dots, 1)$. For the functions $u(\lambda) = l(\lambda \mathbf{e})$ and $v_z(\lambda) = h(\frac{1}{\lambda} z)$ with $z \gg 0$, we have

$$u(\lambda) \le \sup_{\lambda' > 0} u(\lambda) = \lim_{\lambda' \to 0} l(\lambda' e) = \lim_{y \to 0} l(y) = M,$$

$$\inf_{\lambda > 0} v_z(\lambda) = \inf_{\mu > 0} h(\mu z) = \lim_{\mu \to +\infty} h(\mu z) = \lim_{\min_i y_i \to +\infty} h(y) = H.$$

So

$$(h \diamond l)(z) = \inf_{y \gg 0} l(y) h\left(\frac{z}{y}\right) \le \inf_{\lambda > 0} u(\lambda) v_z(\lambda) \le M \inf_{\lambda > 0} v_z(\lambda) = MH. \tag{4.5}$$

Thus

$$\lim_{z \to +0} (h \diamond l)(z) \le MH. \tag{4.6}$$

We now prove the reverse inequality.

It follows from condition 3 that there exists numbers $\mu > 0$ and $\varepsilon > 0$ such that $h(u) \ge (1 + \varepsilon)(1/\gamma) MH$ whenever $\min u_i \le \mu$. Thus, if $\min_i (z/y)_i \le \mu$, then

$$l(y) h\left(\frac{z}{y}\right) \ge \gamma h\left(\frac{z}{y}\right) \ge (1 + \varepsilon) MH.$$

Thus $\inf_{y \gg 0, \ \min (z/y)_i \le \mu} l(y) h(z/y) > MH$. Applying (4.5) we can conclude that

$$\begin{aligned} MH \ge (h \diamond l)(z) &= \inf_{y \gg 0} l(y) h\left(\frac{z}{y}\right) \\ &= \min\left(\inf_{y \gg 0, \ \min (z/y)_i \le \mu} l(y) h\left(\frac{z}{y}\right), \inf_{y \gg 0, (z/y) \gg \mu \mathbf{e}} l(y) h\left(\frac{z}{y}\right) \right) \\ &= \inf_{y \gg 0, (z/y) \gg \mu \mathbf{e}} l(y) h\left(\frac{z}{y}\right). \end{aligned}$$

Let $z \in \mathbb{R}^I_{++}$ and $z_\mu = (1/\mu) z$. We have

$$(h \diamond l)(z) = \inf_{y \gg 0, \ z/y \gg \mathbf{e}} l(y) h\left(\frac{z}{y}\right) = \inf_{0 \ll y \ll z_\mu} l(y) h\left(\frac{z}{y}\right). \tag{4.7}$$

Since l is decreasing we have $l(y) \ge l(z_\mu)$ for $0 \ll y \ll z_\mu$. So

$$(h \diamond l)(z) \ge \inf_{0 \ll y \ll z_\mu} l(z_\mu) h\left(\frac{z}{y}\right) = l(z_\mu) \inf_{u \gg \mu \mathbf{e}} h(u).$$

It follows from Lemma 4.1 that

$$\inf_{u \geq \mu \mathbf{e}} h(u) = \lim_{\min_i u_i \to +\infty} h(u) = H,$$

so $(h \diamond l)(z) \geq Hl(z_\mu)$. Thus

$$\lim_{z \to 0} (h \diamond l)(z) \geq H \lim_{z \to 0} l(z_\mu) = H \lim_{y \to 0} l(y) = HM. \tag{4.8}$$

It follows from (4.5) and (4.8) that (4.4) holds. □

Remark 4.2. Condition 3 of the theorem holds if

$$\frac{\liminf_{\min_i z_i \to 0} h(z)}{\lim_{\min_i z_i \to +\infty} h(z)} > \frac{\lim_{z \to 0} l(z)}{\lim_{\min_i z_i \to +\infty} l(z)}.$$

It is clear that this inequality holds if $\lim_{\min_i z_i \to 0} h(z) = +\infty$.

5. The modified penalty function. We now study the following constrained optimization problem:

$$(P): \quad f_o(x) \longrightarrow \inf \quad \text{subject to} \quad x \in X,\ f_1(x) \leq 0, \tag{5.1}$$

where $X \subset \mathbb{R}^n$ and $f_i : X \to \mathbb{R}$, $i = 0, 1$.

Remark 5.1. The more general problem

$$f_o(x) \longrightarrow \inf \quad \text{subject to} \quad x \in X,\ g_i(x) \leq 0,\ i \in I, \tag{5.2}$$

can be represented in the form of (5.1) with $f_1(x) = \sup_{i \in I} g_i(x)$.

We will require the following assumption.

Assumption 5.1. $\inf_{x \in X} f_o(x) := \gamma > 0$.

Let X_o be the set of all feasible solutions for (P):

$$X_o = \{x \in X : f_1(x) \leq 0\}. \tag{5.3}$$

The set X_o can also be represented in the following form:

$$X_o = \{x \in X : f_1^+(x) = 0\},$$

where $f_1^+(x) = \max\{f_1(x), 0\}$. Thus the problem (P) is equivalent to the following problem:

$$(P^+): \qquad f_o(x) \longrightarrow \inf \quad \text{subject to} \quad f_1^+(x) = 0. \tag{5.4}$$

It follows from Assumption 5.1 that $f_o^+(x) := \max\{f_o(x), 0\} = f_o(x)$ for all $x \in X$. Let

$$F^+(x, d_o, d) = (d_o f_o(x), d f_1^+(x)), \qquad x \in X,\ d_o,\ d > 0.$$

We now consider the function p_1 defined on $\mathbb{R}^2_+$ by $p_1(y_o, y_1) = y_o + y_1$. Clearly p_1 is an IPH function. The function p_1 generates the *classical* penalty function $\mathcal{L}^+$ for the problem (P):

$$\mathcal{L}^+(x, d_o, d) = d_o f_o(x) + d f_1^+(x) \equiv p_1(F^+(x, d_o, d)), \quad x \in X,\ d_o,\ d > 0.$$

Suppose now that we have an arbitrary continuous IPH function p defined on $\mathbb{R}^2_+$.

DEFINITION 5.1. *The function* $\mathcal{L}_p^+$

$$\mathcal{L}_p^+(x, d_o, d) = p(F^+(x, d_o, d)), \qquad x \in X,\ d_o > 0, d > 0,$$

is called the modified penalty function for the problem (P)*, corresponding to the function* p*. Let*

$$q_p(d_o, d) = \inf_{x \in X} p(d_o f_o(x), d f_1^+(x)) \equiv \inf_{x \in X} \mathcal{L}_p^+(x, d_o, d), \quad d_o > 0, d > 0. \tag{5.5}$$

DEFINITION 5.2. *The problem*

$$(D_p): \qquad q_p(1, d) \longrightarrow \sup \quad \textit{subject to} \quad d > 0,$$

where q_p *is defined by* (5.5)*, is called the dual problem to the problem* (P)*, corresponding to the function* p*.*

We will denote by M_P the value of the initial problem (P) and by M_{D_p} the value of the dual problem:

$$M_P = \inf\{f_o(x) : x \in X_o\}, \qquad M_{D_p} = \sup\{q_p(1, d) : d > 0\}.$$

Note that the function $d \mapsto q_p(1, d)$ is increasing and $M_{D_p} = \lim_{d \to +\infty} q_p(1, d)$.

We can express the function $q_p(d_o, d)$ defined by (5.5) in the following form:

$$q_p(d_o, d) = \inf_{x \in X} \mathcal{L}_p^+(x, d_o, d) = \min\left\{ \inf_{x \in X_o} \mathcal{L}_p^+(x, d_o, d), \inf_{x \notin X_o} \mathcal{L}_p^+(x, d_o, d) \right\}. \tag{5.6}$$

Let

$$X_1 = \{x \in X : f_1(x) > 0\} = \{x \in X : x \notin X_o\} \tag{5.7}$$

and

$$r_p(d_o, d) = \inf_{x \in X_1} p(d_o f_o(x), d f_1(x)). \tag{5.8}$$

PROPOSITION 5.1. *If Assumption* 5.1 *holds, then*

$$q_p(d_o, d) = \min\left\{ \inf_{x \in X_o} \mathcal{L}_p^+(x, d_o, d), r_p(d_o, d) \right\}.$$

Proof. This follows directly from (5.6) and (5.8). □

We now study the function r_p defined by (5.8) on $\mathbb{R}^2_{++}$. Since p is an IPH function it follows that r_p is also an IPH function. Recall (see Definition 2.2) that the support set of the function p coincides with the support set of its restriction to the cone $\mathbb{R}^2_{++}$. We will denote this restriction by the same letter p. We now describe the support set $\operatorname{supp}(r_p)$ of r_p in terms of the set $\operatorname{supp}(p)$.

PROPOSITION 5.2. *Let* p *be an IPH function and let* f_o *and* f_1 *be, respectively, the objective and constraint functions of the problem* (P)*. Furthermore, let the set* X_1 *and the function* r_p *be defined by* (5.7) *and* (5.8)*, respectively. Then the following holds:*

$$\operatorname{supp}(r_p) = \bigcap_{x \in X_1} (f_o(x_o), f_1(x)) \cdot \operatorname{supp}(p), \tag{5.9}$$

where the product $a \cdot U$ is defined by (2.3)*.*

Proof. We have for $d_o, d > 0$:

$$r_p(d_o, d) = \inf_{x \in X_1} p(d_o f_o(x), d f_1(x)) = \inf_{x \in X_1} q_p^x(d_o, d),$$

where

$$q_p^x(d_o, d) = p((f_o(x), f_1(x)) \cdot (d_o, d)) = p((f_o(x), f_1^+(x)) \cdot (d_o, d)).$$

It follows from Propositions 2.1 and 2.2 that

$$\operatorname{supp}(r_p) = \bigcap_{x \in X_1} \operatorname{supp}(q_p^x) = \bigcap_{x \in X_1} (f_o(x), f_1(x)) \cdot \operatorname{supp}(p). \qquad \square$$

6. Perturbation functions. We now study the perturbation function (see, for example, [4, 5, 6, 14, 13] and references therein) for the problem (P) defined by (5.1).

DEFINITION 6.1. *The function β defined on $\mathbb{R}_+ = \{y \in \mathbb{R} : y \geq 0\}$ by*

$$\beta(y) = \inf\{f_o(x) : x \in X,\ f_1(x) \leq y\} \tag{6.1}$$

is called the perturbation function of the problem (P)*.*

The value $\beta(0)$ of the perturbation function at the origin coincides with the value M_P of the problem (P). We also have

$$\inf_{y>0} \beta(y) = \inf_{y>0} \inf_{x \in X,\ f_1(x) \leq y} f_o(x) = \inf_{x \in X} f_o(x).$$

Since $\inf_{x \in X} f_o(x) = \gamma > 0$ (by Assumption 5.1) we have $\inf_{y>0} \beta(y) = \gamma > 0$. It follows directly from the definition that the perturbation function is decreasing: $y_1 \geq y_2 \implies \beta(y_1) \leq \beta(y_2)$.

Assumption 6.1. Let X_o and X_1 be the sets defined by (5.3) and (5.7), respectively. There exists a sequence $x_k \in X_1$ such that $f_1(x_k) \to 0$ and $f_o(x_k) \to M_P$, where $M_P = \inf_{x \in X_o} f_o(x)$ is the value of the problem (5.1).

If Assumption 6.1 holds, then for each $y > 0$ we have $\inf_{x \in X_1, f_1(x) \leq y} f_o(x) \leq M_P$, so

$$\beta(y) = \inf_{x \in X_1, f_1(x) \leq y} f_o(x), \qquad y > 0. \tag{6.2}$$

PROPOSITION 6.1. *Let p be an IPH function defined on $\mathbb{R}^2_{++}$ and let $U = \operatorname{supp}(p)$. Let Assumptions* 5.1 *and* 6.1 *hold. Then, for the function r_p defined by* (5.8)*, we have*

$$\operatorname{supp}(r_p) = \bigcap_{y>0} (\beta(y), y) \cdot U.$$

Proof. Let

$$A = \bigcap_{x \in X_1} (f_o(x), f_1(x)) \cdot U \quad \text{and} \quad B = \bigcap_{y>0} (\beta(y), y) \cdot U.$$

It follows from Proposition 5.2 that $\operatorname{supp}(r_p) = A$. We now check that $B \subset A$. Let $x \in X_1$ and $y = f_1(x)$. Then (see (6.2))

$$\beta(y) = \inf_{x' \in X_1, f_1(x') \leq y} f_o(x') \leq f_o(x)$$

and therefore $(f_o(x), f_1(x)) \geq (\beta(y), y)$. It follows from Lemma 2.1 that $(f_o(x), f_1(x)) \cdot U \supset (\beta(y), y) \cdot U$ so

$$A = \bigcap_{x \in X_1} (f_o(x), f_1(x)) \cdot U \supset \bigcap_{y = f_1(x),\ x \in X_1} (\beta(y), y) \cdot U \supset B.$$

We now prove that $A \subset B$. Let $y > 0$. It follows from (6.2) that for each sufficiently small $\varepsilon' > 0$ there exists a vector $x \in X_1$ such that $f_1(x) \leq y$ and $f_o(x) - \varepsilon' \leq \beta(y)$. It follows from Lemma 2.1 that $(\beta(y), y) \cdot U \supset (f_o(x) - \varepsilon', f_1(x)) \cdot U$. Therefore,

$$\begin{aligned} B &= \bigcap_{y>0} (\beta(y), y) \cdot U \\ &\supset \bigcap_{x \in X_1} (f_o(x) - \varepsilon', f_1(x)) \cdot U. \end{aligned}$$

Since $\varepsilon' > 0$ is an arbitrary number it follows that $B \supset \bigcap_{x \in X_1} (f_o(x), f_1(x)) \cdot U = A$. □

COROLLARY 6.1. $h_{r_p} = \beta \diamond h_p$.

Proof. This follows immediately from Proposition 4.3. □

REMARK 6.1. The perturbation function β does not depend on the IPH function p, and the associated function h_p does not depend on the problem (P) (that is, on the functions f_o and f_1).

Let $M = \lim_{y \to +0} \beta(y) = \sup_{y>0} \beta(y)$. Since $X_o \equiv \{x \in X : f_1(x) \leq 0\} \subset \{x \in X : f_1(x) \leq y\}$, it follows that $\beta(y) \leq \inf_{x \in X_o} f_o(x) = M_P$; therefore, $M \leq M_P = \beta(0)$. Hence, the equality

$$M_P = \lim_{y \to +0} \beta(y) \tag{6.3}$$

holds if and only if β is lower semicontinuous at the point zero.

Conditions ensuring lower semicontinuity of the perturbation function are well known (see, for example, [12]). A simple sufficient condition has the following form: if f_o is continuous and there exists $y > 0$ such that the set $X_y = \{x \in X : f_1(x) \leq y\}$ is compact, then β is lower semicontinuous and (6.3) holds.

We shall assume below that both Assumptions 5.1 and 6.1 hold.

THEOREM 6.1. *Let h_p be the associated function for p and assume*

$$\lim_{z \to +\infty} h_p(z) = 1 \quad \textit{and} \quad \lim_{z \to +0} h_p(z) = +\infty.$$

Then

$$\sup_{z>0} h_{r_p}(z) = \sup_{d>0} r_p(1, d) = \inf_{x \in X_o} f_o(x) \tag{6.4}$$

if and only if the perturbation function β is lower semicontinuous at the point zero.

Proof. This follows directly from Corollary 6.1, Theorem 4.1, Remark 4.2, and Proposition 3.3. □

The conditions in Theorem 6.1 are given in terms of the associated function h_p of an IPH function p. Applying Propositions 3.5 and 3.6 we can present conditions guaranteeing the validity of (6.4) in terms of the function p itself.

THEOREM 6.2. *Let p be a continuous IPH function defined on the cone $\mathbb{R}^2_+$ with $p(1, 0) = 1$ and $\lim_{u \to +\infty} p(1, u) = +\infty$. Then* (6.4) *holds if and only if the function β is lower semicontinuous at the point zero.*

Let us now consider the dual problem (see Definition 5.2) to the problem (P), where M_{D_p} is the value of the dual problem.

LEMMA 6.1. *Let p be an IPH function defined on $\mathbb{R}^2_+$ with $p(1,0) = 1$. Then $M_{D_p} \leq M_P$ and, for all $x \in X_o$ and $d > 0$,*

$$\mathcal{L}_p^+(x, 1, d) \equiv p(f_o(x), df_1^+(x)) = f_o(x).$$

Proof. Let $d > 0$ and $x \in X_o$. Since $f_1^+(x) = 0$ and p is positively homogeneous with $p(1,0) = 1$ we have

$$f_o(x) = p(f_o(x), 0) = p(f_o(x), df_1^+(x)) = \mathcal{L}_p^+(x, 1, d).$$

Also

$$\mathcal{L}_p^+(x, 1, d) \geq \inf_{x' \in X} p(f_o(x'), d_1 f_1^+(x')) = q_p(1, d).$$

Thus $M_P = \inf_{x \in X_o} f_o(x) \geq \sup_{d>0} q_p(1, d) = M_{D_p}$. □

LEMMA 6.2. *Let p be a continuous IPH function defined on the cone $\mathbb{R}^2_+$ with $p(1,0) = 1$ and $\lim_{u \to +\infty} p(1, u) = +\infty$. Then $M_{D_p} = M$, where $M = \lim_{y \to +0} \beta(y)$.*

Proof. By Proposition 5.1 and Lemma 6.1 we have

$$q_p(1, d) = \min\left\{\inf_{x \in X_o} \mathcal{L}_p^+(x, 1, d), r_p(1, d)\right\} = \min\{M_P, r_p(1, d)\}. \tag{6.5}$$

Propositions 3.5 and 3.6 show that

$$\lim_{z \to +\infty} h_{r_p}(z) = 1 \quad \text{and} \quad \lim_{z \to +0} h_{r_p}(z) = +\infty. \tag{6.6}$$

Applying Corollary 6.1 and Theorem 4.1 we can conclude that

$$\lim_{d \to +0} h_{r_p}(d) = \lim_{y \to +0} \beta(y) = M.$$

It follows from Proposition 3.3 that

$$\sup_{d>0} r_p(1, d) = \lim_{d \to +0} h_{r_p}(d) = M. \tag{6.7}$$

Since $M_P \geq M$ we have, by applying (6.5) and (6.7), that

$$M_{D_p} = \lim_{d \to +\infty} q_p(1, d) = \lim_{d \to +\infty} \min\{M_P, r_p(1, d)\} = \min\{M_P, M\} = M. \qquad \square$$

REMARK 6.2. Let $\lim_{y \to +0} \beta(y) = M_P$. Then $q_p(1, d) = r_p(1, d)$ for all $d > 0$. Indeed, since the function $d \mapsto r_p(1, d)$ is increasing it follows from (6.7) that $r_p(1, d) \leq M_P$ for all $d > 0$. Applying (6.5) we can conclude that $q_p(1, d) = r_p(1, d)$ for all $d > 0$.

THEOREM 6.3. *Let p be a continuous IPH function defined on the cone $\mathbb{R}^2_+$ with $p(1,0) = 1$ and $\lim_{u \to +\infty} p(1, u) = +\infty$. Then $M_{D_p} = M_P$ if and only if the perturbation function β is lower semicontinuous at the point zero.*

Proof. This follows directly from Lemma 6.2. □

7. Exact penalty functions. Consider the optimization problem (P) defined by (5.1). In this section we will discuss the existence of an (exact) penalty parameter, that is, a number $d > 0$ such that $M_{D_p} = q_p(1, d)$, for a given IPH function p.

If Assumptions 5.1 and 6.1 hold and the perturbation function β is lower semicontinuous at the origin, then (see Remark 6.2) $q_p(1, d) = r_p(1, d)$, where r_p is defined by (5.8). It has been shown (see Corollary 6.1) that the associated-to-r_p function h_{r_p} can be presented as the multiplicative inf-convolution of the perturbation function β and the function h_p: $h_{r_p} = \beta \diamond h_p$. We will use this formula in the study of the problem under consideration.

We need the following assumption.

Assumption 7.1. The perturbation function $\beta(y)$ is lower semicontinuous at the origin.

If Assumption 7.1 holds, then (see Theorem 6.3) $M_P = M_{D_p}$ for each continuous IPH function p defined on $\mathbb{R}^2_+$ with properties $p(1, 0) = 1$ and $\lim_{y\to+\infty} p(1, y) = +\infty$.

PROPOSITION 7.1. *Let Assumptions* 5.1, 6.1, *and* 7.1 *hold. Let* p *be a continuous IPH function defined on* $\mathbb{R}^2_+$ *with* $p(1, 0) = 1$ *and* $\lim_{y\to+\infty} p(1, y) = +\infty$. *Then* $r_p(1, \bar{d}) = M_P$ *if and only if the associated function* h_{r_p} *is constant on the segment* $[0, M_P/\bar{d}_1]$*:*

$$h_{r_p}(y) = M_P, \quad 0 \le y \le \frac{M_P}{\bar{d}_1}.$$

Proof. Assume that there exists $\bar{d}$ such that $r_p(1, \bar{d}) = M_P$. Since the function $r_p(1, d)$ is increasing and $\sup_{d>0} r_p(1, d) = M_P$, it follows that $r_p(1, d) = M_P$ for all $d \ge \bar{d}$. Thus we have for $d/d_0 \ge \bar{d}$,

$$r_p(d_0, d) = d_0 r_p(1, d/d_0) = d_0 M_P.$$

For the support set s_{r_p} of the function r_p the following is valid (see Theorem 2.3):

$$s_{r_p} = \left\{ \ell = (l_0, l_1) : r_p\left(\frac{1}{l_0}, \frac{1}{l_1}\right) \ge 1 \right\}.$$

Consider the point $(l_0, l_1) = (M_P, M_P/\bar{d})$. We have $(\frac{1}{l_1})(\frac{1}{l_0})^{-1} = \bar{d}$. Thus

$$r_p\left(\frac{1}{l_0}, \frac{1}{l_1}\right) = \frac{1}{l_0} M_P = 1.$$

Hence $(M_P, M_P/\bar{d}) \in s_{r_p}$. The set s_{r_p} is normal so $\{(l_0, l_1) : l_0 \le M_P, l_1 \le M_P/\bar{d}\} \subset s_{r_p}$. By the definition of the associated function, we have

$$h_{r_p}(y) = \sup\{\alpha : (\alpha, y) \in s_{r_p}\}.$$

So if $0 \le y \le M_P/\bar{d}$, then $h_{r_p}(y) \ge M_P$. On the other hand, $h_{r_p}(y)$ is a decreasing function with $h_{r_p}(0) = M_P$. Thus $h_{r_p}(y) = M_P$ if $y \le M_P/\bar{d}$.

Assume now that $h_{r_p}(y) = M_P$ for $0 \le y \le \bar{y} = M_P/\bar{d}$. Since

$$h_{r_p}(\bar{y}) = \sup\{\alpha : (\alpha, \bar{y}) \in s_{r_p}\},$$

s_{r_p} is closed, and $h_{r_p}(\bar{y}) < +\infty$, we can deduce that $(M_P, \bar{y}) \in s_{r_p}$. Thus

$$r_p\left(\frac{1}{M_P}, \frac{\bar{d}}{M_P}\right) = \frac{1}{M_P} r_p(1, \bar{d}) \ge 1;$$

that is, $r_p(1,\bar{d}) \geq M_P$. On the other hand, $r_p(1,d) \leq M_P$ for all d. Thus $r_p(1,\bar{d}) = M_P$. $\square$

EXAMPLE 7.1. Let $p(\alpha, y) = \max\{\alpha, y\}$ and β be the perturbation function of the problem (P). Assume that β is continuous. Then (see Example 4.1) $h_{r_p}(z) = (\beta \diamond h_p)(z) = \beta(z)$. We have, by applying Theorem 2.1 and the definition of the associated function,

$$r_p(1,d) = \max\{\langle l, y\rangle : l \in \operatorname{supp}(r_p)\} = \max_{z>0} \min(h_{r_p}(z), zd) = \max_{z>0} \min(\beta(z), zd).$$

Assume that the perturbation function β is strictly decreasing for sufficiently small y, that is, that $\inf\{f_o(x) : \ f_1(x) \leq y_1\} > \inf\{f_o(x) : \ f_1(x) \leq y_2\}$ whenever $y_1 < y_2$. Then $M_P = \lim_{y\to 0} \beta(y) > \beta(z)$ for all $z > 0$. Since $\max_{z>0} \min(\beta(z), zd) = \beta(z_d)$ where z_d is a solution of the equation $\beta(z) = zd$, we have

$$q_p(1,d) = r_p(1,d) = \max_{z>0} \min(\beta(z), zd) = \beta(z_d) < M_P.$$

Thus there is no $d > 0$ such that $r_p(1,d) = M_P$.

We now consider the convolution function p_k $(k > 0)$ defined by

$$p_k(\delta, y) = \left(\delta^k + y^k\right)^{\frac{1}{k}} \qquad (\delta > 0,\ y > 0). \tag{7.1}$$

For the sake of simplicity we shall denote the function r_{p_k} by $r_{[k]}$ and its associated function $h_{r_{[k]}}$ by $h_{[k]}$. The following assertion will be useful in the study of both conditions for the exact penalization and estimations of penalty parameters.

LEMMA 7.1. *Let Assumptions* 5.1, 6.1, *and* 7.1 *hold. Let $k > 0$ and $p_k = p$ be the function defined by* (7.1). *Let $h_{[k]}$ be the associated function to r_{p_k}. Then $M_P = h_{[k]}(z)$ if and only if*

$$M_P \leq \beta(y) \frac{z}{(z^k - y^k)^{\frac{1}{k}}} \quad \textit{for} \quad 0 < y < z. \tag{7.2}$$

Proof. It easily follows from Example 3.2 that

$$h_{[p]}(y) = \begin{cases} \frac{y}{(y^k-1)^{\frac{1}{k}}}, & y > 1, \\ +\infty, & y \leq 1. \end{cases}$$

Since $\lim_{y\to+\infty} h_p(y) = 1$ and $\lim_{y\to+0} h_p(y) = +\infty$, we can apply Theorem 6.1, which shows that

$$M_P = M_{D_p} = \sup_{z>0} h_{[k]}(z) = \lim_{z\to 0} h_{[k]}(z). \tag{7.3}$$

By Corollary 6.1 we have

$$h_{[k]}(z) = \inf_{y>0} \beta(y) h_p\left(\frac{z}{y}\right) = \inf_{0<y<z} \beta(y) h_p\left(\frac{z}{y}\right) = \inf_{0<y<z} \beta(y) \frac{z}{(z^k - y^k)^{\frac{1}{k}}}. \tag{7.4}$$

It follows from (7.3) that $M_P \geq h_{[k]}(z')$ for all z', so $M_P = h_{[k]}(z)$ if and only if $M_P \leq h_{[k]}(z)$; that is,

$$M_P \leq \inf_{0<y<z} \beta(y) \frac{z}{(z^k - y^k)^{\frac{1}{k}}}. \tag{7.5}$$

Clearly, (7.5) is equivalent to (7.2). □

THEOREM 7.1. *Let Assumptions* 5.1, 6.1, *and* 7.1 *hold. Let* $k > 0$ *and the function* p_k *be defined by* (7.1). *Then there exists a number* $\tilde{d} > 0$ *such that* $M_P = q_p(1, \tilde{d})$ *if and only if*

$$\liminf_{y \to +0} \frac{\beta(y) - \beta(0)}{y^k} > -\infty. \tag{7.6}$$

Proof. It follows from Proposition 7.1 that $M_P = r_{[k]}(1, \tilde{d})$ for some $\tilde{d} > 0$ if and only if there exists $z > 0$ such that $M_P = h_{[k]}(z)$. According to Lemma 7.1, this equality holds if and only if

$$\frac{\beta(y)}{M_P} \geq \left(1 - \left(\frac{y}{z}\right)^k\right)^{\frac{1}{k}} \quad \text{for} \quad 0 < y < z. \tag{7.7}$$

Since $M_P = \beta(0)$ we can represent (7.7) in the following form:

$$\frac{\beta(y) - \beta(0)}{y^k} \geq \frac{\beta(0)}{z^k} \frac{(1 - u^k)^{\frac{1}{k}} - 1}{u^k} \quad \text{for} \quad 0 < y < z, \tag{7.8}$$

where $u = y/z$. Clearly (7.6) holds if and only if there exists z such that (7.8) is valid. □

REMARK 7.1. The notion of calmness (see, for example, [4, 6] and references therein) has been used to study exact penalization with classical penalty functions generated by the IPH function p_1 (where $p_1(y_1, y_2) = y_1 + y_2$). The family (P_y) of perturbed problems

$$f_o(x) \longrightarrow \min \quad \text{subject to} \quad x \in X, \quad f_1^+(x) := \max(f_1(x), 0) \leq y \qquad (y > 0),$$

is said to be calm at the point zero if (7.6) with $k = 1$ holds.

Using Theorem 7.1 it is possible to derive the well-known result (see, for example, [4, 6]) that there exists a number $\tilde{d} > 0$ such that $M_P = q_{p_1}(1, \tilde{d})$ if and only if the family (P_y) is calm.

For many problems the exact penalization with respect to the classical penalty function p_1 can be accomplished only with very large penalty parameters, which leads to ill-conditioned unconstrained optimization problems. Applying penalization with respect to other convolution functions p we can sometimes decrease penalty parameters providing exact penalization. We shall next study this question for the convolution function $p_{\frac{1}{2}}$. First we consider the arbitrary $k > 0$ and describe the least penalty parameter, that is, the least number, for which the equality $M_P = r_{[k]}(1, d)$ holds.

Assumption 7.2. Assume that the perturbation function $\beta(y)$ is continuous on $[0, +\infty)$ and $\beta(y) < M_P$ for $y > 0$.

Let Assumption 7.2 hold, $k > 0$, and

$$v_k(y) = \frac{y}{(1 - (\beta(y) M_P^{-1})^k)^{\frac{1}{k}}} \qquad (y > 0). \tag{7.9}$$

Let

$$\varphi_k(z) = \inf_{0 < y < z} v_k(y). \tag{7.10}$$

Since $v_k(y)$ is continuous for $y > 0$, it follows that $\varphi_k(z) = \inf_{0<y\le z} v_k(y)$ and $\varphi_k(z)$ is continuous for $z > 0$. Clearly φ_k is a decreasing function.

Lemma 7.2. *Let $k > 0$ and* (7.6) *be valid. If Assumptions* 5.1, 6.1, *and* 7.2 *hold, then the least exact penalty parameter $\bar{d}_k$ of the problem (P) with respect to the function p_k is equal to $1/\bar{z}_k$, where $\bar{z}_k$ is the solution of the equation $z = \varphi_k(z)$.*

Proof. It follows from Proposition 7.1 that $r_{[k]}(1,d) = M_P$ if and only if $h_{[k]}(z) = M_P$ with $z = 1/d$. Lemma 7.1 demonstrates that the equality $M_P = h_{[k]}(z)$ is valid if and only if (7.2) holds; that is,

$$\frac{M_P}{\beta(y)} \le \frac{z}{(z^k - y^k)^{\frac{1}{k}}} \quad \text{for} \quad 0 < y < z.$$

It easy to check that this inequality is equivalent to the following:

$$z \le v_k(y) \quad \text{for} \quad 0 < y < z, \tag{7.11}$$

where v_k is defined by (7.9).

Thus $M_P = h_{[k]}(z)$ if and only if $z \le \varphi_k(z)$. Since (7.6) holds it follows from Theorem 7.1 that there exists $z > 0$ such that $M_P = h_{[k]}(z)$; that is, $z \le \varphi_k(z)$. Thus $r_{[k]}(1,d) = M_P$ if and only if $z \le \varphi_k(z)$, so the least exact penalty parameter $\bar{d}_k$ is equal to the inverse to the greatest element $\bar{z}_k$ of the set $\{z : z \le \varphi_k(z)\}$. Since φ_k is a continuous decreasing function we can deduce that the equation $z = \varphi_k(z)$ has the unique solution. It is clear that this solution coincides with $\bar{z}_k$. □

Let $\beta(y) < M_P$ for $y > 0$. Set

$$u(y) = \frac{1 + \sqrt{\beta(y) M_P^{-1}}}{1 - \sqrt{\beta(y) M_P^{-1}}} \qquad (y > 0). \tag{7.12}$$

Since β is a decreasing function it follows that u is a decreasing function as well.

Lemma 7.3. *Let $\beta(y) < M_P$ for $y > 0$, and let v_k, φ_k, and u be functions defined by* (7.9), (7.10), *and* (7.12), *respectively. Then*

$$\varphi_{\frac{1}{2}}(z) \ge \varphi_1(z) u(z) \qquad (z > 0).$$

Proof. We have

$$v_1(y) = \frac{y}{1 - \beta(y) M_P^{-1}}, \qquad v_{\frac{1}{2}}(y) = \frac{y}{\left(1 - \sqrt{\beta(y) M_P^{-1}}\right)^2}.$$

Hence

$$v_{\frac{1}{2}}(y) = u(y) v_1(y) \qquad (y > 0).$$

Let $z > 0$ and $0 < y < z$. Since $\varphi_1(z) \le v_1(y)$ and u is a decreasing function it follows that

$$\varphi_{\frac{1}{2}}(z) = \inf_{0<y<z} v_{\frac{1}{2}}(y) \ge \varphi_1(z) \inf_{0<y<z} u(y) = \varphi_1(z) u(z). \qquad \square$$

Lemma 7.4. *Suppose Assumptions* 5.1, 6.1, *and* 7.2 *hold and*

$$\liminf_{y \to +0} \frac{\beta(y) - \beta(0)}{y} > -\infty.$$

Let $\bar{d}_k$ be the least exact penalty parameter with respect to the function p_k and $\bar{z}_k = 1/\bar{d}_k$ $(k = 1, \frac{1}{2})$. Let $\mu \in (0,1)$ be a number such that

$$\beta(z) \geq \mu M_P \quad \textit{for} \quad z \in (0, \lambda\bar{z}_1), \tag{7.13}$$

where

$$\lambda = \frac{1+\sqrt{\mu}}{1-\sqrt{\mu}}. \tag{7.14}$$

Then

$$\bar{d}_{\frac{1}{2}} \leq \frac{1-\sqrt{\mu}}{1+\sqrt{\mu}}\bar{d}_1.$$

Proof. Let λ be the number defined by (7.14). It easily follows from (7.12), (7.13), and (7.14) that $u(z) \geq \lambda$ for $0 < z \leq \lambda\bar{z}_1$. Hence (see Lemma 7.3)

$$\varphi_{\frac{1}{2}}(z) \geq \lambda\varphi_1(z) \quad \text{for} \quad 0 < z \leq \ \lambda\bar{z}_1. \tag{7.15}$$

Since $\varphi_1(\bar{z}_1) = \bar{z}_1$ it follows that $\lambda\varphi_1(\bar{z}_1) = \lambda\bar{z}_1$. Since $\varphi_{\frac{1}{2}}$ and φ_1 are decreasing functions and (7.15) holds, we can deduce that $\bar{z}_{\frac{1}{2}} \geq \lambda\bar{z}_1$. Thus $\bar{d}_{\frac{1}{2}} \leq \frac{1}{\lambda}\bar{d}_1$. □

THEOREM 7.2. *Assume all conditions in Lemma 7.4 hold. Then*

$$\bar{d}_{\frac{1}{2}} \leq \frac{1-\sqrt{\gamma M_P^{-1}}}{1+\sqrt{\gamma M_P^{-1}}}\bar{d}_1, \tag{7.16}$$

where $\gamma = \inf_{x\in X} f_0(x) = \inf_{z>0}\beta(z)$.

Proof. The proof follows from Lemma 7.4 since $\beta(z) \geq \mu M_P$ for all z, where $\mu = \gamma M_P^{-1}$. □

Theorem 7.2 allow us to draw the following conclusion:

The exact penalty parameter $\bar{d}_{\frac{1}{2}}$ is always less than $\bar{d}_1$. If the perturbation function $\beta(y)$ changes fairly slowly (that is, the constrained minimum M_P is not very far from the unconstrained minimum γ), then the penalization by the convolution function $p_{\frac{1}{2}}$ can be accomplished with a substantially smaller exact penalty parameter than that of the classical penalty function p_1.

The following simple example confirms this conclusion.

EXAMPLE 7.2. Let $0 < b < c < a$ be real numbers and $X = [0, c]$. Consider the problem

$$(a-x)^2 \longrightarrow \min \quad \text{subject to} \quad x - b \leq 0, \ x \in X. \tag{7.17}$$

Let $k = 1/2$ and $r_{[\frac{1}{2}]} = r_{p_k}$. We have $r_{[\frac{1}{2}]}(1,d) = \min_{b\leq x\leq c} h^2(x)$, where $h(x) = (a-x) + d^{\frac{1}{2}}(x-b)^{\frac{1}{2}}$. Since h is a concave positive function, it easily follows that

$$r_{[\frac{1}{2}]}(1,d) = \begin{cases} [(a-c) + d^{\frac{1}{2}}(c-b)^{\frac{1}{2}}]^2 & \text{if} \quad 0 < d \leq c-b, \\ (a-b)^2 & \text{if} \quad d > c-b, \end{cases}$$

and that $\sup_{d>0} r_{[\frac{1}{2}]}(1,d) = (a-b)^2$ is attained at the point $\bar{d}_{\frac{1}{2}} = c - b$. Note that $\bar{d}_{[\frac{1}{2}]}$ does not depend on a.

Consider now the classical penalty function with $k = 1$. It is easy to check that $\sup_{d>0} r_{[1]}(1, d) = (a - b)^2$ is attained at the point $\bar{d}_1 = 2(a - b)$. Thus $\bar{d}_1 \to +\infty$ as $a \to +\infty$.

Proposition 7.2. *The estimation* (7.16) *is asymptotically sharp in the following sense: for each $\varepsilon > 0$ there exists a problem (P) such that the difference between expressions in the right-hand side and the left-hand side of* (7.16) *is less than ε.*

Proof. Consider the problem (7.17) from Example 7.2. We have

$$M_P = (a - b)^2, \quad \gamma = (c - a)^2, \quad \bar{d}_1 = 2(a - b), \quad \bar{d}_{\frac{1}{2}} = c - b.$$

Hence the inequality (7.16) can be presented in the following form:

$$c - b \le \frac{1 - \frac{a-c}{a-b}}{1 + \frac{a-c}{a-b}} 2(a - b). \tag{7.18}$$

Note that the difference between expressions in the right-hand side and left-hand side in (7.18) is equal to $(c - b)^2(2a - c - b)^{-1}$, so this difference tends to zero as $c - b \to 0$. □

Remark 7.2. Consider the following problem (P_c):

$$f_o(x) + c \longrightarrow \min \quad \text{subject to} \quad f_1(x) \le 0,$$

which is equivalent to problem (P). Clearly, both problems (P) and (P_c) have the same exact penalty parameter $\bar{d}_1$ with respect to the classical penalty function p_1. At the same time, they have different exact penalty parameters $\bar{d}_{\frac{1}{2}}$ with respect to the convolution function $p_{\frac{1}{2}}$. Let $\bar{d}_{\frac{1}{2}}(c)$ be the least penalty parameter with respect to $p_{\frac{1}{2}}$ for the problem (P_c). It follows from (7.16) that $\bar{d}_{\frac{1}{2}}(c)$ tends to zero as $c \to +\infty$. Note that the corresponding unconstrained optimization problem can become ill-conditioned for very large c.

Remark 7.3. Consider the classical convolute function p_1 and the coresponding penalty parameter $\bar{d}_1$. It is well known (see [3]) that $\bar{d}_1$ can be estimated from below by the optimal Lagrange multiplier λ of the problem (P). Clearly, the optimal Lagrange multiplier of the problem (P_c) (see Remark 7.2) coincides with λ. It follows from Remark 7.2 that the estimation of the exact penalty parameter $\bar{d}_{\frac{1}{2}}$ by the Lagrange multiplier λ is impossible.

We conclude the paper by showing that it is possible to find an IPH function p and a number d such that for the modified penalty function $\mathcal{L}_p^+$ generated by p we have $M_P = \inf_{x \in X} \mathcal{L}_p^+(x, 1, d) \equiv q_p(1, d)$.

Theorem 7.3. *Let Assumptions* 5.1, 6.1, *and* 7.1 *hold. Then for the problem (P) defined by* (5.1) *there exists an IPH function p and a number $d > 0$ such that*

$$M_P = q_p(1, d),$$

where q_p is defined by (5.5).

Proof. Let β be the perturbation function of the problem (P). Consider the lower semicontinuous hull $\bar{\beta}$ of the function β:

$$\bar{\beta}(y) = \max\left(\beta(y), \liminf_{y' \to y, y' \neq y} \beta(y')\right).$$

Clearly $\bar{\beta}$ is decreasing and lower semicontinuous. Since β is lower semicontinuous at the origin we can conclude that $\bar{\beta}(0) = \beta(0) = M_P$. Let

$$g(y) = \begin{cases} +\infty & \text{if} \quad 0 < y \le 1, \\ M_P/\bar{\beta}(y^{-1}) & \text{if} \quad y > 1. \end{cases}$$

Clearly g is a decreasing function. Since $\bar{\beta}$ is lower semicontinuous it follows that g is upper semicontinuous. We have also

$$\lim_{y \to +\infty} g(y) = \lim_{u \to +0} \frac{M_P}{\bar{\beta}(u)} = 1.$$

Since g is upper semicontinuous we can conclude (see Proposition 3.2) that there exists a normal closed set $U \subset \mathbb{R}^2_{++}$ such that $g = g_U$. Consider an IPH function $\bar{p}$ defined on $\mathbb{R}^2_{++}$ by

$$\bar{p}(y) = \sup\{\min(l_1 y_1, l_2 y_2) : l \in U\}.$$

Since the function $y \to \bar{p}(1, y)$ is increasing it follows that $a = \lim_{y \to +0} p(1, y) < +\infty$. Let the function p be defined on $\mathbb{R}^2_+$ by

$$p(y_1, y_2) = \begin{cases} \bar{p}(y_1, y_2) & \text{if} \quad y_1, y_2 > 0, \\ 0 & \text{if} \quad y_1 = 0, \\ a y_1 & \text{if} \quad y_2 = 0. \end{cases}$$

It is easy to check that p is a continuous IPH function and $h_p(y) = g(y)$ for $y > 0$. Since $\lim_{y \to +\infty} h_p(y) = \lim_{y \to +\infty} g(y) = 1$ it follows from Proposition 3.5 that $a = p(0, 1) = 1$. The equality $h_p(y) = +\infty$ for $y \le 1$ shows that $\lim_{u \to +\infty} p(1, u) = +\infty$. It follows from Theorem 6.2 that $M_P = \sup\{h_{r_p}(z) : z > 0\}$ so $h_{r_p}(z) \le M_P$ for all $z > 0$. On the other hand, we have

$$h_{r_p}(z) = \inf_{y > 0} \beta(y) h_p\left(\frac{z}{y}\right) = \inf_{y > 0} \beta(y) g\left(\frac{z}{y}\right) = \inf_{y < z} \beta(y) \frac{M_P}{\bar{\beta}(\frac{y}{z})}.$$

Let $z = 1$. If the function β is continuous at a point y, then $\beta(y) = \bar{\beta}(y)$; otherwise $\beta(y) \ge \bar{\beta}(y)$. So

$$h_{r_p}(1) = \inf_{y < 1} \frac{\beta(y)}{\bar{\beta}(y)} M_P = M_P.$$

Thus $h_{r_p}(z) = M_P$ for $z \le 1$. It follows from Proposition 7.1 that there exists a number $\bar{d} > 0$ such that $r_p(1, \bar{d}) = M_P$. □

Acknowledgments. The authors are very grateful to two anonymous referees for constructive comments on an earlier version of this paper. These comments have enabled the authors to significantly improve the last section of the paper.

REFERENCES

[1] M. Yu. Andramonov, *An Approach to Constructing Generalized Penalty Functions*, Research Report 21/97, SITMS, University of Ballarat, Ballarat, VIC, Australia, 1997.

[2] A. Auslender, R. Cominetti, and M. Haddou, *Asymptotic analysis for penalty and barrier methods in convex and linear programming*, Math. Oper. Res., 22 (1997), pp. 43–62.

[3] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[4] J. V. Burke, *Calmness and exact penalization*, SIAM J. Control Optim., 29 (1991), pp. 493–497.

[5] J. V. Burke, *An exact penalization viewpoint of constrained optimization*, SIAM J. Control Optim., 29 (1991), pp. 968–998.

[6] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley-Interscience, New York, 1983; reprinted as Classics Appl. Math. 5, SIAM, Philadelphia, PA, 1990.

[7] C. Chen and O. L. Mangasarian, *Smoothing methods for convex inequalities and linear complementarity problems*, Math. Programming, 71 (1995), pp. 51–69.

[8] C. J. Goh and X. Q. Yang, *A sufficient and necessary condition for nonconvex constrained optimization*, Appl. Math. Lett., 10 (1997), pp. 9–12.

[9] S. S. Kutateladze and A. M. Rubinov, *Minkowski Duality and Its Applications*, Nauka, Novosibirsk, Russia, 1976 (in Russian).

[10] L. S. Lasdon, *Optimization Theory for Large Systems*, Macmillan, London, 1970.

[11] M. Minoux, *Programmation mathematique, theorie et algorithmes Dunod*, Bordas et G.N.E.T.-E.N.S.T., Paris, 1989.

[12] D. Pallaschke and S. Rolewicz, *Foundations of Mathematical Optimization*, Kluwer Academic, Norwell, MA, 1997.

[13] R. T. Rockafellar, *Conjugate Duality and Optimization*, CBMS-NSF Regional Conf. Ser. in Appl. Math. 16, SIAM, Philadelphia, 1974.

[14] R. T. Rockafellar, *Lagrange multipliers and optimality*, SIAM Rev., 35 (1993), pp. 183–238.

[15] A. M. Rubinov, *Some properties of increasing convex-along-rays functions*, Proc. Centre Math. Appl. Austral. Nat. Univ., 36 (1999), pp. 153–167.

[16] A. M. Rubinov and B. M. Glover, *Duality for increasing positively homogeneous functions and normal sets*, Rech. Opér., 32 (1998), pp. 105–123.

[17] A. M. Rubinov, B. M. Glover, and X. Q. Yang, *Extended Lagrange and penalty functions in continuous optimization*, Optimization, to appear.

[18] I. Singer, *Abstract Convex Analysis*, Wiley-Interscience, New York, 1997.

[19] Yu. Yevtushenko and V. Zhadan, *Exact auxiliary functions in optimization problems*, U.S.S.R. Comput. Math. Math. Phys., 30 (1990), pp. 31–42.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 315–330

A REGULARIZED SMOOTHING NEWTON METHOD FOR BOX CONSTRAINED VARIATIONAL INEQUALITY PROBLEMS WITH P_0-FUNCTIONS*

HOU-DUO QI†

Abstract. Based on Qi, Sun, and Zhou's smoothing Newton method, we propose a regularized smoothing Newton method for the box constrained variational inequality problem with P_0-function (P_0 BVI). The proposed algorithm generates an infinite sequence such that the value of the merit function converges to zero. If P_0 BVI has a nonempty bounded solution set, the iteration sequence must be bounded. This result implies that there exists at least one accumulation point. Under CD-regularity, we prove that the proposed algorithm has a superlinear (quadratic) convergence rate without requiring strict complementarity conditions. The main feature of our global convergence results is that we do not assume a priori the existence of an accumulation point. This assumption is used widely in the literature due to the possible unboundedness of level sets of various adopted merit functions. Preliminary numerical results are also reported.

Key words. smoothing Newton's method, semismoothness, global convergence, superlinear convergence

AMS subject classifications. 90C33, 65K10, 65H10

PII. S1052623497324047

1. Introduction.

Consider the box constrained variational inequality problem with P_0-function (P_0 BVI for abbreviation), which is described as follows. Find $x^* \in X$ such that

$$F(x^*)^T(x - x^*) \geq 0 \quad \text{for all} \quad x \in X, \tag{1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable P_0-function and

$$X := \{x \in \mathbb{R}^n |\ a \leq x \leq b\},$$

and $a \in \{\mathbb{R} \cup \{-\infty\}\}^n$, $b \in \{\mathbb{R} \cup \{\infty\}\}^n$, and $a < b$. If $a = 0$, $b = \infty$, then BVI becomes the well-known nonlinear complementarity problem (NCP). BVI, which is also called the mixed complementarity problem, covers a large class of problems, e.g., convex programming problems and monotone variational inequality problems (MVIP). We refer the interested reader to the survey papers by Harker and Pang [13] and Ferris and Pang [10] for various applications. Recently, P_0 BVI/NCP has raised much interest among researchers [1, 2, 4, 5, 9, 22, 23, 26].

Let $\Pi_X(x)$ denote the Euclidean projection of x on X. It is well known that solving BVI is equivalent to solving the Robinson's normal equation

$$E(x) := F(\Pi_X(x)) + x - \Pi_X(x) = 0 \tag{2}$$

in the sense that if $x^* \in \mathbb{R}^n$ is a solution of (2), then $y^* := \Pi_X(x^*)$ is a solution of (1) and, conversely, if y^* is a solution of (1), then $x^* := y^* - F(y^*)$ is a solution of (2)

*Received by the editors July 8, 1997; accepted for publication (in revised form) December 9, 1998; published electronically December 22, 1999.

http://www.siam.org/journals/siopt/10-2/32404.html

†Institute of Computational Mathematics and Scientific/Engineering Computing, Chinese Academy of Sciences, P.O. Box 2719, Beijing, China. Current address: School of Mathematics, University of New South Wales, Sydney 2052 NSW, Australia (hdqi@maths.unsw.edu.au).

[24]. We note that $E(x)$ is a nonsmooth equation. By introducing the Gabriel–Moré smoothing function for $\Pi_X(x)$, we can approximate $E(\cdot)$ by

$$\bar{G}(u,x) := F(p(u,x)) + x - p(u,x), \qquad (u,x) \in \mathbb{R}^n \times \mathbb{R}^n, \tag{3}$$

where for each $i \in N := \{1,\dots,n\}$, $p_i(u,x)$ is derived via the Gabriel–Moré smoothing function; see [12, 22] for such a procedure. We note that for some popular choices, $p(u,x)$ is continuously differentiable except at the point $(u,x) \in \mathbb{R}^n \times \mathbb{R}^n$ with some $i \in N$ such that $u_i = 0$. It is also noted that for any $(u,x) \in \mathbb{R}^n \times \mathbb{R}^n$, $p(u,x) \in X$ [12, Lemma 2]. Hence for $\bar{G}$ to be well defined, it is sufficient to assume that F is defined on X only. This is an interesting feature of the function $\bar{G}$. Based on the function

$$\bar{H}(z) := \begin{pmatrix} u \\ \bar{G}(z) \end{pmatrix},$$

where $z = (u,x) \in \mathbb{R}^n \times \mathbb{R}^n$, Qi, Sun, and Zhou proposed a smoothing Newton method with global and superlinear/quadratic convergence results for the three most popular choices of the Gabriel–Moré smoothing function. The outstanding feature of their method is that, in each iteration, they use a slightly modified Newton direction based on $\bar{H}$. This modification is crucial to the design of their algorithm. They showed that if F is a uniform P-function, their method generates an infinite sequence converging to the unique solution of BVI. In fact their method is well defined for P-functions. We note that their method may not be well defined for P_0 BVI with some free variables since in this case the underlying Jacobian might be singular. Fortunately, we will show in this paper that the drawback can be overcome by some regularization techniques on F.

Regularization techniques were used recently by Facchinei to study the structure of solution set of P_0 nonlinear complementarity problems [8] and by Facchinei and Kanzow to propose an inexact regularization method for P_0 NCP and to study the trajectory property of the regularized problems [9]. The regularization in the sense of *Tikhonov* is to replace the function $F(x)$ by $F_u(x)$, where

$$F_u(x) := F(x) + \mathrm{diag}(u)x \quad \text{and} \quad u \in \mathbb{R}^n_{++}.$$

Consequently, regularization methods try to solve, instead of the original problem, a sequence of the regularized problems obtained by replacing F in the original problems by F_u and let u converge to $\mathbf{0}$. The following desirable property obtained by Facchinei and Kanzow [9, Lemma 3.2] will be used in our subsequent analysis.

PROPOSITION 1.1. *Let $u \in \mathbb{R}^n_{++}$ be arbitrary and $F(x)$ be a continuously differentiable P_0-function in $\mathbb{R}^n$. Then the Jacobian matrices $F'_u(x)$ are P-matrices for all $x \in \mathbb{R}^n$. In particular, the function F_u is a P-function.*

So it is natural to consider the regularized version of the function $\bar{H}$ in the sense of *Tikhonov*:

$$H(z) := \begin{pmatrix} u \\ G(z) \end{pmatrix},$$

where $z = (u,x) \in \mathbb{R}^n \times \mathbb{R}^n$ and

$$G(z) := \bar{G}(z) + \mathrm{diag}(u)x.$$

We then apply the Qi–Sun–Zhou method to the function H. The regularization allows us to establish stronger convergence results. More precisely, let $\{z^k = (u^k, x^k)\}$ be a sequence generated by the algorithm; we will show $u^k \to \mathbf{0}$ and $\|H(z^k)\| \to 0$. The only additional assumption for those results is that $F(x)$ is a P_0-function on X. Note that the solution set of the corresponding P_0 BVI may be empty. We also note that such convergence results hold without any restriction on the value choice of a or b as required in [22]. If we further assume that the P_0 BVI has a nonempty bounded solution set, then the generated sequence $\{z^k\}$ remains bounded; hence it has at least one accumulation point, the projection of which on X is a solution of the BVI. In particular, the regularization method presented in this paper can handle problems with free variables. This is an interesting feature of our method and makes the method applicable to MVIP since it is well known that an MVIP can be reduced to a P_0 BVI (necessarily with some free variables) by enlarging the dimension of the problem [13]. We summarize that if F is assumed to be a P_0-function on X and the solution set is nonempty and bounded, then our regularized smoothing Newton method is able to find a solution of BVI.

The paper is organized as follows. In section 2 we give some definitions and review the Gabriel–Moré smoothing procedure. Section 3 includes the algorithm itself with some properties. In section 4 we establish the global, superlinear/quadratic convergence of the algorithm. We report some preliminary numerical results in section 5. Conclusions are drawn in section 6.

We introduce some notation here. Let $\mathbb{R}_{++}$ denote the set of all positive real numbers, and let $\mathbb{R}^n_{++}$ be the set of all vectors in $\mathbb{R}^n$ whose entries are positive. If $u \in \mathbb{R}^n$, $\operatorname{diag}(u)$ is the diagonal matrix whose ith diagonal element is u_i. For a continuously differentiable function $\Phi : \mathbb{R}^n \to \mathbb{R}^m$, we denote the Jacobian of Φ at $x \in \mathbb{R}^n$ by $\Phi'(x)$, whereas the transposed Jacobian is denoted by $\nabla\Phi(x)$. When $m = 1$, $\nabla\Phi(x)$ is viewed as a column vector. $\|\cdot\|$ denotes the Euclidean norm. Let $\mathbf{0}$ denote the zero vector in $\mathbb{R}^n$ and e be the vector of all ones in $\mathbb{R}^n$. Let $N := \{1, \dots, n\}$ and $\mathcal{S}$ be the solution set of P_0 BVI.

2. Preliminaries. In this section, we restate briefly the procedure of the Gabriel–Moré smoothing method to choose the function $p(u, x)$ involved in $G(u, x)$ [22] and review some definitions that will be used in the subsequent analysis.

Let $\rho : \mathbb{R} \to \mathbb{R}_+$ be a density function with a bounded absolute mean. For any three numbers $c \in \mathbb{R} \cup \{-\infty\}$, $d \in \mathbb{R} \cup \{\infty\}$ with $c \le d$ and $e \in \mathbb{R}$, let $\operatorname{mid}(c, d, e)$ denote the median function, the projection of e on $[c, d]$. Then the Gabriel–Moré smoothing function $\phi(\mu, c, d, w)$ for $\Pi_{[c,d]\cap\mathbb{R}}(w)$ [12] is defined by

$$\phi(\mu, c, d, w) = \int_{-\infty}^{\infty} \operatorname{mid}(c, d, w - \mu s)\rho(s)ds, \qquad (\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}.$$

If $c = -\infty$ and/or $d = \infty$, the value of ϕ takes the limit of ϕ as $c \to -\infty$ and/or $d \to \infty$, correspondingly. It is easy to see

$$\phi(0, c, d, w) = \Pi_{[c,d]}(w).$$

Now we define the ith component of the function $p(u, x) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ by

$$p_i(u, x) := \phi(|u_i|, a_i, b_i, x_i), \qquad i \in N.$$

In fact, our algorithm guarantees $u > 0$ at all iterations. We define $p(u, x)$ in the whole space in order to state its semismooth property, which in turn plays an important role in the local convergence analysis. We also need some properties of ϕ with $\mu > 0$. So let $\phi_{cd} : \mathbb{R}_{++} \times \mathbb{R} \to \mathbb{R}$ be defined by

$$\phi_{cd}(\mu, w) := \phi(\mu, c, d, w), \qquad (\mu, w) \in \mathbb{R}_{++} \times \mathbb{R},$$

and for any given $\mu \in \mathbb{R}_{++}$, let $\phi_{\mu cd} : \mathbb{R} \to \mathbb{R}$ be defined by

$$\phi_{\mu cd}(w) := \phi(\mu, c, d, w), \qquad w \in \mathbb{R}.$$

Then we have the following continuity properties; see [12, Lemma 2.3] for (i) and [22, Lemma 2.2] for (ii).

LEMMA 2.1. (i) *For any given* $\mu > 0$, *the mapping* $\phi_{\mu cd}(\cdot)$ *is continuously differentiable with* $\phi'_{\mu cd}(w) \in [0, 1]$ *for any* $w \in \mathbb{R}$.

(ii) *The mapping* $\phi_{cd}(\cdot)$ *is Lipschitz continuous on* $\mathbb{R}_{++} \times \mathbb{R}$.

There are three popular choices for the density function $\rho(s)$, which lead to three well-known smoothing functions ϕ, namely, the neural networks smoothing function, the Chen–Harker–Kanzow–Smale (CHKS) smoothing function [3, 15, 16], and the uniform smoothing function. For example, let the density function be

$$\rho(s) = \frac{2}{(s^2 + 4)^{3/2}}.$$

Then the CHKS smoothing function is

$$\phi(\mu, c, d, w) = \frac{c + \sqrt{(c - w)^2 + 4\mu^2}}{2} + \frac{d - \sqrt{(d - w)^2 + 4\mu^2}}{2}, \tag{4}$$

where $(\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}$. Apparently, $\phi_{cd}(\cdot)$ is continuously differentiable at any $(\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}$. If $c = 0$ and $d = \infty$, then the smoothing function in (4) reduces to the CHKS smoothing NCP function:

$$\phi(\mu, 0, \infty, w) = \frac{\sqrt{w^2 + 4\mu^2} + w}{2}, \qquad (\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}.$$

The CHKS smoothing function will be used in the numerical experiments of our Algorithm 3.3. Since neither of the other two smoothing functions is used explicitly in this paper, we do not state their analytic expression; for detailed discussion on those functions see Examples 2.1–2.3 in [22]. We note that all those smoothing functions give rise to the following properties [22, Theorem 3.1].

PROPOSITION 2.2. (i) $\phi_{cd}(\cdot)$ *is continuously differentiable at* $(\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}$.

(ii) H *is semismooth at any* $z \in \mathbb{R}^{2n}$.

(iii) *If for some point* $z \in \mathbb{R}^{2n}$, F' *is Lipschitz continuous around* $p(z) \in \mathbb{R}^n$, *then* H *is strongly semismooth at* z.

If (i) in Proposition 2.2 holds, then the approximation function $G(u, x)$ is continuously differentiable at any $(u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. We will extract this property as a basic assumption on ϕ in Assumption 3.1. The concept of semismoothness was originally introduced by Mifflin [18] for functionals and later was extended by Qi and Sun [21] to the vector-valued functions $\Phi : \mathbb{R}^{m_1} \to \mathbb{R}^{m_2}$. A locally Lipschitz continuous vector-valued function $\Phi : \mathbb{R}^{m_1} \to \mathbb{R}^{m_2}$ has a generalized Jacobian $\partial\Phi(x)$ in the sense of Clarke [6]. Now we recall some definitions.

DEFINITION 2.3. *Suppose* $m_1 = m_2$; *we say* Φ *is CD-regular at a point* x *if all* $V \in \partial\Phi(x)$ *are nonsingular.*

DEFINITION 2.4. *The function* $F : X \subset \mathbb{R}^n \to \mathbb{R}^n$ *is said to be (over the set* X*) a*

- *P_0-function if for all $x, y \in X$ with $x \neq y$*

$$\max_{\substack{i\in N\\ x_i\neq y_i}} (x_{i_0} - y_{i_0})[F_{i_0}(x) - F_{i_0}(y)] \geq 0;$$

- *P-function if for all $x, y \in X$ with $x \neq y$*

$$\max_{i\in N}(x_i - y_i)[F_i(x) - F_i(y)] > 0;$$

- *uniform P-function if there exists a constant $\mu > 0$ such that for all $x, y \in X$*

$$\max_{i\in N}(x_i - y_i)[F_i(x) - F_i(y)] \geq \mu\|x - y\|^2;$$

- *monotone function if for all $x, y \in X$*

$$(x - y)^T(F(x) - F(y)) \geq 0.$$

Those function classes are closely related to each other; for example, a monotone function is necessarily a P_0-function. Although Proposition 1.1 showed that $F_u(x)$ is a P-function for $u \in \mathbb{R}^n_{++}$ if F is a P_0-function, a counterexample in [9] shows that it is not necessarily a uniform P-function.

3. Regularized smoothing Newton method.

In this section, we apply the Qi–Sun–Zhou smoothing Newton method to our regularized function H. First we state assumptions that are crucial to our method.

ASSUMPTION 3.1. (i) *The function $\phi_{cd}(\cdot)$ is continuously differentiable at any $(\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}$.*

(ii) *$F(x)$ is a P_0-function on X.*

We note that (i) in Assumption 3.1 is used only to guarantee the differentiability of function H at any $(u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$, while (ii) guarantees the nonsingularity of the Jacobian of H.

PROPOSITION 3.2. *Suppose that Assumption* 3.1 *holds for a chosen smoothing function $\phi(\mu, c, d, w)$ with $(\mu, w) \in \mathbb{R}_{++} \times \mathbb{R}$. Then*

(i) *the mapping $H(\cdot)$ is continuously differentiable at any $z = (u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$ and*

$$H'(z) = \begin{pmatrix} I & 0 \\ G'_u(z) & G'_x(z) \end{pmatrix}, \tag{5}$$

where

$$G'_u(z) := (F'(p(z)) - I)D(u) + \mathrm{diag}(x),$$
$$G'_x(z) := F'(p(z))C(x) + \mathrm{diag}(u) + I - C(x)$$

and $D(u) = \mathrm{diag}(d_i(u), i \in N)$, $C(x) = \mathrm{diag}(c_i(x), i \in N)$, $d_i(u) = \partial p_i(u, x)/\partial u_i$, $c_i(x) = \partial p_i(u, x)/\partial x_i$, and $c_i(x) \in [0, 1], i \in N$.

(ii) *$H'(z)$ is nonsingular at any $z = (u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$.*

Proof. (i) Since Assumption 3.1 holds for $\phi(\cdot)$, it is easy to know that $H(\cdot)$ is continuously differentiable at any $z = (u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. By direct computation we have (5). From Lemma 2.1(i), $c_i(x) \in [0, 1], i \in N$.

(ii) It is known [19, Corollary 5.3 and Theorem 5.8] that $F'(x)$ being a P_0-matrix (i.e., all of its principal minors are nonnegative) for all x in an open box $X \subset \mathbb{R}^n$

is equivalent to F being a P_0-function on X. Now let $z = (u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. Since $F(x)$ is a P_0-function on X and $p(z) \in X$, $F(p(z))$ is a P_0-matrix. Now we assume that there exists $v \in \mathbb{R}^n$ such that

$$(F'(p(z))C(x) + \mathrm{diag}(u) + I - C(x))v = 0. \tag{6}$$

If $C(x)v \neq 0$, then, by noting $c_i(x) \geq 0$ for all $i \in N$, for any index $i \in N$ with $(C(x)v)_i \neq 0$, we have

$$\begin{aligned}[C(x)v]_i[F'(p(z))C(x)v]_i &= -[C(x)v]_i[(\mathrm{diag}(u) + I - C(x))v]_i\\ &= -(u_i + 1)c_i(x)v_i^2 - [C(x)v]_i^2 < 0,\end{aligned}$$

which contradicts the fact that $F'(p(z))$ is a P_0-matrix. Hence $C(x)v = 0$; then (6) implies $(\mathrm{diag}(u) + I)v = 0$, which in turn implies $v = 0$ due to the fact $u \in \mathbb{R}^n_{++}$. This proved that $G'_x(z)$ is nonsingular. Hence $H'(z)$ is nonsingular at any $z = (u, x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. □

Before stating the Qi–Sun–Zhou smoothing Newton method, we need the following parameter setting and some functions which are developed in [22].

Choose $\bar{u} \in \mathbb{R}^n_{++}$ and $\gamma \in (0, 1)$ such that $\gamma\|\bar{u}\| < 1$. Let $\bar{z} := (\bar{u}, 0) \in \mathbb{R}^n \times \mathbb{R}^n$. Define the merit function $\psi : \mathbb{R}^{2n} \to \mathbb{R}_+$ by

$$\psi(z) := \|H(z)\|^2$$

and define $\beta : \mathbb{R}^{2n} \to \mathbb{R}_+$ by

$$\beta(z) := \gamma \min\{1, \psi(z)\}.$$

Let

$$\Omega := \{z = (u, x) \in \mathbb{R}^n \times \mathbb{R}^n |\ u \geq \beta(z)\bar{u}\}.$$

Then, because for any $z \in \mathbb{R}^{2n}, \beta(z) \leq \gamma < 1$, it follows that for any $x \in \mathbb{R}^n$,

$$(\bar{u}, x) \in \Omega.$$

Algorithm 3.3 (regularized smoothing Newton method).

(S.0) *Initialization. Choose constants* $\delta, \gamma \in (0, 1)$, $\sigma \in (0, 1/2)$, *and* $\bar{u} \in \mathbb{R}^n_{++}$ *such that* $\gamma\|\bar{u}\| < 1$. *Let* $u^0 := \bar{u}, x^0 \in \mathbb{R}^n$ *be an arbitrary point. Let* $z^0 := (u^0, x^0), \bar{z} := (\bar{u}, 0)$, *and* $k := 0$.

(S.1) *Termination criterion. If* $H(z^k) = 0$, *then stop. Otherwise, let* $\beta_k := \beta(z^k)$.

(S.2) *Modified Newton direction. Compute* $\Delta z^k := (\Delta u^k, \Delta x^k) \in \mathbb{R}^n \times \mathbb{R}^n$ *by*

$$H(z^k) + H'(z^k)\Delta z^k = \beta_k \bar{z}. \tag{7}$$

(S.3) *Line search strategy. Let* ℓ_k *be the smallest nonnegative integer* ℓ *satisfying*

$$\psi(z^k + \delta^\ell \Delta z^k) \leq [1 - 2\sigma(1 - \gamma\|\bar{u}\|)\delta^\ell]\psi(z^k). \tag{8}$$

Define $z^{k+1} := z^k + \delta^{\ell_k}\Delta z^k$.

(S.4) *Update.* $k := k + 1$*; return to step* (S.1).

Since we have assumed that Assumption 3.1 is satisfied, $H(\cdot)$ is continuously differentiable at any $z^k \in \mathbb{R}^n_{++} \times \mathbb{R}^n$, and $H'(z^k)$ is nonsingular with the following relation:

$$H'(u,x) = \bar{H}'(u,x) + \begin{pmatrix} 0 & 0 \\ \operatorname{diag}(x) & \operatorname{diag}(u) \end{pmatrix}$$

for any $(u,x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. We can view $H'(z)$ as a diagonal perturbation of $\bar{H}(z)$. This observation makes it easy to understand that H inherits many properties of $\bar{H}$, especially properties related to Algorithm 3.3. For more comments on Algorithm 3.3, see Remarks in [22]. The result below states that Algorithm 3.3 is well defined and the generated sequence remains in Ω. Since the proof is similar to that of Proposition 4.5 in [22], we omit it.

PROPOSITION 3.4. *Suppose that Assumption* 3.1 *holds. Then Algorithm* 3.3 *is well defined and generates an infinite sequence* $\{z^k = (u^k, x^k)\}$. *Moreover,*

$$u^k \in \mathbb{R}^n_{++} \quad \textit{and} \quad \{z^k\} \subset \Omega.$$

4. Convergence analysis. In this section, we will prove that Algorithm 3.3 generates an infinite sequence such that the merit function converges to zero, and the projection of any accumulation point on X is a solution of P_0 BVI. Moreover, if the solution set of BVI is nonempty and bounded, then Algorithm 3.3 is able to solve problem (1). First, for any given $u \in \mathbb{R}^n$, we define $\psi_u(x) : \mathbb{R}^n \to \mathbb{R}_+$,

$$\psi_u(x) := \|G(u,x)\|^2.$$

It is easy to see that for any fixed $u \in \mathbb{R}^n_{++}$, ψ_u is continuously differentiable with the gradient given by

$$\nabla\psi_u(x) = 2(G'_x(u,x))^T G(u,x),$$

where $G'_x(u,x) = F'(p(z))C(x) + \operatorname{diag}(u) + I - C(x)$ and $C(x)$ is defined as in Proposition 3.2. By repeating the proof of (ii) in Proposition 3.2, $G'_x(u,x)$ is nonsingular at any point $(u,x) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. We note that for any $z = (u,x) \in \mathbb{R}^n \times \mathbb{R}^n$,

$$\psi(z) = \|u\|^2 + \psi_u(x). \tag{9}$$

LEMMA 4.1. *If* $\mathcal{S}$, *the solution set of* P_0 *BVI, is nonempty and bounded, then the set*

$$\Gamma := \{x \in \mathbb{R}^n \,|\, \psi_{\mathbf{0}}(x) = 0\}$$

is also nonempty and bounded.

Proof. First, notice that $\psi_{\mathbf{0}}(x) = \|E(x)\|^2$, where $E(x)$ is defined by (2). If $y \in \mathcal{S}$, then $x := y - F(y) \in \Gamma$ [24]. Hence the nonemptiness of $\mathcal{S}$ implies the nonemptiness of Γ. Now we assume $\mathcal{S}$ is bounded. Let $x \in \Gamma$; then $\Pi_X(x) \in \mathcal{S}$ [24] and

$$\begin{aligned} \|x\| &= \|\Pi_X(x) - F(\Pi_X(x))\| \\ &\le \|F(\Pi_X(x))\| + \|\Pi_X(x)\| \\ &\le \sup_{y \in \mathcal{S}} (\|F(y)\| + \|y\|). \end{aligned}$$

The boundedness of $\mathcal{S}$, together with the continuity of $F(\cdot)$, implies the boundedness of Γ. ◻

The following lemma is key to our global convergence. The proof technique is taken from [9].

LEMMA 4.2. *If F is a P_0-function on X, for any $u \in \mathbb{R}^n_{++}$ and $r > 0$, define the level set*

$$L_u(r) := \{x \in \mathbb{R}^n | \ \psi_u(x) \le r\}\,. \tag{10}$$

Then, for any $u_1 \ge u_2 \in \mathbb{R}^n_{++}$ and $r > 0$, the set

$$L(r) := \cup_{u \in [u_1,u_2]} L_u(r)$$

is bounded.

Proof. Suppose to the contrary that the lemma is false. Then for some fixed $r > 0$, we can find a sequence $\{(u^k, x^k)\}$ such that $u^k \in [u_1, u_2]$ and

$$\psi_{u^k}(x^k) \le r \quad \text{and} \quad \|x^k\| \to \infty. \tag{11}$$

Let $z^k = (u^k, x^k) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$. It is easy to prove that

$$|\text{mid}(a_i, b_i, x_i^k)| \to \infty \ \Rightarrow \ |x_i^k| \to \infty \ \text{ and } \ |x_i^k - \text{mid}(a_i, b_i, x_i^k)| \to 0, \ i \in N. \tag{12}$$

Since $p_i(z), i \in N$, is Lipschitz continuous with Lipschitz constant $L' > 0$ (Lemma 2.1), we have

$$|p_i(u^k, x^k) - \text{mid}(a_i, b_i, x_i^k)| = |p_i(u^k, x^k) - p_i(\mathbf{0}, x^k)| \le L'|u_i^k|, \ i \in N. \tag{13}$$

Hence (12) and (13) imply that for all k large enough,

$$|\text{mid}(a_i, b_i, x_i^k)| \to \infty \ \Rightarrow \ |x_i^k - p_i(u^k, x^k)| \le 2L'|u_i^k|, \ i \in N. \tag{14}$$

Define index set J by $J := \{i| \ \{p_i(z^k)\} \text{ is unbounded}, i \in N\}$. The set J is not empty, because otherwise

$$\psi_{u^k}(x^k) = \|F(p(z^k)) + \text{diag}(u^k)x^k + x^k - p(z^k)\| \to \infty.$$

Let $\tilde{z}^k = (\tilde{u}^k, \tilde{x}^k) \in \mathbb{R}^n_{++} \times \mathbb{R}^n$ be defined by

$$\tilde{u}_i^k = \begin{cases} u_i^k & \text{if } i \notin J, \\ 0 & \text{if } i \in J, \end{cases} \qquad i \in N,$$

and

$$\tilde{x}_i^k = \begin{cases} x_i^k & \text{if } i \notin J, \\ 0 & \text{if } i \in J, \end{cases} \qquad i \in N.$$

Then

$$p_i(\tilde{z}^k) = \begin{cases} p_i(z^k) & \text{if } i \notin J, \\ \text{mid}(a_i, b_i, 0) & \text{if } i \in J, \end{cases} \qquad i \in N.$$

Hence $\{\|p(\tilde{z}^k)\|\}$ is bounded. Since $p(\tilde{z}^k) \in X$ and F is a P_0-function on X, we get

$$\begin{aligned} 0 \le & \max_{\substack{i \in N \\ p_i(z^k) \ne p_i(\tilde{z}^k)}} (p_i(z^k) - p_i(\tilde{z}^k))[F_i(p(z^k)) - F_i(p(\tilde{z}^k))] \\ &= \max_{i \in J} (p_i(z^k) - p_i(\tilde{z}^k))[F_i(p(z^k)) - F_i(p(\tilde{z}^k))] \\ &= (p_j(z^k) - p_j(\tilde{z}^k))[F_j(p(z^k)) - F_j(p(\tilde{z}^k))], \end{aligned} \tag{15}$$

where j is one of the indices for which the max is attained, which we have, without loss of generality, assumed to be independent of k. Since $j \in J$, we have that by passing to a subsequence if necessary

$$|p_j(z^k)| \to \infty. \tag{16}$$

We now consider two cases.

Case 1. $p_j(z^k) \to \infty$. It follows from (13) that

$$\mathrm{mid}(a_j, b_j, x_j^k) \to \infty,$$

which implies in turn by (12) that $x_j^k \to \infty$. Hence (14) holds for index j. Since $F_j(p(\tilde{z}^k))$ is bounded by the continuity of F_j, (15) implies that $F_j(p(z^k))$ does not tend to $-\infty$. Thus

$$F_j(p(z^k)) + u_j^k x_j^k + x_j^k - p_j(z^k) \to \infty,$$

since $u_j^k x_j^k \to \infty$.

Case 2. $p_j(z^k) \to -\infty$. Then once again (12) and (13) imply that $x_j^k \to -\infty$. It follows from (15) that

$$F_j(p(z^k)) \le 2F_j(p(\tilde{z}^k))$$

for all k sufficiently large. Hence by (14)

$$F_j(p(z^k)) + u_j^k x_j^k + x_j^k - p_j(z^k) \to -\infty,$$

since $u_j^k x_j^k \to -\infty$.

In either case, we get $\psi_{u^k}(x^k) \to \infty$, which contradicts (11). This completes the proof. □

COROLLARY 4.3. *Suppose that F is a P_0-function on X. Then for any $u \in \mathbb{R}^n_{++}$, the function $\psi_u(x)$ is coercive, i.e.,*

$$\lim_{\|x\|\to\infty} \psi_u(x) = \infty.$$

Proof. Let $u_1 = u_2 = u$ in Lemma 4.2. Then $L_u(r)$ is bounded for any $r > 0$ or, equivalently, $\psi_u(x)$ is coercive. □

The next result can be proved by noticing the results in Lemma 4.1 and Corollary 4.3 above and by following the similar proof lines of [9, Theorem 5.4].

LEMMA 4.4. *Let F be a P_0-function and assume that the solution set $\mathcal{S}$ of P_0 BVI is nonempty and bounded. Suppose that there is a sequence $\{(u^k, x^k)\} \subset \mathbb{R}^n_{++} \times \mathbb{R}^n$ (not necessarily generated by Algorithm* 3.3*) such that*

$$u^k \to \mathbf{0} \quad \text{and} \quad \psi_{u^k}(x^k) \to 0.$$

Then the sequence $\{x^k\}$ remains bounded and every accumulation point of $\{x^k\}$ is a solution of $\psi_{\mathbf{0}}(x) = 0$.

Now we state our global convergence result.

THEOREM 4.5. *Suppose that Assumption* 3.1 *is satisfied. Then the following conditions hold.*

(1) *An infinite sequence* $\{z^k\}$ *is generated by Algorithm* 3.3 *and*

$$\lim_{k\to\infty}\psi(z^k)=0 \qquad \textit{and} \qquad \lim_{k\to\infty}u^k=\mathbf{0}. \tag{17}$$

Hence each accumulation point of $\{z^k\}$ *is a solution of* $H(z)=0$.

(2) *The sequence* $\{z^k\}$ *is bounded if* P_0 *BVI has a nonempty and bounded solution set* $\mathcal{S}$. *Hence there exists at least one accumulation point, say,* $\tilde{z}$, *of* $\{z^k\}$ *satisfying* $H(\tilde{z})=0$.

Proof. (1) Since Assumption 3.1 is satisfied, it follows from Proposition 3.4 that an infinite sequence $\{z^k\}$ is generated by Algorithm 3.3. If (17) holds, it follows from the continuity of H that each accumulation point of $\{z^k\}$ is a solution of $H(z)=0$. Now we show (17) is valid. From the design of Algorithm 3.3, $\psi(z^{k+1})<\psi(z^k)$ for all $k\geq 0$. Hence the two sequences $\{\psi(z^k)\}$ and $\{\beta(z^k)\}$ are monotonically decreasing. Since $\psi(z^k),\beta(z^k)\geq 0$ $(k\geq 0)$, there exist $\tilde{\psi},\tilde{\beta}\geq 0$ such that $\psi(z^k)\to\tilde{\psi}$ and $\beta(z^k)\to\tilde{\beta}$ as $k\to\infty$. Suppose that the first equality in (17) is false, i.e., $\tilde{\psi}>0$. It is easy to see

$$\tilde{\beta}=\gamma\min\{1,\tilde{\psi}\}>0.$$

Since $\{z^k\}\subset\Omega$, we claim that

$$u^k\geq\tilde{\beta}\bar{u} \quad \text{for all} \ \ k\geq 0. \tag{18}$$

Note that the boundedness of $\{\psi(z^k)\}$ implies the boundedness of u^k. Let $\hat{u}\in\mathbb{R}^n_{++}$ be sufficiently large such that $u^k\leq\hat{u}$ for all $k\geq 0$. Let

$$L(\psi(z^0)):=\cup_{u\in[\tilde{\beta}\bar{u},\hat{u}]}L_u\left(\psi(z^0)\right),$$

where $L_u(\psi(z^0))$ is defined by (10). It must hold that $x^k\in L(\psi(z^0))$, since x^k belongs to the set $L_{u^k}(\psi(z^0))$ and $u^k\in[\tilde{\beta}\bar{u},\hat{u}]$. It follows from Lemma 4.2 that the set $L(\psi(z^0))$ is bounded. Hence $\{x^k\}$ is bounded. By noticing the boundedness of $\{u^k\}$, we have proved that the sequence $\{z^k\}$ is bounded. Let $\tilde{z}=(\tilde{u},\tilde{x})\in\mathbb{R}^n\times\mathbb{R}^n$ be an accumulation point of $\{z^k\}$. By taking a subsequence if necessary, we may assume that $\{z^k\}$ converges to $\tilde{z}$. It is easy to see that

$$\tilde{\psi}=\psi(\tilde{z}),\ \beta(\tilde{z})=\tilde{\beta} \quad \text{and} \quad \tilde{z}\in\Omega,\ \tilde{u}\geq\tilde{\beta}\bar{u}>\mathbf{0}.$$

Then from Proposition 3.2, $H'(\tilde{z})$ exists and is nonsingular. By repeating the proof of [22, Theorem 5.1], we can find a nonnegative integer ℓ such that for all k sufficiently large,

$$\psi(z^{k+1})\leq[1-2\sigma(1-\gamma\|\bar{u}\|)\delta^\ell]\psi(z^k).$$

This contradicts the fact that the sequence $\{\psi(z^k)\}$ converges to $\tilde{\psi}>0$. So we proved that $\tilde{\psi}=0$, which necessarily implies $u^k\to\mathbf{0}$.

(2) Now we further assume that P_0 BVI has a nonempty and bounded solution set $\mathcal{S}$. We note that $\psi_u(x)$ is continuous for both variables u and x. The results in (17) and (9) imply that

$$\lim_{k\to\infty}\psi_{u^k}(x^k)=\lim_{k\to\infty}(\psi(x^k)-\|u^k\|^2)=0.$$

It is easy to see that the sequence $\{(u^k,x^k)\}$ satisfies all the conditions in Lemma 4.4. Hence the sequence $\{x^k\}$ is bounded and any accumulation point $\tilde{x}$ of $\{x^k\}$ is a solution of $\psi_{\mathbf{0}}(x) = 0$. Hence $\tilde{z} := (\mathbf{0}, \tilde{x})$ is a solution of $H(z) = 0$. This completes our proof. □

Qi, Sun, and Zhou established superlinear/quadratic convergence results for Algorithm 3.3 with $H(z)$ replaced by $\bar{H}(z)$ under the nonsingularity assumption made on $\partial \bar{H}(z)$. If z^* is a solution of $H(z) = 0$, then it is easy to know that $\partial H(z^*) = \partial \bar{H}(z^*)$. We refer to [22] for a discussion on the nonsingularity of all $V \in \partial \bar{H}(z^*)$. Hence local conditions guaranteeing the Qi–Sun–Zhou method to be superlinearly/quadratically convergent should also lead to the same convergence rate for Algorithm 3.3. This observation makes the proof of the superlinear and quadratic convergence result below for Algorithm 3.3 go along the same lines as that in [22, Theorem 7.1]. We omit the proof.

THEOREM 4.6. *Suppose that Assumption* 3.1 *is satisfied and z^* is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm* 3.3. *Suppose that H is semismooth (strongly semismooth, respectively) at z^* and z^* is a CD-regular point with respect to H. Then the whole sequence $\{z^k\}$ converges to z^*,*

$$\|z^{k+1} - z^*\| = o(\|z^k - z^*\|) \ (= O(\|z^k - z^*\|^2), \quad \textit{respectively})$$

and

$$u_i^{k+1} = o(u_i^k) \ (= O(u_i^k)^2, \quad \textit{respectively}), \qquad i \in N.$$

The convergence results established above are quite satisfactory. Even though we do not know whether the P_0 BVI has a solution, Algorithm 3.3 produces a sequence such that the value sequence of the merit function converges to its global minimum; meanwhile the smoothing parameter converges to zero, ensuring that any accumulation point, say, $z^* = (0, x^*)$ if exists, is a solution of $H(z) = 0$. Hence the projection $\Pi_X(x^*)$ of x^* on X is a solution of BVI. We note that those results hold under only the assumption that F is a P_0-function on X. Moreover, if the solution set of P_0 BVI is nonempty and bounded, Algorithm 3.3 is able to solve the above problem. We stress that the proof of the boundedness of the iteration sequence is not from proving that the corresponding level set is bounded, but from Lemma 4.4, which is the corresponding BVI part of the Facchinei and Kanzow result on NCP [9]. Both of the following conditions imply that the merit function $\|H(z)\|^2$ has bounded level sets [22, Theorems 6.1–6.2] and hence imply the boundedness of the solution set.

COROLLARY 4.7. *Suppose one of the following conditions holds:*

(i) *X is bounded, i.e., $a_i > -\infty, b_i < \infty$ for all $i \in N$;*

(ii) *F is a uniform P-function on X.*

Then the solution set of BVI is bounded.

5. Preliminary numerical results. In this section, we present some numerical experiments for the nonmonotone line search version of Algorithm 3.3. We replaced the monotone Armijo condition (8) with the nonmonotone line search version:

(S.3′) Let ℓ_k be the smallest nonnegative integer ℓ satisfying

$$z^k + \delta^\ell \Delta z^k \in \Omega \tag{19}$$

and

$$\psi(z^k + \delta^\ell \Delta z^k) \le \mathcal{W} - 2\sigma(1 - \gamma\|\bar{u}\|)\delta^\ell \psi(z^k), \tag{20}$$

Table 1
Numerical results for NCPs.

Example	Dim.	x^0	$It\#1/\#2$	$NF\#1/\#2$	$FF\#1/\#2$
Prob. 1	4	0	6/6	9/9	1.1e-14/5.6e-18
	4	e-F(e)	6/4	8/6	5.0e-17/2.7e-20
Prob. 2	10000	e	5/5	6/6	1.7e-25/1.1e-21
	10000	0	5/5	6/6	1.1e-21/1.1e-21
Prob. 3	10000	e	17/5	30/6	4.6e-14/1.1e-21
	10000	0	17/5	30/6	4.9e-14/1.1e-21
Prob. 4	5	e	16/16	17/17	5.8e-21/2.8e-24
	5	0	19/19	20/20	8.8e-22/7.6-26
Prob. 5	4	0-F(0)	7/5	8/7	7.8e-18/8.8e-15
	4	-10e	7/5	9/6	3.7e-20/2.4e-19
Prob. 6	10	0	10/10	11/11	4.5e-15/5.0e-15
	10	e	7/8	8/9	7.8e-13/1.4e-22
	10	10e	7/7	8/8	7.0e-13/1.6e-13
Prob. 7a	4	e	4/8	5/11	2.6e-18/3.4e-23
Prob. 7b	4	e	7/5	7/11	1.0e-15/2.7e-21
Prob. 8	4	-e	6/5	10/9	1.9e-18/4.2e-16
	4	e-F(e)	7/6	9/8	6.6e-19/3.2e-19
Prob. 9	42	0	9/15	13/28	5.3e-18/8.2e-26
	42	e	9/14	15/24	5.9e-15/8.3e-20
	42	-e	10/10	14/15	2.2e-18/8.5e-25
Prob. 10	50	a	27/14	89/28	1.8e-18/1.2e-19
	50	0	29/17	92/41	1.9e-18/5.5e-15
Prob. 12	10	0	6/6	11/10	1.4e-15/9.9e-20

where $\mathcal{W}$ is any value satisfying

$$\psi(z^k) \le \mathcal{W} \le \max_{j=0,1,\ldots,M^k} \psi(z^{k-j})$$

and M^k are nonnegative integers bounded above for all k such that the occurrence of nonnegative indices does not happen. Define $z^{k+1} := z^k + \delta^{\ell_k}\Delta z^k$. The reason why we choose a nonmonotone line search here is that in most cases it increases the stability of algorithms. The requirement (19), which automatically holds for Algorithm 3.3, is to guarantee the global convergence of the algorithm. In the implementation we choose $\mathcal{W}$ as follows:

(1) Set $\mathcal{W} = \psi(z^0)$ at the beginning of the algorithm.

(2) Keep the value of $\mathcal{W}$ fixed as long as

$$\psi(z^k) \le \min_{j=0,1,\ldots,5} \psi(z^{k-j}). \tag{21}$$

(3) If (21) is not satisfied at the kth iteration, set $\mathcal{W} = \psi(z^k)$.

For a detailed description of the above nonmonotone line search technique and its motivation, see [7].

We choose the CHKS smoothing function in G. The function has been widely used in continuation methods. Algorithm 3.3 with the above steps was implemented in Matlab on an SGI INDIGO workstation. In all computations, the termination criterion is

$$\psi(z^k) \le 10^{-12}$$

and the parameter setting is

$$\sigma = 0.5 \times 10^{-3}, \;\; \delta = 0.5, \;\; \bar{u} = (0.1, \ldots, 0.1), \;\; \gamma = 0.2 \times \min\{1, 1/\|\bar{u}\|\}.$$

Table 2
Numerical results for VIPs.

Example	Dim.	x^0	It	NF	FF
5.1	5	(0 0 100 0 0)	22	33	1.2e-17
		(100 0 0 0 0)	28	47	7.8e-18
		(1 2 3 4 5)	12	15	7.3e-27
		(0 0 0 0 0)	11	17	8.5e-27
		(1 1 1 1 1)	12	19	1.8e-18
5.2	3	(0 0 0)	4	6	6.0e-16
		(0.5 0.5 0.5)	4	5	2.6e-22
		(1 2 3)	4	5	5.8e-23
		(4 3 2)	4	5	2.8e-20
5.3	10	e	19	43	3.4e-19
		0	20	44	1.4e-18
5.4	4	0	7	11	8.1e-18
		e	5	6	1.4e-13
		(-4 -13 -7 -5)	6	10	1.7e-16
5.5	3	(1 1 0)	6	7	4.2e-15
		(4 3 2)	8	9	1.2e-13
		(1 1 1)	7	8	1.3e-17
		(1 2 3)	8	9	3.5e-15
5.6	1	0	15	16	1.4e-13
		e	15	16	9.2e-13
5.7	2	0	4	5	3.4e-16
		e	3	4	2.4e-13
		10e	6	7	2.2e-19
		-e	5	6	1.5e-13

Our numerical results are summarized in Tables 1 and 2, where we present the following data:

Example:	Number of test examples,
Dim:	Dimension of the test problem,
x^0:	Starting point,
It:	Number of iterations needed until termination,
NF :	Number of function evaluations needed until termination,
FF:	Value of $\psi(\cdot)$ at the final iterate.

The test problems are separated into two groups. One group consists of 11 nonlinear complementarity problems, which were also tested by Qi, Sun, and Zhou [22] with the same problem names. Hence we do not give the detailed description of the problems in this group. Numerical results on those problems are summarized in Table 1. The other groups of the test problems are variational inequality problems VIP(F, X), which are taken from the literature with a brief description as follows. It is known that, by introducing Lagrange multipliers, VIP(F, X) can be equivalently reformulated as BVI under some reasonable conditions. We stress that the last two small problems in this group were proposed to show the behavior of our regularized method for problems with unbounded or empty solution sets. The numerical results for this group are reported in Table 2.

Example 5.1. The problem was tested by Taji, Fukushima, and Ibaraki [27] and also was tested as the second example by Kanzow and Qi [17]. F is nonlinear, the dimension $n = 5$, and the feasible region in this example is polyhedral and is given by

$$X := \{x \in \mathbb{R}^n | \; Ax \leq b, x \geq 0\},$$

where the data for F, A, b is specified in Table 6 of [27].

Example 5.2. This example is a variational inequality reformulation of the convex optimization problem 35 from the test problem collection [14] by Hock and Schittkowski and was tested in [17]. Its dimension is $n = 3$, and the feasible set is given by

$$X := \{x \in \mathbb{R}^3 |\, 3 - x_1 - x_2 - 2x_3 \geq 0, x \geq 0\}.$$

Example 5.3. The function F is from the Nash–Cournot production problem discussed in [20]; here $n = 10$. The feasible region was given by

$$X := \left\{ x \in \mathbb{R}^{10} |\, \sum_{i=1}^{10} x_i = 10, x \geq 0 \right\}.$$

The problem was initially tested by Solodov and Tseng [25].

Example 5.4. The function F is from the Kojima–Shindo problem discussed in [20]; here $n = 4$. The feasible region was given by

$$X := \left\{ x \in \mathbb{R}^{4} |\, \sum_{i=1}^{4} x_i = 4, x \geq 0 \right\}.$$

The problem was initially tested by Solodov and Tseng [25].

Example 5.5. This is the first example used in the paper by Fukushima [11]. F is a nonlinear function of dimension $n = 3$, and the feasible set, in contrast to all previous examples is nonlinear; more precisely, it is given by

$$X := \{x \in \mathbb{R}^3 |\, 1 - x_1^2 - 0.4x_2^2 - 0.6x_3^2 \geq 0\}.$$

Example 5.6. This is an one-dimensional problem with no constraints, i.e.,

$$F(x) = e^x \quad \text{and} \quad X = \mathbb{R}.$$

Obviously $F(x) = 0$ has no solution and $F(x)$ is a nonuniform P-function.

Example 5.7. This problem is a two-dimensional linear complementarity problem problem with an unbounded solution set. Here $F(x) = Mx + q$ and

$$M = \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \quad \text{and} \quad q = \begin{pmatrix} -1 \\ 0 \end{pmatrix}.$$

F is a P_0-function and the solution set is given by

$$\mathcal{S} = \{(x_1, x_2) |\, (0, x_2), x_2 \geq 1\} \cup \{(x_1, x_2) |\, x_1 \geq 0, x_2 \geq 0, x_1 + x_2 - 1 = 0\}.$$

If they were available, we took for x^0 the starting point(s) from the literature and set all components of the initial Lagrange multipliers to one when we tested variational inequality problems.

In Table 1, we also give numerical results of the Qi–Sun–Zhou method on those nonlinear complementarity problems for comparison. For example, $It\#1$ stands for the iteration number of our Algorithm 3.3, while $It\#2$ is the iteration number of the Qi–Sun–Zhou algorithm on the same problems. $NF\#1/\#2$ and $FF\#1/\#2$ have the same meaning. As we see from Table 1, our regularized smoothing Newton method behaves similarly to the one in [22]. It is quite reasonable since for NCPs $a_i = 0$ and $b_i = \infty$, for all $i \in N$, i.e., for each $i \in N$, at least one of a_i and b_i is finite.

The property [22, Theorem 2.1] makes the Qi–Sun–Zhou method applicable to P_0 NCP. The numerical results given in Table 2 are quite promising, and most problems are solved only after a small number of iterations. Although Example 5.6 has no solution, we found that the function value sequence $\|F(x^k)\|^2$ converges to zero and the algorithm always terminated somewhere around 10^{-14}, a point satisfying the termination criterion. Regarding Example 5.7 which has an unbounded solution set, we observed that the iterations, from various starting points, converge to the solutions on the segment $\{(x_1, x_2)|\ x_1 \geq 0, x_2 \geq 0, x_1 + x_2 - 1 = 0\}$. We also note that Example 5.5 is the only example where the feasible region is nonlinear.

6. Conclusions. In this paper, we introduced a regularized smoothing Newton method for the solution of box constrained variational inequality problems with P_0-functions. Our numerical results on some selected variational inequality problems show the success of the proposed method. We do not make numerical experiments with the neural network or the uniform smoothing function since we feel that the strong convergence property and the successful numerical results with CHKS smoothing function verified the promise of the method considered. We expect that the method can be used to solve practical large-scale problems efficiently. Recently, Zhou, Sun, and Qi [28] made extensive numerical experiments on a class of regularized smoothing methods, including the one proposed in this paper. Their results show the promise and robustness of the regularization technique. When we were finalizing the paper, we received a new report by Sun [26] that describes a regularization Newton method for the solution of nonlinear complementarity problems. In fact we consider a larger class of problems; in particular, our method can treat problems with free variables.

Acknowledgments. I am very grateful to Defeng Sun and Guanglu Zhou at the University of New South Wales for providing their code of testing NCP problems. Special thanks go to Defeng for his detailed comments on an earlier version of the paper. I thank Guanglu for pointing out an error in Theorem 4.5 of an earlier version of this paper. I also thank Jiming Peng for his constant help and useful discussion. Finally I would like to thank two referees and the associate editor for their valuable comments, which greatly improved the presentation of the paper.

REFERENCES

[1] B. Chen and X. Chen, *A global linear and local quadratic continuation method for variational inequalities with box constraints*, Comput. Optim. Appl., to appear.

[2] B. Chen and X. Chen, *A global and local superlinear continuation-smoothing method for P_0 and R_0 NCP or monotone NCP*, SIAM J. Optim., 9 (1999), pp. 624–645.

[3] B. Chen and P. T. Harker, *A non-interior-point continuation method for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1168–1190.

[4] B. Chen and N. Xiu, *A global linear and local quadratic noninterior continuation method for nonlinear complementarity problems based on Chen–Mangasarian smoothing functions*, SIAM J. Optim., 9 (1999), pp. 605–623.

[5] X. Chen, L. Qi, and D. Sun, *Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities*, Math. Comp., 67 (1998), pp. 519–540.

[6] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983; reprinted as Classics Appl. Math. 5, SIAM, Philadelphia, 1990.

[7] T. De Luca, F. Facchinei, and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Programming, 75 (1996), pp. 407–439.

[8] F. Facchinei, *Structural and stability properties of P_0 nonlinear complementarity problems*, Math. Oper. Res., 23 (1998), pp. 735–745.

[9] F. Facchinei and C. Kanzow, *Beyond monotonicity in regularization methods for nonlinear complementarity problems*, SIAM J. Control Optim., 37 (1999), pp. 1150–1161.

[10] M. C. Ferris and J. S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev., 39 (1997), pp. 669–713.

[11] M. Fukushima, *A relaxed projection method for variational inequalities*, Math. Programming, 35 (1986), pp. 58–70.

[12] S. A. Gabriel and J. J. Moré, *Smoothing of mixed complementarity problems*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J. S. Pang, eds., SIAM, Philadelphia, 1997, pp. 105–116.

[13] P. T. Harker and J. S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.

[14] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econom. and Math. Systems 187, Springer-Verlag, Berlin, 1981.

[15] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 851–868.

[16] C. Kanzow and H. Jiang, *A continuation method for (strongly) monotone variational inequalities*, Math. Programming, 81 (1998), pp. 103–125.

[17] C. Kanzow and H. D. Qi, *A QP-free constrained Newton-type method for variational inequality problems*, Math. Programming, 85 (1999), pp. 81–106.

[18] R. Mifflin, *Semismoothness and semiconvex functions in constrained optimization*, SIAM J. Control Optim., 15 (1977), pp. 959–972.

[19] J. J. Moré and W. C. Rheinboldt, *On P- and S-functions and related classes of n-dimensional nonlinear mappings*, Linear Algebra Appl., 6 (1973), pp. 45–68.

[20] J. S. Pang and S. A. Gabriel, *NE/SQP: A robust algorithm for the nonlinear complementarity problem*, Math. Programming, 60 (1993), pp. 295–337.

[21] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 691–714.

[22] L. Qi, D. Sun, and G. Zhou, *A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities*, Math. Programming, to appear.

[23] G. Ravindran and M. S. Gowda, *Regularization of P_0-functions in box variational inequality problems*, SIAM J. Optim., to appear.

[24] S. M. Robinson, *Normal maps induced by linear transformation*, Math. Oper. Res., 17 (1992), pp. 691–714.

[25] M. V. Solodov and P. Tseng, *Modified projection-type methods for monotone variational inequalities*, SIAM J. Control Optim., 34 (1996), pp. 1814–1830.

[26] D. Sun, *A regularization Newton method for solving nonlinear complementarity problems*, Appl. Math. Optim., 40 (1999), pp. 315–339.

[27] K. Taji, M. Fukushima, and T. Ibaraki, *A globally convergent Newton method for solving strongly monotone variational inequality problems*, Math. Programming, 58 (1993), pp. 369–383.

[28] G. Zhou, D. Sun, and L. Qi, *Numerical experiments for a class of squared smoothing Newton methods for box constrained variational inequality problems*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 421–441.

SIAM J. Optim.
Vol. 10, No. 2, pp. 331–344

THE LAGRANGE MULTIPLIER RULE IN SET-VALUED OPTIMIZATION[*]

ARNULF GÖTZ[†] AND JOHANNES JAHN[†]

Abstract. The known Lagrange multiplier rule is extended to set-valued constrained optimization problems using the contingent epiderivative as differentiability notion. A necessary optimality condition for weak minimizers is derived which is also a sufficient condition under generalized convexity assumptions.

Key words. optimality conditions, convex and set-valued analysis, vector optimization

AMS subject classifications. 49K27, 90C29

PII. S1052623496311697

1. Introduction.

In 1797, Lagrange [14] published his famous multiplier rule, which turned out to be an essential tool in constrained optimization. He developed this rule in connection with problems from mechanics. First he applied this principle to infinite dimensional problems in the calculus of variations and then he extended it to finite dimensional optimization problems. It is well known that the Karush–Kuhn–Tucker conditions in finite dimensional optimization, the maximum principle in optimal control theory, and the extended Kolmogorov criterion in approximation theory can be deduced from a general multiplier rule.

In this paper we consider general optimization problems with a set-valued objective map and a set-valued constraint, and we show that the Lagrange rule remains valid in such a general setting as well.

Throughout this paper we use the following standard assumption:

$$\text{(1.1)}\quad \left\{\begin{array}{l} \text{Let } (X, \|\cdot\|_X) \text{ be a real normed space;} \\ \text{let } (Y, \|\cdot\|_Y) \text{ and } (Z, \|\cdot\|_Z) \text{ be real normed spaces} \\ \text{partially ordered by convex pointed cones } C_Y \subset Y \\ \text{and } C_Z \subset Z\text{, respectively;} \\ \text{let } \hat{S} \text{ be a nonempty subset of } X; \\ \text{and let } F : \hat{S} \to 2^Y \text{ and } G : \hat{S} \to 2^Z \text{ be set-valued maps.} \end{array}\right.$$

Notice that a cone C_Y (or C_Z) is called *pointed* if $C_Y \cap (-C_Y) = \{0_Y\}$ (or $C_Z \cap (-C_Z) = \{0_Z\}$). It is well known that the convex pointed cones C_Y and C_Z induce partial orderings $\leq_{C_Y}$ and $\leq_{C_Z}$ (i.e., reflexive transitive and antisymmetric binary relations being compatible with addition and scalar multiplication) in the spaces Y and Z (for instance, compare [8]).

Under the assumption (1.1) we consider the constrained set-valued optimization problem

$$\text{(1.2)}\quad \left\{\begin{array}{l} \min F(x) \\ \text{subject to the constraints} \\ G(x) \cap (-C_Z) \neq \emptyset, \\ x \in \hat{S}. \end{array}\right.$$

[*] Received by the editors November 5, 1996; accepted for publication (in revised form) April 14, 1999; published electronically December 22, 1999.
http://www.siam.org/journals/siopt/10-2/31169.html

[†] Angewandte Mathematik II, Universität Erlangen-Nürnberg, Martensstrasse 3, D-91058 Erlangen, Germany (jahn@am.uni-erlangen.de).

For simplicity let $S := \{x \in \hat{S} \mid G(x) \cap (-C_Z) \neq \emptyset\}$ denote the feasible set of this problem, which is assumed to be nonempty. If G is single-valued, the constraint in (1.2) reduces to $G(x) \in -C_Z$ or $G(x) \leq_{C_Z} 0_Z$ generalizing equality and inequality constraints. If, in addition, F is single-valued, then the problem (1.2) is a general vector optimization problem.

In the standard optimization theory it is always assumed that the objective function and the function describing the constraints are exactly given. Sometimes these functions are not exactly given or their values may vary in a certain range. Optimization problems with uncertain functions can be found in stochastic optimization and fuzzy set optimization. In set-valued optimization it is assumed that these functions are set-valued and that the ranges (in which the function values can vary) are explicitly known. Hence, there are close relationships between these three types of optimization problems.

As in vector optimization there are different optimality concepts in use. We recall two standard optimality notions (where we use the names given in [15], [16], and [11]).

Definition 1.1. *Let the problem* (1.2) *be given. Let* $F(S) := \bigcup_{x \in S} F(x)$ *denote the image set of* F.

(a) *A pair* $(\bar{x}, \bar{y})$ *with* $\bar{x} \in S$ *and* $\bar{y} \in F(\bar{x})$ *is called a* minimizer *of the problem* (1.2) *(or a* minimizer *of* F *on* S*), if* $\bar{y}$ *is a minimal element (or an Edgeworth–Pareto point) of the set* $F(S)$, *i.e.,*

$$y \in F(S), \qquad y \leq_{C_Y} \bar{y} \quad \Longrightarrow \quad y = \bar{y},$$

or in other words

$$(\{\bar{y}\} - C_Y) \cap F(S) = \{\bar{y}\}.$$

(b) *Let* C_Y *have a nonempty interior* $\operatorname{int}(C_Y)$. *A pair* $(\bar{x}, \bar{y})$ *with* $\bar{x} \in S$ *and* $\bar{y} \in F(\bar{x})$ *is called a* weak minimizer *of the problem* (1.2) *(or a* weak minimizer *of* F *on* S*), if* $\bar{y}$ *is a weakly minimal element of the set* $F(S)$, *i.e.,*

$$(\{\bar{y}\} - \operatorname{int}(C_Y)) \cap F(S) = \emptyset.$$

It is known from vector optimization that the minimality notion is the suitable optimality concept in applications, but in important cases it is not possible to give optimality conditions for it that are necessary and sufficient. On the other hand, the weak minimality notion is not the right tool for applications but in many cases it can be completely characterized (for instance, see [8]). Therefore, we restrict ourselves mainly to the concept of a weak minimizer.

Set-valued optimization problems have been investigated by many authors; for instance, there are papers on optimality conditions (e.g., [2], [19], [3], [4], [6], [16], [18], [11]), duality theory (e.g., [20], [5], [17]), and related topics (e.g., [21], [13]).

For the formulation of a multiplier rule in the nonlinear case one needs an appropriate differentiability concept. As early as 1981 Aubin [1] introduced the (nowadays) so-called contingent derivative which is of great importance in set-valued analysis. But it turned out that this differentiability notion is not the right tool for the formulation of optimality conditions in set-valued optimization (see [11]). Therefore, a modification called contingent epiderivative has been presented (in [11]) which modifies a notion introduced by Aubin [1] as upper contingent derivative. It has been shown in [11] that one gets optimality conditions for unconstrained problems that are necessary and sufficient under suitable assumptions. Therefore, contingent epiderivatives are used for the investigations in this paper.

Definition 1.2. *Let $(E_1, \|\cdot\|_{E_1})$ and $(E_2, \|\cdot\|_{E_2})$ be real normed spaces, let E_2 be partially ordered by a convex cone $K \subset E_2$, let M be a nonempty subset of E_1, and let $H : M \to 2^{E_2}$ be a set-valued map.*

(a) *The set*

$$\mathrm{epi}(H) := \{(x, y) \in E_1 \times E_2 \mid x \in M,\ y \in H(x) + K\}$$

is called the epigraph *of H.*

(b) *Let a pair $(\bar{x}, \bar{y}) \in E_1 \times E_2$ with $\bar{x} \in M$ and $\bar{y} \in H(\bar{x})$ be given. A single-valued map $DH(\bar{x}, \bar{y}) : E_1 \to E_2$ whose epigraph equals the contingent cone to the epigraph of H at $(\bar{x}, \bar{y})$, i.e.,*

$$\mathrm{epi}(DH(\bar{x}, \bar{y})) = T(\mathrm{epi}(H), (\bar{x}, \bar{y})),$$

is called the contingent epiderivative *of H at $(\bar{x}, \bar{y})$.*

Recall that the contingent cone $T(\mathrm{epi}(H), (\bar{x}, \bar{y}))$ consists of all tangent vectors $h := \lim_{n\to\infty} \lambda_n((x_n, y_n) - (\bar{x}, \bar{y}))$ with $(\bar{x}, \bar{y}) = \lim_{n\to\infty}(x_n, y_n)$ ($(x_n, y_n) \in \mathrm{epi}(H)$ for all $n \in \mathbb{N}$) and $\lambda_n > 0$ ($n \in \mathbb{N}$). Properties of the contingent epiderivative can be found in [11].

Since convexity plays an important role in the following investigations, recall the definition of cone-convex maps.

Definition 1.3. *Let E_1 and E_2 be real linear spaces, let E_2 be partially ordered by a convex cone $K \subset E_2$, and let M be a nonempty convex subset of E_1. A set-valued map $H : M \to 2^{E_2}$ is called K-*convex *if for all $x_1, x_2 \in M$ and $\lambda \in [0, 1]$,*

$$\lambda H(x_1) + (1 - \lambda)H(x_2) \subset H(\lambda x_1 + (1 - \lambda)x_2) + K.$$

Moreover, the *dual cone* of C_Y in assumption (1.1) is defined as

$$C_{Y^*} := \{y^* \in Y^* \mid y^*(y) \geq 0 \text{ for all } y \in C_Y\}.$$

(Similarly, the dual cone of C_Z is denoted C_{Z^*}.) The *cone generated* by a nonempty subset M of a real linear space is denoted

$$\mathrm{cone}(M) := \{\lambda x \mid \lambda \geq 0 \text{ and } x \in M\}.$$

On the basis of the concept of contingent epiderivatives we prove in section 2 a multiplier rule as a necessary optimality condition of problem (1.2) and discuss a regularity assumption. In section 3 assumptions are presented which guarantee that this multiplier rule is a sufficient optimality condition as well. The results are based on the presentation in [7].

2. A necessary optimality condition.

We begin our investigations with a generalized Lagrange multiplier rule as a necessary optimality condition for set-valued optimization problems.

Theorem 2.1. *Let the cone C_Y have a nonempty interior $\mathrm{int}(C_Y)$, let the set $\hat{S}$ be convex, and let the maps F and G be C_Y-convex and C_Z-convex, respectively. Assume that $(\bar{x}, \bar{y}) \in X \times Y$ with $\bar{x} \in S$ and $\bar{y} \in F(\bar{x})$ is a weak minimizer of the problem (1.2). Let the contingent epiderivative of (F, G) at $(\bar{x}, (\bar{y}, \bar{z}))$ for an arbitrary $\bar{z} \in G(\bar{x}) \cap (-C_Z)$ exist. Then there are continuous linear functionals $t \in C_{Y^*}$ and $u \in C_{Z^*}$ with $(t, u) \neq (0_{Y^*}, 0_{Z^*})$ so that*

$$t(y) + u(z) \geq 0 \textit{ for all } (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) \textit{ with } x \in \hat{S}$$

and

$$u(\bar{z}) = 0.$$

If, in addition to the above assumptions, the regularity assumption

$$\big\{z \mid (y,z) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))\big(cone(S - \{\bar{x}\})\big)\big\} + cone(C_Z + \{\bar{z}\}) = Z \tag{2.1}$$

is satisfied, then $t \neq 0_{Y^*}$*.*

Proof. In the product space $Y \times Z$ we define for an arbitrary $\bar{z} \in G(\bar{x}) \cap (-C_Z)$ the following set:

$$M := \left[\bigcup_{x \in \hat{S}} D(F,G)(\bar{x},(\bar{y},\bar{z}))(x - \bar{x})\right] + \big(C_Y \times (C_Z + \{\bar{z}\})\big).$$

The proof of this theorem consists of several steps. First, we prove two important properties of this set M and then we apply a separation theorem in order to obtain the multiplier rule. Finally, we show $t \neq 0_{Y^*}$ under the regularity assumption.

(a) We show that the nonempty set M is convex. We prove the convexity for the translated set $M' := M - \{(0_Y, \bar{z})\}$ and immediately get the desired result. For this proof we fix two arbitrary pairs (y_1, z_1), $(y_2, z_2) \in M'$. Then there are elements $x_1, x_2 \in \hat{S}$ with

$$(y_i, z_i) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))(x_i - \bar{x}) + \big(C_Y \times C_Z\big) \quad \text{for} \quad i = 1, 2,$$

resulting in

$$(x_i - \bar{x}, (y_i, z_i)) \in T\big(\mathrm{epi}(F,G), (\bar{x},(\bar{y},\bar{z}))\big) \quad \text{for} \quad i = 1, 2.$$

This contingent cone is convex because the map (F,G) is cone-convex and, therefore, the epigraph $\mathrm{epi}(F,G)$ is a convex set (see [11, Lem. 1]). Then we obtain for all $\lambda \in [0,1]$

$$\lambda(x_1 - \bar{x}, (y_1, z_1)) + (1-\lambda)(x_2 - \bar{x}, (y_2, z_2)) \in T\big(\mathrm{epi}(F,G), (\bar{x},(\bar{y},\bar{z}))\big),$$

implying

$$(\lambda y_1 + (1-\lambda) y_2, \lambda z_1 + (1-\lambda) z_2) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))(\lambda x_1 + (1-\lambda) x_2 - \bar{x}) + (C_Y \times C_Z).$$

Consequently, the set M is convex.

(b) In the next step of the proof we show the equality

$$M \cap \Big[\big(-\mathrm{int}(C_Y)\big) \times \big(-\mathrm{int}(C_Z)\big)\Big] = \emptyset. \tag{2.2}$$

Assume that this equality does not hold. Then there are elements $x \in \hat{S}$ and $(y,z) \in Y \times Z$ with

$$\begin{aligned}(y, z + \bar{z}) \in &\Big[D(F,G)(\bar{x},(\bar{y},\bar{z}))(x - \bar{x}) + \big(C_Y \times (C_Z + \{\bar{z}\})\big)\Big] \\ &\cap \Big[\big(-\mathrm{int}(C_Y)\big) \times \big(-\mathrm{int}(C_Z)\big)\Big],\end{aligned} \tag{2.3}$$

implying

$$(x - \bar{x}, (y, z)) \in T\big(\mathrm{epi}(F,G), (\bar{x},(\bar{y},\bar{z}))\big).$$

This means that there are sequences $(x_n,(y_n,z_n))_{n\in\mathbb{N}}$ of elements in $\operatorname{epi}(F,G)$ and a sequence $(\lambda_n)_{n\in\mathbb{N}}$ of positive real numbers with

$$(\bar{x},(\bar{y},\bar{z}))=\lim_{n\to\infty}(x_n,(y_n,z_n))$$

and

$$(x-\bar{x},(y,z))=\lim_{n\to\infty}\lambda_n\big(x_n-\bar{x},(y_n-\bar{y},z_n-\bar{z})\big). \tag{2.4}$$

Since $y\in-\operatorname{int}(C_Y)$ by (2.3), we conclude $\lambda_n(y_n-\bar{y})\in-\operatorname{int}(C_Y)$ for sufficiently large $n\in\mathbb{N}$ resulting in

$$y_n\in\{\bar{y}\}-\operatorname{int}(C_Y)\ \text{for sufficiently large}\ n\in\mathbb{N}. \tag{2.5}$$

Because of $(x_n,(y_n,z_n))\in\operatorname{epi}(F,G)$ for all $n\in\mathbb{N}$ there are elements $\hat{y}_n\in F(x_n)$ with

$$y_n\in\{\hat{y}_n\}+C_Y\ \text{for all}\ n\in\mathbb{N}.$$

Together with (2.5) we obtain

$$\hat{y}_n\in\{\bar{y}\}-\operatorname{int}(C_Y)-C_Y=\{\bar{y}\}-\operatorname{int}(C_Y)\ \text{for sufficiently large}\ n\in\mathbb{N}$$

or

$$\big(\{\bar{y}\}-\operatorname{int}(C_Y)\big)\cap F(x_n)\neq\emptyset\ \text{for sufficiently large}\ n\in\mathbb{N}. \tag{2.6}$$

Moreover, from (2.3) we conclude $z+\bar{z}\in-\operatorname{int}(C_Z)$, and with (2.4) we obtain

$$\lambda_n(z_n-\bar{z})+\bar{z}\in-\operatorname{int}(C_Z)\ \text{for sufficiently large}\ n\in\mathbb{N}$$

or

$$\lambda_n\Big(z_n-\Big(1-\frac{1}{\lambda_n}\Big)\bar{z}\Big)\in-\operatorname{int}(C_Z)\ \text{for sufficiently large}\ n\in\mathbb{N},$$

implying

$$z_n-\Big(1-\frac{1}{\lambda_n}\Big)\bar{z}\in-\operatorname{int}(C_Z)\ \text{for sufficiently large}\ n\in\mathbb{N}. \tag{2.7}$$

Since $y\neq 0_Y$ (by (2.3)), we conclude with (2.4) that

$$\lambda_n>1\ \text{for sufficiently large}\ n\in\mathbb{N}.$$

By assumption we have $\bar{z}\in-C_Z$ and, therefore, we get from (2.7)

$$z_n\in-C_Z-\operatorname{int}(C_Z)=-\operatorname{int}(C_Z)\ \text{for sufficiently large}\ n\in\mathbb{N}. \tag{2.8}$$

Because of $(x_n,(y_n,z_n))\in\operatorname{epi}(F,G)$ for all $n\in\mathbb{N}$ there are elements $\hat{z}_n\in G(x_n)$ with

$$z_n\in\{\hat{z}_n\}+C_Z\ \text{for all}\ n\in\mathbb{N}.$$

Together with (2.8) we then get

$$\hat{z}_n\in\{z_n\}-C_Z\subset-\operatorname{int}(C_Z)\ \text{for sufficiently large}\ n\in\mathbb{N}$$

and

$$\hat{z}_n \in G(x_n) \cap (-C_Z) \text{ for sufficiently large } n \in \mathbb{N}. \tag{2.9}$$

Hence, for a sufficiently large $n \in \mathbb{N}$ we have $\hat{x}_n \in \hat{S}$, $\big(\{\bar{y}\} - \operatorname{int}(C_Y)\big) \cap F(x_n) \neq \emptyset$ (by (2.6)), and $G(x_n) \cap (-C_Z) \neq \emptyset$ (by (2.9)) and, therefore, $(\bar{x}, \bar{y})$ is not a weak minimizer of the problem (1.2), which is a contradiction to the assumption of the theorem.

(c) In this step we now prove the first part of the theorem. By part (a) the set M is convex and by (b) the equality (2.2) holds. By Eidelheit's separation theorem (e.g., see [9]) there are continuous linear functionals $t \in Y^*$ and $u \in Z^*$ with $(t, u) \neq (0_{Y^*}, 0_{Z^*})$ and a real number $\gamma > 0$ so that

$$t(c_Y) + u(c_Z) < \gamma \leq t(y) + u(z) \qquad \text{for all } c_Y \in -\operatorname{int}(C_Y), \qquad c_Z \in -\operatorname{int}(C_Z), \qquad (y, z) \in M. \tag{2.10}$$

Since $(0, \bar{z}) \in M$, we obtain from (2.10) for $c_Y = 0_Y$

$$u(c_Z) < u(\bar{z}) \text{ for all } c_Z \in -\operatorname{int}(C_Z). \tag{2.11}$$

If we assume that $u(c_Z) > 0$ for a $c_Z \in -\operatorname{int}(C_Z)$, we get a contradiction to (2.11) because C_Z is a cone. Therefore, we obtain

$$u(c_Z) \leq 0 \text{ for all } c_Z \in -\operatorname{int}(C_Z),$$

resulting in $u \in C_{Z^*}$ because $C_Z \subset \operatorname{cl}(\operatorname{int}(C_Z))$. For $(0, \bar{z}) \in M$ and $c_Z = 0_Z$ we get from (2.10)

$$t(c_Y) < u(\bar{z}) \leq 0 \text{ for all } c_Y \in -\operatorname{int}(C_Y) \tag{2.12}$$

(notice that $\bar{z} \in -C_Z$ and $u \in C_{Z^*}$). This inequality implies $t \in C_{Y^*}$. From (2.11) and (2.12) we immediately obtain $u(\bar{z}) = 0$. In order to prove the inequality of the multiplier rule we conclude from (2.10) with $c_Y = 0_Y$ and $c_Z = 0_Z$

$$t(y) + u(z) \geq 0 \text{ for all } (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) \text{ with } x \in \hat{S}.$$

Hence, the first part of the theorem is shown.

(d) Finally, we prove $t \neq 0_{Y^*}$ under the regularity assumption (2.1). For an arbitrary $\hat{z} \in Z$ there are elements $x \in \hat{S}$, $c_Z \in C_Z$ and nonnegative real numbers α and β with

$$\hat{z} = z + \beta(c_Z + \bar{z}) \quad \text{for} \quad (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))\big(\alpha(x - \bar{x})\big).$$

Since $D(F, G)(\bar{x}, (\bar{y}, \bar{z}))$ is positively homogeneous (see [11, Thm. 4], where one does not need the cone-convexity of G), we can write

$$(y, z) = \alpha D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) =: \alpha(\tilde{y}, \tilde{z}).$$

Assume that $t = 0_{Y^*}$. Then we conclude from the multiplier rule

$$\begin{aligned} u(\hat{z}) &= u(z) + \beta u(c_Z + \bar{z}) \\ &= \alpha \underbrace{u(\tilde{z})}_{\geq 0} + \beta \underbrace{u(c_Z)}_{\geq 0} + \beta \underbrace{u(\bar{z})}_{= 0} \\ &\geq 0. \end{aligned}$$

Because $\hat{z}$ is arbitrarily chosen we have

$$u(\hat{z}) \geq 0 \text{ for all } z \in Z,$$

implying $u = 0_{Z^*}$. But this is a contradiction to $(t, u) \neq (0_{Y^*}, 0_{Z^*})$. □

Theorem 2.1 generalizes the Lagrange multiplier rule as a necessary optimality condition to set-valued optimization. Since a minimizer of the problem (1.2) is also a weak minimizer (compare [8, p. 106]), this multiplier rule is a necessary optimality condition for a minimizer as well.

The basic idea for the first part of the proof of Theorem 2.1 has been given by Corley [6] using a different differentiability concept. This idea of proof has also been used by Luc and Malivert [18] (e.g., see Theorem 5.6) for the contingent derivative. They have already proved an optimality condition under a regularity assumption (a generalized Slater condition). The regularity condition in Theorem 2.1 extends the Kurcyusz–Robinson–Zowe regularity assumption (e.g., see [9]) to set-valued optimization problems. It is weaker than a generalized Slater condition (compare Lemma 2.3). Although the regularity condition in Theorem 2.1 also includes the objective map F, one uses only the second component of the contingent epiderivative of (F, G).

In Theorem 2.1 the existence of the contingent epiderivative of (F, G) is assumed. It is still an open problem whether there are close relations between this derivative and the contingent epiderivatives of F and G in the general case.

It is important to note that the maps F and G are assumed to be cone-convex in Theorem 2.1, whereas convexity of the objective function and the constraint function is not needed in the single-valued scalar case (e.g., see [9, Thm. 5.3]). In fact, the cone-convexity is only needed in part (a) of the proof in order to obtain the convexity of the contingent cone. If we would modify the notion of the contingent epiderivative in such a way that we replace the contingent cone by Clarke's tangent cone, which is always convex, we could drop the cone-convexity assumption in Theorem 2.1.

With the following example we illustrate the usefulness of the necessary condition in Theorem 2.1.

Example 2.2. Let $(X, \|\cdot\|_X)$ be a real normed space, and let $f, g, h : X \to \mathbb{R}$ be given functionals with

$$f(x) \leq g(x) \text{ for all } x \in X.$$

Then we consider the set-valued map $F : X \to 2^{\mathbb{R}}$ with

$$F(x) := \{y \in \mathbb{R} \mid f(x) \leq y \leq g(x)\}$$

and the set-valued map $G : X \to 2^{\mathbb{R}}$ (which is actually single-valued) with

$$G(x) := \{h(x)\}.$$

Under these assumptions we investigate the optimization problem

$$(2.13) \qquad \left\{ \begin{array}{l} \min\ F(x) \\ \text{subject to the constraints} \\ G(x) \cap (-\mathbb{R}_+) \neq \emptyset, \\ x \in X. \end{array} \right.$$

This is a special problem of the general type (1.2). Notice that the constraint is equivalent to the inequality $h(x) \leq 0$. If $f = g$ this problem reduces to a standard optimization problem. But if the data of the objective function of a standard problem are not exactly known, it makes sense to replace the objective by a set-valued objective representing fuzzy outcomes. In this example the values of the objective may vary between the values of two known functions.

Next, we assume that $(\bar{x}, f(\bar{x}))$ is a weak minimizer of problem (2.13) and that f and h are continuous at $\bar{x}$ and convex. Since

$$\operatorname{epi}(F,G) = \{(x,(y,z)) \in X \times \mathbb{R}^2 \mid x \in X,\ y \geq f(x),\ z \geq h(x)\},$$

we conclude

$$T(\operatorname{epi}(F,G),(\bar{x},(f(\bar{x}),h(\bar{x})))) = \operatorname{epi}(f,g)'(\bar{x}),$$

i.e., that the contingent derivative of (F,G) at $(\bar{x},(f(\bar{x}),h(\bar{x})))$ exists and equals the directional derivative $(f,h)'(\bar{x}) = (f',h')(\bar{x})$ of (f,h) at $\bar{x}$ (see [10, Cor. 1] for the case of one functional). Consequently, by the previous theorem there are nonnegative numbers t and u with $(t,u) \neq (0,0)$ so that

$$tf'(\bar{x})(x-\bar{x}) + uh'(\bar{x})(x-\bar{x}) \geq 0 \text{ for all } x \in X$$

and

$$uh(\bar{x}) = 0.$$

If $f'(\bar{x})$ and $h'(\bar{x})$ are linear (e.g., in the case of Fréchet differentiability), we even conclude that

$$tf'(\bar{x}) + uh'(\bar{x}) = 0_{X^*}$$

and

$$uh(\bar{x}) = 0.$$

Hence, for the special set-valued optimization problem (2.13) we obtain a classical multiplier rule. Finally, we discuss the regularity condition of Theorem 2.1 for this problem. Assume that for every $z < 0$ there is an $x \in X$ with $z = h'(\bar{x})(x)$. Then $h'(\bar{x})(X) \supset -\mathbb{R}_+$, and because of $h(\bar{x}) \leq 0$ we obtain

$$\begin{aligned} &h'(\bar{x})(\operatorname{cone}(X - \{\bar{x}\})) + \operatorname{cone}(\mathbb{R}_+ + \{h(\bar{x})\}) \\ &= \underbrace{h'(\bar{x})(X)}_{\supset -\mathbb{R}_+} + \begin{cases} \mathbb{R} & \text{if } h(\bar{x}) < 0, \\ \mathbb{R}_+ & \text{if } h(\bar{x}) = 0, \end{cases} \\ &= \mathbb{R}. \end{aligned}$$

Hence, the general regularity condition (2.1) is satisfied in this case.

The next lemma shows that a generalization of the well-known Slater condition implies the extended Kurcyusz–Robinson–Zowe constraint qualification.

LEMMA 2.3. *Let* $\operatorname{int}(\hat{S}) \neq \emptyset$*; let* $\bar{x} \in S$, $\bar{y} \in F(\bar{x})$, *and* $\bar{z} \in G(\bar{x}) \cap (-C_Z)$ *be arbitrarily given; and let the contingent epiderivative of* (F,G) *at* $(\bar{x},(\bar{y},\bar{z}))$ *exist. If there is an* $\hat{x} \in \operatorname{int}(\hat{S})$ *with*

$$\bar{z} + z \in -\operatorname{int}(C_Z) \text{ for } (y,z) = D(F,G)(\bar{x},(\bar{y},\bar{z}))(\hat{x}-\bar{x}),$$

then the regularity assumption (2.1) *is fulfilled.*

Proof. Take an arbitrary $\hat{z} \in Z$. Since $D(F,G)(\bar{x},(\bar{y},\bar{z}))$ is positive homogeneous (this result is proved in [11, Thm. 4], where one does not need the cone-convexity of G), we obtain for a sufficiently large $\lambda > 0$ and $\lambda(y,z) = D(F,G)(\bar{x},(\bar{y},\bar{z}))(\lambda(\hat{x}-\bar{x})) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))\big(\text{cone}(S-\{\bar{x}\})\big)$

$$\begin{aligned}\hat{z} &= \lambda z + \lambda\Big[\underbrace{-\bar{z} - z + \frac{1}{\lambda}\hat{z}}_{\in C_Z} + \bar{z}\Big]\\ &\in \big\{\tilde{z} \mid (y,\tilde{z}) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))\big(\text{cone}(S-\{\bar{x}\})\big)\big\} + \text{cone}(C_Z + \{\bar{z}\}).\end{aligned}$$

Hence, we conclude

$$Z \subset \big\{\tilde{z} \mid (y,\tilde{z}) \in D(F,G)(\bar{x},(\bar{y},\bar{z}))\big(\text{cone}(S-\{\bar{x}\})\big)\big\} + \text{cone}(C_Z + \{\bar{z}\}).$$

Because the converse inclusion is trivial, the regularity assumption (2.1) is thus fulfilled. □

The following example shows that the regularity condition (2.1) can be satisfied although the regularity assumption in Lemma 2.3 is not fulfilled.

Example 2.4. We consider $X = Z = L_2[0,1]$ with the natural ordering cone

$$C_Z := \{x \in L_2[0,1] \mid x(t) \geq 0 \text{ almost everywhere on } [0,1]\}$$

(notice that $\text{int}(C_Z) = \emptyset$). Take an arbitrary $a \in L_2[0,1]$ and define the set-valued map $G : X \to 2^Z$ with

$$G(x) = \{-x + a\} + C_Z \text{ for all } x \in X.$$

Then we investigate the constraint of problem (1.2)

$$G(x) \cap (-C_Z) \neq \emptyset, \qquad x \in X,$$

being equivalent to

$$-x + a \in -C_Z, \qquad x \in X.$$

For instance, choose the objective map $F : X \to 2^{\mathbb{R}}$ with

$$F(x) = \{\langle x,x\rangle\} \text{ for all } x \in X$$

($\langle\cdot,\cdot\rangle$ denotes the scalar product in X).

Since $\text{int}(C_Z) = \emptyset$, it is obvious that Lemma 2.3 is not applicable in this case. Therefore, we investigate the regularity assumption (2.1) in Theorem 2.1. For an arbitrary $\bar{x} \in X$ with $\bar{z} := -\bar{x} + a \in -C_Z$, we obtain with

$$\text{epi}(F,G) = \{(x,(y,z)) \in X \times \mathbb{R} \times Z \mid x \in X,\ y \geq \langle x,x\rangle,\ -x + a \leq_{C_Z} z\}$$

the equality

$$T(\text{epi}(F,G),(\bar{x},(\langle\bar{x},\bar{x}\rangle,\bar{z}))) = \text{epi}(2\langle x,\cdot\rangle, -\text{id}),$$

implying

$$D(F,G)(\bar{x},(\bar{y},\bar{z})) = (2\langle x,\cdot\rangle, -\text{id})$$

(id denotes the identity). Then we get

$$-\text{id}(\text{cone}(X - \{\bar{x}\})) + \text{cone}(C_Z + \{\bar{z}\}) = X + \text{cone}(C_Z + \{\bar{z}\}) = X = Z,$$

i.e., the regularity condition (2.1) in Theorem 2.1 is fulfilled.

3. A sufficient optimality condition. In this section we answer the question under which assumptions the multiplier rule in Theorem 2.1 is also a sufficient optimality condition. It is known from standard optimization theory that convexity or generalized concepts like quasi convexity play the essential role. Therefore, we begin with an extension of the quasi convexity concept to set-valued maps.

DEFINITION 3.1. *Let $(X, \|\cdot\|_X)$ and $(Y, \|\cdot\|_Y)$ be real normed spaces, let $\hat{S}$ be a nonempty subset of X, let $\tilde{C}$ be a nonempty subset of Y, and let $F : \hat{S} \to 2^Y$ be a set-valued map whose contingent epiderivative exists at $(\bar{x}, \bar{y})$ with $\bar{x} \in \hat{S}$ and $\bar{y} \in F(\bar{x})$. The map F is called* $\tilde{C}$-quasi-convex *at $(\bar{x}, \bar{y})$ if for all $x \in \hat{S}$*

$$\big(F(x) - \{\bar{y}\}\big) \cap \tilde{C} \neq \emptyset \quad \Longrightarrow \quad \big(\{DF(\bar{x}, \bar{y})(x - \bar{x})\} + C_Y\big) \cap \tilde{C} \neq \emptyset.$$

This notion extends a concept introduced in [12] (see also [9]) for problems in single-valued optimization. The following lemma shows that cone-convexity implies quasi convexity in this set-valued setting.

LEMMA 3.2. *Let $\hat{S}$ be a nonempty convex subset of a real normed space $(X, \|\cdot\|_X)$, let $\tilde{C}$ be a nonempty subset of the real normed space $(Y, \|\cdot\|_Y)$ partially ordered by a convex pointed cone $C_Y \subset Y$, and let a set-valued map $F : \hat{S} \to 2^Y$ be given whose contingent epiderivative exists at $(\bar{x}, \bar{y})$ with $\bar{x} \in \hat{S}$ and $\bar{y} \in F(\bar{x})$. If F is C_Y-convex, then it is also $\tilde{C}$-quasi-convex at $(\bar{x}, \bar{y})$.*

Proof. Choose an arbitrary $x \in \hat{S}$ with

$$\big(F(x) - \{\bar{y}\}\big) \cap \tilde{C} \neq \emptyset.$$

Since F is C_Y-convex, we conclude with [11, Lem. 3]

$$F(x) - \{\bar{y}\} \subset \{DF(\bar{x}, \bar{y})(x - \bar{x})\} + C_Y.$$

Consequently, we obtain

$$\big(\{DF(\bar{x}, \bar{y})(x - \bar{x})\} + C_Y\big) \cap \tilde{C} \neq \emptyset. \qquad \square$$

It is known from standard optimization theory (see [12] and [9]) that the quasi convexity of a certain composite map completely characterizes the sufficiency of a multiplier rule. This idea is extended in the next theorem.

THEOREM 3.3. *Let the cone C_Y have a nonempty interior* $\operatorname{int}(C_Y)$, *and let the contingent derivative of (F, G) exist at $(\bar{x}, (\bar{y}, \bar{z}))$ with $\bar{x} \in S$, $\bar{y} \in F(\bar{x})$, and $\bar{z} \in G(\bar{x})$. Moreover, assume that there are continuous linear functionals $t \in C_{Y^*} \backslash \{0_{Y^*}\}$ and $u \in C_{Z^*}$ with*

$$t(y) + u(z) \geq 0 \text{ for all } (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) \text{ with } x \in \hat{S} \tag{3.1}$$

and

$$u(\bar{z}) = 0. \tag{3.2}$$

Then $(\bar{x}, \bar{y})$ is a weak minimizer of F on

$$\tilde{S} := \big\{x \in \hat{S} \mid G(x) \cap \big(-C_Z + cone(\bar{z}) - cone(\bar{z})\big) \neq \emptyset\big\}$$

if and only if the map $(F, G) : \hat{S} \to 2^Y \times 2^Z$ is $\tilde{C}$-quasi-convex at $(\bar{x}, (\bar{y}, \bar{z}))$ with

$$\tilde{C} := \big(-\operatorname{int}(C_Y)\big) \times \big(-C_Z + cone(\bar{z}) - cone(\bar{z})\big).$$

Proof. First we show under the given assumptions

$$\Big(\big(\{y\}+C_Y\big)\times\big(\{z\}+C_Z\big)\Big)\cap\tilde{C}=\emptyset \text{ for all } (y,z)=D(F,G)(\bar{x},(\bar{y},\bar{z}))(x-\bar{x}) \text{ with } x\in\hat{S}. \tag{3.3}$$

For the proof of this assertion assume that there is an $x \in \hat{S}$ with

$$\Big(\big(\{y\}+C_Y\big)\times\big(\{z\}+C_Z\big)\Big)\cap\tilde{C}\neq\emptyset \quad \text{for} \quad (y,z)=D(F,G)(\bar{x},(\bar{y},\bar{z}))(x-\bar{x}),$$

i.e.,

$$\big(\{y\}+C_Y\big)\cap\big(-\operatorname{int}(C_Y)\big)\neq\emptyset \tag{3.4}$$

and

$$\big(\{z\}+C_Z\big)\cap\big(-C_Z+\operatorname{cone}(\bar{z})-\operatorname{cone}(\bar{z})\big)\neq\emptyset. \tag{3.5}$$

The condition (3.4) implies

$$y\in -C_Y-\operatorname{int}(C_Y)=-\operatorname{int}(C_Y),$$

and with the condition (3.5) we obtain

$$z\in -C_Z-C_Z+\operatorname{cone}(\bar{z})-\operatorname{cone}(\bar{z})=-C_Z+\operatorname{cone}(\bar{z})-\operatorname{cone}(\bar{z}).$$

Consequently, we get with (3.2)

$$t\big(y\big)+u\big(z\big)<0,$$

which contradicts the inequality (3.1). Hence, the set equation (3.3) is satisfied.

Now we come to the actual proof of this theorem. First, we assume that the map (F,G) is $\tilde{C}$-quasi-convex at $(\bar{x},(\bar{y},\bar{z}))$. Then we conclude with the equality (3.3)

$$\Big(\big(F(x)-\{\bar{y}\}\big)\times\big(G(x)-\{\bar{z}\}\big)\Big)\cap\tilde{C}=\emptyset \text{ for all } x\in\hat{S}.$$

Hence, there is no $x\in\hat{S}$ with

$$\big(F(x)-\{\bar{y}\}\big)\cap\big(-\operatorname{int}(C_Y)\big)\neq\emptyset$$

and

$$\big(G(x)-\{\bar{z}\}\big)\cap\big(-C_Z+\operatorname{cone}(\bar{z})-\operatorname{cone}(\bar{z})\big)\neq\emptyset.$$

Consequently, there is no $x\in\hat{S}$ with

$$\big(F(x)-\{\bar{y}\}\big)\cap\big(-\operatorname{int}(C_Y)\big)\neq\emptyset$$

and

$$G(x)\cap\big(-C_Z+\operatorname{cone}(\bar{z})-\operatorname{cone}(\bar{z})\big)\neq\emptyset.$$

This means that $(\bar{x},\bar{y})$ is a weak minimizer of F on $\tilde{S}$.

Finally, we assume that $(\bar{x}, \bar{y})$ is a weak minimizer of F on $\tilde{S}$. Then there is no $x \in \hat{S}$ with

$$\big(F(x) - \{\bar{y}\}\big) \cap \big(- \operatorname{int}(C_Y)\big) \neq \emptyset$$

and

$$G(x) \cap \big(- C_Z + \operatorname{cone}(\bar{z}) - \operatorname{cone}(\bar{z})\big) \neq \emptyset,$$

implying

$$\big(G(x) - \{\bar{z}\}\big) \cap \big(- C_Z + \operatorname{cone}(\bar{z}) - \operatorname{cone}(\bar{z})\big) \neq \emptyset.$$

Then we obtain

$$\Big(\big(F(x) - \{\bar{y}\}\big) \times \big(G(x) - \{\bar{z}\}\big)\Big) \cap \tilde{C} = \emptyset \text{ for all } x \in \hat{S}.$$

Together with the equality (3.3), we conclude that the map (F, G) is $\tilde{C}$-quasi-convex. □

Notice that the set $\operatorname{cone}(\bar{z}) - \operatorname{cone}(\bar{z})$ in Theorem 3.3 equals the one-dimensional linear subspace of Z generated by $\bar{z}$, i.e., $\{\lambda \bar{z} \in Z \mid \lambda \in \mathbb{R}\}$.

Based on the result of Theorem 3.3 we can now formulate the type of quasi convexity that is needed for the multiplier rule to be a sufficient optimality condition.

Corollary 3.4. *Let the cone C_Y have a nonempty interior* $\operatorname{int}(C_Y)$, *and let the contingent derivative of (F, G) exist at $(\bar{x}, (\bar{y}, \bar{z}))$ with $\bar{x} \in S$, $\bar{y} \in F(\bar{x})$ and $\bar{z} \in G(\bar{x})$. If there are continuous linear functionals $t \in C_{Y^*} \backslash \{0_{Y^*}\}$ and $u \in C_{Z^*}$ with*

$$t\big(y\big) + u\big(z\big) \geq 0 \textit{ for all } (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) \textit{ with } x \in \hat{S}$$

and

$$u(\bar{z}) = 0,$$

and if the map $(F, G) : \hat{S} \to 2^Y \times 2^Z$ is $\tilde{C}$-quasi-convex at $(\bar{x}, (\bar{y}, \bar{z}))$ with

$$\tilde{C} := \big(- \operatorname{int}(C_Y)\big) \times \big(- C_Z + \textit{cone}(\bar{z}) - \textit{cone}(\bar{z})\big),$$

then $(\bar{x}, \bar{y})$ is a weak minimizer of the problem (1.2).

Proof. By Theorem 3.3 $(\bar{x}, \bar{y})$ is a weak minimizer of F on

$$\tilde{S} = \big\{x \in \hat{S} \mid G(x) \cap \big(- C_Z + \operatorname{cone}(\bar{z}) - \operatorname{cone}(\bar{z})\big) \neq \emptyset\big\}.$$

For every $x \in S$ we obtain

$$\emptyset \neq G(x) \cap (-C_Z) \subset G(x) \cap \big(- C_Z + \operatorname{cone}(\bar{z}) - \operatorname{cone}(\bar{z})\big),$$

implying $x \in \tilde{S}$. Hence, we have $S \subset \tilde{S}$ and $(\bar{x}, \bar{y})$ is a weak minimizer of the problem (1.2). □

Example 3.5. We investigate the optimization problem in Example 2.2 again. Since F is $\mathbb{R}_+$-convex (notice that f is a convex functional) and G is $\mathbb{R}_+$-convex (notice that h is also a convex functional), the composite map $(F, G) : X \to 2^{\mathbb{R}} \times 2^{\mathbb{R}}$ has the required quasi-convexity property. Hence, if there are real numbers $t > 0$ and $u \geq 0$ with

$$tf'(\bar{x})(x - \bar{x}) + uh'(\bar{x})(x - \bar{x}) \geq 0 \text{ for all } x \in X$$

and

$$uh(\bar{x}) = 0,$$

then $(\bar{x}, f(\bar{x}))$ is a weak minimizer of the optimization problem (2.13) in Example 2.2.

If we combine Theorem 2.1 and Corollary 3.4, we obtain the main result of this paper: a complete characterization of weak minimizers using the Lagrange multiplier rule.

COROLLARY 3.6. *Let the cone C_Y have a nonempty interior* $\operatorname{int}(C_Y)$, *let the set $\hat{S}$ be convex, and let the maps F and G be C_Y-convex and C_Z-convex, respectively. Assume that a pair $(\bar{x}, \bar{y}) \in X \times Y$ with $\bar{x} \in S$ and $\bar{y} \in F(\bar{x})$ is given. Let the contingent epiderivative of (F, G) at $(\bar{x}, (\bar{y}, \bar{z}))$ for an arbitrary $\bar{z} \in G(\bar{x}) \cap (-C_Z)$ exist. Moreover, let the regularity assumption* (2.1) *be satisfied. Then $(\bar{x}, \bar{y})$ is a weak minimizer of the problem* (1.2) *if and only if there are continuous linear functionals $t \in C_{Y^*} \backslash \{0_{Y^*}\}$ and $u \in C_{Z^*}$ with*

$$t\big(y\big) + u\big(z\big) \geq 0 \textit{ for all } (y, z) = D(F, G)(\bar{x}, (\bar{y}, \bar{z}))(x - \bar{x}) \textit{ with } x \in \hat{S}$$

and

$$u(\bar{z}) = 0.$$

4. Conclusion. For general set-valued optimization problems, a Lagrange multiplier rule is shown using the concept of contingent epiderivatives. Under appropriate assumptions this multiplier rule is necessary and sufficient for weak minimizers in the cone-convex case. Since in the standard optimization theory this optimality condition has important applications in optimal control and approximation theory, it would be interesting to see whether the maximum principle or the extended Kolmogorov criterion can be generalized to set-valued problems. The main difficulty arises in the calculation of the contingent epiderivative of the objective and constraint maps.

Acknowledgment. The authors are grateful to the referees for valuable comments.

REFERENCES

[1] J.-P. AUBIN, *Contingent derivatives of set-valued maps and existence of solutions to nonlinear inclusions and differential inclusions*, in Mathematical Analysis and Applications, Part A, L. Nachbin, ed., Academic Press, New York, 1981, pp. 160–229.

[2] J.M. BORWEIN, *Multivalued convexity and optimization: A unified approach to inequality and equality constraints*, Math. Programming, 13 (1977), pp. 183–199.

[3] J.M. BORWEIN, *A Lagrange multiplier theorem and a sandwich theorem for convex relations*, Math. Scand., 48 (1981), pp. 189–204.

[4] J.M. BORWEIN, *Adjoint process duality*, Math. Oper. Res., 8 (1983), pp. 403–434.

[5] H.W. CORLEY, *Existence and Lagrangian duality for maximizations of set-valued functions*, J. Optim. Theory Appl., 54 (1987), pp. 489–501.

[6] H.W. CORLEY, *Optimality conditions for maximizations of set-valued functions*, J. Optim. Theory Appl., 58 (1988), pp. 1–10.

[7] A. GÖTZ, *Die Multiplikatorenregel von Lagrange in der mengenwertigen Optimierung für tangentielle Epiableitungen*, Master's thesis, University of Erlangen-Nürnberg, Erlangen, Germany, 1996.

[8] J. JAHN, *Mathematical Vector Optimization in Partially Ordered Linear Spaces*, Peter Lang, Frankfurt, 1986.

[9] J. JAHN, *Introduction to the Theory of Nonlinear Optimization*, Springer-Verlag, Berlin, 1996.

[10] J. Jahn, *Optimality conditions in set-valued vector optimization*, in Multiple Criteria Decision Making, G. Fandel, T. Gal, and T. Hanne, eds., Springer-Verlag, Berlin, 1997, pp. 22–30.

[11] J. Jahn and R. Rauh, *Contingent epiderivatives and set-valued optimization*, Math. Methods Oper. Res., 46 (1997), pp. 193–211.

[12] J. Jahn and E. Sachs, *Generalized quasiconvex mappings and vector optimization*, SIAM J. Control Optim., 24 (1986), pp. 306–322.

[13] J. Klose, *Sensitivity analysis using the tangent derivative*, Numer. Funct. Anal. Optim., 13 (1992), pp. 143–153.

[14] J.L. Lagrange, *Théorie des fonctions analytiques*, Paris, 1797.

[15] D.T. Luc, *Theory of Vector Optimization*, Springer-Verlag, Berlin, 1989.

[16] D.T. Luc, *Contingent derivatives of set-valued maps and applications to vector optimization*, Math. Programming, 50 (1991), pp. 99–111.

[17] D.T. Luc and J. Jahn, *Axiomatic approach to duality in optimization*, Numer. Funct. Anal. Optim., 13 (1992), pp. 305–326.

[18] D.T. Luc and C. Malivert, *Invex optimization problems*, Bull. Austral. Math. Soc., 46 (1992), pp. 47–66.

[19] W. Oettli, *Optimality conditions for programming problems involving multivalued mappings*, in Modern Applied Mathematics, B. Korte, ed., North–Holland, Amsterdam, 1981.

[20] V. Postolică, *Vectorial optimization programs with multifunctions and duality*, Ann. Sci. Math. Québec, 10 (1986), pp. 85–102.

[21] D. Zhuang, *Regularity and Minimality Properties of Set-Valued Structures in Optimization*, Ph.D. thesis, Dalhousie University, Halifax, Canada, 1989.

SIAM J. Optim.
Vol. 10, No. 2, pp. 345–358

CONVERGENCE PROPERTIES OF NONLINEAR CONJUGATE GRADIENT METHODS*

YUHONG DAI†, JIYE HAN‡, GUANGHUI LIU§, DEFENG SUN¶, HONGXIA YIN‡, AND YA-XIANG YUAN†

Abstract. Recently, important contributions on convergence studies of conjugate gradient methods were made by Gilbert and Nocedal [*SIAM J. Optim.*, 2 (1992), pp. 21–42]. They introduce a "sufficient descent condition" to establish global convergence results. Although this condition is not needed in the convergence analyses of Newton and quasi-Newton methods, Gilbert and Nocedal hint that the sufficient descent condition, which was enforced by their two-stage line search algorithm, may be crucial for ensuring the global convergence of conjugate gradient methods. This paper shows that the sufficient descent condition is actually not needed in the convergence analyses of conjugate gradient methods. Consequently, convergence results on the Fletcher–Reeves- and Polak–Ribière-type methods are established in the absence of the sufficient descent condition.

To show the differences between the convergence properties of Fletcher–Reeves- and Polak–Ribière-type methods, two examples are constructed, showing that neither the boundedness of the level set nor the restriction $\beta_k \geq 0$ can be relaxed for the Polak–Ribière-type methods.

Key words. conjugate gradient method, descent condition, global convergence

AMS subject classifications. 65, 49

PII. S1052623494268443

1. Introduction.

We consider the global convergence of conjugate gradient methods for the unconstrained nonlinear optimization problem

$$\min f(x), \tag{1.1}$$

where $f : R^n \to R^1$ is continuously differentiable and its gradient is denoted by g. We consider only the case where the methods are implemented without regular restarts. The iterative formula is given by

$$x_{k+1} = x_k + \lambda_k d_k, \tag{1.2}$$

where λ_k is a step-length and d_k is the search direction defined by

$$d_k = \begin{cases} -g_k & \text{for } k = 1, \\ -g_k + \beta_k d_{k-1} & \text{for } k \geq 2, \end{cases} \tag{1.3}$$

where β_k is a scalar and g_k denotes $g(x_k)$.

*Received by the editors May 26, 1994; accepted for publication (in revised form) July 9, 1998; published electronically December 22, 1999. This research was partially supported by Chinese NSF grants 19525101 and 19731010 and by the Australian Research Council.

http://www.siam.org/journals/siopt/10-2/26844.html

†LSEC, Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing 100080, People's Republic of China (dyh@lsec.cc.ac.cn, yyx@lsec.cc.ac.cn).

‡Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing 100080, People's Republic of China (jyhan@amath3.amt.ac.cn, hxyin@public.fhnet.cn.net).

§Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston IL, 60208 (guanghui@pluto.ece.nwu.edu).

¶School of Mathematics, University of New South Wales, Sydney 2052, Australia (sun@maths.unsw.edu.au).

The best-known formulas for β_k are the following Fletcher–Reeves, Polak–Ribière, and Hestenes–Stiefel formulas:

$$\beta_k^{\mathrm{FR}} = \|g_k\|^2/\|g_{k-1}\|^2, \tag{1.4}$$

$$\beta_k^{\mathrm{PR}} = g_k^T(g_k - g_{k-1})/\|g_{k-1}\|^2, \tag{1.5}$$

$$\beta_k^{\mathrm{HS}} = g_k^T(g_k - g_{k-1})/d_{k-1}^T(g_k - g_{k-1}), \tag{1.6}$$

where $\|\cdot\|$ denotes the l_2-norm. The Fletcher–Reeves [4] method with an exact line search was proved to be globally convergent on general functions by Zoutendijk [18]. However, the Polak–Ribière [13] and Hestenes–Stiefel [8] methods with the exact line search are not globally convergent; see the counterexample of Powell [14]. Conjugate gradient methods (1.2)–(1.3) with exact line searches satisfy the equality

$$-g_k^T d_k = \|g_k\|^2, \tag{1.7}$$

which directly implies the *sufficient descent condition*

$$-g_k^T d_k \geq c\|g_k\|^2 \tag{1.8}$$

for some positive constant $c > 0$. This condition has been used often in the literature to analyze the global convergence of conjugate gradient methods with inexact line searches. For instance, Al-Baali [1], Touati-Ahmed and Storey [15], Hu and Storey [9], and Gilbert and Nocedal [5] analyzed the global convergence of algorithms related to the Fletcher–Reeves method with the strong Wolfe line search. Their convergence analyses used the sufficient descent condition, which is implied by the strong Wolfe line search and Fletcher–Reeves-type β_k formulas. For algorithms related to the Polak–Ribière methods, Gilbert and Nocedal [5] investigated wide choices of β_k that resulted in globally convergent methods. In particular, they first gave the global convergence result for the Polak–Ribière-type methods $\beta_k = \max\{0, \beta_k^{\mathrm{PR}}\}$ with inexact line searches. In order for the sufficient descent condition to hold, they modified the strong Wolfe line search to the two-stage line search: the first stage is to find a point using the strong Wolfe line search, and the second stage is, when the sufficient descent condition does not hold, to do more line search iterations until a new point satisfying the sufficient descent condition is found. They hinted that the sufficient descent condition may be crucial for conjugate gradient methods.

It is noted that the sufficient descent condition is not needed in the convergence analyses of Newton and quasi-Newton methods. This motivates us to investigate whether the sufficient descent condition is necessary, as it seemed to be, for the global convergence of conjugate gradient methods. In [11], Liu, Han, and Yin have proved the global convergence properties of the Fletcher–Reeves method under weaker conditions than those of [1]. In [3], Dai and Yuan have proved that the Fletcher–Reeves method using the strong Wolfe line search is globally convergent as long as each search direction is downhill. In the next section, we will provide some basic results for general conjugate gradient methods with a descent condition, instead of the sufficient descent condition. In section 3, we will establish the convergence results for the Fletcher–Reeves- and Polak–Ribière-type methods without assuming the sufficient descent condition. To show the differences between the convergence of Fletcher–Reeves-type methods and Polak–Ribière-type methods, two nonconvergence examples are constructed in section 4 for the Polak–Ribière-type methods, showing that neither the boundedness of the level set nor the restriction $\beta_k \geq 0$ can be relaxed in some sense. A brief discussion is given in the last section.

2. Results for general conjugate gradient methods. Throughout this section, we assume that every search direction d_k satisfies the *descent condition*

$$g_k^T d_k < 0 \tag{2.1}$$

for all $k \geq 1$.

We make the following basic assumptions on the objective function.

ASSUMPTION 2.1. (i) *f is bounded below on the level set $\mathcal{L} = \{x | f(x) \leq f(x_1)\}$, where x_1 is the starting point.* (ii) *In some neighborhood $\mathcal{N}$ of $\mathcal{L}$, f is continuously differentiable, and its gradient is Lipschitz continuous; namely, there exists a constant $L > 0$ such that*

$$\|g(x) - g(y)\| \leq L\|x - y\| \qquad \textit{for all } x, y \in \mathcal{N}. \tag{2.2}$$

The step-length λ_k in (1.2) is computed by carrying out a line search. The *Wolfe line search* [16] consists of finding a positive step-length λ_k such that

$$f(x_k + \lambda_k d_k) \leq f(x_k) + \rho \lambda_k g_k^T d_k, \tag{2.3}$$

$$g(x_k + \lambda_k d_k)^T d_k \geq \sigma g_k^T d_k, \tag{2.4}$$

where $0 < \rho < \sigma < 1$. In order to prove global convergence for the Fletcher–Reeves method, [1], [5] and [9] used the *strong Wolfe line search*, which requires λ_k to satisfy (2.3) and

$$|g(x_k + \lambda_k d_k)^T d_k| \leq -\sigma g_k^T d_k. \tag{2.5}$$

The following important result was obtained by Zoutendijk [18] and Wolfe [16, 17].

LEMMA 2.2. *Suppose that Assumption* 2.1 *holds. Consider any iteration method of the form* (1.2)–(1.3)*, where d_k satisfies* (2.1) *and λ_k is obtained by the Wolfe line search. Then*

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty. \tag{2.6}$$

The following theorem is a general and positive result for conjugate gradient methods with the strong Wolfe line search.

THEOREM 2.3. *Suppose that Assumption* 2.1 *holds. Consider any method of the form* (1.2)–(1.3) *with d_k satisfying* (2.1) *and with the strong Wolfe line search* (2.3) *and* (2.5)*. Then either*

$$\liminf_{k \to \infty} \|g_k\| = 0 \tag{2.7}$$

or

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \tag{2.8}$$

Proof. (1.3) indicates that for all $k \geq 2$,

$$d_k + g_k = \beta_k d_{k-1}. \tag{2.9}$$

Squaring both sides of (2.9), we obtain

$$\|d_k\|^2 = -\|g_k\|^2 - 2g_k^T d_k + \beta_k^2 \|d_{k-1}\|^2. \tag{2.10}$$

It follows from this relation and (2.1) that

$$\|d_k\|^2 \geq \beta_k^2 \|d_{k-1}\|^2 - \|g_k\|^2. \tag{2.11}$$

Definition (1.3) implies the following relation:

$$g_k^T d_k - \beta_k g_k^T d_{k-1} = -\|g_k\|^2, \tag{2.12}$$

which, with the line search condition (2.5), shows that

$$|g_k^T d_k| + \sigma |\beta_k| \, |g_{k-1} d_{k-1}| \geq \|g_k\|^2. \tag{2.13}$$

The above inequality and the Cauchy–Schwartz inequality yield

$$(g_k^T d_k)^2 + \beta_k^2 (g_{k-1}^T d_{k-1})^2 \geq c_1 \|g_k\|^4, \tag{2.14}$$

where $c_1 = (1+\sigma^2)^{-1}$ is a positive constant. Therefore, it follows from (2.11) and (2.14) that

$$\begin{aligned} \frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} &= \frac{1}{\|d_k\|^2} \left[(g_k^T d_k)^2 + \frac{\|d_k\|^2}{\|d_{k-1}\|^2} (g_{k-1}^T d_{k-1})^2 \right] \\ &\geq \frac{1}{\|d_k\|^2} \left[(g_k^T d_k)^2 + \beta_k^2 (g_{k-1}^T d_{k-1})^2 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} \|g_k\|^2 \right] \\ &\geq \frac{1}{\|d_k\|^2} \left[c_1 \|g_k\|^4 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} \|g_k\|^2 \right]. \end{aligned} \tag{2.15}$$

If (2.7) is not true, relations (2.15) and (2.6) imply that the inequality

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} \geq \frac{c_1}{2} \frac{\|g_k\|^4}{\|d_k\|^2} \tag{2.16}$$

holds for all sufficiently large k. Now inequality (2.8) follows from (2.16) and (2.6). □

The following result is a direct corollary of the above theorem.

COROLLARY 2.4. *Suppose that Assumption* 2.1 *holds. Consider any method of the form* (1.2)–(1.3) *with* d_k *satisfying* (2.1) *and with the strong Wolfe line search* (2.3) *and* (2.5). *If*

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^t}{\|d_k\|^2} = +\infty \tag{2.17}$$

for any $t \in [0,4]$, *the method converges in the sense that* (2.7) *is true.*

Proof. If (2.7) is not true, it follows from Theorem 2.3 that

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \tag{2.18}$$

Because $\|g_k\|$ is bounded away from zero, and $t \in [0, 4]$, it is easy to see that (2.18) contradicts (2.17). This shows that the corollary is true. □

If a conjugate gradient method fails to converge, one can easily see from the above corollary that the length of the search direction will converge to infinity. Results similar to Corollary 2.4 can also be established using the Zoutendijk condition and the sufficient descent condition (1.8). It should be noted that we have not assumed the sufficient descent condition. Hence our results are powerful tools for our analyses in the next section, where we will concentrate on proving the global convergence of some conjugate gradient methods without assuming the sufficient descent condition (1.8). Another point worth mentioning is that we do not assume the boundedness of the level set.

3. Global convergence. In this section, we establish some global convergence results for the Fletcher–Reeves- and Polak–Ribière-type methods. The general outline of the proofs is that, assuming that the convergence relation (2.7) does not hold, we can derive that $\sum_{k=1}^{\infty} \frac{\|g_k\|^2}{\|d_k\|^2} = +\infty$ or $\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} = +\infty$, which with Corollary 2.4 in turn implies that (2.7) holds, giving a contradiction.

First, we consider the Fletcher–Reeves-type methods of the form (1.2)–(1.3), where β_k is any scalar satisfying

$$\sigma|\beta_k| \le \bar{\sigma}\beta_k^{\mathrm{FR}} \tag{3.1}$$

for all $k \ge 2$, where σ is the parameter defined in (2.4) and $\bar{\sigma} \in (0, 1/2]$ is a constant. In order to prove its global convergence, Hu and Storey [9] had to restrict the parameter $\bar{\sigma}$ to be strictly less than 1/2 to derive the sufficient descent condition. The following result shows that such a restriction can be relaxed while preserving the global convergence.

THEOREM 3.1. *Suppose that Assumption* 2.1 *holds. Consider any method of the form* (1.2)–(1.3) *with the strong Wolfe line search* (2.3) *and* (2.5)*, where* β_k *satisfies* (3.1) *with* $\bar{\sigma} \in (0, 1/2]$*, and*

$$\|g_k\|^2 \sum_{j=2}^{k} \prod_{i=j}^{k} \left(\frac{\beta_i}{\beta_i^{\mathrm{FR}}}\right)^2 \le c_2 k \tag{3.2}$$

for some constant $c_2 > 0$*. Then*

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{3.3}$$

Proof. From (1.3), (1.4), (2.5), and (3.1), we deduce that

$$\begin{aligned}\frac{-g_k^T d_k}{\|g_k\|^2} &= 1 - \beta_k \frac{-g_k^T d_{k-1}}{\|g_k\|^2} = 1 - \left(\frac{\beta_k}{\beta_k^{\mathrm{FR}}}\right) \frac{-g_k^T d_{k-1}}{\|g_{k-1}\|^2} \\ &\le 1 + \left|\frac{\beta_k}{\beta_k^{\mathrm{FR}}}\right| \frac{-\sigma g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2} \\ &\le 1 + \bar{\sigma}\left(\frac{-g_{k-1}^T d_{k-1}}{\|g_{k-1}\|^2}\right) \\ &\le \cdots\end{aligned}$$

$$\leq \sum_{j=0}^{k-2} \bar{\sigma}^j + \bar{\sigma}^{k-1} \left(\frac{-g_1^T d_1}{\|g_1\|^2} \right) = \frac{1-\bar{\sigma}^k}{1-\bar{\sigma}} < \frac{1}{1-\bar{\sigma}}. \tag{3.4}$$

Similarly, we have that

$$\frac{-g_k^T d_k}{\|g_k\|^2} \geq 1 - \bar{\sigma} \frac{1-\bar{\sigma}^{k-1}}{1-\bar{\sigma}} > 0 \tag{3.5}$$

because $\bar{\sigma} \leq 1/2$. Thus, d_k is a descent direction.

On the other hand, it follows from (2.10) that

$$\|d_k\|^2 \leq -2 g_k^T d_k + \beta_k^2 \|d_{k-1}\|^2. \tag{3.6}$$

Using (3.6) recursively and observing that $d_1 = -g_1$, we get that

$$\begin{aligned} \|d_k\|^2 &\leq -2 g_k^T d_k - 2 \sum_{j=2}^{k} \prod_{i=j}^{k} \beta_i^2 g_{j-1}^T d_{j-1} \\ &= -2 g_k^T d_k - 2 \|g_k\|^4 \sum_{j=2}^{k} \prod_{i=j}^{k} \left(\frac{\beta_i}{\beta_i^{\mathrm{FR}}} \right)^2 \left(\frac{g_{j-1}^T d_{j-1}}{\|g_{j-1}\|^4} \right). \end{aligned} \tag{3.7}$$

If the theorem is not true, (3.2) holds and there exists a positive constant γ such that

$$\|g_k\| \geq \gamma \quad \text{for all } k. \tag{3.8}$$

Thus, it follows from the above inequality, (3.4), and (3.7) that

$$\frac{\|d_k\|^2}{\|g_k\|^2} \leq \frac{2}{1-\bar{\sigma}} \left[1 + \frac{\|g_k\|^2}{\gamma^2} \sum_{j=2}^{k} \prod_{i=j}^{k} \left(\frac{\beta_i}{\beta_i^{\mathrm{FR}}} \right)^2 \right]. \tag{3.9}$$

The above relation and (3.2) imply that

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^2}{\|d_k\|^2} = +\infty. \tag{3.10}$$

This, with Corollary 2.4, implies that $\liminf_k \|g_k\| = 0$. This completes our proof. □

The above theorem extends Hu and Storey's [9] result to the case when $\bar{\sigma} = 1/2$. If $\bar{\sigma} \in (0, 1/2)$, we can see from (3.5) that the sufficient descent condition (1.8) holds. If $\bar{\sigma} = 1/2$, however, we only have that

$$\frac{-g_k^T d_k}{\|g_k\|^2} \geq \frac{1}{2^k}, \tag{3.11}$$

which does not imply the sufficient descent condition.

Now we consider methods that are related to the Polak–Ribière and Hestenes–Stiefel algorithms. We need the following assumption.

Assumption 3.2. *The level set* $\mathcal{L} = \{x | f(x) \leq f(x_1)\}$ *is bounded.*

Under Assumptions 2.1 and 3.2, there exists a positive constant $\bar{\gamma}$ such that

$$\|g(x)\| \leq \bar{\gamma} \quad \text{for all } x \in \mathcal{L}. \tag{3.12}$$

Denote $s_{k-1} = x_k - x_{k-1}$ and $u_k = d_k/\|d_k\|$. In [5], Gilbert and Nocedal introduced the following property.

PROPERTY $(*)$. *Consider a method of the form* (1.2)–(1.3), *and suppose that* (3.12) *and* (3.8) *hold. Then we say that the method has Property* $(*)$ *if there exist constants* $b > 1$ *and* $\lambda > 0$ *such that for all* k,

$$|\beta_k| \le b \tag{3.13}$$

and

$$\|s_{k-1}\| \le \lambda \Longrightarrow |\beta_k| \le \frac{1}{2b}. \tag{3.14}$$

Let N^* denote the set of positive integers. For $\lambda > 0$ and positive integer Δ, denote

$$\mathcal{K}_{k,\Delta}^{\lambda} := \{i \in N^* : k \le i \le k + \Delta - 1, \|s_{i-1}\| > \lambda\}.$$

Let $|\mathcal{K}_{k,\Delta}^{\lambda}|$ denote the number of elements of $\mathcal{K}_{k,\Delta}^{\lambda}$ and let $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ denote, respectively, the floor and ceiling operators. The following lemmas are drawn from [5].

LEMMA 3.3 (see [5]). *Suppose that Assumptions* 2.1 *and* 3.2 *hold. Consider any method of the form* (1.2)–(1.3) *with a descent direction* d_k. *If, at the* k*th step,* $\beta_k \ge 0$, *then* $d_k \ne 0$ *and*

$$\|u_k - u_{k-1}\| \le 2\frac{\|g_k\|}{\|d_k\|}. \tag{3.15}$$

LEMMA 3.4 (see [5]). *Suppose that Assumptions* 2.1 *and* 3.2 *hold. Consider the method of* (1.2)–(1.3) *with any line search satisfying* (2.1). *Assume that the method has Property* $(*)$ *and that*

$$\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} < +\infty. \tag{3.16}$$

Assume also that (3.8) *holds. Then there exists* $\lambda > 0$ *such that, for any* $\Delta \in N^*$ *and any index* k_0, *there is a greater index* $k > k_0$ *such that*

$$|\mathcal{K}_{k,\Delta}^{\lambda}| > \frac{\Delta}{2}.$$

The conditions used in Lemma 3.4 are not the same as those used in [5]. In particular, the sufficient descent condition (1.8) used in [5] is here replaced by the descent condition (2.1). Under this weaker condition, we can also establish a similar global convergence result as that in [5].

The next theorem is a global convergence result of conjugate gradient methods with Property $(*)$. It is applicable, for example, to the Polak–Ribière-type method

$$\beta_k = \max\{0, \beta_k^{\mathrm{PR}}\}. \tag{3.17}$$

The proof of the theorem is similar to that in [5].

THEOREM 3.5. *Suppose that Assumptions* 2.1 *and* 3.2 *hold. Consider the method* (1.2)–(1.3) *with the following three properties:* (i) $\beta_k \ge 0$; (ii) *the strong Wolfe line search conditions* (2.3) *and* (2.5) *and the descent condition* (2.1) *hold for all* k; (iii) *Property* $(*)$ *holds. Then the method converges in the sense that* (3.3) *holds.*

Proof. We proceed by contradiction, assuming that the theorem is not true. Then there exists a positive constant γ such that (3.8) holds. Since $\beta_k \geq 0$ and d_k is a descent direction, it follows from Lemma 3.3 that

$$\|u_k - u_{k-1}\| \leq 2\frac{\|g_k\|}{\|d_k\|} \tag{3.18}$$

for all $k \geq 2$. The above inequality, (3.8), and Theorem 2.3 imply that

$$\sum_{k=1}^{\infty} \|u_k - u_{k-1}\|^2 \leq \frac{4}{\gamma^2} \sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty. \tag{3.19}$$

For any two indices l, k, with $l \geq k$, we can write

$$\begin{aligned} x_l - x_{k-1} &= \sum_{i=k}^{l} \|s_{i-1}\| u_{i-1} \\ &= \sum_{i=k}^{l} \|s_{i-1}\| u_{k-1} + \sum_{i=k}^{l} \|s_{i-1}\| (u_{i-1} - u_{k-1}). \end{aligned}$$

This relation and the fact that $\|u_{k-1}\| = 1$ give

$$\sum_{i=k}^{l} \|s_{i-1}\| \leq \|x_1 - x_{k-1}\| + \sum_{i=k}^{l} \|s_{i-1}\| \, \|u_{i-1} - u_{k-1}\|. \tag{3.20}$$

Since f_k decreases with k, we have that $\{x_k\} \subset \mathcal{L}$, which together with Assumption 3.2 implies that there exists a positive constant B such that $\|x_k\| \leq B$ for all $k \geq 1$. Hence

$$\sum_{i=k}^{l} \|s_{i-1}\| \leq 2B + \sum_{i=k}^{l} \|s_{i-1}\| \, \|u_{i-1} - u_{k-1}\|. \tag{3.21}$$

By Corollary 2.4, we can assume that (3.16) holds. Thus the conditions of Lemma 3.4 are satisfied. Let $\lambda > 0$ be given by Lemma 3.4 and define $\Delta := \lceil 8B/\lambda \rceil$. By (3.19), we can find an index $k_0 \geq 1$ such that

$$\sum_{i \geq k_0} \|u_i - u_{i-1}\|^2 \leq \frac{1}{4\Delta}. \tag{3.22}$$

With this Δ and k_0, Lemma 3.4 gives an index $k \geq k_0$ such that

$$|\mathcal{K}_{k,\Delta}^{\lambda}| > \frac{\Delta}{2}. \tag{3.23}$$

Next, for any index $i \in [k, k+\Delta-1]$, by the Cauchy–Schwartz inequality and (3.22),

$$\begin{aligned} \|u_i - u_{k-1}\| &\leq \sum_{j=k}^{i} \|u_j - u_{j-1}\| \\ &\leq (i-k+1)^{1/2} \left(\sum_{j=k}^{i} \|u_j - u_{j-1}\|^2 \right)^{1/2} \\ &\leq \Delta^{1/2} \left(\frac{1}{4\Delta} \right)^{1/2} = \frac{1}{2}. \end{aligned} \tag{3.24}$$

Using this relation and (3.23) in (3.21), with $l = k + \Delta - 1$, we get that

$$2B \geq \frac{1}{2} \sum_{i=k}^{k+\Delta-1} \|s_{i-1}\| > \frac{\lambda}{2} |\mathcal{K}_{k,\Delta}^{\lambda}| > \frac{\lambda\Delta}{4}. \tag{3.25}$$

Thus $\Delta < 8B/\lambda$, which contradicts the definition of Δ. Therefore, the theorem is true. □

4. Nonconvergence examples. In the previous section, we have proved two convergence theorems, namely, Theorems 3.1 and 3.5, for the Fletcher–Reeves- and Polak–Ribière-type methods. Neither of the theorems needs the line search to satisfy the sufficient descent condition (1.8). In this section, we will present two nonconvergence examples for the Polak–Ribière methods.

It can be seen from Theorem 3.1 that the boundedness of the level set is not required in analyzing the Fletcher–Reeves-type methods. Therefore, the convergence results for the Fletcher–Reeves-type methods also apply to noncoercive objective function. In contrast, we are able to construct an example, as included in the following theorem, to show that the boundedness of the level set is necessary for the convergence of Polak–Ribière methods even if line searches are exact. It is easy to see that the theorem is also true for the Polak–Ribière-type method (3.17).

THEOREM 4.1. *Consider the Polak–Ribière method* (1.2), (1.3), *and* (1.5) *with* λ_k *chosen to be any local minimizer of* $\Phi_k(\lambda) = f(x_k + \lambda d_k)$, $\lambda > 0$. *Then there exists a starting point* x_1 *and a function* $f(x)$ *satisfying Assumption* 2.1 *such that the iterations generated by the method satisfy, for all* $k \geq 1$,

$$\beta_{k+1}^{\mathrm{PR}} \geq 0 \tag{4.1}$$

and

$$\|g_k\| = 1. \tag{4.2}$$

Proof. We define

$$\theta_k = \begin{cases} -\dfrac{\pi}{2} & \text{for } k = 0, \\ 0 & \text{for } k = 1, \\ \dfrac{1}{6}\left[1 - \left(-\dfrac{1}{2}\right)^{k-1}\right]\pi & \text{for } k \geq 2 \end{cases} \tag{4.3}$$

and consider the gradients and the search directions given by

$$g_k = (-1)^k \begin{pmatrix} \sin\theta_{k-1} \\ -\cos\theta_{k-1} \end{pmatrix} \tag{4.4}$$

and

$$d_k = \csc\frac{\pi}{2^k} \begin{pmatrix} \cos\theta_k \\ \sin\theta_k \end{pmatrix}, \tag{4.5}$$

where

$$\csc\frac{\pi}{2^k} = \frac{1}{\sin\frac{\pi}{2^k}}.$$

It follows that (4.2) holds for all $k \geq 1$. In addition, (4.4) and (4.5) clearly satisfy the equality

$$g_{k+1}^T d_k = 0. \tag{4.6}$$

Because

$$|\theta_k - \theta_{k-1}| = \frac{\pi}{2^k} \tag{4.7}$$

holds for all $k \geq 1$, it follows from (1.5), (4.2), and (4.4) that

$$\beta_{k+1}^{\mathrm{PR}} = 1 - g_{k+1}^T g_k = 1 + \cos(\theta_k - \theta_{k-1}) = 1 + \cos\frac{\pi}{2^k} = 2\cos^2\frac{\pi}{2^{k+1}}. \tag{4.8}$$

Thus (4.1) also holds for all $k \geq 1$. Further, direct calculations show that

$$\begin{aligned}
-g_{k+1} + \beta_{k+1}^{\mathrm{PR}} d_k &= (-1)^{k+1}\begin{pmatrix} -\sin\theta_k \\ \cos\theta_k \end{pmatrix} + 2\cos^2\frac{\pi}{2^{k+1}}\csc\frac{\pi}{2^k}\begin{pmatrix} \cos\theta_k \\ \sin\theta_k \end{pmatrix} \\
&= \csc\frac{\pi}{2^{k+1}}\left[\sin\frac{\pi}{(-2)^{k+1}}\begin{pmatrix} -\sin\theta_k \\ \cos\theta_k \end{pmatrix} + \cos\frac{\pi}{2^{k+1}}\begin{pmatrix} \cos\theta_k \\ \sin\theta_k \end{pmatrix}\right] \\
&= \csc\frac{\pi}{2^{k+1}}\begin{pmatrix} \cos\left(\theta_k + (-1)^{k+1}\frac{\pi}{2^{k+1}}\right) \\ \sin\left(\theta_k + (-1)^{k+1}\frac{\pi}{2^{k+1}}\right) \end{pmatrix} \\
&= \csc\frac{\pi}{2^{k+1}}\begin{pmatrix} \cos\theta_{k+1} \\ \sin\theta_{k+1} \end{pmatrix} = d_{k+1}. \qquad (4.9)
\end{aligned}$$

This together with $d_1 = -g_1$ imply that if the gradients are given by (4.4), then the Polak–Ribière method will produce the search directions as in (4.5).

Now, we let $\lambda_k = 1/\|d_k\|$ and define

$$x_k = \sum_{i=0}^{k-1}\begin{pmatrix} \cos\theta_i \\ \sin\theta_i \end{pmatrix} \tag{4.10}$$

and

$$f_k = -\sum_{i=0}^{k-1}\sin\frac{\pi}{2^i}. \tag{4.11}$$

Then (1.2) holds and since $\|d_k\| = csc\frac{\pi}{2^k}$ and $g_k^T d_k = -1$, (2.3) and (2.5) hold. Because

$$\lim_{k\to\infty}\theta_k = \frac{\pi}{6} \tag{4.12}$$

and

$$\|x_{k+1} - x_k\| = 1, \tag{4.13}$$

we can see that $\{x_k\}$ has no cluster points and hence that it is easy to construct a function f satisfying Assumption 2.1 such that for all $k \geq 1$,

$$f(x_k) = f_k, \qquad g(x_k) = g_k, \tag{4.14}$$

and λ_k is a local minimizer of $\Phi_k(\lambda)$. Therefore, for the starting point $x_1 = (0, -1)^T$ and the function f, the iterations generated by the Polak–Ribière method satisfy (4.1) and (4.2) for all $k \geq 1$. □

As opposed to Theorem 3.1, Theorem 3.5 does not allow any negative values of β_k. However, as pointed out in Gilbert and Nocedal [5], the Polak–Ribière method can produce negative values of β_k^{PR} even for strong convex objective functions. Therefore, it is interesting to investigate in what range the restriction $\beta_k \geq 0$ in Theorem 3.5 can be relaxed. After further studies of the $n = 2$, $m = 8$ example of Powell [14], we obtain the following result.

Theorem 4.2. *For any given positive constant ε, consider the method* (1.2)–(1.3) *with*

$$\beta_k = \max\{\beta_k^{\text{PR}}, -\varepsilon\} \tag{4.15}$$

and with λ_k chosen to be any local minimizer of $\Phi_k(\lambda) = f(x_k + \lambda d_k)$, $\lambda > 0$. There exists a starting point x_1 and a function $f(x)$ satisfying Assumptions 2.1 *and* 3.2 *such that the sequence of the gradient norms $\{\|g_k\|\}$ generated by the method is bounded away from zero.*

Proof. For any positive constant $\phi \in (0, 1)$, let the steps of the method have the form

$$s_{8j+i} = a_i \begin{pmatrix} 1 \\ b_i \phi^{2j} \end{pmatrix}, \quad s_{8j+4+i} = a_i \begin{pmatrix} -1 \\ b_i \phi^{2j+1} \end{pmatrix}, \quad j \geq 0, \quad i = 1, 2, 3, 4, \tag{4.16}$$

where the numbers $\{a_i; i = 1, 2, 3, 4\}$ are all positive, and consider the values

$$b_1 = -2, \quad b_2 = \frac{6 - 2\phi - 2\phi^2}{2 + 5\phi}, \quad b_3 = -\phi, \quad b_4 = -2.$$

To satisfy the line search condition

$$g_{k+1}^T d_k = 0, \tag{4.17}$$

we assume that the gradients have the form

$$\begin{aligned} &g_{8j+1} = c_1 \begin{pmatrix} b_4 \phi^{2j-1} \\ 1 \end{pmatrix}, \quad g_{8j+i} = c_i \begin{pmatrix} -b_{i-1} \phi^{2j} \\ 1 \end{pmatrix}, \quad i = 2, 3, 4; \\ &g_{8j+5} = c_1 \begin{pmatrix} -b_4 \phi^{2j+1} \\ 1 \end{pmatrix}, \quad g_{8j+4+i} = c_i \begin{pmatrix} b_{i-1} \phi^{2j+1} \\ 1 \end{pmatrix}, \quad i = 2, 3, 4, \end{aligned} \tag{4.18}$$

where $\{c_i; i = 1, 2, 3, 4\}$ are constants. To ensure the conjugacy condition

$$s_k^T (g_{k+1} - g_k) = 0 \tag{4.19}$$

for all $k \geq 1$, we choose each c_i as follows:

$$\begin{aligned} c_1 &= 3\phi(1-\phi)(5-\phi), \quad c_2 = -3(1+\phi)(2+\phi^2), \\ c_3 &= (1+\phi)(2-\phi)(2+5\phi), \quad c_4 = 2(5-\phi)(1-\phi^2). \end{aligned} \tag{4.20}$$

Because $n = 2$, relations (4.17) and (4.19) ensure that each d_k is produced by the Polak–Ribière method. In addition, direct calculations show that $g_k^T s_k < 0$ holds for all $k \geq 1$; namely, each d_k is a descent direction.

Due to symmetry, we can reduce the objective function at every iteration if the following relations hold:

$$f(x_{8j+1}) > f(x_{8j+2}) > f(x_{8j+3}) > f(x_{8j+4}) > f(x_{8j+5}). \tag{4.21}$$

Now, when the first component of x is equal to the first component of x_k, where k is any positive integer, then the values in (4.18) allow the second component of $g(x)$ to be constant, provided that the first components of the points $\{x_{8j+i}; i = 1, 2, \ldots, 8\}$ are all different. Thus, the equation

$$f(x_k) - f^* = (x_k)_2(g_k)_2 \tag{4.22}$$

is satisfied, where f^* is the limit of f_k. Given the limit point $\hat{x}_1 = \lim_{j\to\infty} x_{8j+1}$, we can compute x_{8j+1} in the following way:

$$x_{8j+1} = \hat{x}_1 - \sum_{k=j}^{\infty}\sum_{i=1}^{8} s_{8k+i} = \begin{pmatrix} 0 \\ h\phi^{2j}/(\phi-1) \end{pmatrix} \tag{4.23}$$

and

$$x_{8j+i+1} = x_{8j+i} + s_{8j+i}, \qquad i = 1, 2, \ldots, 7, \tag{4.24}$$

where $h = a_1b_1 + a_2b_2 + a_3b_3 + a_4b_4$. It follows that expression (4.21) is equivalent to the inequalities

$$\begin{aligned} -c_1(a_1b_1 + a_2b_2 + a_3b_3 + a_4b_4) &> -c_2(a_1b_1\phi + a_2b_2 + a_3b_3 + a_4b_4) \\ &> -c_3(a_1b_1\phi + a_2b_2\phi + a_3b_3 + a_4b_4) \\ &> -c_4(a_1b_1\phi + a_2b_2\phi + a_3b_3\phi + a_4b_4) \\ &> -c_1\phi(a_1b_1 + a_2b_2 + a_3b_3 + a_4b_4). \end{aligned} \tag{4.25}$$

These inequalities are consistent because, if

$$a_1 = 10, \quad a_2 = 35\phi, \quad a_3 = 38, \quad a_4 = \phi, \tag{4.26}$$

and if ϕ is small, then the dominant terms of the five lines of (4.25) are 300ϕ, 270ϕ, 240ϕ, 220ϕ, and 300ϕ, respectively. Now, as in Powell [14], we can construct a function satisfying Assumptions 2.1 and 3.2 such that the gradient conditions (4.18) hold.

By direct estimations, we can obtain that the dominant terms of $\{\beta_{4j+1}^{\text{PR}}; i = 1, 2, 3, 4\}$ are

$$-\frac{3}{2}\phi, \quad \frac{4}{25\phi^2}, \quad \frac{10}{9}, \quad \frac{9}{4},$$

respectively, when ϕ is small and j is large. Therefore, for any positive number $\varepsilon > 0$, we have that $\beta_k^{\text{PR}} \geq -\varepsilon$ for all large j, provided that $\phi \in (0, 1)$ is sufficiently small. This completes our proof. □

In [2], the above theorem is proved by using a three-dimensional example, in which line searches choose the first local minimum in every iteration.

5. Discussions. In this paper we have presented some global convergence results for nonlinear conjugate gradient methods, where the step-length is computed by the strong Wolfe conditions under the assumption that all the search directions are descent directions. The sufficient descent condition (1.8) has not been used in our convergence proofs and we have established convergence results for Fletcher–Reeves- and Polak–Ribière-type methods.

We have also provided two examples for which Polak–Ribière-type methods fail to converge. From these examples, we can see that the Fletcher–Reeves-type methods have better convergence properties than the Polak–Ribière-type methods, even though the latter perform better in practice. We believe that the results given in this paper will lead to a deeper understanding of the behavior of nonlinear conjugate gradient methods with inexact line searches.

This paper is a combination of two research reports, [6] and [2]; readers can find a more extensive discussion on the subject of this paper in those reports. See also [7], [10], and [11]. Some recent advances can be found in [7] and [10].

Acknowledgments. The authors are very grateful to Professor Jorge Nocedal for his suggestions in revising this paper. The third author is much indebted to Professor Jorge Nocedal for his grant support and supervision. The authors thank the referees for their extensive comments, which improved the paper.

REFERENCES

[1] M. Al-Baali, *Descent property and global convergence of the Fletcher-Reeves method with inexact linesearch*, IMA J. Numer. Anal., 5 (1985), pp. 121–124.

[2] Y. H. Dai and Y. Yuan, *Further Studies on the PRP Method*, Research Report ICM-95-040, Inst. of Comp. Math. and Sci. Eng. Computing, Chinese Academy of Sciences, Beijing, 1995.

[3] Y. H. Dai and Y. Yuan, *Convergence properties of the Fletcher-Reeves method*, IMA J. Numer. Anal., 16 (1996), pp. 155–164.

[4] R. Fletcher and C. M. Reeves, *Function minimization by conjugate gradients*, Comput. J., 7 (1964), pp. 149–154.

[5] J. C. Gilbert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM J. Optim., 2 (1992), pp. 21–42.

[6] J. Y. Han, G. H. Liu, D. F. Sun, and H. X. Yin, *On the Global Convergence of Nonlinear Conjugate Gradient Methods*, Technical Report 94-011, Inst. of Applied Math., Chinese Academy of Sciences, Beijing, 1994; Acta Math. Appl. Sinica, to appear.

[7] J. Y. Han, G. H. Liu, and H. X. Yin, *Convergence properties of conjugate gradient methods with strong Wolfe linesearch*, Systems Sci. Math. Sci., 11 (1998), pp. 112–116.

[8] M. R. Hestenes and E. Stiefel, *Method of conjugate gradient for solving linear system*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.

[9] Y. F. Hu and C. Storey, *Global convergence result for conjugate gradient methods*, J. Optim. Theory Appl., 71 (1991), pp. 399–405.

[10] G. H. Liu, J. Y. Han, H. D. Qi, and Z. L. Xu, *Convergence analysis on a class of conjugate gradient methods*, Acta Math. Scientia, 18 (1998), pp. 11–16.

[11] G. H. Liu, J. Y. Han, and H. X. Yin, *Global convergence of the Fletcher-Reeves algorithm with inexact linesearch*, Appl. Math. J. Chinese Univ. Ser. B, 10 (1995), pp. 75–82.

[12] J. J. Moré and D. J. Thuente, *On line search algorithms with guaranteed decrease*, ACM Trans. Math. Software, 20 (1994), pp. 286–307.

[13] E. Polak and G. Ribière, *Note sur la convergence de méthodes de directions conjugées*, Revue Française d'Informatique et de Recherche Opérationnelle, 16 (1969), pp. 35–43.

[14] M. J. D. Powell, *Nonconvex minimization calculations and the conjugate gradient method*, in Numerical Analysis, Lecture Notes in Mathematics 1066, D. F. Griffiths, ed., Springer-Verlag, Berlin, 1984, pp. 122–141.

[15] D. Touati-Ahmed and C. Storey, *Efficient hybrid conjugate gradient techniques*, J. Optim. Theory Appl., 64 (1990), pp. 379–397.

[16] P. Wolfe, *Convergence conditions for ascent methods*, SIAM Rev., 11 (1969), pp. 226–235.

[17] P. Wolfe, *Convergence conditions for ascent methods* II*: Some corrections*, SIAM Rev., 13 (1971), pp. 185–188.

[18] G. Zoutendijk, *Nonlinear programming, computational methods*, in Integer and Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, 1970, pp. 37–86.

SIAM J. Optim.
Vol. 10, No. 2, pp. 359–383

ON LOCAL SOLUTIONS OF THE CELIS–DENNIS–TAPIA SUBPROBLEM*

XIONGDA CHEN† and YA-XIANG YUAN†

Abstract. We discuss the distribution of the local solutions of the Celis–Dennis–Tapia (CDT) subproblem, which appears in some trust region algorithms for nonlinear optimization. We also give some examples to show the differences between the CDT subproblem and the single-ball-constraint subproblem. These results show that the complexity of the CDT subproblem does not depend on the complexity of the structure of the dual plane. Thus they provide the possibility to search for the global minimizer in the dual plane.

Key words. trust region subproblem, local solutions, optimality conditions

AMS subject classifications. 65K10, 90C20

PII. S1052623498335018

1. Introduction.

In this paper, we study some theoretical properties of local solutions to the following minimization problem with a quadratic objective and two quadratic constraints:

$$\min_{d\in\mathcal{R}^n} \Phi(d) = \frac{1}{2}d^TBd + g^Td \tag{1.1}$$

subject to

$$\|d\| \le \Delta, \tag{1.2}$$

$$\|A^Td + c\| \le \xi, \tag{1.3}$$

where $g \in \mathcal{R}^n$, $B \in \mathcal{R}^{n\times n}$, $A \in \mathcal{R}^{n\times m}$, $c \in \mathcal{R}^m$, $\Delta > 0$, $\xi \ge 0$, and B is a symmetric matrix. Throughout this paper, the norm $\|\cdot\|$ is the 2-norm. Problem (1.1)–(1.3) is a subproblem of some trust region algorithms for nonlinear programming (see Celis, Dennis, and Tapia [2] and Powell and Yuan [15]), and it is often called the CDT subproblem.

As an important application, the CDT subproblem was used as an inner iteration in the algorithm given by Powell and Yuan [15], whose superlinear convergence property is obtained under certain conditions. However, for general B and A, there is still no satisfactory method with which to find the global solution of problem (1.1)–(1.3) which is required in some trust region algorithms.

The properties of the CDT subproblem have been studied; see Yuan [16] and Peng and Yuan [13] for its extension. Under some additional assumptions, some algorithms have been given to solve it. For example, under the assumption that B is positive definite, different kinds of algorithm are presented by Ecker and Niemi [6], Mehrotra and Sun [12], Phan-Huy-Hao [14], Yuan [17], and Zhang [18]. Instead of the above

[*] Received by the editors March 6, 1998; accepted for publication (in revised form) March 1, 1999; published electronically December 22, 1999. This work was partially supported by Chinese NSF grants 19525101 and 19731001 and by State Key project 96-221-04-02-02.

http://www.siam.org/journals/siopt/10-2/33501.html

[†] State Key Laboratory of Scientific and Engineering Computing, Institute of Computational Mathematics and Scientific and Engineering Computing, Chinese Academy of Sciences, P.O. Box 2719, Beijing 100080, People's Republic of China (cxd@lsec.cc.ac.cn, yyx@lsec.cc.ac.cn).

assumption, under the assumption that $A = I$ and B is semidefinite an algorithm is given by Heinkenschloss [9], which is modified by Chen [3]. A global algorithm for the case $A = I$ and general symmetric B is given by Martínez and Santos [11].

Some approximate methods are given since the CDT problem, used as a subproblem of nonlinear programming algorithms, is needed to obtain a sufficient descent feasible point instead of the global minimizer. See El-Alem and Tapia [7] and Fu, Luo, and Ye [8] for algorithms based on approximations of the feasible region. Byrd and Schnabel [1] and Dennis and Williamson [4] solve the CDT subproblem in the two-dimensional subspace $\{g, (A^T)^+c\}$. These alternative CDT subproblems work in some nonlinear programming to some extent—for example, in the so-called PNCDT method of El-Alem and Tapia [7]. However, it is not clear how to compute the global solution of the CDT problem efficiently.

As a subproblem, ξ can be chosen such that problem (1.2)–(1.3) has feasible points; see Dennis, El-Alem, and Maciel [5] and references therein. If the CDT subproblem has no interior point, it is deduced to a simple case which is discussed in Yuan [16]. So we assume that problem (1.2)–(1.3) has interior points and we do not discuss the choice of ξ.

The rest of this paper is organized as follows. Some basic results are restated in section 2. The structure of the dual plane of the CDT problem is investigated in section 3. The dual function of the CDT problem is extended to a closed region, and some properties of the extend dual function are given, in section 4. Sections 5 and 6 are the main part of this paper. The global minimizer of the CDT problem is divided into three cases. By defining a "related region," we prove that the Lagrangian multipliers corresponding to the global minimizer locate in the related region if the maximizer of the dual function does not correspond to a global minimizer. Then we find the smallest related region. The differences between the local minimizers of the trust region subproblem and those of the CDT problem are presented in section 7. Also shown is that the Lagrangian multipliers of the CDT problem corresponding to local minimizers are permuted in the way of the connected branches of the region where the Hessian has exactly one negative eigenvalue. Conclusions and some possible ways to solve the CDT problem are presented in section 8.

2. Some basic results.

In this section, we restate some fundamental results of the CDT problem. For their proofs, see Yuan [16].

THEOREM 2.1. *Let d^* be a global solution of the problem* (1.1)–(1.3). *Assume that $\xi > \min_{\|d\|\le\Delta} \|A^T d + c\|$. Then there exist nonnegative constants λ, μ such that*

$$(B + \lambda I + \mu AA^T)d^* = -(g + \mu Ac), \tag{2.1}$$

where λ and μ satisfy the complementarity conditions

$$\lambda(\Delta - \|d^*\|) = 0, \tag{2.2}$$

$$\mu(\xi - \|A^T d^* + c\|) = 0. \tag{2.3}$$

Furthermore, the matrix

$$H(\lambda, \mu) = B + \lambda I + \mu AA^T \tag{2.4}$$

has at most one negative eigenvalue if the multipliers λ and μ are unique.

To say that $H(\lambda, \mu)$ has one negative eigenvalue means that the negative eigenvalue of $H(\lambda, \mu)$ is a single eigenvalue. For the case that the multipliers λ and μ are not unique, we have the following result.

THEOREM 2.2. *Assume that the conditions of Theorem* 2.1 *hold. Then there exists* $(\lambda, \mu) \in \Omega$ *such that the matrix* (2.4) *has at most one negative eigenvalue, where* Ω *is the set of Lagrangian multipliers.*

We have the following sufficient optimality condition for a global minimizer of problem (1.1)–(1.3).

THEOREM 2.3. *If* d^* *is a feasible point of* (1.2)–(1.3)*, if there are two multipliers* λ *and* μ *such that* (2.1)–(2.3) *hold, and if the matrix* (2.4) *is positive semidefinite, then* d^* *is a global solution of the problem* (1.1)–(1.3).

3. Structure of dual plane. A dual algorithm is given in Yuan [17] for solving subproblem (1.1)–(1.3) with B positive definite, based on the equivalent problem:

$$\min_{d \in \mathcal{R}^n} \Phi(d) = \frac{1}{2} d^T B d + g^T d \tag{3.1}$$

subject to

$$\|d\|^2 \leq \Delta^2, \tag{3.2}$$

$$\|A^T d + c\|^2 \leq \xi^2. \tag{3.3}$$

Similar to the single-ball constrained trust region subproblem, the CDT subproblem may be hard when the Hessian of the Lagrangian is positive semidefinite but not positive definite. Furthermore, the Hessian at the global solution may have one negative eigenvalue (see Theorem 2.1). The dual problem for (3.1)–(3.3) can also be defined when the Hessian of the Lagrangian is singular. This will be discussed in the next section.

First, we consider the case in which the Hessian of the Lagrangian is nonsingular. The Hessian of the Lagrangian is (2.4), where $\lambda \geq 0$, $\mu \geq 0$ are the Lagrangian multipliers of problem (3.1)–(3.3), and they are also the dual variables. Using the notations of Yuan [17], we define the vector

$$d(\lambda, \mu) = -H(\lambda, \mu)^{-1}(g + \mu A c), \tag{3.4}$$

which satisfies the first equation of the well-known KKT system (2.1)–(2.3) of problem (3.1)–(3.3). We also define the Lagrangian dual function of problem (3.1)–(3.3) as

$$\Psi(\lambda, \mu) = \Phi(d(\lambda, \mu)) + \frac{\lambda}{2}(\|d(\lambda, \mu)\|^2 - \Delta^2) + \frac{\mu}{2}(\|A^T d(\lambda, \mu) + c\|^2 - \xi^2) \tag{3.5}$$

and the region

$$\Omega_0 = \{(\lambda, \mu) \in \mathcal{R}^2_+ \mid H(\lambda, \mu) \text{ is positive semidefinite}\}, \tag{3.6}$$

where $d(\lambda, \mu)$ is defined by (3.4) and $\mathcal{R}^2_+ = \{\lambda \geq 0, \mu \geq 0\}$. Direct calculations show that

$$\nabla \Psi(\lambda, \mu) = \frac{1}{2} \begin{pmatrix} \|d(\lambda, \mu)\|^2 - \Delta^2 \\ \|A^T d(\lambda, \mu) + c\|^2 - \xi^2 \end{pmatrix} \tag{3.7}$$

and

$$\nabla^2\Psi(\lambda,\mu) = -\begin{pmatrix} d(\lambda,\mu)^T H(\lambda,\mu)^{-1} d(\lambda,\mu) & d(\lambda,\mu)^T H(\lambda,\mu)^{-1} y(\lambda,\mu) \\ d(\lambda,\mu)^T H(\lambda,\mu)^{-1} y(\lambda,\mu) & y(\lambda,\mu)^T H(\lambda,\mu)^{-1} y(\lambda,\mu) \end{pmatrix}, \tag{3.8}$$

where $y(\lambda,\mu)$ is the vector

$$y(\lambda,\mu) = A(A^T d(\lambda,\mu) + c). \tag{3.9}$$

In order to study the dual function, we define the region

$$\Omega(\varepsilon) = \{(\lambda,\mu) \in \Omega_0 \mid \ \mathrm{dist}((\lambda,\mu), \partial\Omega_0) \geq \varepsilon\}, \tag{3.10}$$

where $\varepsilon > 0$, $\mathrm{dist}(\cdot,\cdot)$ is the 2-norm distance function, and $\partial\Omega$ denotes the boundary of a region Ω. It is easy to see that Ω_0 and $\Omega(\varepsilon)$ are convex sets. First we show a property of the Hessian on $\Omega(\varepsilon)$.

Lemma 3.1. *For any $(\lambda,\mu) \in \Omega(\varepsilon)$, $H(\lambda,\mu)$ is positive definite.*

Proof. Define $\mathcal{B}_\eta$ to be the Euclidean ball in R^2 with radius η and

$$X \oplus Y = \{x + y \mid x \in X, y \in Y\} \tag{3.11}$$

for two sets X and Y. We have, for any $(\lambda,\mu) \in \Omega(\varepsilon)$,

$$(\lambda,\mu) \oplus \mathcal{B}_{\frac{\varepsilon}{2}} \subset \Omega(\varepsilon) \oplus \mathcal{B}_{\frac{\varepsilon}{2}} \subset \Omega\left(\frac{\varepsilon}{2}\right), \tag{3.12}$$

which implies that $(\lambda - \frac{\varepsilon}{2}, \mu) \in \Omega(\frac{\varepsilon}{2})$ and $H(\lambda - \frac{\varepsilon}{2}, \mu)$ is positive semidefinite. Therefore,

$$H(\lambda,\mu) = H\left(\lambda - \frac{\varepsilon}{2}, \mu\right) + \frac{\varepsilon}{2} I \tag{3.13}$$

is positive definite. □

From the above lemma, the dual function $\Psi(\lambda,\mu)$, its gradient, and its Hessian are well defined in $\mathrm{int}\Omega_0$, the interior of Ω_0, and the region $\Omega(\varepsilon)$ for any $\varepsilon > 0$. Thus by the concavity of $\Psi(\lambda,\mu)$ (from (3.8)), we can obtain the maxima of $\Psi(\lambda,\mu)$ on $\Omega(\varepsilon)$. If there is an $\varepsilon > 0$ such that

$$(\lambda_+, \mu_+) = \arg \max_{(\lambda,\mu)\in\Omega(\varepsilon)} \Psi(\lambda,\mu) \in \ \mathrm{int}\Omega(\varepsilon), \tag{3.14}$$

then $H(\lambda_+, \mu_+)$ is positive definite and we can prove the following theorem. Actually, this theorem holds for any positive definite $H(\lambda_+, \mu_+)$ if $(\lambda_+, \mu_+) \in \Omega_0$.

Theorem 3.2. *Suppose that (λ_+, μ_+) is any point defined by* (3.14)*. Then the global solution of* (3.1)–(3.3) *is $d(\lambda_+, \mu_+)$ given by* (3.4).

Proof. Because $H(\lambda_+, \mu_+)$ is positive definite, $d(\lambda_+, \mu_+)$ is well defined by (3.4), and (λ_+, μ_+) is a local maximizer of $\Psi(\lambda,\mu)$ in $\mathcal{R}^2_+$. Therefore, we have that

$$\nabla\Psi(\lambda_+, \mu_+) \leq 0 \tag{3.15}$$

and

$$(\lambda_+, \mu_+)^T \nabla\Psi(\lambda_+, \mu_+) = 0. \tag{3.16}$$

This shows that $d(\lambda_+, \mu_+)$ is the global solution of (3.1)–(3.3) and (λ_+, μ_+) are the corresponding Lagrangian multipliers. □

Since $\arg\max \Psi(\lambda,\mu)$ is a convex set, it includes an interior point of Ω_0 if it strictly includes a segment. If $\arg\max \Psi(\lambda,\mu)$ on $\mathrm{int}\Omega_0$ is a segment and there is a point of this segment in the interior of Ω_0, there exists a sufficient small $\epsilon >$ such that (3.14) holds. Thus it follows from Theorem 3.2 that there exists a global solution of (3.1)–(3.3) with a positive definite Hessian of the Lagrangian. Otherwise, the Hessian of the Lagrangian might not be positive semidefinite. By the theorems stated in section 2, the Hessian $H(\lambda,\mu)$ at the global solution of (3.1)–(3.3) has at most one negative eigenvalue, and the corresponding Lagrangian multipliers (λ,μ) may locate in the region

$$\Omega_1 = \{(\lambda,\mu) \in \mathcal{R}^2_+ \mid H(\lambda,\mu) \text{ has one negative eigenvalue}\}. \tag{3.17}$$

Next we investigate the structure of Ω_1. Let

$$\Omega_1 = \bigcup_{k\in\mathcal{K}} \Omega_{1k}, \tag{3.18}$$

where $\mathcal{K}$ is an index set; $\Omega_{1k}, k \in \mathcal{K}$, are connected sets; and Ω_{1k} and Ω_{1j} are disconnected for any $k \neq j$, $k,j \in \mathcal{K}$. Whether $\mathcal{K}$ is a finite index set makes no difference to our discussions by the location theorem given in section 5. By defining

$$u_{\lambda k} = \sup\nolimits_{(\lambda,\mu)\in\Omega_{1k}} \lambda, \tag{3.19}$$

$$u_{\mu k} = \sup\nolimits_{(\lambda,\mu)\in\Omega_{1k}} \mu, \tag{3.20}$$

$$l_{\lambda k} = \inf\nolimits_{(\lambda,\mu)\in\Omega_{1k}} \lambda, \tag{3.21}$$

and

$$l_{\mu k} = \inf\nolimits_{(\lambda,\mu)\in\Omega_{1k}} \mu, \tag{3.22}$$

we have the following lemma,

LEMMA 3.3. *If there are $(\lambda_1,\mu_1) \in \Omega_{1k}$, $(\lambda_2,\mu_2) \in \Omega_{1j}$, where $k \neq j$ and $\lambda_1 > \lambda_2$, then we have*

$$l_{\lambda k} \geq u_{\lambda j}, \tag{3.23}$$

$$u_{\mu k} \leq l_{\mu j}. \tag{3.24}$$

Moreover, (3.23) *and* (3.24) *are both equalities or strict inequalities.*

Proof. It is easy to show that the set $\{\lambda \mid \exists(\lambda,\mu) \in \Omega_{1k}\}$ is a segment for any fixed μ, as is $\{\mu \mid \exists(\lambda,\mu) \in \Omega_{1k}\}$ for any fixed λ. If $l_{\lambda k} < u_{\lambda j}$, there is a $\lambda_0 \in (l_{\lambda k}, u_{\lambda j})$. By the above definitions, there are $\mu_k^0, \mu_j^0 \in R$ such that $(\lambda_0,\mu_k^0) \in \Omega_{1k}$ and $(\lambda_0,\mu_j^0) \in \Omega_{1j}$. Without loss of generality, let $\mu_k^0 < \mu_j^0$. Denote $\rho_i(B)$ as the ith eigenvalue of B. We have

$$\rho_n(H(\lambda_0,\mu_k^0)) \leq \rho_n(H(\lambda_0,\mu)) \leq \rho_n(H(\lambda_0,\mu_j^0)) < 0, \tag{3.25}$$

$$0 \leq \rho_{n-1}(H(\lambda_0,\mu_k^0)) \leq \rho_{n-1}(H(\lambda_0,\mu)) \leq \rho_{n-1}(H(\lambda_0,\mu_j^0)) \tag{3.26}$$

for all $\mu \in (\mu_k^0, \mu_j^0)$. Thus $(\lambda_0, \mu) \in \Omega_1$ for all $\mu \in (\mu_k^0, \mu_j^0)$, which implies that Ω_{1k} and Ω_{1j} are connected. The contradiction proves (3.23), and (3.24) can be proved similarly.

If $l_{\lambda k} > u_{\lambda j}$ and $u_{\mu k} = l_{\mu j}$, $H(u_{\lambda j}, l_{\mu j})$ is positive semidefinite and $H(l_{\lambda k}, u_{\mu k}) = H(u_{\lambda j}, l_{\mu j}) + (l_{\lambda k} - u_{\lambda j})I$ is positive definite, contradicting the definitions (3.19)–(3.22).

Suppose that $l_{\lambda k} = u_{\lambda j}$ and $u_{\mu k} < l_{\mu j}$. Since for any sufficiently small $\varepsilon > 0$,

$$(l_{\lambda k} - \varepsilon, \mu) \notin \Omega_{1l} \tag{3.27}$$

for any $\mu \in (u_{\mu k}, l_{\mu j})$ and $l \in \mathcal{K}$, $H(l_{\lambda k} - \varepsilon, \mu)$ has at least two negative eigenvalues. It also can be shown that $H(l_{\lambda k} + \varepsilon, \mu)$ is positive definite. Thus taking $\varepsilon \to 0$, we obtain that $H(l_{\lambda k}, \mu)$ has zero eigenvalues with multiplicity at least two for $\mu \in (u_{\mu k}, l_{\mu j})$.

Since $\det(H(l_{\lambda k}, \mu))$ is a polynomial with zeros with multiplicity at least two for $\mu \in (u_{\mu k}, l_{\mu j})$, $\det(H(l_{\lambda k}, \mu))$ has zeros with multiplicity at least two for all $\mu \geq 0$, which means that the dimension of $\text{Null}(H(l_{\lambda k}, \mu))$ is no less than two. Therefore, $H(u_{\lambda j} - \varepsilon, \mu)$ has at least two negative eigenvalues for all $\varepsilon > 0$ and $\mu > l_{\mu j}$, contradicting the definition of Ω_{1j}. □

In the following, we denote $\Omega_{1k} \succ \Omega_{1j}$ if (3.23)–(3.24) hold. Moreover, from the above proof, there is at most one segment in the intersection of any positive-slope straight line in $\mathcal{R}_+^2$ and any connected branch of Ω_1.

Definition 3.4. *Two connected branches $\Omega_{1k} \succ \Omega_{1j}$ of Ω_1 are called consecutive connected branches if there is no other connected branch Ω_{1l} of Ω_1 such that*

$$\Omega_{1k} \succ \Omega_{1l} \succ \Omega_{1j}, \tag{3.28}$$

and they are called two adjoint connected branches if (3.23) *and* (3.24) *hold as equalities.*

The following lemma tells us the more detailed structure of the border of Ω_0.

Lemma 3.5. *For any two adjoint connected branches Ω_{1k} and Ω_{1j} of Ω_1, $\Omega_{1k} \succ \Omega_{1j}$, $(l_{\lambda k}, u_{\mu k})$ is an extreme point of the convex set Ω_0.*

Proof. It suffices to prove that $\partial\Omega_0 \bigcap \partial(\Omega_{1k} \bigcup \Omega_{1j})$ is not a segment in any neighborhood of $(l_{\lambda k}, u_{\mu k})$. We prove this lemma by contradiction. Suppose

$$(l_{\lambda k}, u_{\mu k}) \in \overline{\Omega_{1k}} \bigcap \overline{\Omega_{1j}}, \tag{3.29}$$

$$(\bar{\lambda}_k, \bar{\mu}_k) \in \partial\Omega_0 \bigcap \overline{\Omega}_{1k}, \tag{3.30}$$

and

$$(\bar{\lambda}_j, \bar{\mu}_j) \in \partial\Omega_0 \bigcap \overline{\Omega}_{1j} \tag{3.31}$$

are in a straight line. Then there exists $0 < \delta < 1$, such that

$$H(l_{\lambda k}, u_{\mu k}) = \delta H(\bar{\lambda}_k, \bar{\mu}_k) + (1 - \delta) H(\bar{\lambda}_j, \bar{\mu}_j). \tag{3.32}$$

Since $(\bar{\lambda}_k - \varepsilon, \bar{\mu}_k - \varepsilon) \in \Omega_{1k}$ for $\varepsilon > 0$ sufficient small, $H(\bar{\lambda}_k - \varepsilon, \bar{\mu}_k - \varepsilon)$ has exactly one negative eigenvalue. Taking $\varepsilon \to 0+$, we can show that $H(\bar{\lambda}_k, \bar{\mu}_k)$ is positive semidefinite and has one multiple zero eigenvalue. The fact is also true for $H(\bar{\lambda}_j, \bar{\mu}_j)$. Suppose that

$$z_1 \in \ \text{Null}(H(\bar{\lambda}_k, \bar{\mu}_k)) \tag{3.33}$$

and

$$z_2 \in \text{ Null}(H(\bar{\lambda}_j, \bar{\mu}_j)), \tag{3.34}$$

where Null$(\cdot)$ denotes the null space of a matrix. Since $H(l_{\lambda k}, u_{\mu k})$ is positive semidefinite, for any $\varepsilon > 0$, $H((1+\varepsilon)l_{\lambda k}, (1+\varepsilon)u_{\mu k})$ is positive definite. For any Ω_{1l} satisfying $\Omega_{1l} \succeq \Omega_{1k}$,

$$l_{\lambda l} > (1-\varepsilon)l_{\lambda k}, \tag{3.35}$$

while for any Ω_{1l} satisfying $\Omega_{1j} \succeq \Omega_{1l}$,

$$u_{\mu l} > (1-\varepsilon)u_{\mu k}, \tag{3.36}$$

for any $\varepsilon > 0$. So for any $\varepsilon > 0$, $((1-\varepsilon)l_{\lambda k}, (1-\varepsilon)u_{\mu k}) \notin \Omega_{1j}$ for all j, and hence $H((1-\varepsilon)l_{\lambda k}, (1-\varepsilon)u_{\mu k})$ has at least two negative eigenvalues. Let $\varepsilon \to 0+$; we can see that $H(l_{\lambda k}, u_{\mu k})$ is positive semidefinite and has zero eigenvalues with multiplicity at least two.

Suppose span$\{v_1, v_2\} \subset$ Null$(H(l_{\lambda k}, u_{\mu k}))$. Then for all $z \in$ span$\{v_1, v_2\}$,

$$z^T H(l_{\lambda k}, u_{\mu k})z = z^T(\delta H(\bar{\lambda}_k, \bar{\mu}_k) + (1-\delta)H(\bar{\lambda}_j, \bar{\mu}_j))z = 0. \tag{3.37}$$

The above equality implies that $z^T H(\bar{\lambda}_k, \bar{\mu}_k)z = 0$ and $z^T H(\bar{\lambda}_j, \bar{\mu}_j)z = 0$, so $z_1 = z_2$, and

$$\text{Null}(H(\bar{\lambda}_k, \bar{\mu}_k)) = \text{ Null}(H(\bar{\lambda}_j, \bar{\mu}_j)), \tag{3.38}$$

and the dimension of Null$(H(l_{\lambda k}, u_{\mu k}))$ is equal to 1, contradicting the fact that $H(l_{\lambda k}, u_{\mu k})$ has zero eigenvalues with multiplicity two. □

4. Definitions on the boundary.

In this section, we deal with the boundary of Ω_0. First we define the dual function on the boundary of Ω_0 based on the definitions in intΩ_0. Assuming that $(\bar{\lambda}, \bar{\mu}) \in \partial\Omega_0$ with $H(\bar{\lambda}, \bar{\mu})$ singular, we define

$$\Psi(\bar{\lambda}, \bar{\mu}) = \lim_{\varepsilon \to 0+} \Psi(\bar{\lambda} + \varepsilon, \bar{\mu}). \tag{4.1}$$

In summary, $\Psi(\bar{\lambda}, \bar{\mu})$ is the right limit of $\Psi(\cdot, \cdot)$ at the point $(\bar{\lambda}, \bar{\mu})$ along the line $\mu = \bar{\mu}$. Because $\Psi(\cdot, \cdot)$ is continuous in intΩ_0, definition (4.1) also holds for the interior point of Ω_0.

Lemma 4.1. *Equation* (4.1) *is well defined.*

Proof. Equation (3.5) can be rewritten as

$$\Psi(\lambda, \mu) = -\frac{1}{2}(g + \mu Ac)^T H(\lambda, \mu)^{-1}(g + \mu Ac) - \frac{\lambda}{2}\Delta^2 - \frac{\mu}{2}(\xi^2 - \|c\|^2) \tag{4.2}$$

when $(\lambda, \mu) \in$ intΩ_0. Therefore,

$$\Psi(\lambda, \mu) \le -\frac{\lambda}{2}\Delta^2 - \frac{\mu}{2}(\xi^2 - \|c\|^2), \tag{4.3}$$

which shows that $\Psi(\lambda, \mu)$ is locally upper bounded in intΩ_0.

If (2.1) is inconsistent, the right-hand side of (4.1) is $-\infty$. Otherwise, suppose that $g + \bar{\mu}Ac = H(\bar{\lambda}, \bar{\mu})v$ for some $v \in \mathcal{R}^n$. It is easy to see that for any positive semidefinite matrix A,

$$\lim_{\varepsilon \to 0+}(A + \varepsilon I)^{-1}A = A^+ A. \tag{4.4}$$

Therefore, the following limit exists:

$$\begin{array}{rl} & \lim_{\varepsilon\to 0+} d(\bar{\lambda}+\varepsilon,\bar{\mu}) \\ = & \lim_{\varepsilon\to 0+} -H(\bar{\lambda}+\varepsilon,\bar{\mu})^{-1}(g+\bar{\mu}Ac) \\ = & \lim_{\varepsilon\to 0+} -H(\bar{\lambda}+\varepsilon,\bar{\mu})^{-1}H(\bar{\lambda},\bar{\mu})\bar{g} \\ = & -H(\bar{\lambda},\bar{\mu})^{+}H(\bar{\lambda},\bar{\mu})\bar{g}. \end{array} \tag{4.5}$$

$d(\bar{\lambda}+\varepsilon,\bar{\mu})$ is uniformly bounded when $\varepsilon\to 0+$. Suppose there are two sequences $\{\lambda_{1k}\}$ and $\{\lambda_{2k}\}$ such that

$$\lim_{\lambda_{1k}\to\bar{\lambda}+}\Psi(\lambda_{1k},\bar{\mu}) \neq \lim_{\lambda_{2k}\to\bar{\lambda}+}\Psi(\lambda_{2k},\bar{\mu}). \tag{4.6}$$

By the mean value theorem, we have

$$\Psi(\lambda_{1k},\bar{\mu})-\Psi(\lambda_{2k},\bar{\mu})=\frac{1}{2}(\lambda_{1k}-\lambda_{2k})\|d(\lambda_m,\bar{\mu})\|. \tag{4.7}$$

The right-hand side of (4.7) vanishes by the boundedness of $d(\cdot,\bar{\mu})$, contradicting (4.6). This completes our proof. □

Thus $\Psi(\lambda,\mu)$ is defined on the closed set Ω_0 and can take a finite value or $-\infty$, but not $+\infty$. However,

$$\max_{(\lambda,\mu)\in\Omega_0}\Psi(\lambda,\mu) \tag{4.8}$$

may be $+\infty$, and

$$\arg\max_{(\lambda,\mu)\in\Omega_0}\Psi(\lambda,\mu) \tag{4.9}$$

may lie at the infinity on the dual space $\mathcal{R}^2_+$. In section 5, we will see that this case can be handled in the same way as the finite case. Since $\Psi(\cdot,\cdot)$ is well defined on the closed set Ω_0, we define the set

$$S=\left\{(\lambda,\mu)\in\Omega_0 \mid (\lambda,\mu)=\arg\max_{\Omega_0}\Psi(\lambda,\mu)\right\}. \tag{4.10}$$

Because $\Psi(\lambda,\mu)$ is concave in intΩ_0 and also concave on Ω_0 by Lemma 4.2 given below, its maxima point set may be a segment on the $\partial\Omega_0$. Then Lemma 3.5 implies that this segment belongs to only one connected branch.

LEMMA 4.2. *$\Psi(\lambda,\mu)$ is a concave function on the closed set Ω_0.*

Proof. Let (λ_1,μ_1) and (λ_2,μ_2) be two points in Ω_0. Then we have that $H(\lambda_1+\varepsilon,\mu_1)$ and $H(\lambda_2+\varepsilon,\mu_2)$ are positive definite for all $\varepsilon>0$, and hence so is $H(\frac{\lambda_1+\lambda_2+2\varepsilon}{2},\frac{\mu_1+\mu_2}{2})$. Since $\Psi(\cdot,\cdot)$ is concave in intΩ_0,

$$\Psi\left(\frac{\lambda_1+\lambda_2+2\varepsilon}{2},\frac{\mu_1+\mu_2}{2}\right)\geq\frac{1}{2}(\Psi(\lambda_1+\varepsilon,\mu_1)+\Psi(\lambda_2+\varepsilon,\mu_2)). \tag{4.11}$$

Taking limits on both sides of the above inequality we deduced that $\Psi(\lambda,\mu)$ is concave on Ω_0. □

Assuming $(\bar{\lambda},\bar{\mu})\in\partial\Omega_0$, we define

$$d(\bar{\lambda},\bar{\mu})=\lim_{\varepsilon\to 0+}d(\bar{\lambda}+\varepsilon,\bar{\mu}). \tag{4.12}$$

If (2.1) is inconsistent for $(\bar{\lambda}, \bar{\mu})$, the right-hand side of (4.1) goes to $-\infty$. In this case $d(\bar{\lambda}, \bar{\mu})$ is undefined in (4.12). Suppose (2.1) is consistent at $(\bar{\lambda}, \bar{\mu})$. Then we can choose, for convenience, the minimum norm least square solution of (2.1). In the following, we will see that it is important that the limit (4.12) satisfies the property stated in Lemma 4.3 instead of the definition (4.12) itself.

LEMMA 4.3. *Assuming that* (2.1) *holds at* $(\bar{\lambda}, \bar{\mu}) \in \partial\Omega_0$*, we have the following property:*

$$\lim_{\varepsilon \to 0+} H(\bar{\lambda}, \bar{\mu}) d(\bar{\lambda} + \varepsilon, \bar{\mu}) = -(g + \bar{\mu}Ac). \tag{4.13}$$

Proof. First we have

$$\begin{aligned} H(\bar{\lambda}, \bar{\mu}) d(\bar{\lambda} + \varepsilon, \bar{\mu}) &= -H(\bar{\lambda}, \bar{\mu}) H(\bar{\lambda} + \varepsilon, \bar{\mu})^{-1}(g + \bar{\mu}Ac) \\ &= -H(\bar{\lambda}, \bar{\mu}) \left(H(\bar{\lambda}, \bar{\mu}) + \varepsilon I\right)^{-1} (g + \bar{\mu}Ac). \end{aligned} \tag{4.14}$$

It is easy to see that for any positive semidefinite matrix A,

$$\lim_{\varepsilon \to 0+} A(A + \varepsilon I)^{-1} = AA^{+}, \tag{4.15}$$

where A^{+} is the Moore–Penrose generalized inverse of A. Thus, it follows from (4.14) and (4.15) that

$$\begin{aligned} \lim\nolimits_{\varepsilon \to 0+} H(\bar{\lambda}, \bar{\mu}) d(\bar{\lambda} + \varepsilon, \bar{\mu}) &= -H(\bar{\lambda}, \bar{\mu}) H(\bar{\lambda}, \bar{\mu})^{+}(g + \bar{\mu}Ac) \\ &= -(g + \bar{\mu}Ac), \end{aligned} \tag{4.16}$$

which gives (4.13). □

Since we have

$$\begin{aligned} & \lim\nolimits_{\varepsilon \to 0+}(g + \bar{\mu}Ac)^T H(\bar{\lambda} + \varepsilon, \bar{\mu})^{-1}(g + \bar{\mu}Ac) \\ = \; & \lim\nolimits_{\varepsilon \to 0+}(g + \bar{\mu}Ac)^T H(\bar{\lambda} + \varepsilon, \bar{\mu})^{-1} H(\bar{\lambda} + \varepsilon, \bar{\mu}) H(\bar{\lambda} + \varepsilon, \bar{\mu})^{-1}(g + \bar{\mu}Ac) \\ = \; & \lim\nolimits_{\varepsilon \to 0+}(g + \bar{\mu}Ac)^T H(\bar{\lambda} + \varepsilon, \bar{\mu})^{+} H(\bar{\lambda} + \varepsilon, \bar{\mu}) H(\bar{\lambda} + \varepsilon, \bar{\mu})^{+}(g + \bar{\mu}Ac) \\ = \; & (g + \bar{\mu}Ac)^T H(\bar{\lambda}, \bar{\mu})^{+}(g + \bar{\mu}Ac), \end{aligned} \tag{4.17}$$

the following result follows from our extended definitions given in (4.1).

LEMMA 4.4. *For* $(\bar{\lambda}, \bar{\mu}) \in \Omega_0$*,*

$$\Psi(\bar{\lambda}, \bar{\mu}) = \begin{cases} -\infty & \textit{if } (2.1) \textit{ is inconsistent,} \\ -\frac{1}{2}(g + \bar{\mu}Ac)^T H(\bar{\lambda}, \bar{\mu})^{+}(g + \bar{\mu}Ac) - \frac{\lambda}{2}\Delta^2 - \frac{\mu}{2}(\xi^2 - \|c\|^2) & \\ & \textit{otherwise} \end{cases} \tag{4.18}$$

and

$$d(\bar{\lambda}, \bar{\mu}) = \begin{cases} \textit{undefined} & \textit{if } (2.1) \textit{ is inconsistent,} \\ -H(\bar{\lambda}, \bar{\mu})^{+}(g + \bar{\mu}Ac) & \textit{otherwise.} \end{cases} \tag{4.19}$$

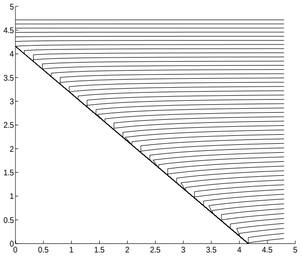


Fig. 5.1. *Example* 5.1.

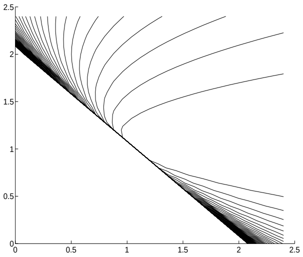


Fig. 5.2. *Example* 5.2.

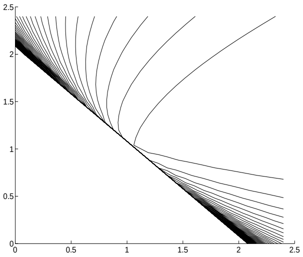


Fig. 5.3. *Example* 5.3.

5. Location of global solution.

In this section, we study the relations between the set S defined by (4.10) and the Lagrangian multipliers $(\bar{\lambda}, \bar{\mu})$ at the global solutions of problem (1.1)–(1.3). First, we consider the following cases:

- There exists a $(\lambda_+, \mu_+) \in S$ satisfying (3.14). So, $d(\lambda_+, \mu_+)$ is a global solution of problem (3.1)–(3.3) due to Theorem 3.2. In this case, S may be a singleton or a segment.
- A segment of S lies in $\partial\Omega_{1k}$ for some $k \in \mathcal{K}$. (k is unique due to Lemma 3.5.) Theorem 5.4, given below, states that we have obtained the desired global solution.
- S is a singleton such that $S \subset \partial\Omega_1$. For this case, in this section we give the locating branches in which the global solution lies, which generally includes two or three connected branches of Ω_1. In this case, the Hessian of any global solution of problem (3.1)–(3.3) might not be positive semidefinite. And we still cannot determine in which connected branch the global solution lies. This is the hard case of subproblem (1.1)–(1.3).

The following examples show the last two cases.

Example 5.1. *A segment maxima of the dual function.* Let

$$B = \mathrm{diag}(-4, -2), A = \mathrm{diag}(1, 2), g = (0, 4)', c = (0, 3)', \Delta = 3, \xi = \sqrt{6}.$$

Then $d = (\pm\sqrt{5}, -2)'$ with the Lagrangian multiplier $(4 - \mu, \mu)$ and the Hessian of Lagrangian $\mathrm{diag}(0, 2 + 3\mu)$, where $\mu \in [0, 4]$.

The following two examples show the hardness of the last case.

Example 5.2. Let

$$B = \mathrm{diag}(-2, 2), A = \mathrm{diag}(1, 1), g = (2, 0)', c = (-2, 0)', \Delta = 2, \xi = 1.$$

Then $d = (2, 0)'$ with the Lagrangian multiplier $(1, 0)$ and the Hessian of Lagrangian $\mathrm{diag}(-1, 3)$.

Example 5.3. Let

$$B = \mathrm{diag}(-1, -2), A = \mathrm{diag}(1, 1), g = (-4, 6)', c = (0, -6)', \Delta = 5, \xi = 5.$$

Then $d = (4, 3)'$ with the Lagrangian multiplier $(1, 1)$ and the Hessian of Lagrangian $\mathrm{diag}(1, 0)$.

The contours of the dual functions of the above examples are given in Figures 5.1, 5.2, and 5.3.

If $S \subset \Omega_{1k}$ is a segment for some $k \in \mathcal{K}$, we already get the solution by adding a null-space-step of the Hessian of the Lagrangian by Theorem 5.4.

THEOREM 5.4. *If $S \subset \partial\Omega_{1k}$ is a segment, then there exists a solution of* (2.1)–(2.3) *where the Hessian of the Lagrangian is positive semidefinite.*

Proof. Let $(\bar{\lambda}, \bar{\mu})$ and $(\hat{\lambda}, \hat{\mu})$ be two different points of S. By the definition of S, we have

$$\lim_{i\to+\infty} \Psi(\bar{\lambda}_i, \bar{\mu}) = \max_{\Omega_0} \Psi(\lambda, \mu), \tag{5.1}$$

$$\lim_{i\to+\infty} \Psi(\hat{\lambda}_i, \hat{\mu}) = \max_{\Omega_0} \Psi(\lambda, \mu), \tag{5.2}$$

where $\bar{\lambda}_i - \bar{\lambda} \to 0+$ and $\hat{\lambda}_i - \hat{\lambda} \to 0+$. Since $\Psi(\cdot,\cdot)$ approximates to a constant in a neighborhood of S, the above two relations give that

$$\lim_{i\to+\infty} \begin{pmatrix} \|d(\bar{\lambda}_i, \bar{\mu})\|^2 - \Delta^2 \\ \|A^T d(\bar{\lambda}_i, \bar{\mu}) + c\|^2 - \xi^2 \end{pmatrix}^T \begin{pmatrix} \bar{\lambda} - \hat{\lambda} \\ \bar{\mu} - \hat{\mu} \end{pmatrix} = 0 \tag{5.3}$$

and

$$\lim_{i\to+\infty} d(\bar{\lambda}_i, \bar{\mu})(\bar{\lambda}_i - \hat{\lambda}_i) - y(\bar{\lambda}_i, \bar{\mu})(\bar{\mu} - \hat{\mu}) = 0. \tag{5.4}$$

Since $S \subset \partial\Omega_{1k}$, $H(\bar{\lambda}, \bar{\mu})$, and $H(\hat{\lambda}, \hat{\mu})$ are singular. By the concavity of $\Psi(\cdot,\cdot)$, we have that $H(\lambda, \mu)$ is singular and positive semidefinite for all (λ, μ) in the segment between $(\bar{\lambda}, \bar{\mu})$ and $(\hat{\lambda}, \hat{\mu})$. For any $0 \neq v \in \text{Null}(H(\bar{\lambda}, \bar{\mu}))$ we have that

$$v^T H(\bar{\lambda}, \bar{\mu}) v = v^T H(\hat{\lambda}, \hat{\mu}) v = 0, \tag{5.5}$$

which implies that

$$\|v\|^2(\bar{\lambda} - \hat{\lambda}) + \|A^T v\|^2(\bar{\mu} - \hat{\mu}) = 0. \tag{5.6}$$

Therefore,

$$\lim_{i\to+\infty} v^T(d(\bar{\lambda}_i, \bar{\mu})(\bar{\lambda}_i - \hat{\lambda}_i) - y(\bar{\lambda}_i, \bar{\mu})(\bar{\mu} - \hat{\mu})) = 0, \tag{5.7}$$

which, together with (5.4) and (5.6), gives that

$$\lim_{i\to+\infty} \begin{pmatrix} \|d(\bar{\lambda}_i, \bar{\mu}) + tv\|^2 - \Delta^2 \\ \|A^T(d(\bar{\lambda}_i, \bar{\mu}) + tv) + c\|^2 - \xi^2 \end{pmatrix}^T \begin{pmatrix} \bar{\lambda} - \hat{\lambda} \\ \bar{\mu} - \hat{\mu} \end{pmatrix} = 0 \tag{5.8}$$

for all $t \in \mathcal{R}$. Since the right-hand sides of (5.1) and (5.2) are not $-\infty$, it can be seen that (2.1) is consistent at $(\bar{\lambda}, \bar{\mu})$ and $(\hat{\lambda}, \hat{\mu})$. Thus

$$\lim_{i\to\infty} d(\bar{\lambda}_i, \bar{\mu}) = d(\bar{\lambda}, \bar{\mu}). \tag{5.9}$$

Since $(\bar{\lambda}, \bar{\mu}) \in S$, by the reasons mentioned in Theorem 3.2, we see that $\|d(\bar{\lambda}, \bar{\mu})\| \leq \Delta$. Therefore, we can choose t_i such that

$$\lim_{i\to+\infty} \|d(\lambda_i, \mu) + t_i v\| = \Delta. \tag{5.10}$$

Let $\bar{d}$ be any limit point of $\{d(\lambda_i, \mu) + t_i v\}$, so

$$\|\bar{d}\| = \Delta, \tag{5.11}$$

$$\|A^T \bar{d} + c\| = \xi, \tag{5.12}$$

and

$$H(\bar{\lambda}, \bar{\mu})\bar{d} = -(g + \bar{\mu}Ac), \tag{5.13}$$

which implies that $\bar{d}$ is a global solution, $(\bar{\lambda}, \bar{\mu})$ is the corresponding pair of Lagrangian multipliers, and $H(\bar{\lambda}, \bar{\mu})$ is positive semidefinite. □

6. Location of global solution: Hard case. We now consider the hard case of problem (1.1)–(1.3), in which S is a singleton on $\partial\Omega_0$ and the Hessian at the global solution may have one negative eigenvalue. In order to determine the region where the solution locates, we introduce the following definition. In all the following discussion we assume that $S = \{(\lambda_+, \mu_+)\} \subset \partial\Omega_0$ is a singleton.

DEFINITION 6.1. *Define two sets*

$$\mathcal{L} = \{\lambda_e < \lambda_+ \mid ri\{\{\lambda = \lambda_e\} \cap \Omega_1\} = \emptyset\} \cup \{0\} \tag{6.1}$$

and

$$\mathcal{M} = \{\mu_e < \mu_+ \mid ri\{\{\mu = \mu_e\} \cap \Omega_1\} = \emptyset\} \cup \{0\}. \tag{6.2}$$

For $\lambda_e \in \mathcal{L}$ *and* $\mu_e \in \mathcal{M}$*, the set*

$$\Omega(\lambda_e, \mu_e) = \left(\overline{\Omega}_1 \cup S\right) \cap \{\lambda \geq \lambda_e, \mu \geq \mu_e\} \tag{6.3}$$

is called a related region of S*.*

Because S might not be in Ω_1 or even $\overline{\Omega}_1$, the term $\overline{\Omega}_1 \cup S$ must occur in (6.3). Here, $ri\{\{\lambda = \lambda_e\} \cap \Omega_1\} = \emptyset$ implies that it is impossible for $H(\lambda_e, \mu)$ to have exactly one negative eigenvalue for any μ. In the latter case, for the λ-direction, we may have the following two cases:

(i) $\lambda_+ > 0$, and for all $0 \leq \lambda_e < \lambda_+$, $ri\{\{\lambda = \lambda_e\} \cap \Omega_1\} \neq \emptyset$;

(ii) $\lambda_+ = 0$.

Similarly, we have two cases for the μ-direction. If $\lambda_e = 0$ or $\mu_e = 0$, the related region is the same as $\overline{\Omega}_1 \cup S$ in the λ-direction or in the μ-direction. Therefore, there exists a Lagrange multiplier at the global solution lies in the related region in the λ-direction or in the μ-direction. Thus, we need only to consider the case when $\lambda_e \neq 0$ and $\mu_e \neq 0$. In this case we have that $\lambda_+ \neq 0$ and $\mu_+ \neq 0$.

First, we need the following lemma to prove our location theorem.

LEMMA 6.2. *Assume that* S *is a singleton. If the triple* (λ^*, μ^*, d^*) *satisfies KKT system* (2.1)–(2.3) *with* $(\lambda^*, \mu^*) \in \Omega_0$*, and if either of the statements*

(i) $\lambda^* \neq 0$*,*

(ii) $\mu^* \neq 0$ *and* $\det(B + \lambda^* I + \mu AA^T)$ *does not vanish identically for* $\mu \geq \mu^*$

holds, then

$$S = \{(\lambda^*, \mu^*)\}. \tag{6.4}$$

Proof. It suffices to prove, for any fixed $(\lambda, \mu) \in \text{int}\Omega_0$,

$$\Psi(\lambda^*, \mu^*) \geq \Psi(\lambda, \mu). \tag{6.5}$$

We assume, first, that $\lambda^* \neq 0$. For any $\varepsilon > 0$, it is easy to see that d^* is a global solution of the subproblem

$$\min_{d \in \mathcal{R}^n} \bar{\Phi}(d) = \frac{1}{2} d^T B d + \bar{g}^T d \tag{6.6}$$

subject to

$$\|d\| \leq \Delta, \tag{6.7}$$

$$\|A^T d + c\| \leq \xi, \tag{6.8}$$

where $\bar{g} = g - \varepsilon d^*$ and the corresponding pair of Lagrangian multipliers is $(\lambda^* + \varepsilon, \mu^*)$. Define that

$$\bar{d}(\lambda, \mu) = -H(\lambda, \mu)^{-1}(\bar{g} + \mu A c) \tag{6.9}$$

and

$$\bar{\Psi}(\lambda, \mu) = -\frac{1}{2}(\bar{g} + \mu A c)^T H(\lambda, \mu)^{-1}(\bar{g} + \mu A c) - \frac{\lambda}{2}\Delta^2 - \frac{\mu}{2}(\xi^2 - \|c\|^2) \tag{6.10}$$

when $H(\lambda, \mu)$ is positive definite. If $H(\lambda, \mu)$ is singular, $\Psi(\lambda, \mu)$ can be defined as in (4.1). Then we have

$$\nabla \bar{\Psi}(\lambda^* + \varepsilon, \mu^*) \leq 0 \tag{6.11}$$

and

$$(\lambda^* + \varepsilon, \mu^*)^T \nabla \bar{\Psi}(\lambda^* + \varepsilon, \mu^*) = 0. \tag{6.12}$$

Since the Hessian $H(\lambda^* + \varepsilon, \mu^*)$ is positive definite, (6.11) and (6.12) imply that $(\lambda^* + \varepsilon, \mu^*)$ is a stationary point of $\bar{\Psi}(\lambda, \mu)$. Since $\bar{\Psi}(\lambda, \mu)$ is concave in Ω_0, $(\lambda^* + \varepsilon, \mu^*)$ is a global maximizer of $\bar{\Psi}(\lambda, \mu)$ on Ω_0, i.e.,

$$\bar{\Psi}(\lambda^* + \varepsilon, \mu^*) \geq \bar{\Psi}(\lambda, \mu) \text{ for all } (\lambda, \mu) \in \text{ int}\Omega_0. \tag{6.13}$$

For the left-hand side of (6.13), we have

$$\begin{aligned} & \lim_{\varepsilon \to 0+} \bar{\Psi}(\lambda^* + \varepsilon, \mu^*) \\ = \ & \lim_{\varepsilon \to 0+} -\tfrac{1}{2} d^{*T} H(\lambda^* + \varepsilon, \mu^*) d^* - \tfrac{\lambda^* + \varepsilon}{2}\Delta^2 - \tfrac{\mu^*}{2}(\xi^2 - \|c\|^2) \\ = \ & -\tfrac{1}{2} d^{*T} H(\lambda^*, \mu^*) d^* - \tfrac{\lambda^*}{2}\Delta^2 - \tfrac{\mu^*}{2}(\xi^2 - \|c\|^2). \end{aligned} \tag{6.14}$$

Because (λ^*, μ^*, d^*) satisfies the KKT system, it follows that

$$\begin{aligned} & -\tfrac{1}{2} d^{*T} H(\lambda^*, \mu^*) d^* \\ = \ & -\tfrac{1}{2} d^{*T} H(\lambda^*, \mu^*) H^+(\lambda^*, \mu^*) H(\lambda^*, \mu^*) d^* \\ = \ & -\tfrac{1}{2}(g + \mu^* A c)^T H^+(\lambda^*, \mu^*)(g + \mu^* A c). \end{aligned} \tag{6.15}$$

Equations (6.14), (6.15), and (4.18) imply that

$$\lim_{\varepsilon \to 0+} \bar{\Psi}(\lambda^* + \varepsilon, \mu^*) = \Psi(\lambda^*, \mu^*). \tag{6.16}$$

For any $(\lambda,\mu)\in \mathrm{int}\Omega_0$, $H(\lambda,\mu)$ is positive definite. Thus, it is easy to see that

$$\lim_{\varepsilon\to 0+}\bar{\Psi}(\lambda+\varepsilon,\mu)=\Psi(\lambda,\mu)\quad\text{ for all } (\lambda,\mu)\in\ \mathrm{int}\Omega_0. \tag{6.17}$$

Therefore, (6.5) follows from (6.13), (6.16), and (6.17).

Now we consider the case that $\mu^*\neq 0$, and $\det(B+\lambda^* I+\mu AA^T)\not\equiv 0$ for $\mu\geq\mu^*$. Let $\bar{g}=g-\varepsilon AA^Td^*$, where d^* is a global solution of (6.6)–(6.8) with $(\lambda^*,\mu^*+\varepsilon)$ the corresponding pair of Lagrangian multipliers. Our assumption implies that there exists a small ε^* such that

$$B+\lambda^* I+(\mu^*+\varepsilon)AA^T \tag{6.18}$$

is positive definite for all $0<\varepsilon<\varepsilon^*$. Then, we also have equalities (6.15) and (6.17), and, similarly, we have

$$\begin{aligned} &\lim\nolimits_{\varepsilon\to 0+}\bar{\Psi}(\lambda^*,\mu^*+\varepsilon)\\ = \quad &-\tfrac{1}{2}{d^*}^T H(\lambda^*,\mu^*)d^*-\tfrac{\lambda^*}{2}\Delta^2-\tfrac{\mu^*}{2}(\xi^2-\|c\|^2). \end{aligned} \tag{6.19}$$

Thus we can prove the same result. □

Moreover, with the additional assumption that d^* is feasible for both constraints, we can prove the result of Lemma 6.2 without the assumption that $\det(H(\lambda^*,\mu))\not\equiv 0$ for $\mu>\mu^*$. Suppose the condition $\det(H(\lambda^*,\mu))\not\equiv 0$ for $\mu>\mu^*$ fails, i.e., $\det(B+\lambda^* I+\mu AA^T)\equiv 0$ for $\mu\geq\mu^*$. Let μ_c be the minimal μ such that $(\lambda^*,\mu)\in\Omega_0$, i.e., $H(\lambda^*,\mu)$ is positive semidefinite for $\mu=\mu_c$ and is not positive semidefinite for $\mu<\mu_c$. By Lemma 4.2, $\Psi(\lambda^*,\mu)$ is a concave function on $\mu\in[\mu_c,+\infty)$. By the arguments of Theorem 3.2,

$$\frac{d\Psi(\lambda^*,\mu)}{d\mu}\,|_{\mu=\mu^*}\ \leq 0, \tag{6.20}$$

which holds as an inequality only if $\mu^*=\mu_c$. Hence μ^* is the maximizer of function $\Psi(\lambda^*,\cdot)$ on $[\mu_c,+\infty)$. Thus, by (3.15) and (3.16), (λ^*,μ^*) is the maximizer of $\Psi(\lambda,\mu)$.

In the following, we discuss some properties of the so-called shifted problem.

We consider the "shifted" problem $\widehat{P}$:

$$\min\ \ \hat{\Phi}(\hat{d})=\frac{1}{2}\hat{d}^T(B+\lambda_e I+\mu_e AA^T)\hat{d}+(g+\mu_e Ac)^T\hat{d} \tag{6.21}$$

subject to

$$\|\hat{d}\|^2\leq\Delta^2, \tag{6.22}$$

$$\|A^T\hat{d}+c\|^2\leq\xi^2. \tag{6.23}$$

Actually, except for a constant, the objective function $\hat{\Phi}(d)$ is the sum of the original objective function $\Phi(d)$ and a penalty term $\frac{1}{2}(\lambda_e\|d\|^2+\mu_e\|A^Td+c\|^2)$.

The dual function of $\widehat{P}$ is

$$\hat{\Psi}(\hat{\lambda},\hat{\mu})=\hat{\Phi}(\hat{d})+\frac{\hat{\lambda}}{2}(\|\hat{d}\|^2-\Delta^2)+\frac{\hat{\mu}}{2}(\|A^T\hat{d}+c\|^2-\xi^2), \tag{6.24}$$

where

$$\hat{d}=\hat{d}(\hat{\lambda},\hat{\mu})=-(\hat{B}+\hat{\lambda}I+\hat{\mu}AA^T)^{-1}(\hat{g}+\hat{\mu}Ac), \tag{6.25}$$

$\hat{B} = B + \lambda_e I + \mu_e AA^T$, $\hat{g} = g + \mu_e Ac$, and $\hat{\lambda} \geq 0, \hat{\mu} \geq 0$ are the multipliers of problem $\widehat{P}$. The Hessian of the Lagrangian is $\hat{H}(\hat{\lambda}, \hat{\mu}) = \hat{B} + \hat{\lambda} I + \hat{\mu} AA^T$. We also define the regions,

$$\widehat{\Omega}_0 = \{(\hat{\lambda}, \hat{\mu}) \in \mathcal{R}^2_+ \mid \hat{H}(\hat{\lambda}, \hat{\mu}) \text{ is positive semidefinite}\}, \tag{6.26}$$

and

$$\widehat{\Omega}_1 = \{(\hat{\lambda}, \hat{\mu}) \in \mathcal{R}^2_+ \mid \hat{H}(\hat{\lambda}, \hat{\mu}) \text{ has one negative eigenvalue}\}. \tag{6.27}$$

It is easy to show that

$$\widehat{\Omega}_0 = (\Omega_0 \cap \{\lambda \geq \lambda_e, \mu \geq \mu_e\}) - (\lambda_e, \mu_e) \tag{6.28}$$

and

$$\widehat{\Omega}_1 = (\Omega_1 \cap \{\lambda \geq \lambda_e, \mu \geq \mu_e\}) - (\lambda_e, \mu_e). \tag{6.29}$$

The statement (6.29) also holds for the connected branches $\widehat{\Omega}_{1j}$ and Ω_{1j} of the regions $\widehat{\Omega}_1$ and Ω_1, respectively, if $\Omega_{1j} \cup \{\lambda \geq \lambda_e, \mu \geq \mu_e\} \neq \emptyset$. We also have that two connected branches $\widehat{\Omega}_{1j}$ and $\widehat{\Omega}_{1k}$ of $\widehat{\Omega}_1$ are consecutive or adjoint if and only if their counterparts, Ω_{ij} and Ω_{1k}, the connected branches of Ω_1, are consecutive or adjoint (assuming that all the connected branches are well defined). In other words, the translated dual plane holds all these properties of the dual plane of original problem if the dual variables satisfy $\lambda \geq \lambda_e$ and $\mu \geq \mu_e$. Similarly to (3.7),

$$\nabla \hat{\Psi}(\hat{\lambda}, \hat{\mu}) = \frac{1}{2} \begin{pmatrix} \|\hat{d}(\hat{\lambda}, \hat{\mu})\|^2 - \Delta^2 \\ \|A^T \hat{d}(\hat{\lambda}, \hat{\mu}) + c\|^2 - \xi^2 \end{pmatrix}. \tag{6.30}$$

From the KKT conditions of the original problem and those of problem $\widehat{P}$, we have

$$d(\lambda + \lambda_e, \mu + \mu_e) = \hat{d}(\lambda, \mu). \tag{6.31}$$

So, by (6.30) and (3.7), the following equation holds:

$$\nabla \Psi(\lambda + \lambda_e, \mu + \mu_e) = \nabla \hat{\Psi}(\lambda, \mu). \tag{6.32}$$

Moreover, the difference between the dual functions of these two problems is only a constant depending on λ_e and μ_e:

$$\Psi(\lambda + \lambda_e, \mu + \mu_e) = \hat{\Psi}(\lambda, \mu) + \frac{\lambda_e}{2} \Delta^2 + \frac{\mu_e}{2} (\xi^2 - \|c\|^2). \tag{6.33}$$

By definition (4.1), the equality (6.33) holds also for $(\lambda, \mu) \in \partial \Omega_0$. Hence,

$$(\lambda_+ - \lambda_e, \mu_+ - \mu_e) = \arg \max_{(\hat{\lambda}, \hat{\mu}) \in \widehat{\Omega}_0} \widehat{\Psi}(\hat{\lambda}, \hat{\mu}). \tag{6.34}$$

Now we are ready to prove the main result of this section.

THEOREM 6.3. *If S is a singleton, there exist multipliers (λ, μ) in the related region of S such that (λ, μ) are the corresponding Lagrangian multipliers of a global minimizer of* (1.1)–(1.3).

Proof. If there is a feasible point d_* such that the triple (λ_+, μ_+, d_*) solves (2.1)–(2.3), then the global solution of problem (3.1)–(3.3) is d_*.

Suppose $\hat{d}_g$ is a global solution of problem (6.21)–(6.23), and $(\hat{\lambda}_g, \hat{\mu}_g) \in \widehat{\Omega}_1$ are the corresponding Lagrangian multipliers. If we suppose $\lambda_e \neq 0$, then we have $\hat{\lambda}_g \neq 0$. Otherwise, since

$$ri\{\{\hat{\lambda} = \hat{\lambda}_g\} \cap \Omega_1\} = \emptyset, \tag{6.35}$$

$\widehat{H}(\hat{\lambda}_g, \hat{\mu}_g)$ is positive semidefinite, and then $\widehat{\Psi}$ reaches its maximum at two points $(\hat{\lambda}_g, \hat{\mu}_g)$ and $(\lambda_+ - \lambda_e, \mu_+ - \mu_e)$, which means that Ψ also reaches its maximum at two points. This contradicts the assumption that S is a singleton. If $\hat{\lambda}_g = 0$, then $\lambda_e = 0$ and $\lambda_e + \hat{\lambda}_g = 0$. Then we have

$$(\hat{\lambda}_g + \lambda_e)(\|d(\hat{\lambda}_g + \lambda_e, \hat{\mu}_g + \mu_e)\| - \Delta) = \hat{\lambda}_g(\|\hat{d}(\hat{\lambda}_g, \hat{\mu}_g)\| - \Delta) = 0 \tag{6.36}$$

and, similarly,

$$(\hat{\mu}_g + \mu_e)(\|A^T d(\hat{\lambda}_g + \lambda_e, \hat{\mu}_g + \mu_e) + c\| - \xi) = \hat{\mu}_g(\|A^T \hat{d}(\hat{\lambda}_g, \hat{\mu}_g) + c\| - \xi) = 0. \tag{6.37}$$

This implies that

$$\hat{\Phi}(\hat{d}_g) = \Phi(\hat{d}_g) + \frac{\lambda_e}{2}\Delta^2 + \frac{\mu_e}{2}(\xi^2 - \|c\|^2). \tag{6.38}$$

Moreover,

$$d(\lambda_g, \mu_g) = \hat{d}(\hat{\lambda}_g, \hat{\mu}_g) \tag{6.39}$$

follows (6.31) with (λ_g, μ_g) the corresponding Lagrangian multipliers. Furthermore, for any feasible $d \in \mathcal{R}^n$ of the original problem, we have that

$$\hat{\Phi}(\hat{d}_g) \leq \hat{\Phi}(d). \tag{6.40}$$

Expressions (6.38) and (6.40) imply that

$$\begin{aligned} \Phi(d(\lambda_g, \mu_g)) \quad &\leq \Phi(d) + \lambda_e(\|d\|^2 - \Delta^2) + \mu_e(\|A^T d + c\|^2 - \xi^2) \\ &\leq \Phi(d). \end{aligned} \tag{6.41}$$

The above inequality indicates that $d(\lambda_g, \mu_g)$ is a global solution of the original problem. This completes our proof. □

The above theorem illustrates the relation between the location of the Lagrangian multipliers and the maxima of the dual function on the region where the Hessian of the Lagrangian is positive semidefinite. From Definition 6.1, we can see that the choices of λ_e and μ_e are independent of each other. It can be seen that the larger λ_e and μ_e are, the smaller the related region is. Now we choose the minimal related region of S, i.e., we find the maxima (or supremum) of $\mathcal{L}$ and $\mathcal{M}$.

First, we consider the set $\mathcal{M}$. For $\varepsilon > 0$, we consider all the indices j such that

$$\Omega_{1j} \cap \{\mu_+ - \varepsilon < \mu < \mu_+\} \neq \emptyset. \tag{6.42}$$

By Lemma 3.3, this set can be divided into three cases.

Case 1. For sufficiently small $\varepsilon > 0$, there is no such j. Then, we may choose $\mu_e = \mu_+ - \varepsilon$ for sufficiently small $\varepsilon > 0$. Actually, the global solution lies in $\{\mu \geq \mu_+\}$.

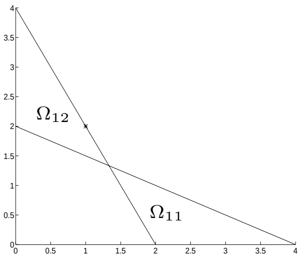


FIG. 6.1. (a) *of Example* 6.4.

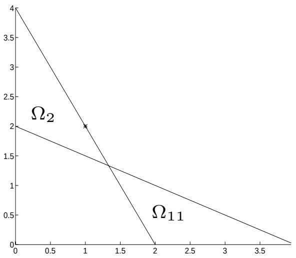


FIG. 6.2. (b) *of Example* 6.4.

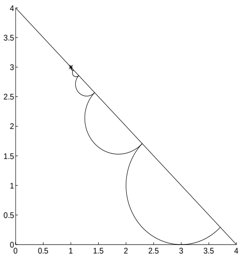


FIG. 6.3. (c) *of Example* 6.4.

Case 2. For sufficiently small $\varepsilon > 0$, there is exactly one such j. Then we easily see that the best choice of μ_e is $\mu_e = l_{\mu j}$.

Case 3. For any small $\varepsilon > 0$, there are infinitely many j such that (6.42) holds. Then μ_e can be chosen as any $l_{\mu j}$, which implies, actually, that the global solution lies in $\{\mu \geq l_{\mu j}\}$ for all j. Since the supremum of all these indices is μ_+, then the global solution lies in $\{\mu \geq \mu_+\}$ as in the first case.

These three cases show that the number of existent connected branches in $\{\mu < \mu_+\}$ is at least one. See Figures 6.1, 6.2, and 6.3.

Example 6.4. (a) Let

$$B = \operatorname{diag}(-2, -4), \qquad A = \operatorname{diag}\left(\frac{\sqrt{2}}{2}, \sqrt{2}\right).$$

The related region of $(4/3, 4/3)$ is $\Omega_{11} \cup \Omega_{12}$.

(b) Let

$$B = \operatorname{diag}(-2, -2, -4), \qquad A = \operatorname{diag}\left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, \sqrt{2}\right).$$

The related region of $(4/3, 4/3)$ is Ω_{11}, and the Hessian with Lagrangian multiplier in Ω_2 has two negative eigenvalues.

(c) We would like to show an example where there are infinitely many connected branches near one point. However, we could not find such an example or prove its nonexistence.

For the λ-direction, the case is slightly different. Considering Case 2, if there exists k such that $\Omega_{1k} \cap \{\lambda_+ - \varepsilon < \lambda < \lambda_+\} \neq \emptyset$, then the following statement may fail:

$$ri\{\{\lambda = \lambda_{\lambda k}\} \cap \Omega_1\} = \emptyset. \tag{6.43}$$

That is to say, $B + l_{\lambda k} I + \mu A A^T$ can be singular and have one negative eigenvalue in an interval $(\underline{\mu_s}, \overline{\mu_s})$. Therefore, we cannot set $\lambda_e = l_{\lambda k}$. In this case,

$$\det(B + l_{\lambda k} I + \mu A A^T) = 0 \tag{6.44}$$

for $\mu \in (\underline{\mu_s}, \overline{\mu_s})$. Since $\det(B + l_{\lambda k} I + \mu A A^T)$ is a polynomial of μ, the above relation implies that

$$\det(B + l_{\lambda k} I + \mu A A^T) = 0 \quad \text{for all}\, \mu \geq 0. \tag{6.45}$$

Thus for any $\lambda < \lambda_{\lambda k}$, $B + \lambda I + \mu AA^T$ has at least one negative eigenvalue for all μ, which implies that $l_{\lambda k'} = 0$ and $u_{\mu k'} = +\infty$ if there is a connected branch $\Omega_{1k'}$ such that $\Omega_{1k} \succ \Omega_{1k'}$. Therefore, there is at most one $\Omega_{1k'}$ such that $\Omega_{1k} \succ \Omega_{1k'}$. So we can set $\lambda_e = 0$ and there are at most two connected branches in the region $\{0 \le \lambda \le \lambda_+\}$.

For example, let

$$B = \operatorname{diag}(-5, -8, -3), \quad A = \begin{pmatrix} 1 & 0 \\ 0 & 2 \\ 0 & 0 \end{pmatrix}; \tag{6.46}$$

then $\lambda = 3$ is the singular line and $(\underline{\mu_s}, \overline{\mu_s}) = (\frac{5}{4}, 2)$. In section 7, we will present an example to show that the Hessian at the global solution may have one negative eigenvalue and can be singular.

From the above analyses, there are at most three connected branches of Ω_1 in the minimal related region of S. In the case that there are three connected branches in the minimal related region of S, we will have that (λ_+, μ_+) is an adjoint point of two connected branches of Ω_1. Moreover, there must be a singular line of the Lagrangian Hessian in the related region. Thus there are indices $k, k' \in \mathcal{K}$ such that

$$l_{\lambda k} = \lambda_+ = u_{\lambda k'} \tag{6.47}$$

$$u_{\mu k} = \mu_+ = l_{\mu k'}, \tag{6.48}$$

and

$$\det(B + l_{\lambda k'} I + \mu AA^T) \equiv 0 \quad \text{for all}\, \mu \ge 0. \tag{6.49}$$

Here, it follows from (6.47)–(6.48) that Ω_{1k} and $\Omega_{1k'}$ are two adjoint connected branches.

7. Distribution of local solutions.

7. Distribution of local solutions. In this section we show that, for the CDT problem, there may exist two local solutions whose corresponding Lagrangian multipliers lie in the same connected branch Ω_{1k} defined by (3.18) of the region where the Hessian of the Lagrangian possesses exactly one negative eigenvalue. It is also possible that the Hessian of the Lagrangian can have one negative eigenvalue and a zero eigenvalue. The following example shows that there may exist two local solutions in one connected branch of Ω_1.

Example 7.1. Let

$$B = \begin{pmatrix} -34/9 & \\ & -3 \end{pmatrix}, \quad A = \begin{pmatrix} 4/3 & \\ & 1 \end{pmatrix}, \quad g = \begin{pmatrix} 24 \\ 27 \end{pmatrix}, \quad c = \begin{pmatrix} -10 \\ 13 \end{pmatrix}, \tag{7.1}$$

$\Delta = 13$, and $\xi = 10$.

The global and local-nonglobal solutions of problem (3.1)–(3.3) are

$$d_g = \begin{pmatrix} 0 \\ -13 \end{pmatrix}, \quad d_l = \begin{pmatrix} 12 \\ -5 \end{pmatrix}, \tag{7.2}$$

with the Lagrangian multipliers $(\lambda_g, \mu_g) = (12/13, 9/5)$ and $(\lambda_l, \mu_l) = (28/51, 94/51)$, respectively. Then we easily see, from Figures 7.1 and 7.2, that the two points

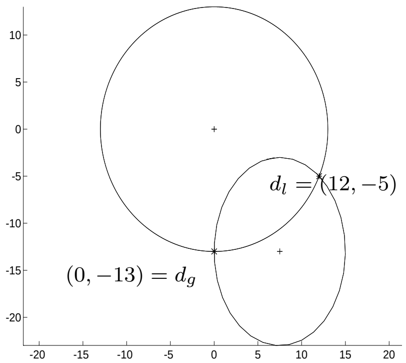


FIG. 7.1. *Primal space.*

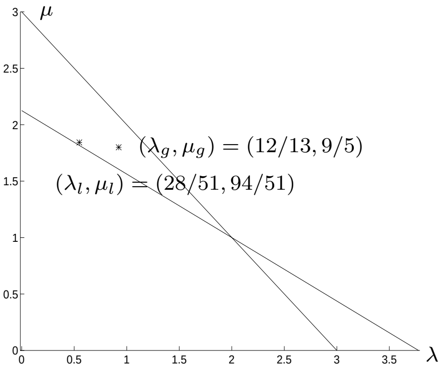


FIG. 7.2. *Dual space.*

$(12/13, 9/5)$, $(28/51, 94/51)$ are in the same connected branch. For the single-ball-constrained quadratic minimization, there is at most one local-nonglobal solution; see Martínez [10].

To show the distribution of local solutions, first we need a lemma.

LEMMA 7.2. *Assume that*

(a) $d(\lambda_1^*, \mu_1^*)$ *and* $d(\lambda_2^*, \mu_2^*)$ *are two stationary points of problem* (3.1)–(3.3)*, i.e., satisfy the KKT system,* (2.1)–(2.3)*;*

(b) $\lambda_2^* \geq \lambda_1^* \geq 0$ *and* $\mu_2^* \geq \mu_1^* \geq 0$*;*
then

$$\Phi(d(\lambda_2^*, \mu_2^*)) \leq \Phi(d(\lambda_1^*, \mu_1^*)). \tag{7.3}$$

The equality in (7.3) *holds if and only if*

$$\lambda_2^* = \lambda_1^* \tag{7.4}$$

and

$$(\mu_2^* - \mu_1^*)A^T(d(\lambda_2^*, \mu_2^*) - d(\lambda_1^*, \mu_1^*)) = 0. \tag{7.5}$$

Proof. For simplicity, we use the notations

$$H_1 = H(\lambda_1^*, \mu_1^*), \qquad H_2 = H(\lambda_2^*, \mu_2^*), \tag{7.6}$$

and

$$g_1 = g + \mu_1^* Ac, \qquad g_2 = g + \mu_2^* Ac. \tag{7.7}$$

Let $d_i = d(\lambda_i^*, \mu_i^*)$, $i = 1, 2$. Then we get

$$H_1 d_1 = -g_1, \qquad H_2 d_2 = -g_2. \tag{7.8}$$

Using (7.8), we have

$$\Phi(d_i) = -\frac{1}{2} {d_i}^T H_i d_i - \frac{\lambda_i^*}{2}\Delta^2 - \frac{\mu_i^*}{2}(\xi^2 - \|c\|^2) \tag{7.9}$$

for $i = 1, 2$. Hence

$$\begin{aligned}
&\Phi(d_1) - \Phi(d_2) \\
&= \tfrac{1}{2} d_2^T H_2 d_2 - \tfrac{1}{2} d_1^T H_1 d_1 + \tfrac{1}{2}(\lambda_2^* - \lambda_1^*)\Delta^2 + \tfrac{1}{2}(\mu_2^* - \mu_1^*)(\xi^2 - \|c\|^2) \\
&\geq \tfrac{1}{2} d_1^T g_1 - \tfrac{1}{2} d_2^T g_2 + \tfrac{1}{2}(\lambda_2^* - \lambda_1^*) d_1^T d_2 \\
&\qquad + \tfrac{1}{2}(\mu_2^* - \mu_1^*)((A^T d_1 + c)^T (A^T d_2 + c) - \|c\|^2) \\
&= \tfrac{1}{2} d_1^T g_1 - \tfrac{1}{2} d_2^T g_2 + \tfrac{1}{2} d_1^T (H_2 - H_1) d_2 + \tfrac{1}{2}(\mu_2^* - \mu_1^*)(d_1 + d_2)^T Ac \\
&= \tfrac{1}{2} d_1^T g_1 - \tfrac{1}{2} d_2^T g_2 - \tfrac{1}{2} d_1^T g_2 + \tfrac{1}{2} d_2^T g_1 + \tfrac{1}{2}(d_1 + d_2)^T (g_2 - g_1) \\
&= 0.
\end{aligned} \tag{7.10}$$

The equality in (7.3) holds if and only if the equality in (7.10) holds, which is equivalent to

$$(\lambda_2^* - \lambda_1^*)(\Delta^2 - d_1{}^T d_2) = 0 \tag{7.11}$$

and

$$(\mu_2^* - \mu_1^*)(\xi^2 - (A^T d_1 + c)^T (A^T d_2 + c)) = 0. \tag{7.12}$$

If $\lambda_2^* > \lambda_1^*$, (7.11) gives

$$d_1{}^T d_2 = \Delta^2 = d_1{}^T d_1 = d_2{}^T d_2, \tag{7.13}$$

which implies that $d_1 = d_2$. Then,

$$(\lambda_2^* - \lambda_1^*) d_1 + (\mu_2^* - \mu_1^*) A(A^T d_1 + c) = 0, \tag{7.14}$$

and any triple (λ, μ, d), with (λ, μ) in the straight line joining (λ_2^*, μ_2^*) and (λ_1^*, μ_1^*), is also a solution to the KKT system. Similar to equation (2.48) of Yuan [16],

$$d_1{}^T A(A^T d_1 + c) \geq 0. \tag{7.15}$$

Relations (7.14), (7.15) and $\mu_2^* \geq \mu_1^*$ give

$$(\lambda_2^* - \lambda_1^*)\|d_1\|^2 \leq 0, \tag{7.16}$$

which contradicts $\lambda_2^* > \lambda_1^*$ and (7.13). Therefore, (7.11) is true if and only if $\lambda_2^* = \lambda_1^*$.

If $\mu_2^* > \mu_1^*$, the equality in (7.12) gives

$$(A^T d_1 + c)^T (A^T d_2 + c) = \xi^2 = \|A^T d_1 + c\|^2 = \|A^T d_2 + c\|^2, \tag{7.17}$$

which implies $A^T d_1 + c = A^T d_2 + c$, i.e., $A^T(d_1 - d_2) = 0$. On the other hand, if $A(d_1 - d_2) = 0$, we have $A^T d_1 + c = A^T d_2 + c$, which implies

$$(A^T d_1 + c)^T (A^T d_2 + c) = \|A^T d_1 + c\|^2 = \|A^T d_2 + c\|^2. \tag{7.18}$$

Since $\mu_2^* > \mu_1^* \geq 0$, we have

$$\|A^T d_2 + c\| = \xi; \tag{7.19}$$

therefore $(A^T d_1 + c)^T (A^T d_2 + c) = \xi^2$. Thus we see that (7.12) is equivalent to (7.5). □

In Theorem 7.4, we have two points satisfying (b) of Lemma 7.2. Furthermore, Theorem 7.4 states that these two points are not local solutions at the same time while both the Hessians are not singular. This assumption cannot be moved, as the following example shows. In this example we show that there may exist a global solution of (3.1)–(3.3) with its Hessian having one negative eigenvalue and being singular.

Example 7.3. In this example, (3.1)–(3.3) have the global solution (λ_g, μ_g, d_g) satisfying $\lambda_g = 0$, both constraints are active at d_g and the Hessian $H(\lambda_g, \mu_g)$ has one negative eigenvalue.

$$B = \begin{pmatrix} -1 & \\ & -4 \end{pmatrix}, \quad A = \begin{pmatrix} 1 & \\ & 1 \end{pmatrix}, \quad g = \begin{pmatrix} 2 \\ -10 \end{pmatrix}, \quad c = \begin{pmatrix} 0 \\ 3 \end{pmatrix}, \tag{7.20}$$

$\Delta = \sqrt{2}$ and $\xi = \sqrt{5}$; then $d_g = (-1, -1)^T$ with its Lagrangian multipliers $(\lambda_g, \mu_g) = (0, 3)$ is the global solution of (3.1)–(3.3). Then the problem is

$$\min_{\bar{d} \in \mathcal{R}^3} \quad \frac{1}{2}\bar{d}^T \bar{B} \bar{d} + \bar{g}^T \bar{d} \tag{7.21}$$

subject to

$$\|\bar{d}\|^2 \le \Delta^2, \tag{7.22}$$

$$\|\bar{A}^T \bar{d} + \bar{c}\|^2 \le \xi^2, \tag{7.23}$$

where $\bar{d} = (d_1, d_2, d_3)^T$, $\bar{B} = \text{diag}(B, 0)$, $\bar{A}^T = (A^T, 0)$, $\bar{g}^T = (g^T, 0)$, and $\bar{c}^T = (c^T, 0)$. Its global solution is $\bar{d}_g = (-1, -1, 0)^T$, with the Lagrangian multipliers $(\lambda_g, \mu_g) = (0, 3)$ and the Hessian $H_g = \text{diag}(2, -1, 0)$. So the Hessian H_g at the global solution has one negative eigenvalue and is singular. However, for the single-ball-constrained quadratic minimization, there is no local solution where the Hessian has one negative eigenvalue and is singular (see Martínez [10]).

THEOREM 7.4. *It is not possible for two local solutions $d(\lambda_1^*, \mu_1^*)$ and $d(\lambda_2^*, \mu_2^*)$ of problem* (3.1)–(3.3) *to satisfy $\lambda_1^* > \lambda_2^* > 0$, $\mu_1^* \ge \mu_2^* > 0$, with $H(\lambda_i^*, \mu_i^*), i = 1, 2$, being a nonsingular matrix with exactly one negative eigenvalue.*

Proof. Suppose that (λ_1^*, μ_1^*) and (λ_2^*, μ_2^*) are the Lagrangian multipliers satisfying the above assumption.

Consider the following problem $(\tilde{P})$:

$$\min_{\tilde{d} \in \mathcal{R}^n} \frac{1}{2}\tilde{d}^T \tilde{B} \tilde{d} + \tilde{g}^T \tilde{d} \tag{7.24}$$

subject to

$$\|\tilde{D}^T \tilde{d} + \tilde{c}\| \le \tilde{\xi}, \tag{7.25}$$

where

$$\begin{aligned} \tilde{B} &= B + \lambda_2^* I + \mu_2^* A A^T, \\ \tilde{g} &= g + \mu_2^* A c, \\ \tilde{D} &= (\tau_1 I + \tau_2 A A^T)^{\frac{1}{2}}, \\ \tilde{c} &= \tau_2 \tilde{D}^{-1} A c, \\ \tilde{\xi} &= (\tilde{c}^T \tilde{c} - \tau_2 \xi^2 + \tau_2 c^T c + \tau_1 \Delta^2)^{\frac{1}{2}}, \end{aligned} \tag{7.26}$$

and $\tau_1 = \lambda_1^* - \lambda_2^*$ and $\tau_2 = \mu_1^* - \mu_2^*$.

Since $\lambda_1^* > \lambda_2^* > 0$ and $\mu_1^* \geq \mu_2^* > 0$, $\tilde{D}$ and $\tilde{c}$ in (7.26) are well defined. Suppose $\tilde{t}$ is the Lagrangian multiplier of $(\tilde{P})$; then the KKT system of $(\tilde{P})$ is

$$(\tilde{B} + \tilde{t}\tilde{D}\tilde{D}^T)\tilde{d}^* = -(\tilde{g} + \tilde{t}\tilde{D}\tilde{c}) \tag{7.27}$$

and

$$\tilde{t}(\|\tilde{D}^T\tilde{d}^* + \tilde{c}\| - \tilde{\xi}) = 0. \tag{7.28}$$

It can be verified that $d(\lambda_1^*, \mu_1^*)$ and $d(\lambda_2^*, \mu_2^*)$ are two stationary points of $(\tilde{P})$ with multipliers $\tilde{t} = 1$ and $\tilde{t} = 0$, respectively. Set

$$\phi(\tilde{t}) = \frac{1}{2}(\|\tilde{D}^T\tilde{d}(\tilde{t}) + \tilde{c}\|^2 - \tilde{\xi}^2). \tag{7.29}$$

By direct calculations and from the result given by Martínez [10], we have

$$\begin{aligned}
&\phi'(\tilde{t})|_{\tilde{t}=0} \\
&= -(\tilde{g} - \tilde{B}\tilde{D}^{-T}\tilde{c})^T\tilde{H}^{-1}\tilde{D}\tilde{D}^T\tilde{H}^{-1}\tilde{D}\tilde{D}^T\tilde{H}^{-1}(\tilde{g} - \tilde{B}\tilde{D}^{-T}\tilde{c})|_{\tilde{t}=0} \\
&= -(\tau_1 d_1^* + \tau_2 y_1^*)^T H(\lambda_1^*, \mu_1^*)^{-1}(\tau_1 d_1^* + \tau_2 y_1^*) \\
&= (\tau_1, \tau_2)\nabla^2\Psi(\lambda_1^*, \mu_1^*)\begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \\
&< 0,
\end{aligned} \tag{7.30}$$

where $\tilde{H} = \tilde{B} + \tilde{t}\tilde{D}\tilde{D}^T$, $d_1^* = d(\lambda_1^*, \mu_1^*)$, and $y_1^* = A(A^T d_1^* + c)$. Since there are two stationary points with the Hessian one negative eigenvalue, from Lemma 4.3 in Martínez [10], (7.30) is a strict inequality.

Let $\delta d = d_2^* - d_1^*$, where $d_2^* = d(\lambda_2^*, \mu_2^*)$; then

$$H(\lambda_1^*, \mu_1^*)\delta d = \tau_1 d_1^* + \tau_2 y_1^* \tag{7.31}$$

follows by direct calculations. Hence

$$\begin{aligned}
\delta d^T H(\lambda_1^*, \mu_1^*)\delta d \quad &= \delta d^T H(\lambda_1^*, \mu_1^*)H(\lambda_1^*, \mu_1^*)^{-1}H(\lambda_1^*, \mu_1^*)\delta d \\
&= -(\tau_1, \tau_2)\nabla^2\Psi(\lambda_1^*, \mu_1^*)\begin{pmatrix} \tau_1 \\ \tau_2 \end{pmatrix} \\
&> 0.
\end{aligned} \tag{7.32}$$

Define

$$S^1 = \left\{ d \mid \langle \frac{d}{\|d\|}, \frac{\delta d}{\|\delta d\|} \rangle \geq 1 - \varepsilon \right\} \cup \{0\} \tag{7.33}$$

and

$$S^2 = \left\{ d \mid \langle \frac{d}{\|d\|}, \frac{\delta d}{\|\delta d\|} \rangle \leq 1 - \frac{1}{2}\varepsilon \right\} \cup \{0\}, \tag{7.34}$$

where

$$\varepsilon^{\frac{1}{2}} = \frac{\delta d^T H_* \delta d}{8\|\delta d\|^2\|H_*\|} \tag{7.35}$$

and $H_* = H(\lambda_1^*, \mu_1^*)$. Since H_* has one negative eigenvalue and $\delta d \neq 0$, (7.35) is well defined. It is easy to verify that S^1 and S^2 are closed sets. Moreover, if $d \in S^1$,

$$\left\| \frac{d}{\|d\|} - \frac{\delta d}{\|\delta d\|} \right\| \leq 2\varepsilon \tag{7.36}$$

and

$$\begin{aligned} &\tfrac{1}{\|d\|^2} d^T H_* d \\ &= \left(\tfrac{d}{\|d\|} - \tfrac{\delta d}{\|\delta d\|} + \tfrac{\delta d}{\|\delta d\|}\right)^T H_* \left(\tfrac{d}{\|d\|} - \tfrac{\delta d}{\|\delta d\|} + \tfrac{\delta d}{\|\delta d\|}\right) \\ &\geq \tfrac{\delta d^T H_* \delta d}{\|\delta d\|^2} - 2\sqrt{2\varepsilon}\|H_*\| - 2\varepsilon\|H_*\| \\ &\geq 0, \end{aligned} \tag{7.37}$$

while if $d \in S^2$,

$$d^T \left(I - \frac{\delta d \delta d^T}{\|\delta d\|^2}\right) d \geq 0. \tag{7.38}$$

Let

$$\theta_1 = \min\{d^T H_* d \mid \|d\| = 1, d \in S^1\}, \tag{7.39}$$

$$\theta_2 = \min\left\{d^T \left(I - \frac{\delta d \delta d^T}{\|\delta d\|^2}\right) d \mid \|d\| = 1, d \in S^2\right\}, \tag{7.40}$$

and $\theta = \min\{\theta_1, \theta_2\}$. It can be verified that $\theta > 0$ since S^1, S^2, and $\{d \mid \|d\| = 1\}$ are all closed sets. From Lemma 2.3 of Yuan [16], there are $\alpha_1 \geq 0$ and $\alpha_2 \geq 0$ with $\alpha_1 + \alpha_2 = 1$ such that

$$\alpha_1(H_* - \theta I) + \alpha_2 \left(I - \theta I - \frac{\delta d \delta d^T}{\|\delta d\|^2}\right) \tag{7.41}$$

is positive semidefinite. Since neither $(H_* - \theta I)$ nor $(I - \theta I - \frac{\delta d \delta d^T}{\|\delta d\|^2})$ is positive semidefinite, $\alpha_1 \neq 0$ and $\alpha_2 \neq 0$. Let $m_0 = \frac{\alpha_2}{\alpha_1}$; the matrix $H_* + m_0(I - \frac{\delta d \delta d^T}{\|\delta d\|^2})$ is positive semidefinite. Now the problem (P_p)

$$\min_d \quad \frac{1}{2} d^T B_p d + g_p^T d \tag{7.42}$$

subject to

$$\|d\| \leq \Delta, \tag{7.43}$$

$$\|A^T d + c\| \leq \xi, \tag{7.44}$$

where $B_p = B + m_0(I - \frac{\delta d \delta d^T}{\|\delta d\|^2})$ and $g_p = g - (B_p - B)d_1^*$, possesses two global solutions

$$(\lambda_1^*, \mu_1^*, d_1^*) \text{ and } (\lambda_2^*, \mu_2^*, d_2^*), \tag{7.45}$$

both with positive semidefinite Hessian. That their objective function value must be the same contradicts Lemma 7.2. □

Remark. From Theorem 7.4, all the local solutions with the multipliers in $\mathrm{int}\Omega_1$ are permuted in the way the connected branches of Ω_1 are. As Example 7.1 shows, in one connected branch of Ω_1, there may exist a global and a local solution simultaneously. It is important to give the characteristic of the global solution and hence construct an algorithm with which to find the global solution instead of the local solution.

8. Conclusions and future work. We investigated the dual plane of the CDT subproblem which is related to a matrix pencil with two parameters. We also extended the general Lagrangian dual function from the region where the Lagrangian Hessian is positive definite to its closure. The location and permutation of the Lagrangian multipliers were studied and the differences between the CDT subproblem and the trust region subproblem were presented.

We have given various results on the locations of the corresponding Lagrange multipliers. These results may be used in the construction of numerical methods for the CDT subproblem based on identifying the multipliers. The main result shows that the Lagrangian multipliers corresponding to a global minimizer of the CDT problem locates in finitely many, often two or three, connected branches of Ω_1 if there is no global minimizer with the Hessian of the Lagrangian positive semidefinite. Roughly speaking, if we define the degree of the nonpositive definite of a symmetric matrix as the number of its negative eigenvalues, an important property of the CDT problem is that its complexity is not related to the nonpositive degree of the Hessian.

For some trust region methods, the trial step can be any sufficient descent feasible direction instead of the global minimizer or a local minimizer. Thus, it is interesting to search for efficient algorithms to compute approximate global minimizers of the CDT subproblem in the primal space.

Acknowledgment. We would like to thank two anonymous referees for their valuable comments which improved the paper.

REFERENCES

[1] R. H. Byrd and R. B. Schnabel, *Approximate solution of the trust region problem by minimization over two-dimensional subspace*, Math. Programming Ser. A, 40 (1988), pp. 247–263.

[2] M. R. Celis, J. E. Dennis, and R. A. Tapia, *A trust region strategy for nonlinear equality constrained optimization*, in Numerical Optimization, P. T. Boggs, R. H. Byrd, and R. B. Schnabel, eds., SIAM, Philadelphia, 1985, pp. 71–82.

[3] Z. Chen, *Some algorithms for a class of CDT subproblems*, in Operations Research and Its Applications, Lecture Notes in Oper. Res. 1, D. Z. Du, X. Zhang, and W. Wang, eds., World Publishing Corporation, Beijing, 1996, pp. 108–114.

[4] J. E. Dennis and K. A. Williamson, *A robust trust region algorithm for nonlinear programming*, Technical Report, Math. Sci. Dept., Rice University, Houston, TX, 1991.

[5] J. E. Dennis, Jr., M. El-Alem, and M. C. Maciel, *A global convergence theory for general trust-region-based algorithms for equality constrained optimization*, SIAM J. Optim., 7 (1997), pp. 177–207.

[6] J. G. Ecker and R. D. Niemi, *A dual method for quadratic programs with quadratic constraints*, SIAM J. Appl. Math., 28 (1975), pp. 568–576.

[7] M. M. El-Alem and R. A. Tapia, *Numerical experience with a polyhedral-norm CDT trust region algorithm*, J. Optim. Theory Appl., 85 (1995), pp. 575–591.

[8] M. Y. Fu, Z. Q. Luo, and Y. Y. Ye, *Approximation Algorithms for Quadratic Programming*, University of Iowa, http://dollar.biz.uiowa.edu/col/ye/paper.html (1996).

[9] M. Heinkenschloss, *On the solution of a two ball trust region subproblem*, Math. Programming, 64 (1994), pp. 249–276.

[10] J. M. Martínez, *Local minimizers of quadratic functions on Euclidean balls and spheres*, SIAM J. Optim., 4 (1994), pp. 159–176.

[11] J. M. Martínez and S. A. Santos, *A trust region strategy for minimization on arbitrary domains*, Math. Programming, 68 (1995) pp. 267–302.

[12] S. Mehrotra and J. Sun, *A method of analytic centers for quadratically constrained convex quadratic programs*, SIAM J. Numer. Anal., 28 (1991), pp. 529–544.

[13] J. M. Peng and Y. Yuan, *Optimality conditions for the minimization of a quadratic with two quadratic constraints*, SIAM J. Optim., 7 (1997), pp. 579–594.

[14] E. Phan-huy-Hao, *Quadratically constrained quadratic programming: Some applications and a method for solution*, Zeitschrift Oper. Res., 26 (1982), pp. 105–119.

[15] M. J. D. Powell and Y. Yuan, *A trust region algorithm for equality constrained optimization*, Math. Programming, 49 (1991), pp. 189–211.

[16] Y. Yuan, *On a subproblem of trust region algorithms for constrained optimization*, Math. Programming, 47 (1990), pp. 53–63.

[17] Y. Yuan, *A dual algorithm for minimizing a quadratic function with two quadratic constraints*, J. Comput. Math., 9 (1991), pp. 348–359.

[18] Y. Zhang, *Computing a Celis-Dennis-Tapia trust-region step for equality constrained optimization*, Math. Programming, 55 (1992), pp. 109–124.

SIAM J. Optim.
Vol. 10, No. 2, pp. 384–404

STOCHASTIC COMPARISON ALGORITHM FOR DISCRETE OPTIMIZATION WITH ESTIMATION[*]

WEI-BO GONG[†], YU-CHI HO[‡], AND WENGANG ZHAI[§]

Abstract. In this paper we study a class of discrete optimization problems, where the objective function for a given configuration can be expressed as the expectation of a random variable. In such problems, only samples of the random variables are available for the optimization process. An iterative algorithm called the stochastic comparison (SC) algorithm is developed. The convergence of the SC algorithm is established based on an examination of the quasi-stationary probabilities of a time-inhomogeneous Markov chain. We also present some numerical experiments.

Key words. discrete optimization, random search, time-inhomogeneous Markov chain

AMS subject classifications. 60J10, 65C05, 90C27

PII. S1052623495290684

1. Introduction.

Discrete optimization plays an increasingly important role in system design and analysis. Examples include the configuration design of distributed computer systems, routing design in communication networks, and scheduling problems in communication networks, production systems, and transportation systems. The common features of these discrete optimization problems are (1) the number of feasible alternatives increases exponentially with the system size and (2) it is usually not possible to obtain an analytic expression for the objective function being optimized. To tackle the first difficulty, one has to relax the goal. That is, instead of asking for the true optimal solution, one must be satisfied with algorithms that provide a good design with high probability (rather than 100% certainty) in realistic time. Simulated annealing (SA) is one such algorithm, and it has been successful in solving many practical problems (see, for example, [1]). However, for SA to work well, it needs a good neighborhood structure and accurate estimates of the objective function values. Experiments show that a poor choice of a neighborhood structure and the use of rough estimates of the objective function values can lead to poor performance (cf. [14], [20], [7]). When one does not have an analytic expression for the objective function, one must usually resort to Monte Carlo simulation to obtain estimates. The quality of these estimates depends on the length of the simulation run. Much theoretical work has been done to evaluate the performance of SA when the objective function is noisy. Two types of errors have been classified: instantaneous error (the difference between the true and the observed objective functions) and accumulated error (the sum of a sequence of the instantaneous errors). Grover [15] presented an early analysis on the effect of the instantaneous errors. Durand and White [8] analyzed equilibrium properties for bounded instantaneous errors. Gelfand and Mitter [9], [10] showed that, under certain conditions, slowly decreasing state-independent Gaussian

[*] Received by the editors August 21, 1995; accepted for publication (in revised form) July 6, 1998; published electronically December 22, 1999. This work was supported in part by National Science Foundation grants CCR-9506764, NCR-9523807, and EEC-9527422, by U.S. Army Research Office contracts DAAH04-95-0148 and DAAG55-98-1-0042, and by Air Force Rome Laboratory contract F30602-97-C-0125.

http://www.siam.org/journals/siopt/10-2/29068.html

[†] Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA 01003 (gong@ecs.umass.edu).

[‡] Division of Applied Sciences, Harvard University, Cambridge, MA 02138 (ho@hrl.harvard.edu).

[§] Ascend Communications, Inc., 1 Robbins Road, Westford, MA 01886 (zhai@casc.com).

noise does not affect asymptotic convergence. Romeo and Sangiovanni-Vincentelli [23] gave conditions on the errors such that transition probabilities of a noisy annealing process converge to those of the accurate process as $T \to 0$. Greening [13] proved a slightly more general result by relating the error range and the cooling temperature. While these studies show that SA can work with Monte Carlo estimates, it would take extensive computer time to simulate a complex system to get sufficiently accurate estimates. Another algorithm, proposed by Yan and Mukai [27], takes this point into consideration. We call their algorithm the stochastic ruler (SR) algorithm. Yan and Mukai proved that, under fairly general conditions, the sequence of feasible alternatives visited by the SR algorithm converges, in probability, to the optimum. In the SR algorithm the estimated objective function is compared with a random number called the stochastic ruler. The original SR algorithm used a random variable uniformly distributed over an interval $[a, b]$ as the ruler. This requires some a priori knowledge regarding the range of the objective function values to determine an appropriate interval size for the ruler. Too big a ruler reduces the sensitivity of the algorithm and slows down the optimization process. Conversely, too small a ruler may not be able to distinguish the best solutions from other good solutions that fall outside the range of the ruler.

In this paper, we propose a new algorithm that we call the stochastic comparison (SC) algorithm. The algorithm is designed for discrete optimization problems with large unstructured search spaces. It has its roots in the ranking and selection procedures well known in statistics (see, for example, [16], [25], [11]). The typical ranking and selection procedures, however, deal only with small search spaces and are not concerned with the convergence of the type of iterative procedure that we propose. Bandit theory also deals with similar problems, but again, such theory deals with much smaller problems (see, for example, [21], [26], and references therein). Our work belongs to the general class of random search algorithms (see, e.g., [24, Chap. 5] and references therein). An initial, much shorter version of this work was presented in [12].

The remainder of the paper is organized as follows. In section 2, we present the discrete optimization with estimation problem and discuss the existing SA and SR algorithms. Section 3 proposes the SC algorithm and establishes an equivalence result to pave the way for the convergence proof. Section 4 analyzes the time-homogeneous Markov chain generated by the SC algorithm when the "testing sequence" is constant. The convergence proof of the SC algorithm via the analysis of the underlying time-inhomogeneous Markov chain is given in section 5. Numerical examples are presented in section 6. We close with a brief discussion in section 7.

2. The problem and existing algorithms.

2.1. The optimization problem.

Because we were originally motivated by a computer system configuration design problem, we will call each element of the search space a *configuration.* The optimization problem is to find a configuration, i (not necessarily unique), from a discrete finite set of alternatives, S, that minimizes an objective function, $g(i)$, i.e.,

$$\min_{i \in S} \{g(i)\}$$

with $g : S \to R$ and $S = \{1, 2, \ldots, s\}$. That is, we wish to find a global optimal configuration $i \in S^*$, where S^* is the global optimal set, given by

$$S^* = \{i \in S \mid \; g(i) \leq g(j) \; \forall j \in S\}. \tag{1}$$

Let us denote the cardinality of the solution space S by $|S|$ (note that $|S| = s$). A key assumption is that $|S|$ is very large. We will also assume, as is often the case in practice, that we do not have an analytic expression for the objective function $g(i)$ and that it can only be evaluated via Monte Carlo simulation. Let $H(i)$ be a sample estimate of $g(i)$ and assume that $g(i) = E[H(i)]$ $\forall i \in S$ (i.e., $H(i)$ is unbiased). Further, assume that the variance of the estimate is finite, i.e., that $E[H(i) - E[H(i)]]^2 < \infty$ $\forall i \in S$.

The unbiasedness assumption is generally required in order to prove various theoretical properties (e.g., convergence) of stochastic optimization schemes. This is the case for the SR algorithm [27], which largely motivated our work here. Unbiasedness is satisfied, for example, when the simulation system used to obtain $H(i)$ has a regenerative structure. In that case, regenerative simulation provides unbiased, independent, identically distributed (i.i.d.) samples of the objective function (see, for example, [2, p. 95]). More concretely, consider a GI/GI/1 queue, where the quantities belong to single busy periods and are i.i.d. random variables. However, performance measures estimates are often given as ratios of such random variables. For example, the estimate of the mean waiting time in a single server queue is the ratio of the expected total waiting time of customers within a busy period to the expected number of customers during that busy period. In this case, a regenerative estimate of the ratio is biased. The following technique can be employed to compare two competing configurations i and j in our optimization scheme.

Suppose the ratio estimator for configuration i is $h(i) = \frac{R_i}{Y_i}$ and the estimator for configuration j is $h(j) = \frac{R_j}{Y_j}$. Instead of comparing $h(i)$ to $h(j)$ directly, we should instead compare $R_i Y_j$ to $R_j Y_i$. Since the regenerative estimates for R_i, R_j, Y_i, and Y_j are unbiased, and R_i and Y_j are independent (as are R_j and Y_i), we have $E[R_i Y_j] = E[R_i]E[Y_j]$ and $E[R_j Y_i] = E[R_j]E[Y_i]$, and consequently, $E[R_i Y_j] < E[R_j Y_i]$ is equivalent to $\frac{E[R_i]}{E[Y_i]} < \frac{E[R_j]}{E[Y_j]}$. In other situations, bias reduction techniques, such as jackknifing (cf. [2]) can be used to reduce the bias. Finally, experimental results demonstrate that algorithms developed under unbiasedness assumptions are nevertheless useful in practice with slightly biased estimates.

2.2. SA and SR algorithms. The SA and SR algorithms are two generic random search algorithms designed to solve the discrete optimization problem (1). To briefly describe these algorithms, we first introduce the following standard definitions and assumption.

DEFINITION 2.1. *For each $i \in S$, there exists a subset $N(i)$ of $S \setminus \{i\}$, which is called the* set of neighbors *of i.*

DEFINITION 2.2. *A function R: $S \times S \to [0, 1]$ is said to be a* generating probability *for S and N if*

1. *$R(i, j) > 0 \Longleftrightarrow j \in N(i)$ and*
2. *$\sum_{j \in S} R(i, j) = 1$ for $i, j \in S$.*

ASSUMPTION 2.1. *For any pair $(i, j) \in S \times S$, j is reachable from i; i.e., there exists a finite sequence $\{n_m\}_{m=0}^{\ell}$ for some ℓ, such that $i_{n_0} = i, i_{n_\ell} = j$, and $i_{n_{m+1}} \in N(i_{n_m})$ for $m = 0, 1, 2, \ldots, \ell - 1$.*

For SA, it has been proved [22] that a real sequence $\{T_k\}_{k=0}^{\infty}$ satisfying $T_k = \frac{\gamma}{\log(k+k_0+1)}$, $k = 0, 1, 2, \ldots$, for some positive numbers γ and k_0 will guarantee that the algorithm will converge to a global optimum. T_k is called the *temperature* at the kth iteration of the sequence and $\{T_k\}_{k=0}^{\infty}$ is called the *cooling schedule*. For SR, it has been proved [27] that an integer sequence $\{M_k\}_{k=0}^{\infty}$ satisfying $M_k = \lfloor c \log_\sigma(k+k_0+1) \rfloor, k =$

$0, 1, 2, \ldots$ ($\lfloor \xi \rfloor$ denotes the greatest integer that is smaller than or equal to ξ), for some positive numbers c, σ, and k_0 will guarantee that the algorithm will converge to a global optimum. We call M_k the kth *testing number* and $\{M_k\}_{k=0}^{\infty}$ the *testing sequence.* Note that to guarantee the convergence of these algorithms, the temperature or the testing number has to change as slowly as a logarithmic function of the iteration number k [17], [22], [27].

The standard way to prove the above results is to note that the sequence of configurations visited by the algorithm forms a Markov chain. That is, if we let X_k denote the configuration visited by the algorithm at the kth iteration, then $\{X_k\}_{k=1}^{\infty}$ is a Markov chain. Then, to prove the convergence of the algorithm, all one has to do is to show that the probability vector $e(k) = [e_1(k) \ldots e_s(k)]$ with $e_i(k) \overset{\triangle}{=} \Pr\{X_k = i\}$ for $i = 1, \ldots, s$ converges to an *optimal probability vector* $e^* = [e_1^* \ldots e_s^*]$, i.e., that

$$e_i^* > 0 \quad \text{for} \quad i \in S^*,$$

$$e_i^* = 0 \quad \text{for} \quad i \notin S^*.$$

This is typically done using the theory of weak and strong ergodicity of time-inhomogeneous Markov chains [1], [17], [19], [22].

For SA, the one-step transition probabilities of the Markov chain $\{X_k\}$ for a given temperature T are

$$P_{ij}(T) = \begin{cases} R(i,j)\min[1, e^{-\{g(j)-g(i)\}/T}] & \text{if} \quad j \in N(i); \\ 1 - \sum_{n \in N(i)} P_{in}(T) & \text{if} \quad j = i; \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Observe that $P_{ij}(T) > 0$, even when $g(j) > g(i)$. In other words, SA visits poor configurations with positive probability in order to jump out of local minima. However, it does so less frequently as the optimization proceeds, so as to mimic the physical "annealing" process in steel, which provided the inspiration for the algorithm.

The SR algorithm works by comparing sample estimates of the objective function to an a priori chosen random variable $\Theta(a, b)$. $\Theta(a, b)$, the stochastic ruler, has a uniform distribution over $[a, b]$, the range of "good" values of the sample estimates. Defining

$$P(i, a, b) = P[H(i) \le \Theta(a, b)],$$

then the one-step state transition probabilities for a given testing number M are given by

$$P_{ij}(M) = \begin{cases} R(i,j)\{P(j,a,b)\}^M & \text{if} \quad j \in N(i); \\ 1 - \sum_{n \in N(i)} P_{in}(M) & \text{if} \quad j = i; \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

That is, $P_{ij}(M)$ is the probability that the search goes from configuration i to configuration j when the testing number is M.

2.3. Practical issues and motivation. The development of the SC algorithm was motivated by certain practical limitations of the existing SA and SR algorithms. First, our experiments show that SA does not converge when the objective function estimates are noisy. This suggests that SA needs long simulation runs to get improved estimates. SR is more robust with respect to estimation error. This is a consequence of the robustness of Monte Carlo estimates for order statistics. As a result, algorithms based on order statistics can significantly reduce the required simulation time [18]. Second, the SA algorithm must often visit poor configurations so as not to overlook the possibility that there might be a very good configuration surrounded by poor configurations. This, however, is only beneficial when one can identify a good neighborhood structure, which is often a difficult task. Similarly, for the SR algorithm, one has to choose the size of the stochastic ruler, which can also be difficult in practice, where one often has very limited information about the configuration space.

Motivated by the shortcomings of the SA and SR algorithms, we propose the SC algorithm. The algorithm is designed for large-scale optimization problems driven by noisy estimates of the objective function. Like the SR algorithm, the SC algorithm exploits the robustness of order statistics to deal with the noisy estimates. It differs from the SR algorithm in that, instead of comparing candidate configurations to a stochastic ruler, it directly compares the current configuration to a candidate configuration. The SC algorithm, therefore, does not require any knowledge whatsoever about the structure of the search space. This does mean, however, that convergence is only guaranteed when any configuration (in the whole configuration space) can be reached from any other in one step. In other words, we have eliminated the neighborhood structure for the sake of convergence.

Our SC algorithm is actually a stochastic version of the crude random search algorithm proposed by Brooks [3], the main difference being that Brooks's algorithm assumes that the exact values of the objective function are available, and it therefore has a straightforward convergence proof. While a good neighborhood structure can speed up the search process of algorithms like SR, a poor neighborhood structure can hurt performance. Eliminating the use of a neighborhood structure is not a limitation for many practical optimization problems, like the computer configuration design problem mentioned earlier, since for such problems a good neighborhood structure is generally very difficult to identify. While it is possible to eliminate the neighborhood structure in the SR algorithm (by taking the whole configuration space as the neighborhood), it is easy to show that when this is done, the SC algorithm is more efficient than the SR algorithm, at least when the objective function estimation error is small. This is because, when the objective function is known exactly, the SC algorithm becomes the usual crude random search, while the SR algorithm still compares the objective function values against a *random* ruler to decide when to move. In this case, the SR algorithm is not as efficient as the SC algorithm; when there is no uncertainty, there is no need to introduce randomness in comparing alternative configurations.

One may question the effectiveness of searching a huge configuration space without exploiting the neighborhood structure that may be present. The following calculation, however, demonstrates that blind random search can be very effective, especially when the noise in the estimated objective function is small. Suppose that the objective function $g(i)$ $\forall i \in S$ is known exactly. At each iteration a blind random search compares the current configuration against a configuration chosen uniformly from the configuration space, i.e., $R(j,i) = 1/(|S|-1)$ $\forall j \in S \setminus \{i\}$. The configuration giving better performance becomes the new "current configuration." If the cardinality of the search space, $|S|$, is sufficiently large, then after k iterations of the algorithm, the probability of finding a configuration whose performance is in the top $\alpha\%$ (i.e., $(1-\alpha)\%$ of the configurations give worse performance) is approximately

$$P_\alpha(k) \approx 1 - \left(1 - \frac{1}{|S|-1}\right)^{0.01\alpha|S|k} \approx 1 - \exp(-0.01\alpha k).$$

The derivation of this approximation is straightforward. Note that $\frac{1}{|S|-1}$ is the probability of choosing one of the top $\alpha\%$ configurations, thus $(1-\frac{1}{|S|-1})$ is the probability that this configuration is not chosen, and $(1-\frac{1}{|S|-1})^{0.01\alpha|S|}$ is the approximate probability that none of the top $\alpha\%$ configurations are chosen. Therefore, after k iterations of the algorithm, the probability that the current configuration is not in the top $\alpha\%$ is $(1-\frac{1}{|S|-1})^{0.01\alpha|S|k}$, and $1-(1-\frac{1}{|S|-1})^{0.01\alpha|S|k}$ is the probability that the current configuration is in the top $\alpha\%$.

As an example, suppose that the size of the configuration space $|S|$ is large and that we are looking for a configuration whose performance is in the top $\alpha = 0.5\%$ of all configurations. After only $k = 1000$ iterations of a blind random search, the probability that we have found such a configuration is $P_{0.5}(1000) \approx 1 - e^{-5} = 0.99326$—a virtual certainty. This suggests that, for certain applications, a crude random search (i.e., without neighborhood structure) can indeed be very useful. This property of the blind random search algorithm is the inspiration for the SC algorithm to be presented next.

3. The stochastic comparison algorithm.

3.1. An alternative problem. A key feature of the SC algorithm is that we do not try to minimize $E[H(i)]$ directly. Instead, we try to maximize an alternative objective function, which we call the *sigma-probability* function. The sigma-probability function, which we denote by $sp(\cdot)$, is defined for each configuration $i \in S$ by

$$sp(i) = \textstyle\sum_{j \in S\setminus\{i\}} \Pr\left[H(i) < H(j)\right].$$

The SC algorithm, therefore, seeks to identify a member of the optimum set $\bar{S}^*$, where

$$\bar{S}^* = \{i \in S \mid \ sp(i) \geq sp(j)\ \forall j \in S\}.$$

Let $W_i = H(i) - g(i)$ denote the estimation error, and assume that it satisfies the following conditions.

Assumption 3.1.

1. *$\{W_i, i \in S\}$ are i.i.d.*

2. *Each $W_i, i \in S$, has a symmetric continuous probability density function with a zero mean.*

Given Assumption 3.1, we will now show that the optimization problem above is equivalent to the original discrete optimization problem in the sense that $\bar{S}^* = S^*$.

Theorem 3.1. *Under Assumption* 3.1*,*

$$E[H(i)] < E[H(j)] \iff sp(i) > sp(j) \qquad \forall i \neq j,\ i,j \in S.$$

Proof. Note that we can express $H(i)$ as $H(i) = g(i) + W_i$, where $g(i) = E[H(i)]\ \forall i \in S$. Let $c = g(i) - g(j)$.

By Assumption 3.1, the differences $W_j - W_i$, $W_k - W_i$, and $W_k - W_j\ \forall k \in S\setminus\{i,j\}$ are identically distributed random variables with a symmetric continuous density function. Let ξ be a random variable with the same density. Then we have

$$\begin{aligned} sp(i) &= \Pr[H(i) < H(j)] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[H(i) < H(k)] \\ &= \Pr[W_j - W_i > g(i) - g(j)] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[W_k - W_i > g(i) - g(k)] \\ &= \Pr[\xi > c] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[\xi > c + g(j) - g(k)] \end{aligned}$$

and

$$\begin{aligned} sp(j) &= \Pr[H(j) < H(i)] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[H(j) < H(k)] \\ &= \Pr[W_j - W_i < g(i) - g(j)] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[W_k - W_j > g(j) - g(k)] \\ &= \Pr[\xi < c] + \textstyle\sum_{k\in S\setminus\{i,j\}} \Pr[\xi > g(j) - g(k)]. \end{aligned}$$

Since $\Pr[\xi < c] = 1 - \Pr[\xi \geq c] = 1 - \Pr[\xi > c]$, we have

$$sp(i) - sp(j) = 2\Pr[\xi > c] - 1 + \sum_{k\in S\setminus\{i,j\}} \{\Pr[\xi > c + g(j) - g(k)] - \Pr[\xi > g(j) - g(k)]\}.$$

Note that the right-hand side of the above equation is a monotonic decreasing function of c, and that $sp(i) - sp(j) = 0$ when $c = 0$. Therefore, we have $sp(i) > sp(j) \iff E[H(i)] < E[H(j)]$. □

The following corollary is an immediate consequence of Theorem 3.1.

COROLLARY 3.1. *Given $H(i),\ H(j)\ \forall i \neq j,\ i,j \in S$, we have under Assumption*

$$E[H(i)] < E[H(j)] \iff \Pr[H(i) < H(j)] > \Pr[H(j) < H(i)].$$

Proof. Let $S = \{i, j\}$ and apply Theorem 3.1. □

We will find the following lemma useful later in the paper.

LEMMA 3.1. *Given $H(i),\ H(j),\ H(k)\ \forall i \neq j, j \neq k, k \neq i,\ i,j,k \in S$, and Assumption 3.1, the following two conditions are equivalent:*

1. $E[H(i)] < E[H(j)] < E[H(k)]$;
2. $\Pr[H(i) < H(k)] > \Pr[H(j) < H(k)]$ *and* $\Pr[H(i) < H(k)] > \Pr[H(i) < H(j)]$.

Proof. Using the same notation as in the proof of Theorem 3.1, we have

$$\Pr[H(i) < H(k)] = \Pr[W_i - W_k < g(k) - g(i)] = \Pr[\xi < g(k) - g(i)],$$

$$\Pr[H(j) < H(k)] = \Pr[W_j - W_k < g(k) - g(j)] = \Pr[\xi < g(k) - g(j)],$$

$$\Pr[H(i) < H(j)] = \Pr[W_i - W_j < g(j) - g(i)] = \Pr[\xi < g(j) - g(i)].$$

The results of the lemma follow immediately. □

To guarantee that the SC algorithm converges, we require that any configuration be reachable from any other in one step. That is, we require the following.

ASSUMPTION 3.2. *$R(i,j) > 0\ \forall i,j \in S$ and $i \neq j$.*

3.2. Implementation of the SC algorithm. Let S be the configuration space, k the iteration number, X_k the configuration accepted at iteration k, and M_k the testing number at iteration k. Similar to the SR algorithm, M_k is the number of sample estimates of H that must be obtained for a configuration at the kth iteration. For some configuration i, denote these samples by $H_\ell(i)$ for $\ell = 1, \ldots, M_k$.

The stochastic comparison algorithm.
Data: R, $\{M_k\}$, $i_0 \in S$.
Step 0: Set $X_0 = i_0$ and $k = 0$.
Step 1: Given $X_k = i$, choose a candidate Z_k from $S \setminus \{i\}$ with probability

$$P[Z_k = j | X_k = i] = R(i, j), \qquad j \in S \setminus \{i\}.$$

Step 2: Given $Z_k = j$, set

$$X_{k+1} = \begin{cases} Z_k & \text{if} \quad H_\ell(j) < H_\ell(i) \ \forall \ell = 1, \dots, M_k, \\ X_k & \text{otherwise.} \end{cases}$$

Step 3: Set $k = k + 1$ and go to Step 1.

The implementation of Step 2 can be described as follows. We generate a sample (via Monte Carlo simulation) for both $H(i)$ and $H(j)$. If $H(j) \geq H(i)$, then we immediately reject Z_k, set $X_{k+1} = X_k$, and go to Step 3. On the other hand, if $H(j) < H(i)$, then we generate another pair of samples, and we compare them again. If $H(j) < H(i)$ for all M_k samples, only then do we accept Z_k, set $X_{k+1} = Z_k$, and go to Step 3. For convergence, the testing sequence used by the SC algorithm must satisfy the same conditions as those required by the SR algorithm; i.e., M_k must be such that $M_k = \lfloor c \log_\sigma(k + k_0 + 1) \rfloor, k = 0, 1, 2, \dots,$ for some positive numbers c, σ, and k_0. While the number of comparisons M_k does increase as the algorithm proceeds, it is important to notice that it increases very slowly. As will be seen, this means that the algorithm has the capability to converge to a good solution very quickly, although it is possible that convergence to a true optimum may take a very long time.

3.3. Sketch of the convergence proof. If the objective function can be evaluated exactly, then the SC algorithm converges trivially to an optimal configuration. We will show, however, that the algorithm will also converge when the objective function value has to be estimated.

Due to the i.i.d. assumption of $\{H_\ell(i), \ell = 1, \dots, M_k; i \in S\}$, the state transition probability from i to j is

$$R(i, j) \Pr[H_1(j) < H_1(i), \dots, H_{M_k}(j) < H_{M_k}(i)] = R(i, j)\{\Pr[H(j) < H(i)]\}^{M_k}.$$

Thus, the sequence of configurations visited by the SC algorithm forms a time-inhomogeneous Markov chain $\{X_k\}$. Using the results for time-homogeneous Markov chains presented in [19], we can show convergence.

An outline of our analysis is as follows.

1. Set $M_k = M$ and study the corresponding Markov chain at its steady state (the steady-state probability distribution is denoted by $\pi(M)$).
2. Let M go to infinity and show that
   (a) $\pi(M)$ converges to an optimal probability vector; and
   (b) for large M, $\pi(M)$ is monotonic in M.
3. Show that the Markov chain with $M_k = M$ is weakly ergodic by calculating the coefficient of ergodicity.
4. Show that the Markov chain with $M_k = M$ is strongly ergodic.
5. Show the convergence of the Markov chain $\{X_k\}$ based on its strong ergodicity.

Essentially, the convergence proof for the SC algorithm follows the same lines as those for the SA and SR algorithms; however, the main component of the proof (the second step, which involves showing the convergence of $\pi(M)$ to an optimal probability vector), is very different from the approach used for the SA and SR algorithms. In the next section, section 4, we discuss the quasi-stationary probabilities of the Markov chains that describe the SC algorithm. The convergence of the SC algorithm is proved in section 5. We point out that the Markov chain describing the SC algorithm belongs to the class studied by Connors and Kumar in [6].

4. Markov chain under a constant testing number.

4.1. Markov chain equations. The one-step state transition probabilities of the Markov chain $\{X_k\}$ generated by the SC algorithm for a given testing number M are

$$P_{ij}(M) = \begin{cases} R(i,j)\{\Pr[H(j) < H(i)]\}^M & \text{if} \quad j \neq i, \\ 1 - \sum_{n=1, n\neq i}^{s} P_{in}(M) & \text{if} \quad j = i. \end{cases} \tag{4}$$

To simplify notation, we will let $r_{ij} = R(i,j)$, $p_{ij} = \Pr[H(j) < H(i)]$, and $t_{ij} = r_{ij} p_{ij}^M$ $(i \neq j)$, where $s = |S|$ represents the size of the configuration space. Using our shorthand notation, we can write the one-step transition probabilities as

$$P(M) = \begin{bmatrix} 1 - \sum_{n=2}^{s} t_{1n} & t_{12} & t_{13} & \cdots & t_{1s} \\ t_{21} & 1 - \sum_{n=1,n\neq 2}^{s} t_{2n} & t_{23} & \cdots & t_{2s} \\ t_{31} & t_{32} & 1 - \sum_{n=1,n\neq 3}^{s} t_{3n} & \cdots & t_{3s} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{s1} & t_{s2} & t_{s3} & \cdots & 1 - \sum_{n=1}^{s-1} t_{sn} \end{bmatrix}.$$

Let

$$\begin{aligned} A &= \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ t_{12} & -\sum_{n=1,n\neq 2}^{s} t_{2n} & t_{32} & \cdots & t_{s2} \\ t_{13} & t_{23} & -\sum_{n=1,n\neq 3}^{s} t_{3n} & \cdots & t_{s3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ t_{1s} & t_{2s} & t_{3s} & \cdots & -\sum_{n=1}^{s-1} t_{sn} \end{bmatrix} \\ &= \begin{bmatrix} a_1 & a_2 & a_3 & \cdots & a_s \end{bmatrix} \end{aligned}$$

and

$$b = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \end{bmatrix}^T.$$

We will show that there exists a vector $\pi(M) = [\pi_1(M), \pi_2(M), \ldots, \pi_s(M)]$ that satisfies

$$A\pi^T(M) = b.$$

To do so, define

$$B_m = \begin{bmatrix} a_1 & \cdots & a_{m-1} & b & a_{m+1} & \cdots & a_s \end{bmatrix};$$

i.e., $\{B_m\}_{m=1}^{s}$ is obtained by replacing the mth column of the matrix A by the vector b.

Let $|X|$ represent the determinant of the matrix X. It can be verified that $|A|$ and $\{|B_m|\}_{m=1}^{s}$ are related by

$$|A| = |B_1| + |B_2| + \cdots + |B_s|. \tag{5}$$

By Assumption 3.2, any configuration can be chosen in the next step with positive probability. Therefore, the Markov chain, $\{X_k\}$, generated by the SC algorithm is irreducible and positive recurrent. Furthermore, it has a unique stationary distribution. From Lemma 4.2 and equation (5), we have that $|A| \neq 0$. By Cramér's rule,

$$\pi_1(M) = \frac{|B_1|}{|A|}, \qquad \pi_2(M) = \frac{|B_2|}{|A|}, \cdots, \qquad \pi_s(M) = \frac{|B_s|}{|A|}.$$

Since $\pi(M)$ satisfies $\pi(M) = \pi(M)P(M)$ and $\sum_{i\in S} \pi_i(M) = 1$, it must be the unique stationary probability distribution defined by the state transition probability matrix $[P(M)]$. We also call $\pi(M)$ the *quasi-stationary probability distribution* of the time-inhomogeneous Markov chain $\{X_k\}$ [22].

We now expand each $|B_i|$, $\forall i \in S$, along its ith column. The resulting expansions are given by the following lemma.

LEMMA 4.1. *For $i \in S$, $|B_i|$ can be expanded as follows.*

1. *For $i = 1$,*

$$|B_1| = \begin{vmatrix} -\sum_{n=1,n\neq 2}^{s} t_{2n} & t_{32} & \cdots & t_{s2} \\ t_{23} & -\sum_{n=1,n\neq 3}^{s} t_{3n} & \cdots & t_{s3} \\ \vdots & \vdots & \vdots & \vdots \\ t_{2s} & t_{3s} & \cdots & -\sum_{n=1}^{s-1} t_{sn} \end{vmatrix}.$$

2. *For $1 < i < s$,*

$$|B_i| =$$

$$\begin{vmatrix} -\sum_{n=2}^{s} t_{1n} & \cdots & t_{(i-1)1} & t_{(i+1)1} & \cdots & t_{s1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ t_{1(i-1)} & \cdots & -\sum_{n=1,n\neq i-1}^{s} t_{(i-1)n} & t_{(i+1)(i-1)} & \cdots & t_{s(i-1)} \\ t_{1(i+1)} & \cdots & t_{(i-1)(i+1)} & -\sum_{n=1,n\neq i+1}^{s} t_{(i+1)n} & \cdots & t_{s(i+1)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ t_{1s} & \cdots & t_{(i-1)s} & t_{(i+1)s} & \cdots & -\sum_{n=1}^{s-1} t_{sn} \end{vmatrix}.$$

3. *For $i = s$,*

$$|B_s| = \begin{vmatrix} -\sum_{n=2}^{s} t_{1n} & t_{21} & \cdots & t_{(s-1)1} \\ t_{12} & -\sum_{n=1,n\neq 2}^{s} t_{2n} & \cdots & t_{(s-1)2} \\ \vdots & \vdots & \ddots & \vdots \\ t_{1(s-1)} & t_{2(s-1)} & \cdots & -\sum_{n=1,n\neq s-1}^{s} t_{(s-1)n} \end{vmatrix}.$$

Proof. Expand each $|B_i|$, $\forall i \in S$, along its ith column. The resulting determinant is also denoted by $|B_i|$. If $i = 1$, then statement 1 follows immediately. If $1 < i \leq s$, then for every $\ell = 1, \ldots, i-1$, multiply the ℓth row by -1 and subtract the rest of the rows from the ℓth row. Note that the expansion introduces $i+1$ sign changes, and the multiplication introduces $i-1$ sign changes, so there is no sign change. We obtain 2 and 3 after performing the aforementioned multiplications and subtractions. □

We further expand each $|B_i|$, $\forall i \in S$, into a summation form, such that each term in the summation is a product of $(s-1)$ elements chosen from $\{t_{ij} \mid i, j \in S\}$

according to the expansion theorem. The subscripts of t_{ij} (the i and the j) represent two configurations. To refer to a particular t_{ij}, we have to clearly indicate its subscripts i, j, especially when there are many t_{ij}'s involved. To help keep track of the subscripts, we define the following sets.

Definition 4.1. *For each $i = 1, \dots, s$, we define a permutation set and its index set:*

1. *Permutation set:*
   $PS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^s \mid \{k_j\}_{j=1,j\neq i}^s$ *is a permutation of* $s-1$ *integers in* $\{1, \dots, s\} \setminus \{i\}\}$;
   $PS'_i \triangleq PS_i \setminus \{\{k_j\}_{j=1,j\neq i}^s | k_j = j\}$;
   $IPS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^s \in PS_i\}$;
   $IPS'_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^s \in PS'_i\}$.
2. *Combination set:*
   $QS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^s \mid k_j \in \{1, \dots, s\} \setminus \{j\}, j = 1, \dots, s, j \neq i\}$;
   $QS'_i \triangleq QS_i \setminus \{\{k_j\}_{j=1,j\neq i}^s | k_j = i\}$;
   $IQS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^s \in QS_i\}$;
   $IQS'_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^s \in QS'_i\}$.
3. *Optimum combination set:*
   $OS_i \triangleq \{\{k_j\}_{j=1,j\neq i}^s \in QS_i \mid k_j \in S^*,$ *for each* $j = 1, \dots, s, j \neq i\}$;
   $IOS_i \triangleq \{\ell \mid \ell$ *is an index for each* $\{k_j\}_{j=1,j\neq i}^s \in OS_i\}$.
4. *Nonoptimum combination set:*
   $NS_i \triangleq QS_i \setminus OS_i$;
   $INS_i \triangleq IQS_i \setminus IOS_i$.

For example, if the configuration set $S = \{1, 2, 3\}$ and the global optimum set, as defined in equation (1), $S^* = \{1\}$, then

$PS_1 = \{\{2, 3\}, \{3, 2\}\}$,
$PS'_1 = \{\{2, 3\}, \{3, 2\}\} \setminus \{2, 3\} = \{\{3, 2\}\}$,
$IPS_1 = \{A_1, A_2 | A_1$ represents $\{2, 3\}, A_2$ represents $\{3, 2\}\}$,
$IPS'_1 = \{A_2 | A_2$ represents $\{3, 2\}\}$,
$QS_1 = \{\{1, 1\}, \{1, 2\}, \{3, 1\}, \{3, 2\}\}$,
$QS'_1 = QS_1 \setminus \{1, 1\} = \{\{1, 2\}, \{3, 1\}, \{3, 2\}\}$,
$IQS_1 = \{B_1, B_2, B_3, B_4 | B_1$ represents $\{1, 1\}, B_2$ represents $\{1, 2\}$,
B_3 represents $\{3, 1\}, B_4$ represents $\{3, 2\}\}$,
$IQS'_1 = \{B_2, B_3, B_4 | B_2$ represents $\{1, 2\}, B_3$ represents $\{3, 1\}$,
B_4 represents $\{3, 2\}\}$,
$OS_1 = \{\{1, 1\}\}$,
$IOS_1 = \{C_1 | C_1$ represents $\{1, 1\}\}$.
NS_1, INS_1 can be expressed similarly and are omitted here.

Lemma 4.2. *Let* $\mathcal{R}_i = \prod_{j=1,j\neq i}^s r_{ji}$ *and* $\mathcal{P}_i = \prod_{j=1,j\neq i}^s p_{ji}$ $\forall i \in S$. *Let also* $\mathcal{R}_i^{(\ell)} = \prod_{j=1,j\neq i}^s r_{jk_j}$ *and* $\mathcal{P}_i^{(\ell)} = \prod_{j=1,j\neq i}^s p_{jk_j}$ $\forall i \in S$, *where* $\{k_j\}_{j=1,j\neq i}^s \in QS'_i$ *and* $\ell \in IQS'_i$.

Then the expansion of $|B_i|$, $\forall i \in S$, *has the following properties:*

1. $|B_i| = (-1)^{(s-1)}[\mathcal{R}_i\mathcal{P}_i + \sum_{\ell \in IQS'_i} C_i^{(\ell)}\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}]$, *where* $C_i^{(\ell)}$ *is the number of times that* $\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}$ *appears in the summation*;

2. $(-1)^{(s-1)}|B_i| > 0$;
3. $\forall i \in S^*, \mathcal{P}_i = \mathcal{P}_i^{(\ell)}$, *if* $\ell \in IOS_i$;
4. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_i^{(\ell)}$, *if* $\ell \in INS_i$;
5. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_n$, *if* $n \in S \setminus S^*$;
6. $\forall i \in S^*, \mathcal{P}_i > \mathcal{P}_n^{(\ell)}$, *if* $n \in S \setminus S^*$.

Proof. Let $t_{ii} = -\sum_{n=1,n\neq i}^{s} t_{in}$ and $\mathcal{U}_i^{(\ell)} = \prod_{j=1,j\neq i}^{s} t_{jk_j}$, where $\{k_j\}_{j=1,j\neq i}^{s} \in PS_i'$ and $\ell \in IPS_i'$. Let also $nsc(\ell)$ be the number of sign changes for $\mathcal{U}_i^{(\ell)}$.

When we apply the Laplace expansion theorem for determinants to $|B_i|$, we get

$$|B_i| = (-1)^{s-1}\left[\prod_{j=1,j\neq i}^{s}\left(\sum_{n=1,n\neq j}^{s} t_{jn}\right) + \sum_{\ell\in IPS_i'} (-1)^{s-1+nsc(\ell)} \cdot \mathcal{U}_i^{(\ell)}\right].$$

If we further expand $|B_i|$ into a summation of $\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}$'s, the index set QS_i' will be used. We can find only one $\mathcal{R}_i\mathcal{P}_i$ term from $\prod_{j=1,j\neq i}^{s}(\sum_{n=1,n\neq j}^{s} t_{jn})$, and none from the summands of $\sum_{\ell\in IPS_i'}(-1)^{s-1+nsc(\ell)} \cdot \mathcal{U}_i^{(\ell)}$, since the second index of all $t_{\imath\jmath}$'s in $\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}$ cannot be i simultaneously. Note also that all negative $(-1)^{s-1+nsc(\ell)}\mathcal{R}_i^{(\ell)}\mathcal{P}_i^{(\ell)}$'s for $\ell \in IQS_i'$ (if any) in $\sum_{\ell\in IPS_i'}(-1)^{s-1+nsc(\ell)}\mathcal{U}_i^{(\ell)}$ are canceled because a corresponding term can be found in $\prod_{j=1,j\neq i}^{s}(\sum_{n=1,n\neq j}^{s} t_{jn})$. Therefore, we proved 1 and 2.

To complete the proof, we will need the results of Corollary 3.1, Theorem 3.1, and Lemma 3.1. For all $i \in S^*$ and $\{k_j\}_{j=1,j\neq i}^{s} \in QS_i'$ with its index $\ell \in IQS_i'$, we have

$$\mathcal{P}_i = \prod_{j=1,j\neq i}^{s} p_{ji} = \prod_{j=1,j\neq i}^{s} \Pr[H(i) < H(j)],$$
$$\mathcal{P}_i^{\ell} = \prod_{j=1,j\neq i}^{s} p_{jk_j} = \prod_{j=1,j\neq i}^{s} \Pr[H(k_j) < H(j)].$$

If $\{k_j\}_{j=1,j\neq i}^{s} \in OS_i'$, then $E[H(i)] = E[H(k_j)]\ \forall k_j$, which implies $\Pr[H(i) < H(j)] = \Pr[H(k_j) < H(j)]$. Therefore, 3 is true.

If $\{k_j\}_{j=1,j\neq i}^{s} \in NS_i'$, then $\exists k_{j'}$, such that $E[H(i)] < E[H(k_j)]$, which implies $\Pr[H(i) < H(j)] > \Pr[H(k_{j'}) < H(j)]$. Therefore, 4 is true.

For all $n \in S \setminus S^*$, we have

$$\mathcal{P}_i = \Pr[H(i) < H(n)] \prod_{j=1,j\neq i,n}^{s} \Pr[H(i) < H(j)]$$

and

$$\mathcal{P}_n = \Pr[H(n) < H(i)] \prod_{j=1,j\neq i,n}^{s} \Pr[H(n) < H(j)],$$

since $\Pr[H(i) < H(n)] > \frac{1}{2} > \Pr[H(n) < H(i)]$ and $\Pr[H(i) < H(j)] > \Pr[H(n) < H(j)]$, $\forall j \neq i, n$. Therefore, 5 is true.

Furthermore, we have

$$\mathcal{P}_n^{(\ell)} = \Pr[H(k_i) < H(i)] \prod_{j=1,j\neq i,n}^{s} \Pr[H(k_j) < H(j)],$$

since $\Pr[H(i) < H(n)] > \frac{1}{2} \geq \Pr[H(k_i) < H(i)]$ and $\Pr[H(i) < H(j)] \geq \Pr[H(k_j) < H(j)]$, $\forall j \neq i, n$. Therefore, 6 is true. □

4.2. Convergence of $\boldsymbol{\pi}(M)$ to an optimal probability vector. Let $M \to \infty$ and note that $\mathcal{P}_i^M$ and $[\mathcal{P}_i^{(\ell)}]^M$ $\forall i \in S^*$ and $\forall \ell \in IOS_i$ will dominate all other $[\mathcal{P}_i^{(\ell)}]^M$ $\forall \ell \in INS_i$ and $\mathcal{P}_j^M$, $[\mathcal{P}_j^{(\ell)}]^M$ $\forall j \in S \setminus S^*$ and $\forall \ell \in IQS_j$. Therefore, we have

$$\lim_{M\to\infty} \pi_i(M) = \begin{cases} e_i^* > 0 & \text{if} \quad i \in S^*, \\ 0 & \text{if} \quad i \in S \setminus S^*. \end{cases}$$

Hence as $M \to \infty$, the quasi-stationary probability vector converges to an optimal probability vector.

4.3. Monotone property of the quasi-stationary probabilities. From the form of $\pi_1(M)$, $\pi_2(M), \ldots, \pi_s(M)$, we see that $\exists M^* < \infty$, such that for $M_k > M^*$ the quasi-stationary probabilities have a monotone property, namely,

$$\begin{aligned} \pi_i(M_{k+1}) &> \pi_i(M_k) \quad \text{for} \quad i \in S^*, \\ \pi_i(M_{k+1}) &< \pi_i(M_k) \quad \text{for} \quad i \in S \setminus S^*. \end{aligned}$$

5. Convergence of the SC algorithm. The convergence proof of the SC algorithm is based on theorems in [19] about weak and strong ergodicity of time-inhomogeneous Markov chains.

Let $P_1, P_2, \ldots,$ represent the sequence of one-step state transition probability matrices of a time-inhomogeneous Markov chain, $\{Y_k\}$, with starting probability vector $\mathbf{f}^{(0)}$. Define

$$\mathbf{f}^{(k)} = \mathbf{f}^{(0)} P_1 P_2 \cdots P_k \quad \text{and} \quad \mathbf{f}^{(m,k)} = \mathbf{f}^{(0)} P_{m+1} P_{m+2} \cdots P_{m+k}.$$

We are interested in the limiting behavior of $\mathbf{f}^{(k)}$ and $\mathbf{f}^{(m,k)}$ for any integer $m < k$, as $k \to \infty$. This limiting behavior is captured by the notions of ergodicity. If $\{\mathbf{f}^{(k)}\}$ converges to the same fixed probability vector, $\mathbf{q}$, irrespective of the starting vector $\mathbf{f}^{(0)}$, then we say that the Markov chain is strongly ergodic. Such behavior is often referred to as *loss of memory with convergence*. If, however, for any starting probability vectors $\mathbf{f}^{(0)}$ and $\mathbf{g}^{(0)}$, $\mathbf{f}^{(k)}$ and $\mathbf{g}^{(k)}$ are "close" for sufficiently large k (although $\mathbf{f}^{(k)}$ and $\mathbf{f}^{(k+1)}$ need not to be very "close" for large k), then we say that the Markov chain is weakly ergodic. In this case, the chain has the property of *loss of memory without convergence*.

To give rigorous definitions of weak and strong ergodicity, we first introduce a norm operator $\|\cdot\|$. If $\mathbf{f} = (f_1, f_2, \ldots)$ is a vector, define the norm of $\mathbf{f}$ by

$$\|\mathbf{f}\| = \sum_{i=1}^{\infty} |f_i|.$$

A time-inhomogeneous Markov chain $\{Y_k\}$ is called *weakly ergodic* if, $\forall m$,

$$\lim_{k\to\infty} \sup_{\mathbf{f}^{(0)}, \mathbf{g}^{(0)}} \|\mathbf{f}^{(m,k)} - \mathbf{g}^{(m,k)}\| = 0,$$

where $\mathbf{f}^{(0)}$ and $\mathbf{g}^{(0)}$ are starting probability vectors.

A time-inhomogeneous Markov chain $\{Y_k\}$ is called *strongly ergodic* if there exists a probability vector $\mathbf{q}$ such that, $\forall m$,

$$\lim_{k\to\infty} \sup_{\mathbf{f}^{(0)}} \|\mathbf{f}^{(m,k)} - \mathbf{q}\| = 0,$$

where $\mathbf{f}^{(0)}$ is a starting probability vector.

5.1. Weak ergodicity. We introduce the following notation for the smallest nonzero $R(i,j)$ and the smallest $\Pr[H(j) < H(i)]$:

$$\rho = \min_{i \in S} \min_{j \in S \setminus \{i\}} R(i,j),$$

$$\mu = \min_{i \in S} \min_{j \in S \setminus \{i\}} \Pr[H(j) < H(i)].$$

From Assumption 3.2, we have $\rho > 0$, and from Assumption 3.1 we have $0 < \mu < 1$ ($\Pr[H(j) < H(i)] = \Pr[W_i - W_j < g(i) - g(j)] \in (0,1)$). Choose $\sigma \geq \frac{1}{\mu}$, $0 < c < 1$, k_0, such that $1 \leq c \log_\sigma k_0$ and $M_k = \lfloor c \log_\sigma (k + k_0) \rfloor$ for $k = 0, 1, 2, \ldots$, and recall that $\lfloor \xi \rfloor$ denotes the greatest integer that is smaller than or equal to ξ.

Then we have

$$P_{ij}(M_k) = R(i,j) \Pr[H(j) < H(i)]^{M_k} \geq \rho \Big(\frac{1}{\sigma} \Big)^{M_k}.$$

Define $P(k+1,k) = P(M_k) \cdot P(M_{k+1})$. The coefficient of ergodicity [19] is

$$\alpha[P(k+1,k)] \overset{\triangle}{=} \min_{i,j} \sum_{\ell \in S} \min_{\ell} [P_{i\ell}, P_{j\ell}] \geq \min_{i,j} \min[P_{i\ell'}, P_{j\ell'}] \geq \rho \Big(\frac{1}{\sigma} \Big)^{M_k}.$$

It can be seen that

$$\sum_{k=k^*}^{\infty} \alpha(P(k+1,k)) \geq \sum_{k=k^*}^{\infty} \rho \Big(\frac{1}{\sigma} \Big)^{M_k} \geq \sum_{k=k^*}^{\infty} \rho \Big(\frac{1}{\sigma} \Big)^{c \log_\sigma (k+k_0)} = \sum_{k=k^*}^{\infty} \rho \frac{1}{(k+k_0)^c} \to \infty$$

if $c \leq 1$. Hence, the Markov chain generated by the SC algorithm is weakly ergodic by Theorem V.3.2 of [19] (see the appendix for a statement of the theorem).

5.2. Strong ergodicity. With the monotone property of the quasi-stationary probabilities we have the following lemma.

LEMMA 5.1. *The probability vector*, $\pi(M)$, *defined in* $\pi(M)P(M) = \pi(M)$ *satisfies*

$$\sum_{k=0}^{\infty} \| \pi_i(M_{k+1}) - \pi_i(M_k) \| < \infty.$$

Proof. It follows from the monotone property of π_i that there exists an integer k^* such that, for any $k > k^*$,

$$\begin{aligned} \pi_i(M_{k+1}) \geq \pi_i(M_k) \qquad & \forall i \in S^*, \\ \pi_i(M_{k+1}) \leq \pi_i(M_k) \qquad & \forall i \in S \setminus S^*. \end{aligned}$$

Hence, for any $k \geq k^*$,

$$\| \pi_i(M_{k+1}) - \pi_i(M_k) \| = \sum_{i \in S^*} [\pi_i(M_{k+1}) - \pi_i(M_k)] - \sum_{i \in S \setminus S^*} [\pi_i(M_{k+1}) - \pi_i(M_k)].$$

Note that from $\sum_{i \in S^*} \pi_i(M_k) + \sum_{i \in S \setminus S^*} \pi_i(M_k) = \| \pi(M_k) \| = 1$, we conclude that, for any $k \geq k^*$,

$$\| \pi_i(M_{k+1}) - \pi_i(M_k) \| = 2 \sum_{i \in S^*} [\pi_i(M_{k+1}) - \pi_i(M_k)].$$

Table 6.1

Percentage of objective function values that fall in each interval.

ℓ	1	2	3	4	5
	[10,30]	[30,50]	[50,70]	[70,90]	[90,110]
p_ℓ	3	37	30	15	15

Therefore, we have, for any $\ell \geq k^*$,

$$\sum_{k=k^*}^{\ell} \|\pi_i(M_{k+1}) - \pi_i(M_k)\| = 2\sum_{i\in S^*}[\pi_i(M_{\ell+1}) - \pi_i(M_{k^*})] \leq 2\sum_{i\in S^*}\pi_i(M_{\ell+1}) \leq 2. \qquad \square$$

By Theorem V.4.3 of [19] (see the appendix for a statement of the theorem), we have the following theorem.

Theorem 5.1. *The Markov chain $\{X_k\}$ generated by the SC algorithm, with M_k taking $M_k = \lfloor c\log_\sigma(k+k_0)\rfloor$, is strongly ergodic. Furthermore,*

1. $\lim_{k\to\infty}\sup_{x_0}\|x(\ell,k) - e^*\| = 0$,
2. $\lim_{k\to\infty}\Pr[X_k \in S^*] = 1$,

where $x(\ell,k) = x_0P(\ell,k) = x_0\prod_{i=\ell}^{k-1}P(M_i)$, x_0 is the initial probability vector, e^* is an optimal probability vector, and c,σ,k_0 are some constants.

Proof. By Lemma 5.1 of this paper and Theorem V.4.3 in [19], the Markov chain, $\{X_k\}$, is strongly ergodic. Conclusions 1 and 2 are true. $\square$

To summarize, what we have shown is that the SC algorithm converges to an optimal solution under the following conditions: (i) the estimates of $H(i)$ are unbiased and identically distributed; (ii) the estimation error $W_i = H(i) - g(i)$ is i.i.d. and each $W_i \in S$ has a symmetric continuous probability distribution; (iii) any configuration can be reached from any other in one step; i.e., $R(i,j) > 0\ \forall i,j \in S$; and (iv) the testing sequence M_k satisfies $M_k = \lfloor c\log_\sigma(k+k_0+1)\rfloor, k = 0,1,2,\ldots,$ for some positive numbers c,σ, and k_0.

6. Numerical examples.

6.1. A testbed system. We design a testbed system with one million configurations. To generate the objective function, we divided the interval [10.0,110.0] into five subintervals with equal lengths of 20.0. Then we generated $p_\ell\%$, $\ell = 1,\ldots,5$, from the total configurations with objective function value uniformly distributed in subinterval ℓ. We consider a minimization problem. To make the search more difficult, we allocate fewer points in the first interval ("good interval") than in others. The parameters used are shown in Table 6.1.

Each sample of objective function for configuration i is generated according to $H(i) = g(i) + W_i$, where W_i models the behavior of a Monte Carlo simulator. For these experiments, we take $W_i \sim \text{unif}[-a/2, a/2]\ \forall i \in S$.

6.2. Comparison of the SC and SR algorithms. Using the testbed system described above, we performed some experiments to compare the performance of our SC algorithm to the SR algorithm. The purpose of these experiments is to demonstrate that the SC algorithm is capable of outperforming the SR algorithm in certain practical situations.

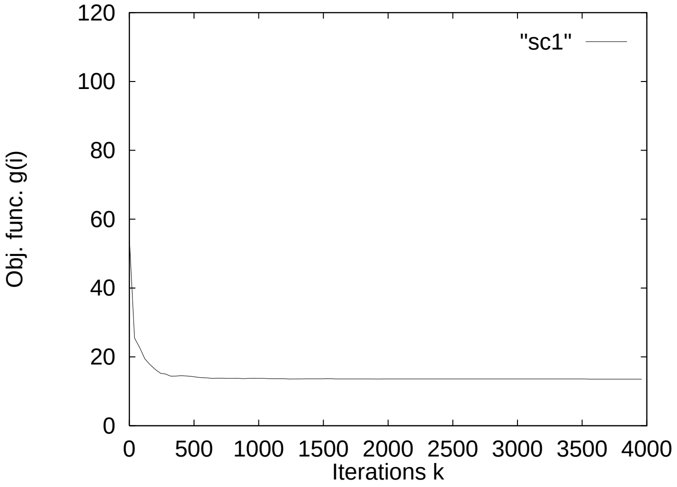


FIG. 1. *Optimization trajectory of SC* $(a = 10)$.

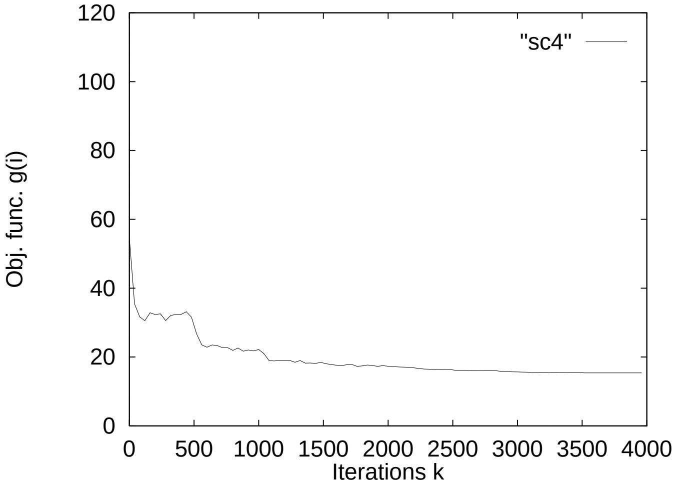


FIG. 2. *Optimization trajectory of SC* $(a = 40)$.

Both the SC and the SR algorithms are guaranteed to converge when the testing number increases logarithmically. However, in practice, the log function increases too fast during the first few iterations. For our experiments, therefore, we used the linear sequence, $M_k = 1 + \lfloor k/500 \rfloor$. It is a reasonable approximation to the logarithm sequence over the finite range of k used in the experiments. While our theoretical results do not guarantee the convergence to the optimum in these experiments, the experiments do reflect what happens in practical optimization. Although our experiments did not indicate any significant sensitivity with respect to the starting configuration, we, nevertheless, decided to initialize the algorithms at the same initial configuration X_0 for all of the experiments. The generating function is given by $R(i, j) = 1/999999 \; \forall i, j \in S, i \neq j$.

We performed the experiments at several different noise levels by letting $a = 0$, 10, 20, 30, 40, 50, 60, 70. Figures 1, 2, and 3 show the performance of the SC algorithm

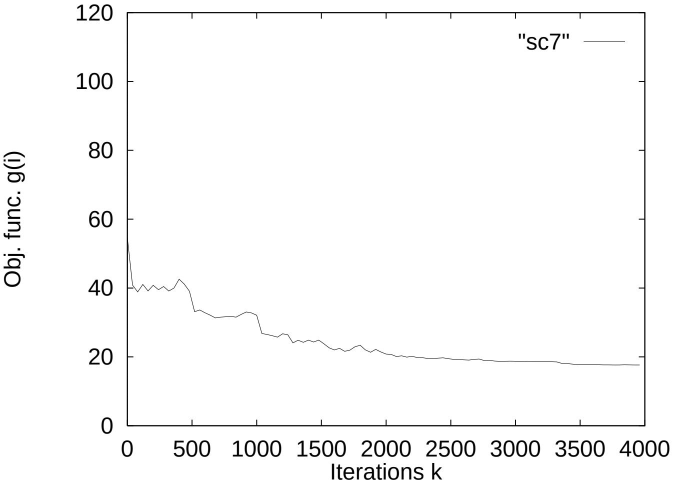


FIG. 3. *Optimization trajectory of SC* $(a = 70)$.

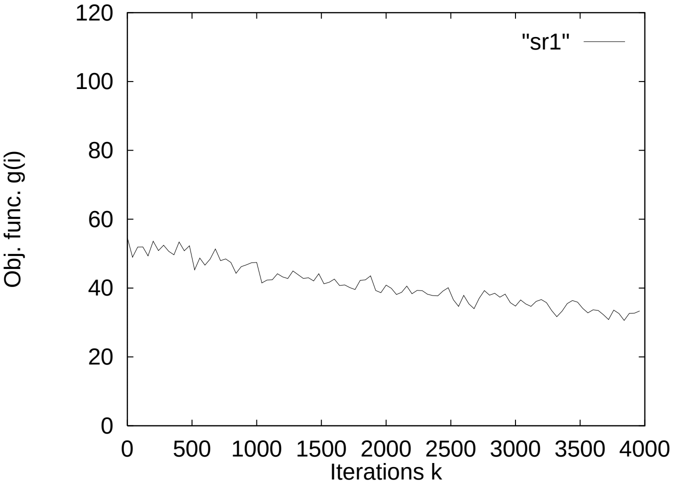


FIG. 4. *Optimization trajectory of SR with a closed neighborhood* $(a = 10)$.

for $a = 10, 40, 70$, respectively. Each curve is an average of 100 replica with the same initial configuration and different random speeds. As can be seen, even at the highest noise level, $a = 70$, the SC algorithm quickly settles down after about 2200 iterations and gives very good performance.

For comparison, we performed the same experiments with the SR algorithm. For these experiments, we chose the same linear testing sequence M_k as used for the SC algorithm. The stochastic ruler was chosen to cover the entire configuration space, i.e., $\Theta(a, b) \sim \text{unif}[0, 120]$. Then we performed two sets of experiments, one with a "closed neighborhood structure" and one with an "open neighborhood structure." The closed neighborhood structure $N(i)$ was defined as follows.

We placed the one million configurations on a circle with equal distance between them. For each configuration, the left 100 configurations and the right 100 configurations are chosen as its neighbor set. During the optimization, a neighbor is chosen

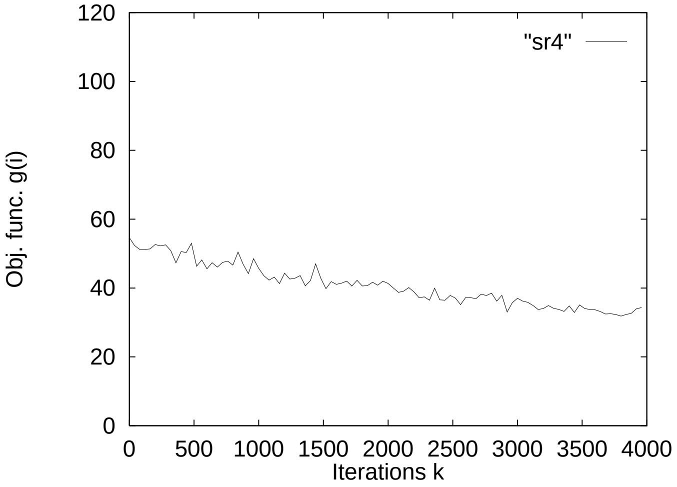


FIG. 5. *Optimization trajectory of SR with a closed neighborhood* $(a = 40)$.

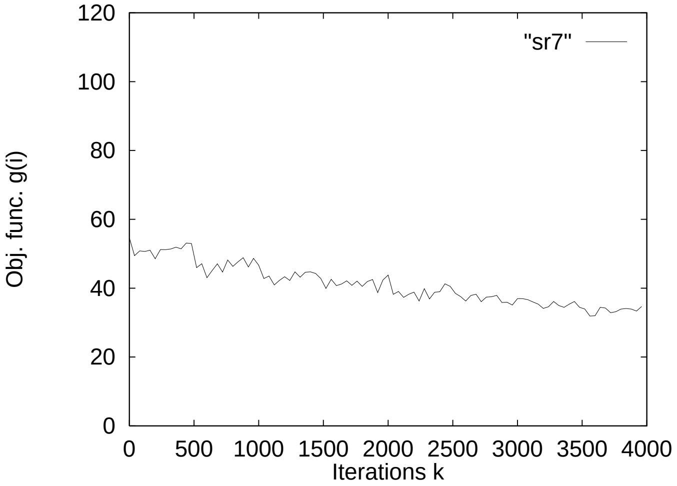


FIG. 6. *Optimization trajectory of SR with a closed neighborhood* $(a = 70)$.

uniformly from the neighbor set of the current configuration. Figures 4, 5, and 6 show the performance of the SR algorithm with the closed neighborhood structure for noise levels $a = 10, 40, 70$, respectively. For the open neighborhood structure a neighbor is chosen uniformly from the entire configuration space, as in the SC algorithm. Figures 7, 8, and 9 show the performance of the SR algorithm with open neighborhood structure for $a = 10, 40, 70$, respectively.

As can be seen from the figures, the SC algorithm performs much better than the SR algorithm on the particular optimization problem examined. We are quick to point out, however, that, in general, it is difficult to “compare” the performance of different optimization algorithms. The particular structure of the problem and the choice of parameters are often crucial. We also emphasize that when there is a good neighborhood structure available, then the SR algorithm, which exploits this structure, may easily outperform the SC algorithm, which does not. For those problems where it is difficult

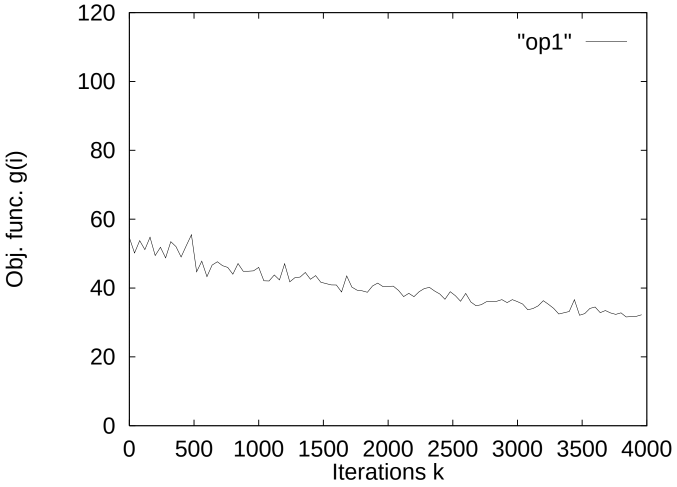


Fig. 7. *Optimization trajectory of SR with an open neighborhood* $(a = 10)$.

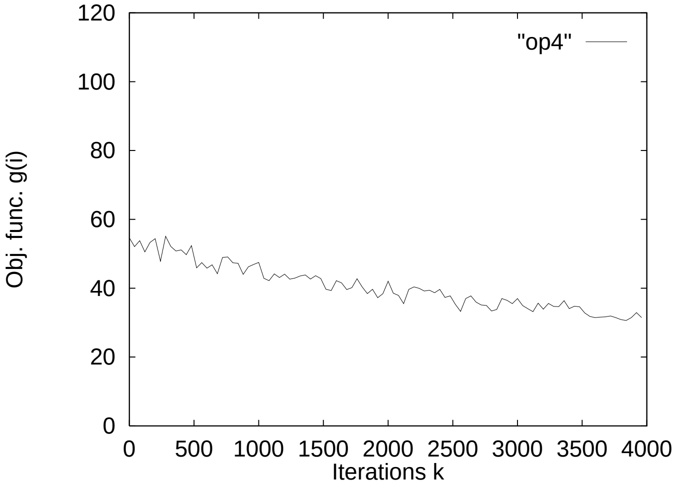


Fig. 8. *Optimization trajectory of SR with an open neighborhood* $(a = 40)$.

to find a useful neighborhood structure (e.g., computer configuration problems), we argue that the SC provides a good alternative.

7. Discussion. In this paper we proposed the SC algorithm as an alternative to the SA and SR algorithms. The SC algorithm is a simple optimization algorithm designed for problems where we have only noisy estimates of the objective function and where the search space is very large. The main advantage of the algorithm is that it is essentially parameter free (except for the choice of the constants k_0, σ, and c in the testing sequence) and requires no knowledge about the structure of the search space. Experimental results show that it converges to a good solution very quickly, even when the estimates of the objective function are very noisy. Unlike the SA and SR algorithms, the SC algorithm does not utilize, nor does it require, that the configuration space have any neighborhood structure. This is important for many of the discrete optimization problems encountered in computer and communication

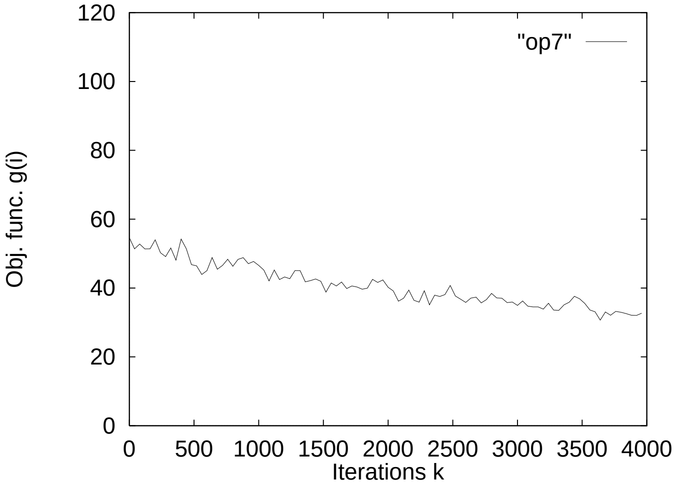


FIG. 9. *Optimization trajectory of SR with an open neighborhood* ($a = 70$).

network design, where it is difficult to find a natural neighborhood structure. In addition to the simple example used in this paper, the SC algorithm has been applied successfully to a realistic computer configuration design problem and to a large-scale transportation scheduling problem. For such problems, it appears to be more practical to apply the SC algorithm, instead of investing effort to identify a good neighborhood structure before beginning the optimization procedure.

Appendix.

THEOREM A.1 (Theorem V.3.2 of [19]). *Let $\{X_n\}$ be a nonstationary Markov chain with transition matrices, $\{P_n\}_{n=1}^{\infty}$. The chain, $\{X_n\}$, is weakly ergodic if and only if there exists a subdivision of $P_1 \cdot P_2 \cdot P_3 \cdots$ into blocks of matrices $[P_1 \cdot P_2 \cdots P_{n_1}] \cdot [P_{n_1+1} \cdot P_{n_1+2} \cdots P_{n_2}] \cdots [P_{n_j+1} \cdot P_{n_j+2} \cdots P_{n_{j+1}}] \ldots$ such that*

$$\sum_{j=0}^{\infty} \alpha(P^{(n_j, n_{j+1})}) = \infty,$$

where $n_0 = 0$.

THEOREM A.2 (Theorem V.4.3 of [19]). *Let $\{P_n\}$ be a sequence of transition matrices corresponding to a nonstationary weakly ergodic Markov chain with $P_n \in \mathcal{A}$ for all n. If there exists a corresponding sequence of left eigenvectors ϕ_n, satisfying*

$$\sum_{j=0}^{\infty} \|\phi_j - \phi_{j+1}\| < \infty,$$

then the chain is strongly ergodic.

Acknowledgments. The authors wish to thank the anonymous reviewers for their many valuable suggestions and comments on the previous versions of the paper. The authors also wish to thank Professor Patrick Kelly and Dr. David Pepyne for their help.

REFERENCES

[1] E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*, John Wiley, New York, 1989.

[2] P. Bratley, B. L. Fox, and L. E. Schrage, *A Guide to Simulation*, 2nd ed., Springer-Verlag, Berlin, 1987.

[3] S. H. Brooks, *A discussion of random methods for seeking maxima*, Oper. Res., 6 (1958).

[4] K. L. Buescher and P. R. Kumar, *Learning by canonical smooth estimation, Part* I*: Simultaneous estimation*, IEEE Trans. Automat. Control, 41 (1996), pp. 545–556.; also available online from http://black.csl.uiuc.edu/ prkumar/postscript_files.html.

[5] K. L. Buescher and P. R. Kumar, *Learning by canonical smooth estimation, Part* II*: Learning and choice of model complexity*, IEEE Trans. Automat. Control, 41 (1996), pp. 557–569; also available online from http://black.csl.uiuc.edu prkumar/postscript_files.html.

[6] D. P. Connors and P. R. Kumar, *Simulated annealing type Markov chains and their order balance equations*, SIAM J. Control Optim., 27 (1989), pp. 1440–1461.

[7] F. Darema, S. Kirkpatrick, and A. Norton, *Parallel algorithms for chip placement by simulated annealing*, IBM J. Res. Develop., 31 (1987), pp. 391–402.

[8] M. Durand and S. R. White, *Permissible Error in Parallel Simulated Annealing*, Technical Report RC 15487, IBM T. J. Watson Research Center, Yorktown Heights, NY, 1990.

[9] S. B. Gelfand and S. K. Mitter, *Simulated annealing with noisy or imprecise energy measurements*, J. Optim. Theory Appl., 62 (1989), pp. 49–62.

[10] S. B. Gelfand and S. K. Mitter, *Simulated annealing type algorithms for multivariate optimization*, Algorithmica, 6 (1991), pp. 419–436.

[11] D. Goldsman and B. Nelson, *Ranking, selection method for stochastic optimization problem*, in Proceedings, 1994 Winter Simulation Conference, Orlando, FL, ACM Press, New York, 1994.

[12] W.-B. Gong, Y. C. Ho, and W. Zhai, *Stochastic comparison algorithm for discrete optimization with estimation*, in Proceedings, 31st CDC, Tucson, AZ, 1992, pp. 795–802.

[13] D. R. Greening, *Simulated Annealing with Inaccurate Cost Functions*, Technical Report, Department of Computer Science, University of California, Los Angeles, 1994.

[14] D. R. Greening and F. Darema, *Rectangular spatial decomposition methods for parallel simulated annealing*, in Proceedings, International Conference on Supercomputing, Crete, Greece, ACM Press, New York, 1989, pp. 295–302.

[15] L. K. Grover, *Simulated Annealing Using Approximate Calculation*, Technical Memorandum 52231-860410-01, AT&T Bell Laboratories, Murray Hill, NJ, 1989.

[16] S. S. Gupta and S. Panchapakesan, *Multiple Decision Procedures: Theory and Methodology of Selecting and Ranking Populations*, John Wiley, New York, 1979.

[17] B. Hajek, *A tutorial survey of theory and applications of simulated annealing*, in Proceedings, 24th CDC, Ft. Lauderdale, FL, 1985, pp. 755–760.

[18] Y. C. Ho, R. Sreenivas, and P. Vakili, *Ordinal Optimization of DEDS*, J. Discrete Event Dyn. Syst., 2 (1992), pp. 61–88.

[19] D. L. Isaacson and R. W. Madsen, *Markov Chains: Theory and Applications*, John Wiley, New York, 1976.

[20] R. Jayaraman and F. Darema, *Error tolerance in parallel simulated annealing techniques*, in Proceedings, International Conference on Computer Design, IEEE Computer Society Press, 1988, pp. 545–548.

[21] T. Lai and S. Yakowit, *Machine learning and nonparametric bandit theory*, IEEE Trans. Automat. Control, 40 (1995), pp. 1199–1209.

[22] D. Mitra, F. Romeo, and A. Sansiovanni-Vincentelli, *Convergence and finite-time behavior of simulated annealing*, Adv. in Appl. Probab., 18 (1986), pp. 747–771.

[23] F. Romeo and A. Sangiovanni-Vincentelli, *A theoretical framework for simulated annealing*, Algorithmica, 6 (1991), pp. 302–345.

[24] R. Rubinstein, *Monte Carlo Optimization, Simulation and Sensitivity of Queueing Networks*, John Wiley, New York, 1986.

[25] T. J. Santner and A. C. Tamhane, *Design of Experiments: Ranking and Selection*, Marcel Dekker, New York, 1984

[26] S. Yakowitz, T. Jayawardena, and S. Li, *Theory for automatic learning under partially observed Markov-dependent noise*, IEEE Trans. Automat. Control, 37 (1992), pp. 1316–1324.

[27] D. Yan and H. Mukai, *Stochastic discrete optimization*, SIAM J. Control Optim., 30 (1992), pp. 594–612.

SIAM J. Optim.
Vol. 10, No. 2, pp. 405–414

A GENERAL PRIMAL-DUAL ENVELOPE METHOD FOR CONVEX PROGRAMMING PROBLEMS[*]

S. E. WRIGHT[†]

Abstract. A general decomposition framework for solving large-scale convex programming problems is described. New algorithms are obtained and several known techniques are recovered as special cases, including Dantzig–Wolfe column generation, the finite envelope method of Rockafellar, and Zhu's primal-dual steepest descent method.

Key words. convex programming, large-scale decomposition, primal-dual methods

AMS subject classification. 90C25

PII. S1052623497330823

1. Introduction.

This paper describes a decomposition framework for large-scale convex programming problems in which there is a natural primal-dual structure. Our framework generalizes the methods proposed by Zhu [11] and Rockafellar [6] for problems of extended linear-quadratic programming (ELQP). The methods and proofs of both Zhu and Rockafellar were extended to convex nonquadratic problems by (respectively) Zhu [10] and Wright [8] by using quadratic lower approximations. The proofs given here combine and refine the arguments of those four papers.

Our aim is to solve the saddle point problem

$$(\mathcal{S}) \qquad \text{minimax } L(u,v) \text{ over all } (u,v) \in U \times V,$$

where L is a finite convex-concave saddle function defined on a product of closed convex sets $U \times V \subset \mathbf{R}^n \times \mathbf{R}^m$. Such minimax problems provide useful reformulations of optimality conditions for extremal problems and also arise in a variety of engineering and economic contexts.

The primal and dual problems associated with $(\mathcal{S})$ are, respectively,

$$(\mathcal{P}) \qquad \text{minimize } f(u) := \sup_V L(u,\cdot) \text{ over } u \in U$$

and

$$(\mathcal{D}) \qquad \text{maximize } g(v) := \inf_U L(\cdot,v) \text{ over } v \in V.$$

We define the *duality* mappings for $(\mathcal{S})$ by

$$u \mapsto F(u) := \operatorname*{argmax}_V L(u,\cdot),$$
$$v \mapsto G(v) := \operatorname*{argmin}_U L(\cdot,v).$$

We assume that these set-valued mappings are nonempty-valued on U and V, respectively. Observe that F and G are single-valued mappings in the case where L is strictly convex-concave; in such cases we shall treat F and G as functions.

[*] Received by the editors December 1, 1997; accepted for publication (in revised form) September 2, 1998; published electronically January 19, 2000.
http://www.siam.org/journals/siopt/10-2/33082.html

[†] Department of Mathematics and Statistics, Miami University, Oxford, OH 45056 (wrightse@muohio.edu).

The family of algorithms we propose relies on the ability to find elements in the sets $F(u)$ and $G(v)$ in an efficient manner. In particular, exact evaluation of the primal and dual objective functions f and g must be easily managed. This would be the case when the optimizations defining $F(u)$ and $G(v)$ are highly separable, as occurs in many large-scale problems.

The general algorithm, and a new algorithm based on it, are introduced in the next section. In section 3 we derive the algorithms of Rockafellar and Zhu as special cases. Section 4 is devoted to a proof of global linear convergence under the assumption of strong convexity. Some concluding remarks are given in section 5.

2. The general algorithm. We propose the following algorithmic framework for solving $(\mathcal{S})$.

Algorithm 2.1. General envelope method.

(1) *Envelope generation: Given a current best guess* $(u_k, v_k) \in U \times V$, *choose sets* $U_k, U_k' \subset U$ *and* $V_k, V_k' \subset V$, *with* U_k *and* V_k *convex, which satisfy the following conditions:*

$$\begin{aligned} u_k \in U_k, \quad U_k' \subset U_k, \qquad U_k' \cap G(v_k) \neq \emptyset, \\ v_k \in V_k, \quad V_k' \subset V_k, \qquad V_k' \cap F(u_k) \neq \emptyset. \end{aligned} \tag{2.1}$$

(2) *Search directions: Calculate saddle points*

$$(\hat{u}_k, v_k') \in \operatorname*{argminimax}_{U_k \times V_k'} L, \qquad (u_k', \hat{v}_k) \in \operatorname*{argminimax}_{U_k' \times V_k} L.$$

(3) *Line searches: Compute (approximately) the linesearch elements*

$$\bar{u}_k \in \operatorname*{argmin}_{[u_k, \hat{u}_k]} f, \quad \bar{v}_k \in \operatorname*{argmax}_{[v_k, \hat{v}_k]} g.$$

(4) *Restart and update: If using the optional "restarts," set*

$$u_{k+1} = \begin{cases} \bar{u}_k \textit{if} & f(\bar{u}_k) \leq f(u_k'), \\ u_k' \textit{if} & f(\bar{u}_k) > f(u_k'), \end{cases}$$

$$v_{k+1} = \begin{cases} \bar{v}_k \textit{if} & g(\bar{v}_k) \geq g(v_k'), \\ v_k' \textit{if} & g(\bar{v}_k) < g(v_k'). \end{cases}$$

Otherwise, set $u_{k+1} = \bar{u}_k$, $v_{k+1} = \bar{v}_k$. *Replace* k *by* $k+1$ *and go to step* 1.

The main idea here is that the primal and dual problems are approximated by replacing U and V in the definitions of $(\mathcal{P})$ and $(\mathcal{D})$ by some choice of subsets. Specifically, the search directions correspond to optimal solutions for the approximating problems

$$\text{minimize } f_k(u) := \sup_{V_k'} L(u, \cdot) \text{ over } u \in U_k \tag{$\mathcal{P}_k$}$$

and

$$\text{maximize } g_k(v) := \inf_{U_k'} L(\cdot, v) \text{ over } v \in V_k. \tag{$\mathcal{D}_k$}$$

The intersection criteria in (2.1) guarantee that these approximations hold exactly at the current iterates u_k and v_k. The method recovers several previously known algorithms, as will be shown in the next section.

Notice that for the algorithm to be well defined, one must of course verify that the problem satisfies suitable hypotheses (such as compactness) to guarantee that the envelope minimax problems in step 2 actually have solutions.

We mention that our convergence proof in section 4 does not actually require U_k and V_k to be convex, but merely star-shaped about the current best guesses u_k and v_k, respectively:

$$\begin{aligned} [u, u_k] &\subset U_k \quad \forall u \in U_k, \\ [v, v_k] &\subset V_k \quad \forall v \in V_k. \end{aligned}$$

This requirement provides a criterion, stated formally in Theorem 4.1, by which the approximate linesearches of step 3 can be judged adequate. The linesearches can be executed in a variety of ways, including a fixed step-length or a modified Armijo rule. We refer the reader to the papers of Zhu [10], [11] for a discussion of possible implementations of such a criterion.

Finally, the optional restarts of step 4 provide the opportunity to use the best points actually calculated so far. For certain choices of search directions, the restarts lead to a slight theoretical improvement in convergence. We include them here for comparison with the algorithm of Zhu [10].

The simplest envelope generation method is given by using the minimal choices of sets satisfying (2.1). The resulting algorithm is new.

ALGORITHM 2.2. DUALITY-DESCENT METHOD WITH INEXACT LINESEARCH.

(1) *Given current best guesses* $u_k \in U$ *and* $v_k \in V$, *choose elements* $u_k' \in G(v_k)$ *and* $v_k' \in F(u_k)$ *and calculate*

$$\hat{u}_k \in \operatorname*{argmin}_{[u_k, u_k']} L(\cdot, v_k'), \qquad \hat{v}_k \in \operatorname*{argmax}_{[v_k, v_k']} L(u_k', \cdot).$$

(2) *Compute (approximately) the linesearch elements*

$$u_{k+1} \in \operatorname*{argmin}_{[u_k, \hat{u}_k]} f, \qquad v_{k+1} \in \operatorname*{argmax}_{[v_k, \hat{v}_k]} g.$$

Replace k *by* $k+1$ *and go to step* 1.

We have omitted the optional restarts for simplicity. Also, the theory under which the restarts provide an advantage does not apply in this particular case.

Observe that Algorithm 2.2 amounts to an inexact linesearch in the primal and dual directions $\hat{u}_k - u_k$ and $\hat{v}_k - v_k$ provided by the duality mappings. As mentioned above, the first step (which in many situations is relatively easy to carry out) provides a criterion for judging the adequacy of the approximate linesearch in step 2. Clearly, an exact linesearch can only improve on this method and removes the requirement for step 1. We state this as a separate algorithm.

ALGORITHM 2.3. DUALITY-DESCENT METHOD WITH EXACT LINESEARCH. *Given a current best guess* $(u_k, v_k) \in U \times V$, *choose* $\tilde{u}_k \in G(v_k)$ *and* $\tilde{v}_k \in F(u_k)$ *and compute (exactly) linesearch elements*

$$u_{k+1} \in \operatorname*{argmin}_{[u_k, \tilde{u}_k]} f, \qquad v_{k+1} \in \operatorname*{argmax}_{[v_k, \tilde{v}_k]} g.$$

Replace k *by* $k+1$ *and repeat.*

The reader should note that Algorithm 2.3 does not fit into the framework of Algorithm 2.1. Nevertheless, a direct comparison with Algorithm 2.2 shows that it must converge at least as fast as that algorithm, under the assumptions used in section 4.

Rockafellar [6] observed that, in the case where F and G are single-valued, the primal-dual direction $(G(v_k)-u_k, F(u_k)-v_k)$ is necessarily a descent direction for the duality gap. However, his convergence proof does not quite cover either version of the duality-descent method. The proof we give in section 4 extends that of Rockafellar to a broader class of methods, as well as to a broader class of saddle functions.

In the next section we give some specializations of the general envelope method which recover several other known algorithms. We prove the main convergence results for our method in section 4.

3. Other special cases. In this section we show that our algorithmic framework generalizes two earlier methods, the *finite envelope method* of Rockafellar [6] and the *primal-dual steepest descent algorithm* of Zhu [10], [11]. Both of these methods are envelope generation techniques with different restrictions on the choice of sets satisfying conditions (2.1).

First we consider the method proposed by Zhu. The statement of this algorithm assumes that the duality mappings F and G are single-valued.

ALGORITHM 3.1. PRIMAL-DUAL STEEPEST DESCENT, PDSD-2.

(1) *Given current best guesses $u_k \in U$ and $v_k \in V$, calculate*

$$\hat{u}_k = G(F(u_k)), \qquad \hat{v}_k = F(G(v_k)).$$

(2) *Compute (approximately) the linesearch elements*

$$\bar{u}_k = \operatorname*{argmin}_{[u_k,\hat{u}_k]} f, \qquad \bar{v}_k = \operatorname*{argmax}_{[v_k,\hat{v}_k]} g.$$

(3) *If using optional restarts, set*

$$u_{k+1} = \begin{cases} \bar{u}_k & \text{if } f(\bar{u}_k) \le f(G(v_k)), \\ G(v_k) & \text{if } f(\bar{u}_k) > f(G(v_k)), \end{cases}$$

$$v_{k+1} = \begin{cases} \bar{v}_k & \text{if } g(\bar{v}_k) \ge g(F(u_k)), \\ F(u_k) & \text{if } g(\bar{v}_k) < g(F(u_k)). \end{cases}$$

Otherwise, set $u_{k+1} = \bar{u}_k$, $v_{k+1} = \bar{v}_k$. Replace k by $k+1$ and go to step 1.

The name of this algorithm stems from the fact that the direction vector $\hat{u}_k - u_k$ represents a "projected steepest descent direction" for f at u_k, where the direction and projection are computed with respect to the inner product determined by the Hessian matrix $\nabla^2_{uu}L(u_k, v_k)$ (when positive definite). Similarly, $\hat{v}_k - v_k$ represents a projected steepest ascent direction. The PDSD-2 variant given here was proposed by Zhu in [10]. The basic idea was introduced earlier by Zhu and Rockafellar [9], who developed a general class of primal-dual projected gradient algorithms, including a primal-dual extension of the conjugate gradient method.

The PDSD-2 algorithm above amounts to envelope generation with the choice

$$U_k \equiv U, \qquad U'_k := \{G(v_k)\}, \qquad V_k \equiv V, \qquad V'_k := \{F(u_k)\}.$$

To see this, note that step 2 of Algorithm 2.1 can now be replaced by the calculation of

$$\hat{u}_k = G(F(u_k)), \qquad \hat{v}_k = F(G(v_k)).$$

Next consider the case where we require the sets used in envelope generation to satisfy $U_k = U'_k$ and $V_k = V'_k$. Then step 1 of Algorithm 2.1 consists of solving a single minimax problem over the restricted domain $U_k \times V_k$. By omitting the restarts we recover the *finite envelope method* of Rockafellar [6].

ALGORITHM 3.2. FINITE ENVELOPE METHOD.

(1) *Given current best guesses* $u_k \in U$ *and* $v_k \in V$, *choose closed convex sets* $U_k \subset U$ *and* $V_k \subset V$ *satisfying the conditions*

$$\begin{aligned} u_k \in U_k, &\qquad G(v_k) \cap U_k \neq \emptyset, \\ v_k \in V_k, &\qquad F(u_k) \cap V_k \neq \emptyset. \end{aligned}$$

(2) *Calculate the saddle point*

$$(\hat{u}_k, \hat{v}_k) = \operatorname*{argminimax}_{U_k \times V_k} L.$$

(3) *Compute (approximately) the linesearch elements*

$$u_{k+1} = \operatorname*{argmin}_{[u_k, \hat{u}_k]} f, \qquad v_{k+1} = \operatorname*{argmax}_{[v_k, \hat{v}_k]} g.$$

Replace k *by* $k+1$ *and go to step* 1.

We mention that Rockafellar's description of this algorithm actually imposes the additional requirement that

$$G(F(u_k)) \in U_k, \qquad F(G(v_k)) \in V_k,$$

although his proof of convergence does not use this fact. However, he goes on to show that when defined in this way, the finite envelope method identifies the "optimal face" after finitely many iterations in the fully quadratic situation.

Algorithm 3.2 can be viewed as a generalization of the classical Dantzig–Wolfe and Benders (or "L-shaped") decompositions for linear programming. The former corresponds to choosing

$$U_k := \operatorname{co}\{u_{k'} : k' \leq k\}, \qquad V_k \equiv V,$$

whereas the latter is given, via the dual, through the choice

$$U_k \equiv U, \qquad V_k := \operatorname{co}\{v_{k'} : k' \leq k\}.$$

Observe that the linesearches are redundant in both of these. A simple basis-counting argument can be used to prove the finite termination of these two methods. It is possible that such an argument could be extended to the case where L is biaffine and U and V are convex polyhedra, as long as suitably large choices of the sets U_k, U'_k, V_k, V'_k are used.

4. Convergence results. In this section, we shall prove global linear convergence of the general envelope method (Algorithm 2.1) for saddle functions of the form

$$L(u, v) = J(u, v) + \frac{p}{2}\|u\|^2 - \frac{q}{2}\|v\|^2,$$

where J is a convex-concave saddle function and p and q are positive real numbers. The quadratic terms guarantee that L admits a unique saddle point on $U \times V$. In addition, they force the duality mappings F and G to be single-valued: we shall treat F and G as functions in this section. This assumption corresponds to strong convexity in the u variables and strong concavity in the v variables.

A comment is perhaps in order here. Imposing strong convexity (via definiteness of the Hessian, say) with respect to the primal variables is a fairly common assumption, but strong concavity in the dual variables is somewhat less usual. This condition is satisfied when augmented Lagrangians or primal-dual barrier representation are used. In such settings, one might use an envelope method to generate a good starting point and then switch to a method giving superlinear local convergence.

Alternatively, to apply the convergence theory given here, some sort of regularization may be needed, possibly in the form of proximal terms introduced for the primal and/or dual variables. (We refer the reader to Rockafellar [4] for a discussion of the proximal point algorithm.) Any separability making envelope methods attractive is not adversely affected by such regularization. For some examples of this in stochastic linear programming, see Rockafellar and Wets [5] and Ruszczyǹski [7].

We also assume that J is differentiable on some open set containing $U \times V$ and has a Lipschitzian gradient:

$$\|\nabla J(u,v) - \nabla J(u',v')\| \leq K \cdot \max\{\|u-u'\|, \|v-v'\|\}$$

for all (u,v) and (u',v') in $U \times V$. The rate of convergence of the algorithm will depend on the parameter $\gamma := K^2/pq$.

We can now state our convergence theorem.

Theorem 4.1. *Suppose that Algorithm* 2.1 *is applied to problem* ($\mathcal{S}$). *Assume that in iteration k the (approximate) linesearches in step* 3 *lead to iterates satisfying*

$$\begin{aligned} f(\bar{u}_k) &\leq f\big(u_k + \lambda_k(\hat{u}_k - u_k)\big), \\ g(\bar{v}_k) &\geq g\big(v_k + \lambda_k(\hat{v}_k - v_k)\big) \end{aligned} \tag{4.1}$$

for some λ_k *with* $0 < \lambda_k \leq \min\{1, 1/\gamma\}$. *Then the new duality gap satisfies*

$$f(u_{k+1}) - g(v_{k+1}) \leq \theta_k[f(u_k) - g(v_k)], \tag{4.2}$$

where $\theta_k = 1 - \lambda_k + \gamma\lambda_k^2 \in (0,1]$.

Observe that the quantity $f(u_k) - g(v_k)$ in inequality (4.2) is the duality gap for the current iterates. By weak duality, this gap provides an estimate for the quality of the objective values for ($\mathcal{P}$) and ($\mathcal{D}$). Because of the quadratic terms in L, the duality gap also allows estimating the distances of the iterates u_k and v_k to the optimal solutions. This is shown in Corollary 4.4 below.

Interestingly, there is a default step-size for λ_k, which can in fact be used for both the primal and dual problems. This is given by

$$\bar{\lambda} := \min\{1, 1/2\gamma\},$$

which is the unique minimizer over $[0,1]$ of the function $\lambda \mapsto 1 - \lambda + \gamma\lambda^2$. According to Theorem 4.1, this default step-size guarantees linear convergence.

Corollary 4.2. *Suppose that in Algorithm* 2.1 *the linesearch criterion* (4.1) *is applied with the default step-size* $\lambda_k = \bar{\lambda}$ *defined above. Then the method converges at a global linear rate in the sense that the iterates satisfy*

$$f(u_{k+1}) - g(v_{k+1}) \leq \bar{\theta}[f(u_k) - g(v_k)],$$

where $\bar{\theta} \in (0,1)$ *is defined as*

$$\bar{\theta} := \begin{cases} \gamma & \text{if } \gamma < 1/2, \\ 1 - 1/4\gamma & \text{if } \gamma \geq 1/2. \end{cases}$$

The default step-size allows us to use fixed-length steps, assuming we know a value for the Lipschitz constant K. Zhu [10], [11] discusses several ways to exploit this information, including a modified form of the Armijo step.

To prove Theorem 4.1 we require a few technical lemmas.

LEMMA 4.3. *If* $\hat{u} \in \operatorname{argmin}_{[\hat{u},u]} L(\cdot, v')$, *then* $f(u) - L(\hat{u}, v') \geq (p/2)\|u - \hat{u}\|^2$.

Proof. Note first that if φ_0 is convex and $\varphi = \varphi_0 + \frac{p}{2}\|\cdot\|^2$, then

$$\varphi(\bar{x}) + \nabla\varphi(\bar{x}) \cdot (x - \bar{x}) + \frac{p}{2}\|x - \bar{x}\|^2 \leq \varphi(x).$$

Applying this to $L(\cdot, v')$ we obtain

$$\begin{aligned} f(u) - L(\hat{u}, v') &\geq L(u, v') - L(\hat{u}, v') \\ &\geq \nabla_u L(\hat{u}, v') \cdot (u - \hat{u}) + \frac{p}{2}\|u - \hat{u}\|^2 \\ &\geq \frac{p}{2}\|u - \hat{u}\|^2. \end{aligned}$$

Here the first inequality follows from the definition of f, the second from the convexity of $L(\cdot, v')$, and the last from the hypothesis. □

Lemma 4.3 leads immediately to an estimate of the distance from any point to the optimal solution, in terms of the duality gap.

COROLLARY 4.4. *Suppose* (u^*, v^*) *is the unique saddle point for* $\mathcal{S}$. *If* $f(u) - g(v) \leq \epsilon$, *then one has*

$$\|u - u^*\|^2 \leq 2\epsilon/p, \qquad \|v - v^*\|^2 \leq 2\epsilon/q.$$

Proof. Applying the preceding lemma with $\hat{u} := u$, $u := u^*$, and $v' = F(\hat{u})$ yields the inequality for $\|u - u^*\|$. The proof for $\|v - v^*\|$ is the same. □

Our next lemma estimates the quality of the lower approximation for f given by using points in the image of the duality mapping F.

LEMMA 4.5. *For any* u *and* $\bar{u}$ *one has* $f(\bar{u}) \leq L(\bar{u}, F(u)) + (K^2/2q)\|\bar{u} - u\|^2$.

Proof. By the definition of F and the concavity of $L(u, \cdot)$ we have

$$0 \geq \nabla_v L(u, F(u)) \cdot (v - F(u)) = [\nabla_v J(u, F(u)) - qF(u)] \cdot (v - F(u)) \tag{4.3}$$

for all v. Consequently,

$$\begin{aligned} &f(\bar{u}) - L(\bar{u}, F(u)) + \frac{q}{2}\|F(\bar{u}) - F(u)\|^2 \\ &\quad = L(\bar{u}, F(\bar{u})) - L(\bar{u}, F(u)) + \frac{q}{2}\|F(\bar{u}) - F(u)\|^2 \\ &\quad = J(\bar{u}, F(\bar{u})) - J(\bar{u}, F(u)) - qF(u) \cdot (F(\bar{u}) - F(u)) \\ &\quad \leq J(\bar{u}, F(\bar{u})) - J(\bar{u}, F(u)) - \nabla_v J(u, F(u)) \cdot (F(\bar{u}) - F(u)) \\ &\quad \leq \nabla_v J(\bar{u}, F(u)) \cdot (F(\bar{u}) - F(u)) - \nabla_v J(u, F(u)) \cdot (F(\bar{u}) - F(u)) \\ &\quad = [\nabla_v J(\bar{u}, F(u)) - \nabla_v J(u, F(u))] \cdot (F(\bar{u}) - F(u)), \end{aligned}$$

where the first inequality follows from (4.3) and the second from the concavity of $J(\bar{u},\cdot)$. Thus

$$\begin{aligned} f(\bar{u}) - L(\bar{u},F(u)) &\le \sup_v \left\{ (\nabla_v J(u,F(u)) - \nabla_v J(\bar{u},F(u)))\cdot v - \frac{q}{2}\|v\|^2 \right\} \\ &= \frac{1}{2q}\|\nabla_v J(u,F(u)) - \nabla_v J(\bar{u},F(u))\|^2. \end{aligned}$$

The desired inequality follows from the Lipschitz assumption for $\nabla_v J(\cdot,F(u))$. □

The above proof is the only place where we use the Lipschitz assumption. It is clear that this assumption can be weakened somewhat. For instance, it only needs to hold separately in u and v. The argument can also be extended to J with locally Lipschitz gradients, but this requires a more careful analysis of the boundedness of the iterates generated.

Our final lemma combines the earlier lemmas to justify the use of a default stepsize.

LEMMA 4.6. *Suppose that* $\hat{u} \in \operatorname{argmin}_{[\hat{u},u]} L(\cdot,v')$ *and* $L(\hat{u},F(u)) \le L(\hat{u},v')$. *Then, for any* $\lambda \in [0,1]$, *one has*

$$f\big(u+\lambda(\hat{u}-u)\big) - f(u) \le [f(u) - L(\hat{u},v')](-\lambda + \gamma\lambda^2). \tag{4.4}$$

Proof. Combining Lemma 4.5 (with $\bar{u} := u + \lambda(\hat{u}-u)$) and the convexity of $L(\cdot,F(u))$ we get

$$\begin{aligned} &f(u+\lambda(\hat{u}-u)) \\ &\quad \le L(u+\lambda(\hat{u}-u),F(u)) + (K^2/2q)\|u - [u+\lambda(\hat{u}-u)]\|^2 \\ &\quad \le L(u,F(u)) + \lambda[L(\hat{u},F(u)) - L(u,F(u))] + (K^2/2q)\|u - [u+\lambda(\hat{u}-u)]\|^2 \\ &\quad = f(u) + \lambda[L(\hat{u},F(u)) - f(u)] + (K^2/2q)\lambda^2\|\hat{u}-u\|^2. \end{aligned}$$

Thus

$$\begin{aligned} f(u+\lambda(\hat{u}-u)) - f(u) &\le -\lambda[f(u) - L(\hat{u},F(u))] + (K^2/2q)\lambda^2\|\hat{u}-u\|^2 \\ &\le -\lambda[f(u) - L(\hat{u},v')] + (K^2/2q)\lambda^2\|\hat{u}-u\|^2, \end{aligned}$$

where the second inequality is due to the hypothesis. Recalling that $\gamma = K^2/pq$ and applying Lemma 4.3 to the right-hand side above yields the inequality (4.4). □

We can now prove our convergence theorem.

Proof of Theorem 4.1. Since $(\hat{u}_k, v'_k)$ solves the left-hand minimax problem in step 2 of Algorithm 2.1, we see that $\hat{u}_k$, u_k, and v'_k satisfy the hypotheses of Lemma 4.6. Thus (4.4) holds when all the variables are subscripted by k. Combining this with (4.1) and the fact (via step 4) that $f(u_{k+1}) \le f(\bar{u}_k)$ leads to

$$f(u_{k+1}) - f(u_k) \le (-\lambda_k + \gamma\lambda_k^2)[f(u_k) - L(\hat{u}_k, v'_k)].$$

Similar arguments give

$$g(v_k) - g(v_{k+1}) \le (-\lambda_k + \gamma\lambda_k^2)[L(u'_k,\hat{v}_k) - g(v_k)].$$

Adding these one obtains

$$\begin{aligned} &f(u_{k+1}) - g(v_{k+1}) + \lambda_k(1-\gamma\lambda_k)[L(u'_k,\hat{v}_k) - L(\hat{u}_k,v'_k)] \\ &\qquad \le (1 - \lambda_k + \gamma\lambda_k^2)[f(u_k) - g(v_k)]. \end{aligned} \tag{4.5}$$

Now observe that we have

$$L(\hat{u}_k, v'_k) = \inf_{u\in U_k} \sup_{v\in V'_k} L(u,v) \leq \inf_{u\in U'_k} \sup_{v\in V_k} L(u,v) = L(u'_k, \hat{v}_k),$$

where the inequality follows from the requirement (in step 1) that $U'_k \subset U_k$ and $V'_k \subset V_k$. Consequently, we see that

$$0 \leq L(u'_k, \hat{v}_k) - L(\hat{u}_k, v'_k).$$

This, together with the assumption that $\lambda_k \leq 1/\gamma$, combines with (4.5) to yield (4.2), as desired. □

We close this section with a final observation on a slight improvement that is sometimes possible.

Corollary 4.7. *Under the hypotheses of Theorem* 4.1*, in any iteration where*

$$f(u_{k+1}) - g(v_{k+1}) \leq L(u'_k, \hat{v}_k) - L(\hat{u}_k, v'_k), \tag{4.6}$$

the decrease in the duality gap can be improved to

$$f(u_{k+1}) - g(v_{k+1}) \leq \frac{\theta_k}{2-\theta_k}[f(u_k) - g(v_k)]. \tag{4.7}$$

Proof. Combining (4.6) with (4.5) gives us (4.7). □

The inequality (4.6) in Corollary 4.7 holds in the situation where the optional restarts of step 4 are employed and the saddle points of step 2 satisfy

$$f(u'_k) = L(u'_k, \hat{v}_k), \quad g(v'_k) = L(\hat{u}_k, v'_k).$$

This is the case when $U_k = U$ and $V_k = V$, as occurs in the primal-dual steepest descent algorithm of Zhu (Algorithm 3.1 in section 3). The coefficient given here is essentially the same as that found by Zhu.

Notice that in Rockafellar's finite envelope method (Algorithm 3.2 in section 3), the right-hand side of inequality (4.6) is always zero. Consequently, the optional restarts yield no theoretical advantage in this case, nor for Algorithms 2.2 and 2.3 of section 2.

5. Conclusion. Decomposition methods are especially useful for large-scale problems consisting of many smaller optimization problems with a few coupling constraints between them. These have an inherent potential for parallelization. Thus envelope methods are appealing for generating an improved solution on the basis of a known feasible point.

Global linear convergence is guaranteed for envelope methods when applied to problems with strongly convex-concave Lagrangians. This result extends those of Rockafellar [6] and Zhu [10], [11] to a larger class of methods and objective functions. As in those papers, the close relationship to steepest descent/ascent methods makes the possibility of a superlinear convergence proof unlikely, even under the assumption of a strongly convex-concave Lagrangian.

The required strong dual concavity may be present in formulations involving augmented Lagrangians or barrier functions. For large-scale linear programs such reformulations can lead to very simple duality mappings F and G; the low overhead per iteration of the duality-descent method (Algorithms 2.2 and 2.3) makes it particularly attractive in this case. One might also combine an envelope method with a regularization scheme, as suggested by Rockafellar and Wets [5] or Ruszczyński [7]. In any event, an envelope method might be used as a "crash routine," providing a good starting guess for a method having superior local convergence but poor global convergence.

A point of interest especially deserving of attention is parallel "nested" decomposition. In the context of stochastic linear programs, several versions of the L-shaped method have been proposed which pass dual information back and forth across the underlying scenario tree [1], [2], [3], [7]. In effect, the method is applied simultaneously on several different levels of decoupling. Similar opportunities exist for the other forms of envelope methods and should be explored.

REFERENCES

[1] J. R. Birge, *Decomposition and partitioning methods for multistage stochastic linear programs*, Oper. Res., 33 (1985), pp. 989–1007.

[2] J. R. Birge, C. J. Donohue, D. F. Holmes, and O. G. Svintsitski, *A parallel implementation of the nested decomposition algorithm for multistage stochastic linear programs,* Math. Programming Ser. B, 75 (1996), pp. 327–352.

[3] A. J. King and S. E. Wright, *A Flexible Partition L-Shaped Method for Multistage Stochastic Programming*, preprint, 1998.

[4] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.

[5] R. T. Rockafellar and R. J.-B. Wets, *A Lagrangian finite generation technique for solving linear-quadratic problems in stochastic programming*, Math. Programming Stud., 28 (1986), pp. 63–93.

[6] R. T. Rockafellar, *Computational schemes for solving large-scale problems in extended linear-quadratic programming*, Math. Programming, 48 (1990), pp. 447–474.

[7] A. Ruszczyǹski, *Parallel decomposition of multistage stochastic programming problems*, Math. Programming, Ser. A, 58 (1993), pp. 201–228.

[8] S. E. Wright, *Convergence and Approximation for Primal-Dual Methods in Large-Scale Optimization*, Ph.D. thesis, University of Washington, Seattle, WA, 1990.

[9] C. Y. Zhu and R. T. Rockafellar, *Primal-dual projected gradient algorithms for extended linear-quadratic programming*, SIAM J. Optim., 3 (1993), pp. 751–783.

[10] C. Zhu, *Solving large-scale minimax problems with the primal-dual steepest descent algorithm*, Math. Programming, Ser. A, 67 (1994), pp. 53–76.

[11] C. Y. Zhu, *On the primal-dual steepest descent algorithm for extended linear-quadratic programming*, SIAM J. Optim., 5 (1995), pp. 114–128.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 415–426

THE ANALYTIC CENTER QUADRATIC CUT METHOD FOR STRONGLY MONOTONE VARIATIONAL INEQUALITY PROBLEMS*

HANS-JAKOB LÜTHI† AND BENNO BÜELER†

Abstract. Convergence of an algorithm for strongly monotone variational inequality problems (VIPs) is investigated. At each iteration, the algorithm adds a quadratic cut through the analytic center of the consequently shrinking convex set. It is shown that the sequence of analytic centers converges to the unique solution in $\mathcal{O}(1/\sqrt{k})$, where k is the number of iterations.

Key words. analytic center, quadratic cuts, variational inequalities

AMS subject classification. 90C25

PII. S1052623498338321

1. Introduction. Variational inequality problems (VIPs) provide a convenient mathematical framework for discussing a number of interesting problems such as optimization problems, saddle point problems, or equilibrium problems. It has been known for many years that a specific ellipsoid algorithm solves strongly monotone VIPs with polynomial-time complexity; see Lüthi [6].[1] In practice, however, the ellipsoid method is not convincing. More recently, Nesterov and Nemirovskii [8] suggested a path-following approach with pseudopolynomial-time complexity for a class of monotone VIPs. In this approach, higher order derivatives are required, whereas first-order information suffices in the case of the ellipsoid cutting plane method. In recent years, a number of authors have studied various linear or quadratic cut methods for solving VIPs [3, 7, 9] or convex feasibility problems [5, 2].

In particular, Nesterov and Vial [9] have introduced a homogeneous analytic center cutting plane method (HACCPM), which solves monotone VIPs in a conic setting and with pseudopolynomial-time complexity, when measured by the dual gap function. They also presented a Lipschitzian and monotone example for which ACCPM does not converge to the solution of the VIP. Here ACCPM denotes the original analytic center cutting plane method, which does not embed the problem in a conic space; see Sonnevend [10] and Goffin, Haurie, and Vial [1].

Assuming a strongly monotone and Lipschitzian operator, a first pseudopolynomial complexity bound for ACCPM was derived by Goffin, Marcotte, and Zhu [3], yielding $\|x_k - x^*\| = \mathcal{O}((\ln(k)/k)^{1/4})$. It is noteworthy, however, that in their approach the "condition" number of the initial feasible convex set directly influences the complexity bound.

While avoiding a conic embedding, an algorithm with a better complexity bound can be devised by exploiting the curvature information of strongly monotone VIPs.

*Received by the editors May 5, 1998; accepted for publication (in revised form) May 12, 1999; published electronically January 19, 2000.
http://www.siam.org/journals/siopt/10-2/33832.html

†Institute for Operations Research, Swiss Federal Institute of Technology, Zürich, Switzerland (luethi@ifor.math.ethz.ch, bueeler@ifor.math.ethz.ch).

[1] For the complexity discussion herein we assume a computer device with arbitrary precision. Namely we assume that all arithmetic operations require an amount of time which is independent of the number of digits needed in the representation of the numbers. In such a computational model the time complexity is proportional to the number of elementary arithmetic operations.

We show in this paper that the iterates of the resulting analytic center quadratic cut method (ACQCM) converge $\|x_k - x^*\| = \mathcal{O}(k^{-1/2})$ to the unique solution of the VIP.

The following notation is used. Let $A, B \in \mathbb{R}^{n\times n}$ be symmetric. We then call A p.s.d. if it is positive semidefinite and p.d. if it is positive definite. By $A \geq B$ we mean that $A - B$ is p.s.d. and by $A > B$ that $A - B$ is p.d. Furthermore, let A be p.d.; we then define $\|x\|_A := \langle x, Ax\rangle^{1/2}$ for all $x \in \mathbb{R}^n$. Finally, by $\mathbb{1}$ we denote the unit matrix of appropriate dimension.

2. Description of the ACQCM. Given a closed, convex set $P \subset \mathbb{R}^n$ and a continuous map $f : P \to \mathbb{R}^n$, we want to solve the VIP

$$\text{find } x^* \in P, \text{ such that } \langle f(x^*), x - x^*\rangle \geq 0 \quad \forall x \in P. \tag{2.1}$$

Throughout the paper we make the following assumptions.

Assumption 2.1.

(i) *P has a nonempty interior and it can be described by an m-self-concordant barrier function F_0; furthermore, without loss of generality (w.l.o.g.) P can be assumed to be compact.*

(ii) *f is bounded on P, i.e.,* $\max_{x\in P} \|f(x)\| \leq L < \infty$.

(iii) *f is strongly monotone; i.e., there exists $\alpha > 0$ such that*

$$\langle f(y) - f(x), y - x\rangle \geq \alpha\|y - x\|^2 \quad \forall x,\, y \in P. \tag{2.2}$$

(iv) $F_0''(x) > 0\ \forall x \in \operatorname{int} P$.

Note that (i) is satisfied if $P = \{x \in \mathbb{R}^n | s_i(x) \geq 0,\ i = 1, \ldots, m\}$ is defined by m concave constraints $s_i \in \mathcal{C}^3$. In this case $F_0(x) := -\sum_{i=1}^m \ln s_i(x)$ is an m-self-concordant barrier for P. As a further observation assume that the algorithm described below starts with a quadratic cut through $x_0 \in \operatorname{int}(P)$ and call the remaining part $\bar{P}$. Then continuity and strong monotonicity of f imply compactness of $\bar{P}$, and furthermore (ii) and (iv) hold as well. In particular, using this $\bar{P}$ construction we also encompass the unconstrained case, i.e., when $m = 0$.

In addition, it is well known [4] that for strongly monotone VIPs continuity of f implies the existence of a unique solution $x^* \in P$ of (2.1). Indeed, the existence of a solution x^* and hence continuity of f will not be needed for the algorithm nor for the convergence analysis, as will be discussed at the beginning of section 3.

For a definition and an in-depth discussion of self-concordant functions, see Nesterov and Nemirovskii [8]. Three properties of self-concordant barriers, which will be used in what follows, are given in the following lemma.

Lemma 2.1. *Let F be an m-self-concordant barrier with $F'' > 0$ on* $\operatorname{dom} F$, *and let $x, y \in \operatorname{dom} F$.*

(i) *Then we know from Nesterov and Vial (Lemma 9 in [9]) that*

$$\langle F'(y) - F'(x), y - x\rangle \geq \frac{\|y - x\|^2_{F''(x)}}{1 + \|y - x\|_{F''(x)}}. \tag{2.3}$$

(ii) *From Nesterov and Nemirovskii* [8, (2.2.1) and Definition 2.3.1] *we have*

$$\|F'(x)\|_{(F''(x))^{-1}} \leq \sqrt{m} \tag{2.4}$$

(iii) *and*

$$\langle F'(x), y - x\rangle \leq m\,. \tag{2.5}$$

Strong monotonicity implies that the solution x^* to (2.1) is unique and also that

$$\begin{aligned} -\langle f(x), x^* - x\rangle - \alpha\|x - x^*\|^2 &\geq -\langle f(x^*), x^* - x\rangle \\ &= \langle f(x^*), x - x^*\rangle \geq 0 \quad \forall x \in P. \end{aligned} \tag{2.6}$$

If we denote by

$$t_k(x) := -\langle f(x_k), x - x_k\rangle - \alpha\|x_k - x\|^2$$

the slack with respect to a quadratic cut through an arbitrary $x_k \in P$, then from (2.6) it immediately follows that the corresponding quadratic cut set

$$C_{x_k} := \{x \in \mathbb{R}^n \mid t_k(x) \geq 0\}$$

contains the solution x^*. C_{x_k} is characterized in the following lemma.

Lemma 2.2. *Let $x_k \in P$; then the set C_{x_k} is a ball with center*

$$z_k := x_k - \frac{f(x_k)}{2\alpha}$$

and radius

$$r_k := \frac{\|f(x_k)\|}{2\alpha}.$$

Furthermore, $z_k = \arg\max_{x \in C_{x_k}} t_k(x)$ and

$$\max_{x\in P} t_k(x) \leq \frac{L^2}{4\alpha}.$$

Proof. The characterization of the ball with center z_k and radius r_k is given by

$$\left\|(x_k - x) - \frac{f(x_k)}{2\alpha}\right\|^2 = \|x_k - x\|^2 - 2\left\langle x_k - x, \frac{f(x_k)}{2\alpha}\right\rangle + \frac{\|f(x_k)\|^2}{4\alpha^2} \leq \frac{\|f(x_k)\|^2}{4\alpha^2},$$

which is equivalent to

$$\alpha\|x - x_k\|^2 + \langle f(x_k), x - x_k\rangle \leq 0.$$

This proves the relations concerning z_k and r_k. Due to its symmetric quadratic nature $t_k(x)$ attains its maximum at z_k, and because $t_k(z_k) = \alpha r_k^2$ and $\|f(x_k)\| \leq L$ the last claim is proved. ◻

It is known (cf. Nesterov and Nemirovskii [8]) that F_0 has a unique minimizer $x_0 \in \operatorname{int}(P)$. For $k \geq 1$, consider the logarithmic barrier

$$F_k(x) := F_0(x) - \sum_{i=0}^{k-1} \ln t_i(x),$$

where x_k, needed for the definition of $t_k(x)$, is defined as the unique minimizer of $F_k(x)$ over $P_k := P \cap_{i=0}^{k-1} C_{x_i}$ and called the *analytic center of* P_k. Based on this notation, the ACQCM presented below can be used to solve strongly monotone VIPs.

ACQCM:

(i) Let $k = 0$, $P_0 := P$, F_0 as in Assumption 2.1, and choose $\varepsilon > 0$.

(ii) Compute the analytic center $x_k = \arg\min_{x\in P_k} F_k(x)$;
let $F_{k+1}(x) = F_k(x) - \ln(t_k(x))$ and $P_{k+1} = P_k \cap C_{x_k}$.

(iii) Stop if $\|x_k - x^*\| \leq \varepsilon$; otherwise set $k := k + 1$ and return to step (ii).

The convergence of ACQCM is discussed in the next section.

3. Convergence analysis.

This section is dedicated to the proof of the following theorem.

THEOREM 3.1. *Let* $\theta := \frac{1}{2}(1+\sqrt{5})$, $a := \ln\left(L\sqrt{\frac{1+\theta}{2\alpha}}\right) + \sqrt{m}\cdot\theta + 1$, *and for all* $k \geq 1, S_k := \sum_{i=0}^{k-1}(t_i(x_k))^{-1}$. *Then, we have for all* $k \geq 1$:

$$\|x_k - x^*\| \; \leq \; \sqrt{\frac{k+m}{\alpha S_k}} \tag{3.1}$$

$$\leq \; \frac{e^a}{\sqrt{\alpha k}}\sqrt{1+\frac{m}{k}}. \tag{3.2}$$

In particular, if $k \geq m$, *then*

$$\|x_k - x^*\| \; \leq \; \frac{L}{\alpha}\sqrt{\frac{1+\theta}{k}}\; e^{\sqrt{m}\cdot\theta+1}.$$

As we will see in the proof of the above theorem, x^* can be replaced by any feasible point $x \in P_k$. Hence, Theorem 3.1 simply states that at iteration k, we can bound the distance from the analytic center x_k to any feasible point $x \in P_k$ by (3.2); i.e., the diameter of P_k is at most two times the bound on $\|x_k - x\|$ as given in (3.2).

It is important to note that the continuity of f is not used in the convergence analysis. We assume it only in order to guarantee the existence of a solution x^* of (2.1). In view of this, ACQCM remains a valid algorithm for noncontinuous operators f.

For example, if we know a priori that a solution exists, or, to the contrary, if we have some structure that allows us to conclude the nonexistence of a solution given, the diameter of P_k is small enough.

Using the above theorem, the stopping criterion $\|x_k - x^*\| \leq \varepsilon$ in step (iii) of ACQCM, which cannot be measured explicitly, can therefore be replaced by two conditions: the algorithm stops either when k exceeds a bound given by L, α, and m or when $f(x_k) = 0$, which implies $P_{k+1} = \{x_k\} = \{x^*\}$.

The proof of Theorem 3.1 is based on four lemmas.

LEMMA 3.2. *Let* $\theta := \frac{1}{2}(1+\sqrt{5})$. *Furthermore, let* $\beta_k := \|x_k - x_{k-1}\|_{F''_{k-1}(x_k)}$ *and* $S_k := \sum_{i=0}^{k-1}(t_i(x_k))^{-1}$ $\forall k \geq 1$. *Then we have for all* $k \geq 1$

$$\beta_k \; \leq \; \theta \tag{3.3}$$

and

$$\|x_k - x_{k-1}\| \; \leq \; \sqrt{\frac{1+\theta}{2\alpha S_k}} \tag{3.4}$$

$$\leq \; \frac{L}{\alpha}\sqrt{\frac{1+\theta}{8}}\frac{1}{\sqrt{k}}. \tag{3.5}$$

Proof. The proof of (3.3) relies on the following two observations. First, x_k is the minimizer of the barrier F_k, hence $0 = F'_k(x_k)$; in view of the definition of the barrier this implies for $k \geq 1$ that

$$F'_{k-1}(x_k) = \frac{t'_{k-1}(x_k)}{t_{k-1}(x_k)}. \tag{3.6}$$

The second observation uses the *centrality* of the quadratic cuts, i.e., $t_{k-1}(x_{k-1}) = 0$. From the Taylor series for the quadratic function t_{k-1} we observe that

$$0 \ = t_{k-1}(x_{k-1}) = t_{k-1}(x_k) + \langle t'_{k-1}(x_k), x_{k-1} - x_k\rangle + \frac{1}{2}\langle x_{k-1} - x_k, t''_{k-1}(x_k)(x_{k-1} - x_k)\rangle.$$

Hence, from the definition of t_{k-1} we obtain for $k \geq 1$

$$\langle t'_{k-1}(x_k), x_k - x_{k-1}\rangle = t_{k-1}(x_k) - \alpha\|x_k - x_{k-1}\|^2. \tag{3.7}$$

Also, from (2.3) we know that

$$\langle F'_{k-1}(x_k) \ - \ F'_{k-1}(x_{k-1}), x_k \ - \ x_{k-1}\rangle \ \geq \ \frac{\beta_k^2}{1+\beta_k}. \tag{3.8}$$

Using $0 = F'_{k-1}(x_{k-1})$ and (3.6) we derive from (3.8)

$$\left\langle \frac{t'_{k-1}(x_k)}{t_{k-1}(x_k)}, x_k - x_{k-1} \right\rangle \ \geq \ \frac{\beta_k^2}{1+\beta_k},$$

which, by (3.7), yields

$$1 - \frac{\alpha\|x_k - x_{k-1}\|^2}{t_{k-1}(x_k)} \ \geq \ \frac{\beta_k^2}{1+\beta_k}. \tag{3.9}$$

Because $x_k \in \operatorname{int} C_{x_{k-1}}$, we know that $t_{k-1}(x_k) > 0$, and hence we obtain $1 \geq \frac{\beta_k^2}{1+\beta_k}$. By definition, $\beta_k \geq 0$, and hence the equivalent relation $\beta_k^2 - \beta_k - 1 \leq 0$ implies $0 \leq \beta_k \leq \frac{1}{2}(1+\sqrt{5})$. This proves the first statement of the lemma.

To prove (3.4) we first treat the case $k = 1$; i.e., we study $\|x_1 - x_0\|$. Note that $\beta_1 \geq 0$, which, by (3.9), implies

$$1 \ \geq \ \frac{\alpha\|x_1 - x_0\|^2}{t_0(x_1)}.$$

Due to $\theta \geq 1$, we find for $k = 1$ a first positive answer:

$$\|x_1 - x_0\|^2 \ \leq \ \frac{t_0(x_1)}{\alpha} \ \leq \ \frac{1+\theta}{2} \cdot \frac{t_0(x_1)}{\alpha} \ = \ \frac{1+\theta}{2\alpha S_1}.$$

As for $k \geq 2$, note that

$$\begin{aligned} F''_{k-1}(x_k) \ &= \ F''_0(x_k) - \sum_{i=0}^{k-2} \frac{t''_i(x_k)}{t_i(x_k)}\mathbb{1} + \sum_{i=0}^{k-2} \frac{t'_i(x_k)(t'_i(x_k))^T}{t_i^2(x_k)} \\ &\geq \ \sum_{i=0}^{k-2} \frac{2\alpha}{t_i(x_k)}\mathbb{1}, \end{aligned}$$

where the last inequality is a consequence of the positive semidefiniteness of both $F''_0(x_k)$ and $t'_i(x_k)(t'_i(x_k))^T$ together with the positivity of $t_i(x_k)$. Therefore,

$$\beta_k^2 \geq \sum_{i=0}^{k-2} \frac{2\alpha}{t_i(x_k)}\|x_k - x_{k-1}\|^2.$$

From (3.9) we have

$$1 - \frac{\alpha\|x_k - x_{k-1}\|^2}{t_{k-1}(x_k)} \;\ge\; \frac{1}{1+\theta}\sum_{i=0}^{k-2}\frac{2\alpha}{t_i(x_k)}\|x_k - x_{k-1}\|^2$$

or, after rewriting,

$$1 \;\ge\; \frac{1}{1+\theta}\left(\sum_{i=0}^{k-2}\frac{2\alpha}{t_i(x_k)} + \frac{\alpha(1+\theta)}{t_{k-1}(x_k)}\right)\|x_k - x_{k-1}\|^2,$$

which, by $1+\theta \ge 2$, finally yields

$$1 \;\ge\; \frac{1}{1+\theta}\left(\sum_{i=0}^{k-1}\frac{2\alpha}{t_i(x_k)}\right)\|x_k - x_{k-1}\|^2.$$

This proves (3.4). The relation (3.5) follows from the bound $t_i(x) \le \frac{L^2}{4\alpha}$ established in Lemma 2.2. ▯

The bound (3.5), i.e., $\|x_k - x_{k-1}\| = \mathcal{O}(k^{-1/2})$, is interesting yet does not suffice to bound $\|x_\infty - x_k\|$. We have not even shown in the above lemma that x_k is a Cauchy sequence. Nevertheless, the relations $\|x_k - x_{k-1}\| = \mathcal{O}(S_k^{-1/2}) = \mathcal{O}(k^{-1/2})$ will play a crucial role in the subsequent analysis, where a lower bound on the growth of the potential $F_k(x_k)$ is derived.

Lemma 3.3. *For $k \ge 1$, we have* $-\sum_{i=0}^{k-1}\ln t_i(x_k) \ge -\sum_{i=0}^{k-1}\ln t_i(x_{i+1}) - k\sqrt{m}\cdot\theta$.

Proof. The proof is based on

$$-\sum_{i=0}^{k-1}\ln t_i(x_k) \;=\; F_k(x_k) - F_0(x_k) \;=\; \underbrace{F_k(x_k) - F_0(x_0)}_{(*)} + \underbrace{F_0(x_0) - F_0(x_k)}_{(**)}.$$

From the definition of the barrier we have

$$F_k(x_k) \;=\; F_{k-1}(x_k) - \ln t_{k-1}(x_k) \;\ge\; F_{k-1}(x_{k-1}) - \ln t_{k-1}(x_k).$$

This in turn implies the following lower bound for (*):

$$F_k(x_k) - F_0(x_0) \;\ge\; -\sum_{i=0}^{k-1}\ln t_i(x_{i+1}). \tag{3.10}$$

To bound the term (**), note that from the convexity of F_0, we have

$$F_0(x_k) - F_0(x_{k-1}) \;\le\; \langle F_0'(x_k), x_k - x_{k-1}\rangle. \tag{3.11}$$

Given a p.d. matrix $A \in \mathbb{R}^{n\times n}$, we have for any vectors $a, b \in \mathbb{R}^n$ the relation $\langle a, b\rangle = \langle A^{-1/2}a, A^{1/2}b\rangle \le \|a\|_{A^{-1}}\|b\|_A$. Thus,

$$\langle F_0'(x_k), x_k - x_{k-1}\rangle \;\le\; \|F_0'(x_k)\|_{(F_0''(x_k))^{-1}}\|x_k - x_{k-1}\|_{F_0''(x_k)} \;\le\; \sqrt{m}\|x_k - x_{k-1}\|_{F_0''(x_k)},$$

where the last inequality is a consequence of (2.4). Since $F_0''(x_k) \le F_{k-1}''(x_k)$, we can conclude from (3.3) that

$$\|x_k - x_{k-1}\|_{F_0''(x_k)} \;\le\; \|x_k - x_{k-1}\|_{F_{k-1}''(x_k)} \;\le\; \theta.$$

In view of (3.11) we have shown $F_0(x_k) - F_0(x_{k-1}) \;\leq\; \sqrt{m}\cdot\theta$ and hence

$$F_0(x_k) - F_0(x_0) \;\leq\; k\sqrt{m}\cdot\theta. \tag{3.12}$$

With (3.10) and (3.12) as bounds for (*) and (**), respectively, the lemma follows. ☐

Based on the above two lemmas we can now establish a lower bound on the growth of $-\sum_{i=0}^{k-1}\ln t_i(x_{i+1})$, which depends only on k and some constants.

LEMMA 3.4. *For $k \geq 1$, we have $\sum_{i=0}^{k-1}\ln t_i(x_{i+1}) \;\leq\; -k\ln k + ck$, where $c = 2\,[\ln(L\sqrt{\frac{1+\theta}{2\alpha}}) + \frac{\sqrt{m}}{2}\cdot\theta + 1]$*

Proof. For convenience we abbreviate in the proof:

$$\tau := L\sqrt{\frac{1+\theta}{2\alpha}}.$$

By the arithmetic-geometric mean inequality, we can deduce from the definition of S_k that

$$S_k \geq k\exp\left(-\frac{1}{k}\sum_{i=0}^{k-1}\ln t_i(x_k)\right). \tag{3.13}$$

We bound $t_k(x_{k+1})$ from above by

$$t_k(x_{k+1}) \leq L\|x_{k+1} - x_k\|$$

and furthermore use (3.4) to bound $\|x_{k+1} - x_k\|$, yielding

$$\ln t_k(x_{k+1}) \;\leq\; \ln L + \ln\sqrt{\frac{1+\theta}{2\alpha}} - \frac{1}{2}\ln S_{k+1}.$$

Using the definition of τ and (3.13) we have

$$\ln t_k(x_{k+1}) \;\leq\; \ln\tau - \frac{1}{2}\ln(k+1) + \frac{1}{2(k+1)}\sum_{i=0}^{k}\ln t_i(x_{k+1}).$$

By Lemma 3.3 we then find

$$\ln t_k(x_{k+1}) \;\leq\; \ln\tau - \frac{1}{2}\ln(k+1) + \frac{1}{2(k+1)}\sum_{i=0}^{k}\ln t_i(x_{i+1}) + \frac{\sqrt{m}}{2}\cdot\theta. \tag{3.14}$$

If we denote $d := \ln\tau + \frac{\sqrt{m}}{2}\cdot\theta$ and define, for $k \geq 1$, $D_k = \sum_{i=0}^{k-1}\ln t_i(x_{i+1})$, then

$$\ln t_k(x_{k+1}) = D_{k+1} - D_k,$$

and (3.14) becomes

$$D_{k+1} - D_k \leq\; d - \frac{1}{2}\ln(k+1) + \frac{D_{k+1}}{2(k+1)}$$

or, equivalently,

$$D_k \geq -d + \frac{1}{2}\ln(k+1) + \left[1 - \frac{1}{2(k+1)}\right]D_{k+1}. \tag{3.15}$$

From Lemma 3.6 stated below, we conclude that

$$D_k \leq -k \ln k + 2(d+1)k$$

since $D_1 := \ln t_0(x_1) \leq 2(d+1)$, which follows from (3.5) for $k = 1$. Finally, we observe from the definitions that $c = 2(d+1)$. □

Combining Lemmas 3.3 and 3.4 immediately yields the following corollary.

COROLLARY 3.5. *Let* $a := \ln\left(L\sqrt{\frac{1+\theta}{2\alpha}}\right) + \sqrt{m} \cdot \theta + 1$; *then for all* $k \geq 1$ *we have*

$$-\sum_{i=0}^{k-1} \ln t_i(x_k) \ \geq \ k \ln k - 2ak, \tag{3.16}$$

and in particular

$$S_k \geq k^2 e^{-2a}.$$

Finally, we have to prove the bound on D_k as used in the proof of Lemma 3.4.

LEMMA 3.6. *For any sequence* $D_1, D_2, \ldots$ *satisfying* (3.15)*, the following inequalities holds for any constant* $C \geq \max\{D_1, 2(d+1)\}$*:*

$$D_k \ \leq \ -k \ln k + kC. \tag{3.17}$$

Proof. For $k = 1$, (3.17) holds, since $D_1 \leq C$. We prove the claim by induction ($k-1 \to k$) for $k > 1$. First note that (3.15) is equivalent to

$$D_k \leq \frac{2k}{2k-1}\left(D_{k-1} + d - \frac{1}{2}\ln k\right). \tag{3.18}$$

By the induction assumption (3.17),

$$D_{k-1} \leq -(k-1)\ln(k-1) + (k-1)C, \tag{3.19}$$

and so from (3.18)

$$D_k \leq \left(\frac{2k}{2k-1}\right)\left(-(k-1)\ln(k-1) + (k-1)C + d - \frac{1}{2}\ln k\right). \tag{3.20}$$

Using

$$\ln(k-1) > \ln k - \frac{1}{(k-1)},$$

which is derived from the concavity of the logarithm, we obtain from (3.20)

$$\begin{aligned}
D_k &< \frac{2k}{2k-1}\left[-\frac{1}{2}(2k-1)\ln k + 1 + (k-1)C + d\right] \\
&= -k\ln k + \frac{2k}{2k-1}(d+1+(k-1)C) \\
&\leq -k\ln k + \frac{2k}{2k-1}\left(\frac{C}{2} + (k-1)C\right) \quad (\text{using } C \geq 2(d+1)) \\
&= -k\ln k + \frac{2k}{2k-1}\cdot\frac{1}{2}(2k-1)C \\
&= -k\ln k + Ck,
\end{aligned}$$

which proves the claim. □

Using Corollary 3.5, Theorem 3.1 can now be proved.

Proof of Theorem 3.1. Since x_k is by definition the minimizer of the barrier F_k, the first-order optimality condition yields $F_k'(x_k) = 0$. In view of the definition of the barrier, this implies for $k \geq 1$ that

$$F_0'(x_k) - \sum_{i=0}^{k-1} \frac{t_i'(x_k)}{t_i(x_k)} \; = \; 0. \tag{3.21}$$

Multiplying (3.21) by $(x_k - x^*)$ yields

$$\langle F_0'(x_k), (x_k - x^*)\rangle - \sum_{i=0}^{k-1} \frac{\langle t_i'(x_k), (x_k - x^*)\rangle}{t_i(x_k)} \; = \; 0. \tag{3.22}$$

Since for the quadratic function t_i

$$\langle t_i'(x_k), (x_k - x^*)\rangle \;\leq\; t_i(x_k) - \alpha\|x^* - x_k\|^2,$$

we conclude from (3.22) and the definition of S_k that

$$\langle F_0'(x_k), (x_k - x^*)\rangle - k + \alpha S_k \|x^* - x_k\|^2 \;\leq\; 0 \tag{3.23}$$

or, by rearranging the terms,

$$\alpha S_k \|x^* - x_k\|^2 \; = k + \langle F_0'(x_k), (x^* - x_k)\rangle \leq k + m, \tag{3.24}$$

where the last inequality is a consequence of property (iii) in Lemma 2.1. This proves inequality (3.1) in Theorem 3.1.

Inequality (3.2) follows directly from (3.1) by using the lower bound on S_k as given in Corollary 3.5. The last assertion is an obvious consequence of (3.2) and the definition of the constant a. □

4. Numerical experiments.

The three examples presented in this section are all based on continuous operators f with polyhedral feasible sets. The first two examples are two-dimensional, and we picture the centers and the cuts of the algorithm. The last example is taken from [11].

Example 1. *Interior solution and integrable operator.* Let

$$\mathsf{f}(x) := \langle c, x\rangle + \frac{1}{2}\langle x, Qx\rangle, \quad \text{where} \quad c := \begin{bmatrix} -3 \\ -0.5 \end{bmatrix} \quad \text{and} \quad Q := \begin{bmatrix} 10 & 0 \\ 0 & 1 \end{bmatrix}.$$

We then have $f := \mathsf{f}'$, and $\alpha = 1$ is the smallest eigenvalue of Q. The first 10 iterations are depicted in Figure 4.1. This example demonstrates the specific advantages of quadratic cuts when the solution lies in the interior of P and the operator f is continuous on P. (Due to the compactness of P, this implies Lipschitz continuity on P.) In such a situation the radii of the quadratic cuts tend to zero and can thereby speed up convergence considerably.

Because Corollary 3.5 directly underlies the convergence proof of Theorem 3.1, it is interesting to look at $-\sum_{i=0}^{k-1} \ln t_i(x_k)$ and its lower bound $k \ln k - 2ak$, as shown in Figure 4.1. Note that $k \ln k - 2ak$ exceeds zero only for $k \geq 1.5 \cdot 10^6$. The comparison between S_k and its lower bound $k^2 e^{-2a}$ is also of interest. For example, we find $S_{10} = 21763$, whereas, based on $L := 7.5$, its lower bound is only $10^2 \cdot e^{-2a} = 6.6 \cdot 10^{-5}$.

In both cases we observe a huge gap between proved lower bound and realized quantities. This suggests that for problems with a continuous operator and a solution in the interior of P, a better convergence rate might be attainable.

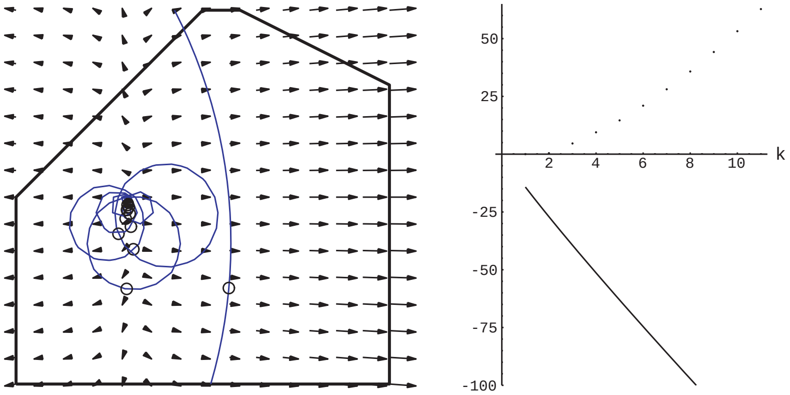


FIG. 4.1. *Left: feasibility set, vector field, and quadratic cuts; right:* $-\sum_{i=0}^{k-1} \ln t_i(x_k)$ *(dotted) and* $k \ln k - 2ak$ *(outlined).*

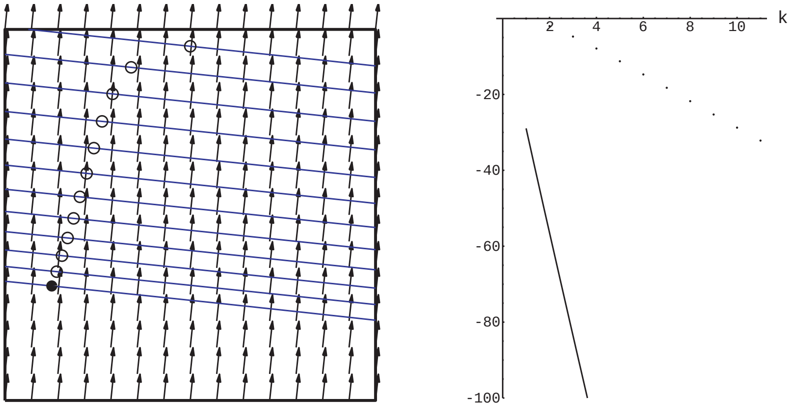


FIG. 4.2. *Left: feasibility set, vector field, and quadratic cuts; right:* $-\sum_{i=0}^{k-1} \ln t_i(x_k)$ *(dotted) and* $k \ln k - 2ak$ *(outlined).*

Example 2. *Small curvature and multiple constraints.* Let

$$f(x) := \begin{bmatrix} 10 \\ 100 \end{bmatrix} + \begin{bmatrix} 1 & 0.01 \\ 0.001 & 0.2 \end{bmatrix} x;$$

we then have $\alpha = 0.2$, the smallest eigenvalue of Q. The first 10 iterations are depicted in Figure 4.2. Here the feasible set is the unit cube, where the inequality $x_2 \geq 0$ is repeated 20 times. The solution lies at the boundary of P, but while in the previous example the radii of the quadratic cuts shrink when approaching the solution, the radii in this example are almost constant and very large; that is, in this example the quadratic cut method behaves somewhat like a linear cut method.

Concerning the relation $S_k \geq k^2 e^{-2a}$, we find $S_{10} = 0.666$, whereas, based on $L := 100$, the lower bound is only $10^2 \cdot e^{-2a} = 2.7 \cdot 10^{-11}$. Again, in this example the gap between S_k and the proven lower bound is large.

Example 3 (taken from Taji, Fukushima, and Ibaraki [11]). We tested ACQCM for two examples described in [11]. In the first example (Example 1 in [11]), the

TABLE 4.1
Example 1 *in* [11] *with* $\rho = 10$ *and* $\alpha = 0.005$.

Iteration	x_1	x_2	x_3	x_4	x_5	$\|x_k - x^*\|$	$\sqrt{\frac{k+m}{\alpha S_k}}$
0	8.6304	8.6304	8.6304	8.6304	8.6304	14.8261	
10	2.5966	2.3288	2.5078	1.5332	1.5028	1.0895	80.5572
20	2.1444	2.0312	1.9927	1.9172	1.9968	0.1695	33.5782
30	2.0114	2.0064	2.0082	1.9774	2.0048	0.0278	11.3023
40	2.0121	1.9927	2.0033	1.9906	2.0020	0.0174	3.4291
50	1.9975	1.9952	2.0007	2.0020	2.0046	0.0074	1.4724

constraint set S and the mapping F are taken, respectively, as

$$S = \left\{ x \in \mathbb{R}^5 \,\middle|\, 50 \geq \sum_{i=1}^{5} x_i \geq 10, x_i \geq 0, \quad i = 1, 2, \ldots, 5 \right\}$$

and

$$F(x) = Mx + \rho C(x) + q,$$

where M is a 5×5 asymmetric p.d. matrix and $C(x)$ is a nonlinear mapping with components $C_i(x) = \arctan(x_i - 2), i = 1, 2, \ldots, 5$. The parameter ρ is used to vary the degree of asymmetry and nonlinearity. The data for Example 1 are taken from [11]. Numerical results for this example are shown in Table 4.1.

In the second example, the constraint set S takes the form

$$S = \{x \in \mathbb{R}^n | Ax \leq b, x \geq 0\},$$

and the mapping F is given by

$$F(x) = Mx + D(x) + q,$$

where M is an $n \times n$ asymmetric p.d. matrix and $D(x)$ is a nonlinear mapping with components $D_i(x) = d_i x_i^4$, where d_i are positive constants. Again, the data for Example 2 are taken from [11]. Numerical results for this example are shown in Table 4.2.

In both cases the algorithm returned an approximate solution with $\|x_k - x^*\| \leq 10^{-2}$ after 50 quadratic cuts. Convergence may appear slow in comparison with [11], where after 5 iterations the problem was solved up to 10^{-6}. But note that for each cut we used only one evaluation of f, whereas in [11] a linearized variational problem was solved in each iteration using Lemke's LCP-algorithm (making use of the Jacobian matrix of f). In particular, if the feasible region is nonpolyhedral, then the approach in [11] is difficult to apply, whereas in our case the performance remains the same.

In Tables 4.1 and 4.2 we show the convergence of ACQCM together with its actual and estimated accuracy in terms of S_k; see (3.1).

5. Final remarks. The analysis in this paper is restricted to exact centers. In this sense the ACQCM suggested in this paper is only a conceptual algorithm. We are currently looking at approximate centers, with the aim of bounding the number of Newton steps instead of the number of iterations in ACQCM.

Table 4.2
Example 2 *in* [11] *with* $\alpha = 0.005$.

Iteration	x_1	x_2	x_3	x_4	x_5	$\|x_k - x^*\|$	$\sqrt{\frac{k+m}{\alpha S_k}}$
0	10.2860	4.0387	12.0730	4.5102	14.1316	15.8613	
10	10.7696	3.8432	1.0148	0.2303	5.3239	2.2442	412.0752
20	9.1794	4.4419	0.0249	0.0136	5.0148	0.4116	92.8726
30	9.1033	4.7810	0.0009	0.0005	5.0005	0.0634	20.9305
40	9.0616	4.8885	0.0000	0.0000	5.0000	0.0518	4.7562
50	9.0776	4.8461	0.0000	0.0000	5.0000	0.0066	0.8610

Acknowledgments. We are very grateful to M. Cochand and A. Gaillard for their careful proofreading of the manuscript, to Y. Nesterov for his constructive and helpful comments, and to L. Finschi and R. Negri for support in the final preparation of the manuscript. Furthermore, we would like to thank a referee for suggesting a simplified proof of Lemma 3.4.

REFERENCES

[1] J.-L. Goffin, A. Haurie, and J.-P. Vial, *Decomposition and nondifferentiable optimization with the projective algorithm*, Management Sci., 38 (1992), pp. 284–302.

[2] J.-L. Goffin, Z.-Q. Luo, and Y. Ye, *On the complexity of a column generation algorithm for convex or quasiconvex feasibility problems*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer, New York, 1993.

[3] J.-L. Goffin, P. Marcotte, and D. Zhu, *An analytic center cutting plane method for pseudomonotone variational inequalities*, Oper. Res. Lett., 20 (1997), pp. 1–6.

[4] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and Their Applications*, Academic Press, New York, 1980.

[5] Z.-Q. Luo and J. Sun, *An analytic center based column generation algorithm for convex quadratic feasibility problems*, SIAM J. Optim., 9 (1999), pp. 217–235.

[6] H.-J. Lüthi, *On the solution of variational inequalities by the ellipsoid method*, Math. Oper. Res., 10 (1985), pp. 515–522.

[7] T. Magnanti and G. Perakis, *A unifying geometric solution framework and complexity analysis for variational inequalities*, Math. Programming, 71 (1996), pp. 327–351.

[8] Yu. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[9] Y. Nesterov and J.-P. Vial, *Homogeneous analytic center cutting plane methods for convex problems and variational inequalities*, SIAM J. Optim., 9 (1999), pp. 707–728.

[10] G. Sonnevend, *An "analytic center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming*, in Proceedings of the 12th IFIP Conference, Budapest, 1985, A. Prekopa, J. Szelezsan, and B. Strazicky, eds., Lecture Notes in Control and Inform. Sci. 84, Springer, Berlin, 1985, pp. 866–876.

[11] K. Taji, M. Fukushima, and T. Ibaraki, *A globally convergent newton method for solving strongly monotone variational inequalities*, Math. Programming, 58 (1993), pp. 369–383.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 427–442

SIMULTANEOUS PRIMAL-DUAL RIGHT-HAND-SIDE SENSITIVITY ANALYSIS FROM A STRICTLY COMPLEMENTARY SOLUTION OF A LINEAR PROGRAM[*]

HARVEY J. GREENBERG[†]

Abstract. This paper establishes theorems about the simultaneous variation of right-hand sides and cost coefficients in a linear program from a strictly complementary solution. Some results are extensions of those that have been proven for varying the right-hand side of the primal or the dual, but not both; other results are new. In addition, changes in the optimal partition and what that means in economic terms are related to the basis-driven approach, notably to the theory of compatibility. In addition to new theorems about this relation, the transition graph is extended to provide another visualization of the underlying economics.

Key words. linear programming, sensitivity analysis, computational economics, interior point methods, parametric programming, optimal partition

AMS subject classifications. 90C05, 90C31, 49Q12

PII. S1052623496310333

1. Introduction.

Consider the primal-dual pair of linear programs:

$$P : \min\{cx : x \geq 0, Ax \geq b\}, \qquad D : \max\{\pi b : \pi \geq 0, \pi A \leq c\},$$

where x is a column vector in R^n of *levels*; b is a *column vector* in R^m of *right-hand sides*; c is a *row vector* in R^n of *objective coefficients*; π is a row vector in R^m of *prices*; and A is an $m \times n$ matrix.

This paper concerns the simultaneous variation of right-hand sides and objective coefficients (dual right-hand sides), which we call *rim data*: $r = (b, c)$. The change is of the form θh, where $\theta > 0$ and h is a nonzero *direction vector*. We have traditionally been concerned with the effect a change has on the optimality of a basis [2]. Here we suppose we have a strictly complementary solution, which is generally not basic (unless the primal-dual solution is unique). A key property of a strictly complementary solution is that it identifies the *optimal partition*. While we define this formally in the next section, it is a unique partition of the rows and columns of the linear program matrix, A, into "active" and "inactive" parts, somewhat analogous to a partition into "basic" and "nonbasic" activities. We are interested in the following questions:

- Must the optimal partition change for any positive value of θ? If so, what is the new optimal partition? If not, for what range does this partition remain optimal?
- How does this relate to basic ranges?
- How does this relate to the differential Lagrangian?
- How does the optimal objective value change as a function of θ?

Previous results [1, 10, 12] answered most of these questions when b or c change separately, but some of those proofs do not have natural extensions to deal with their simultaneous variation, and we shall consider the "decoupling principle" mentioned in [9].

[*] Received by the editors October 8, 1996; accepted for publication (in revised form) March 7, 1999; published electronically February 10, 2000.
http://www.siam.org/journals/siopt/10-2/31033.html

[†] Mathematics Department, University of Colorado at Denver, P.O. Box 173364, Denver, CO 80217-3364 (hgreenbe@carbon.cudenver.edu, http://www.cudenver.edu/~hgreenbe/).

The rest of this paper is organized as follows. In the next section, we briefly give the terms and concepts needed for the main results. (In general, the technical terms used throughout this paper are defined in the *Mathematical Programming Glossary* [6].) Then, we consider the first set of questions concerning the optimal partition, both when it does not change and when it does. In doing so, we shall relate this to the differential Lagrangian, and we shall derive the piecewise quadratic form of the objective value from a new vantage point. Finally, we relate the optimal partition change (if any) to basis-driven sensitivity analysis, notably to the *theory of compatible bases* (see [4]).

2. Terms and concepts. Let $P(b)$ and $D(c)$ denote the primal and dual polyhedra, respectively. For $(x, \pi) \in P(b) \times D(c)$, we associate *surplus variables*, $s = Ax - b$, and *reduced costs*, $d = c - \pi A$. Let $P^*(r)$ and $D^*(r)$ denote the primal and dual optimality regions, respectively, which we suppose are not empty. The *support set* of a nonnegative vector, v, is denoted $\sigma(v) = \{k : v_k > 0\}$. Then, primal-dual optimality can be represented by *complementary slackness*: $\sigma(x) \cap \sigma(d) = \emptyset$ and $\sigma(\pi) \cap \sigma(s) = \emptyset$. As shown by Goldman and Tucker [3], there must exist a *strictly complementary* solution, whereby the support sets span the rows and columns: $\sigma(\pi) \cup \sigma(s) = \{1, \ldots, m\}$ and $\sigma(x) \cup \sigma(d) = \{1, \ldots, n\}$. This defines the (unique) *optimal partition*, obtained from any strictly complementary (i.e., interior) solution.

Although the optimal partition was discovered in 1956 [3] and has been shown to be an important part of algorithm design [14, 16] and sensitivity analysis [5], it has not become familiar enough to appear in the linear programming textbooks. For that reason we consider a small example to illustrate the optimal partition and related concepts. Later, after presenting the theory in sections 3 and 4, we shall consider another example pertaining to electricity generation from competing sources.

Example. min $-x_1$: $x \geq 0$, $-x_1 \geq -b_1$, $-x_2 \geq -b_2$. The primal optimality region is the line segment, $[(b_1, 0), (b_1, b_2)]$, whose relative interior simply excludes the extreme points. The optimal partition has $\sigma(x) = \{1, 2\}, \sigma(d) = \emptyset, \sigma(s) = \{2\}$, and $\sigma(\pi) = \{1\}$. As long as c does not change, this partition remains optimal (for all $b > 0$). If c changes such that $\Delta c_2 \neq 0$, one of the two extreme points becomes uniquely optimal, and the optimal partition must change immediately. That is, suppose we have the perturbed problem

$$\min\ (-1 + \Delta c_1)x_1 + \Delta c_2 x_2\colon\ x \geq 0,\ -x_1 \geq -b_1,\ -x_2 \geq -b_2,$$

where $\Delta c_1 < 1$. Then,

$$\begin{array}{lcl} \Delta c_2 > 0 & \rightarrow & x^* = (b_1, 0), \\ \Delta c_2 < 0 & \rightarrow & x^* = (b_1, b_2). \end{array}$$

In the first case, the optimal partition changes to $\sigma(x) = \{1\}$ and $\sigma(d) = \{2\}$ (no change in $\sigma(s)$ and $\sigma(\pi)$). In the second case, we have $\sigma(s) = \emptyset$ and $\sigma(\pi) = \{1, 2\}$ (no change in $\sigma(x)$ and $\sigma(d)$).

We call the rows in $\sigma(\pi)$ *active* because they never have surplus in any optimal solution (i.e., $s_i = 0\ \forall i \in \sigma(\pi)$), and for each row we have an optimal solution where its price is positive (namely, the π obtained). Similarly, we call the columns in $\sigma(x)$ active because they never have a positive reduced cost (i.e., $d_j = 0\ \forall j \in \sigma(x)$), and for each column we have an optimal solution where its level is positive (namely, the x obtained). The complementary rows and columns are called *inactive*. The rows in $\sigma(s)$ never have a positive price, and each inactive row has a positive surplus in at least one optimal solution (namely, the s obtained). The columns in $\sigma(d)$ never have a positive level, and each inactive column has positive reduced cost in at least one optimal solution (namely, the d obtained).

In [5] several problems were presented to illustrate how the optimal partition provides the information sought, and that this is not available from just one optimal basic solution (unless it is unique). The postoptimal sensitivity analysis examples included job shop scheduling (critical path problem) and peer group identification (DEA). The examples went on to show how the optimal partition helps with debugging, such as finding irreducible infeasible subsystems or all implied equalities with less computational effort than a simplex method due to knowing when a level is positive in some optimal solution.

Partition A according to the optimal partition:

$$\begin{array}{rll}
A = & \left[\begin{array}{cc} B & N \\ B^* & N^* \end{array}\right] & \begin{array}{l} \leftarrow \sigma(\pi) = \text{rows active in some optimal solution} \\ \leftarrow \sigma(s) = \text{rows inactive in all optimal solutions} \end{array} \\
 & \begin{array}{cc} \uparrow & \uparrow \end{array} & \\
 & \quad\ \ \sigma(d) = \text{columns inactive in all optimal solutions} & \\
 & \sigma(x) = \text{columns active in some optimal solution.} &
\end{array}$$

Partition the rim data vectors conformally: $b = \binom{b_N}{b_B}$ and $c = (c_B \ \ c_N)$. Also, $x = \binom{x_B}{x_N}$, $s = \binom{s_N}{s_B}$, $\pi = (\pi_N \ \pi_B)$, and $d = (d_B \ d_N)$.

Let us extend the previous example to illustrate this notation:

$$\min \ -x_1 + x_3\colon \ x \geq 0, \ -x_1 \geq -b_1, \ -x_2 \geq -b_2, \ -x_1 - x_2 - x_3 \geq -b_3,$$

where $b_1 + b_2 > b_3 > \max\{b_1, b_2\}$. A strictly complementary optimal solution is $x = (b_1, \frac{1}{2}(b_3 - b_1), 0)^t$, $d = (0, 0, 1)$, $s = (0, b_2 - \frac{1}{2}(b_3 - b_1), \frac{1}{2}(b_3 - b_1))^t$, and $\pi = (1, 0, 0)$. The optimal partition, revealed by this solution, has only one active row, $\{1\}$ $(= \sigma(\pi))$, and two active columns, $\{1, 2\}$ $(= \sigma(x))$. Thus, the induced partitions are as follows:

$$\begin{array}{rcll}
 & & \begin{array}{cc|c} \overbrace{\qquad\qquad}^{\text{Active}} & & \overbrace{\quad}^{\text{Inactive}} \end{array} & \\
A & = & \left[\begin{array}{cc|c} -1 & 0 & 0 \\ \hline 0 & -1 & 0 \\ -1 & -1 & -1 \end{array}\right] & \begin{array}{l} \}\ \text{Active} \\ \}\ \text{Inactive,} \end{array} \quad \left(\begin{array}{c} b_1 \\ \hline b_2 \\ b_3 \end{array}\right) \ \left(\begin{array}{c} 0 \\ \hline + \\ + \end{array}\right) = s, \quad \left(\begin{array}{c} + \\ \hline 0 \\ 0 \end{array}\right) = \pi^t, \\
c & = & (\begin{array}{cc|c} -1 & 0 & 0 \end{array}), & \\
x^t & = & (\begin{array}{cc|c} + & + & 0 \end{array}), & \\
d & = & (\begin{array}{cc|c} 0 & 0 & + \end{array}). &
\end{array}$$

Using the optimal partition, the original linear programs are equivalent to the following primal-dual pair:

$$\begin{array}{ll}
\underline{\text{Primal}} & \underline{\text{Dual}} \\
\min c_B x_B + c_N x_N : & \max \pi_N b_N + \pi_B b_B : \\
\ B\, x_B + N\, x_N - s_N = b_N, & \ \pi_N B + \pi_B B^* + d_B = c_B, \\
\ B^* x_B + N^* x_N - s_B = b_B, & \ \pi_N N + \pi_B N^* + d_N = c_N, \\
\ \ x, \ s \geq 0, & \ \ \pi, \ d \geq 0.
\end{array}$$

Maintaining the partition conditions, $x_N = 0, s_N = 0, \pi_B = 0$, and $d_B = 0$, we define the following primal and dual *polyhedral conditions*, which we shall use later:

$$\begin{aligned}
\mathcal{P}(b; \overline{r}) &= \{(x_B, 0) : x_B \geq 0, \ B x_B = b_N, \ B^* x_B \geq b_B\}, \\
\mathcal{D}(c; \overline{r}) &= \{(\pi_N, 0) : \pi_N \geq 0, \ \pi_N B = c_B, \ \pi_N N \leq c_N\},
\end{aligned}$$

where the current rim data value, $\bar{r}$, determines the partition, B, N. (While $\mathcal{P}(\bar{b};\bar{r}) = P^*(\bar{r})$ and $\mathcal{D}(\bar{c};\bar{r}) = D^*(\bar{r})$, we use $\mathcal{P}(b;\bar{r})$ and $\mathcal{D}(c;\bar{r})$ to denote the same polyhedral conditions for $(b, c) \neq \bar{r}$, keeping the partition fixed at B, N.) Their relative interiors are the strictly complementary solutions:

$$\begin{aligned} ri(\mathcal{P}(b;\bar{r})) &= \{(x_B, 0) : x_B > 0,\ Bx_B = b_N,\ B^* x_B > b_B\}, \\ ri(\mathcal{D}(c;\bar{r})) &= \{(\pi_N, 0) : \pi_N > 0,\ \pi_N B = c_B,\ \pi_N N < c_N\}. \end{aligned}$$

We say $h = (\delta b, \delta c)$ is *admissible* if the linear program has an optimal solution for $r + \theta h$ for some $\theta > 0$. The set of admissible directions, say $\mathcal{H}$, is composed of those h for which the primal and dual feasibility conditions hold:

$$\mathcal{H} = \{(\delta b, \delta c) \in R^{m+n} : \exists \theta > 0, x \geq 0, \pi \geq 0 \ni Ax \geq b + \theta \delta b \text{ and } \pi A \leq c + \theta \delta c\}.$$

A basis, $\mathcal{B}$, is optimal at r if its associated primal and dual solutions are feasible. (We use $\mathcal{B}$, not to be confused with the active submatrix, B, in an optimal partition. In general, $\mathcal{B} \neq B$ unless the solution is unique.) For $h \in R^{m+n}$, we say $\mathcal{B}$ is *compatible* with h (and h with $\mathcal{B}$) if $\mathcal{B}$ is also optimal for $r + \theta h$ for some $\theta > 0$. Its *range of compatibility* is $\rho(\mathcal{B}; h) = \sup\{\theta$: $\mathcal{B}$ is optimal for $r + \theta h\}$. (Note: $\mathcal{B}$ is optimal throughout $[r, r + \rho(\mathcal{B}; h)h]$.) Let $H(\mathcal{B})$ denote the set of directions compatible with $\mathcal{B}$:

$$H(\mathcal{B}) = \{h \in R^{m+n} : \rho(\mathcal{B}; h) > 0\}.$$

One of the fundamental theorems of (basic) compatibility [4] is $\mathcal{H} = \bigcup_{\mathcal{B}} H(\mathcal{B})$. We shall relate this to a new theory of compatibility in connection with the optimal partition. Also, we denote the *basic spectrum*: $\rho^*(h) = \sup\{\rho(\mathcal{B}; h)$: $\mathcal{B}$ is optimal for $r\}$.

Given $h \in \mathcal{H}$, the objective value is $z(r + \theta h)$, as θ increases from zero. Suppose $\mathcal{B}$ is a compatible basis (one must exist) with (x, π) the associated basic solution. Then, since the basis remains optimal in $[0, \rho(\mathcal{B}; h)]$, the optimal value is quadratic:

$$z(r + \theta h) = z(r) + \theta(\delta c_{\mathcal{B}} x_{\mathcal{B}} + \pi_{\mathcal{N}} \delta b_{\mathcal{N}}) + \theta^2(\delta c_{\mathcal{B}} \mathcal{B}^{-1} \delta b_{\mathcal{N}}),$$

where $\mathcal{N}$ is the complement of $\mathcal{B}$ (following notation analogous to the partition, but induced by basic status). We shall prove a similar result holds when the optimal partition does not change.

We say z has *constant functional form* if the coefficients are constant. In particular, z has constant functional form on $[0, \rho^*(h)]$ $\forall h \in \mathcal{H}$. Further, if either $\delta b = 0$ or $\delta c = 0$, the quadratic term is zero and $z(r + \theta h) - z(r)$ is linear in θ. In this case, we call the range of θ for which z has constant functional form a *linearity interval*. It has already been proven [1, 10] that the break points of the linearity intervals correspond precisely to where the optimal partition changes (which is not necessarily the same as when the basis must change—see [7] for an example). Here we extend this to the more general rim variation, where the functional form is piecewise quadratic.

3. The optimal partition for the perturbation. Define the range for which the optimal partition does not change for a given direction (h):

$$\tau(h) \equiv \sup\{\theta : \text{ the optimal partition does not change throughout } [r,\ r + \theta h]\}.$$

In this definition, the left endpoint of the line segment is closed, so if the partition must change at r (for any $\theta > 0$), $\tau(h) = 0$. If $0 < \tau(h) < \infty$, the optimal partition is invariant on $[r,\ r + \tau(h)h)$, but it could change at $r + \tau(h)h$.

LEMMA 3.1. *Suppose $h = \{\delta b, \delta c\}$ is an admissible direction and $(\Delta b, \Delta c) = \theta h$ for $\theta > 0$ such that $r + \theta h$ has a primal-dual solution. Then, the optimal partition for $r+\theta h$ is the same as the optimal partition for r if and only if $ri(\mathcal{P}(\Delta b; r) \times \mathcal{D}(\Delta c; r)) \neq \emptyset$. Further, when the optimal partition is the same at both endpoints, it remains the same throughout the line segment, $[r,\ r + \theta h]$.*

Proof. The first part follows from the uniqueness of the optimal partition, determined by any strictly complementary solution. To show the optimal partition remains invariant on the line segment, $[r,\ r + \theta h]$, let (x^0, π^0) be a strictly complementary solution in $P^*(r) \times D^*(r)$, and let (x', π') be a strictly complementary solution in $P^*(r + \theta h) \times D^*(r + \theta h)$. Suppose $r'' = \alpha r + (1 - \alpha)(r + \theta h)$ for some $\alpha \in [0, 1]$, and define $(x, \pi) = \alpha(x^0, \pi^0) + (1 - \alpha)(x', \pi')$. Since the optimal partition for r and $r + \theta h$ is the same, we have

$$\begin{aligned} x_B &= \alpha x_B^0 + (1-\alpha)x'_B > 0 \quad \text{and} \quad x_N = \alpha x_N^0 + (1-\alpha)x'_N = 0;\\ \pi_N &= \alpha \pi_N^0 + (1-\alpha)\pi'_N > 0 \quad \text{and} \quad \pi_B = \alpha \pi_B^0 + (1-\alpha)\pi'_B = 0. \end{aligned}$$

Thus, $\sigma(x) = \sigma(x^0)$ and $\sigma(\pi) = \sigma(\pi^0)$. Further,

$$\begin{aligned} B\,x_B &= B[\alpha x_B^0 + (1-\alpha)x'_B] &&= \alpha b_N + (1-\alpha)b'_N &&= b''_N,\\ B^* x_B &= B^*[\alpha x_B^0 + (1-\alpha)x'_B] &&> \alpha b_B + (1-\alpha)b'_B &&= b''_B,\\ \pi_B B &= [\alpha \pi_B^0 + (1-\alpha)\pi'_B]B &&= \alpha c_B + (1-\alpha)c'_B &&= c''_B,\\ \pi_N N &= [\alpha \pi_N^0 + (1-\alpha)\pi'_N]N &&< \alpha c_N + (1-\alpha)c'_N &&= c''_N. \end{aligned}$$

Thus, $\sigma(s) = \sigma(s^0)$ and $\sigma(d) = \sigma(d^0)$, so (x, π) is a strictly complementary solution for the linear program defined by r'', and it has the same partition. This must therefore be the optimal partition, since it is unique. □

Suppose $h = (\delta b, \delta c)$ is an admissible direction, so $\theta^* h$ is an admissible change for some $\theta^* > 0$. If the optimal partition for $r + \theta^* h$ is the same as it is for r, Lemma 3.1 establishes that it is the same for $r + \theta h\ \forall \theta \in [0, \theta^*]$. In that case, the objective value changes with constant functional form. To see this, use the construction in the proof: $(x, \pi) = \alpha(x^0, \pi^0) + (1 - \alpha)(x', \pi')$, where (x^0, π^0) is strictly complementary for r, (x', π') is strictly complementary for $r + \theta^* h$, and $\alpha = 1 - \theta/\theta^*$. Then, since the optimal partition is the same, (x, π) is strictly complementary for $r + \theta h$, and

$$\begin{aligned} z(r + \theta h) &= (c + \theta\delta c)[(1 - \theta/\theta^*)x + \theta/\theta^* x']\\ &= z(r) + \theta[c_B(x'_B - x_B^0)/\theta^* + \delta c_B\, x_B^0] + \theta^2\, \delta c_B(x'_B - x_B^0)/\theta^*. \end{aligned}$$

This proves the following generalization of the linear case [1, 10, 12, 13].

THEOREM 3.2 (optimal value function). *If the optimal partition does not change at r for the admissible change direction h, then z has constant functional form.*

Further, Lemma 3.1 extends to the following convexity property.

THEOREM 3.3 (optimal partition convexity). *If the optimal partition is the same throughout $[r,\ r + h^1]$ as it is throughout $[r,\ r + h^2]$, it is the same throughout $[r, r + \alpha h^1 + (1 - \alpha)h^2]\ \forall \alpha \in [0, 1]$.*

Proof. Let (x^k, π^k) be a strictly complementary solution for $k = 1, 2$, so they satisfy the primal-dual conditions:

$$\begin{aligned} B\,x_B^k &= b_N + \delta b_N^k, & \pi_N^k B &= c_B + \delta c_B^k,\\ B^* x_B^k &> b_B + \delta b_B^k, & \pi_N^k N &< c_N + \delta c_N^k,\\ x_B^k &> 0,\ x_N^k = 0, & \pi_N^k &> 0,\ \pi_B^k = 0. \end{aligned}$$

Define $(x,\pi) = \alpha(x^1,\pi^1) + (1-\alpha)(x^2,\pi^2)$. Multiply the above by α for $k=1$ and by $1-\alpha$ for $k=2$ to satisfy the following for $h = \alpha h^1 + (1-\alpha)h^2 = (\Delta b, \Delta c)$:

$$\begin{array}{ll} B\,x_B = b_N + \Delta b_N, & \pi_N B = c_B + \Delta c_B, \\ B^* x_B > b_B + \Delta b_B, & \pi_N N < c_N + \Delta c_N, \\ x_B > 0,\ x_N = 0, & \pi_N > 0,\ \pi_B = 0. \end{array}$$

So, (x,π) is a strictly complementary solution for $r + \alpha h^1 + (1-\alpha)h^2$ with the same partition. It follows from Lemma 3.1 that the optimal partition remains the same throughout $[r,\ r + \alpha h^1 + (1-\alpha)h^2]$. □

In the special case that $h^1 = (\Delta b, 0)$ and $h^2 = (0, \Delta c)$, Theorem 3.3 on optimal partition convexity can be strengthened to the following *decoupling principle*.

Corollary 3.4. *The optimal partition does not change in* $[r,\ r + (\Delta b, \Delta c)]$ *if and only if it does not change in* $[r,\ r + (\Delta b, 0)] \cup [r,\ r + (0, \Delta c)]$.

Proof. If the optimal partition does not change in $[r,\ r + (\Delta b, \Delta c)]$, the following primal-dual system has a solution:

$$\begin{array}{ll} B\,x_B = b_N + \Delta b_N, & \pi_N B = c_B + \Delta c_B, \\ B^* x_B > b_B + \Delta b_B, & \pi_N N < c_N + \Delta c_N, \\ x_B > 0, & \pi_N > 0. \end{array}$$

Let (x', π') be a solution, and let (x^0, π^0) be a strictly complementary solution for r. Then, (x', π^0) is a strictly complementary solution for $r + (\Delta b, 0)$, and (x^0, π') is a strictly complementary solution for $r + (0, \Delta c)$. These imply that the optimal partition does not change in $[r,\ r + (\Delta b, 0)] \cup [r,\ r + (0, \Delta c)]$.

Conversely, if the optimal partition does not change in $[r,\ r + (\Delta b, 0)]$, there exists x' to satisfy the primal conditions, and if the optimal partition does not change in $[r,\ r + (0, \Delta c)]$, there exists π' to satisfy the dual conditions. Since the partitions are the same, (x', π') is a strictly complementary solution for $r + (\Delta b, \Delta c)$, so the partition is the same throughout $[r,\ r + (\Delta b, \Delta c)]$. □

Let the optimal partition be *compatible* with h (and h with it) if $\tau(h) > 0$. Define the *set of compatible directions*: $H = \{h : \tau(h) > 0\}$. Then, we have the following analogy to the *basis compatibility convexity theorem* (see [4]).

Theorem 3.5 (partition compatibility). *The following properties hold for H and τ.*

(1) *H is a nonempty convex cone.*

(2) *τ is quasi-concave on H; i.e., $\tau(\alpha h^1 + (1-\alpha)h^2) \geq \min\{\tau(h^1), \tau(h^2\}$ for $h^1, h^2 \in H$ and $\alpha \in [0,1]$.*

(3) *H satisfies the decoupling principle; i.e., $(\delta b, \delta c) \in H$ if and only if $(\delta b, 0) \in H$ and $(0, \delta c) \in H$.*

Proof. (1) Suppose $h^1, h^2 \in H$ and define $\theta^* = \min\{\tau(h^1), \tau(h^2)\} > 0$. Then, for $\theta \in (0, \theta^*)$, $\exists (x^k, \pi^k)$ to satisfy the strictly complementary primal-dual conditions:

$$\begin{array}{ll} B\,x^k_B = b_N + \theta\delta b^k_N, & \pi^k_N B = c_B + \theta\delta c^k_B, \\ B^* x^k_B > b_B + \theta\delta b^k_B, & \pi^k_N N < c_N + \theta\delta c^k_N, \\ x^k_B > 0, & \pi^k_N > 0 \end{array}$$

for $k = 1, 2$. Define $(x,\pi) = \frac{1}{2}(x^1,\pi^1) + \frac{1}{2}(x^2,\pi^2)$, then multiply the above by $\frac{1}{2}$ and sum to obtain the following:

$$\begin{aligned} B\,x_B &= b_N + \tfrac{1}{2}\theta\delta b_N, & \pi_N B &= c_B + \tfrac{1}{2}\theta\delta c_B,\\ B^* x_B &> b_B + \tfrac{1}{2}\theta\delta b_B, & \pi_N N &< c_N + \tfrac{1}{2}\theta\delta c_N,\\ x_B &> 0, & \pi_N &> 0. \end{aligned}$$

Define $\theta' = \frac{1}{2}\theta$ and $\theta'^* = \frac{1}{2}\theta^*$, and we have the desired result: the optimal partition for $r + \theta'(h^1 + h^2)$ is the same as the optimal partition for r, so $h^1 + h^2 \in H$. To show that H is nonempty, let $h = (b, c)$, so $r + \theta h = (1+\theta)r$. Then, by rescaling ($x' = x/(1+\theta)$ and $\pi' = \pi/(1+\theta)$), the strictly complementary solution has the same partition for all $\theta \geq 0$.

(2) Let $\theta^* \equiv \min\{\tau(h^1), \tau(h^2\} > 0$ and $(x,\pi) = \alpha(x^1,\pi^1) + (1-\alpha)(x^2,\pi^2)$. For $\theta \in (0,\theta^*)$, multiply the first system ($k = 1$) by α, the second ($k = 2$) by $1-\alpha$, and sum to prove that (x,π) is a strictly complementary solution for the partition:

$$\begin{aligned} B\,x_B &= b_N + \theta\delta b_N, & \pi_N B &= c_B + \theta\delta c_B,\\ B^* x_B &> b_B + \theta\delta b_B, & \pi_N N &< c_N + \theta\delta c_N,\\ x_B &> 0, & \pi_N &> 0. \end{aligned}$$

Thus, $\tau(\alpha h^1 + (1-\alpha)h^2) \geq \sup\{\theta : \theta < \theta^*\} = \theta^*$.

(3) Let $h = (\delta b, \delta c) \in H$. Then, $\exists \theta^* > 0$ such that for $\theta \in [0,\theta^*)$, the primal-dual conditions have a strictly complementary solution, say, (x,π) (with the same partition). Let (x^0,π^0) be a strictly complementary solution for r. Then, since these have the same partition, (x,π^0) is a strictly complementary solution for $r+\theta(\delta b, 0)$ and (x^0,π) is a strictly complementary solution for $r+\theta(0,\delta c)$. Conversely, if (x,π^0) is a strictly complementary solution for $r+\theta(\delta b,0)$ and (x^0,π) is a strictly complementary solution for $r+\theta(0,\delta c)$, both having the partition defined by B, it follows that (x,π) is a strictly complementary solution for $r+\theta(\delta b,\delta c)$. □

Now suppose that h is an admissible direction, but the optimal partition changes: $ri(\mathcal{P}(r+\theta h) \times \mathcal{D}(r+\theta h)) = \emptyset \ \forall\ \theta > 0$. The following theorem shows the fundamental relationship the new partition has with the differential linear programs that comprise Mills's differential Lagrangian [11] when A does not change. (Mills's theorem was extended [15, 8] to apply to any linear program, rather than the special case of a game.) Further, this theorem applies generally, even if the optimal partition does not change. The new result is found in part (3), and the proofs [1, 10] of parts (1) and (2) do not extend. (They are included here for self-containment.)

Theorem 3.6 (optimal partition perturbation). *Suppose (x^0,π^0) is a strictly complementary solution for r and $(\delta b,\delta c)$ is an admissible direction. Define the differential linear programs:*

$$\delta P : \min\{(\delta c)x : x \in P^*(r)\}, \qquad \delta D : \max\{\pi(\delta b) : \pi \in D^*(r)\}.$$

Let x^* and π^* be respective strictly complementary solutions. There exists $\theta^* > 0$ such that the following are true for $\theta \in (0,\theta^*)$.

(1) *The optimal partition for $r+\theta(\delta b,0)$ is the same as the optimal partition for δD, and $z(r+\theta(\delta b,0)) = z(r) + \theta\pi^*_N(\delta b_N)$.*

(2) *The optimal partition for $r+\theta(0,\delta c)$ is the same as the optimal partition for δP, and $z(r+\theta(0,\delta c)) = z(r) + \theta(\delta c_B)x^*_B$.*

(3) *The optimal partition for $r+\theta(\delta b,\delta c)$ is determined by $\sigma(x^*)$ from δP and $\sigma(\pi^*)$ from δD. Further, $z(r+\theta(\delta b,\delta c)) = z(r) + \theta(\delta c_B\, x^*_B + \pi^*_N\, \delta b_N) + \theta^2(\delta c_B B^+ \delta b_N)$, where B^+ is any generalized inverse of B.*

Proof. (1) The following proof is from Jansen, Roos, and Terlaky [10]. The dual of δD is $\min\{c\xi : B\xi_B + N\xi_N \geq \delta b_N, \xi_N \geq 0\}$. Since δD has an optimal solution, there is a strictly complementary optimum, say, (ξ, π^*). Consider $x = x^0 + \theta\xi$. Since $x^0_B > 0$, there exists $\theta' > 0$ for which $x_B > 0$ for $\theta \in [0, \theta')$. Further, $x_N = \theta\xi_N \geq 0$, so $x \geq 0$, and we have $[B\ N]x = Bx^0_B + \theta(B\xi_B + N\xi_N) \geq b_N + \theta\delta b_N$. Further, $[B^*\ N^*]x = B^*x^0_B + \theta(B^*\xi_B + N^*\xi_N)$. Since $B^*x^0_B > b_B$, there exists $\theta'' > 0$ such that $[B^*\ N^*]x > b_B + \theta\delta b_B$ for $\theta \in [0, \theta'')$. Let $\theta^* = \min\{\theta', \theta''\} > 0$. So far, we have that (x, π^*) satisfies the primal-dual conditions $\forall\ \theta \in [0, \theta^*)$:

$$\begin{array}{ll} B\,x^0_B + \theta(B\,\xi_B + N\,\xi_N) \geq b_N + \theta\delta b_N, & \pi^*_N B = c_B, \\ B^* x^0_B + \theta(B^*\xi_B + N^*\xi_N) > b_B + \theta\delta b_B, & \pi^*_N N \leq c_N, \\ x^0_B + \theta\xi_B > 0,\ \theta\xi_N \geq 0, & \pi^*_N\ \geq 0. \end{array}$$

We now prove that (x, π^*) is a strictly complementary solution for $r + \theta(\delta b, 0)$, where $\theta > 0$. Suppose $B_{i\bullet}x_B + N_{i\bullet}x_N = b_i + \theta\delta b_i$. Since $B_{i\bullet}x_B = b_i$, we must have $B_{i\bullet}\xi_B + N_{i\bullet}\xi_N = \delta b_i$. This implies $\pi^*_i > 0$ since (ξ, π^*) is strictly complementary for δD and its dual, so $\sigma(\pi^*) = \sim\sigma(s')$. Also, since (ξ, π^*) is strictly complementary, $\sigma(d^*) = \sim \sigma(\xi_N) \cap \sim \sigma(x^0_B) = \sim (\sigma(\xi_N) \cup \sigma(x^0_B)) = \sim \sigma(x')$. Thus, we have proven (x', π^*) is a strictly complementary solution for $r + \theta(\delta b, 0)$, with the same optimal partition as D, for all $\theta \in (0, \theta^*)$. Further, $z(r + \theta(\delta b, 0)) = cx' = cx + \theta c\xi$. We have $cx = z(r)$ and $c\xi = \pi^*(\delta b)$ (from duality), so $z(r + \theta(\delta b, 0)) = z(r) + \theta\pi^*(\delta b)$. Since $\pi_B = 0\ \forall\ \pi \in D^*(r)$, we conclude $z(r + \theta(\delta b, 0)) = z(r) + \theta\pi^*_N(\delta b_N)$. The proof of (2) is similar by constructing $\pi = \pi^0 + \theta\xi'$, where ξ' is the vector of variables for the dual of $\delta P : \max\{\xi' b : \xi'_B \geq 0,\ \xi'_N B + \xi'_B B^* \leq \delta c_B\}$.

We now prove (3). From (1), there exists $\theta' > 0$ such that the optimal partition does not change throughout $(r,\ r + \theta'(\delta b, 0))$, and the set of active columns is $\sigma(x^*)$. (Note from the proof of (1) that the set of active rows does not change.) From (2), there exists $\theta'' > 0$ such that the optimal partition does not change throughout $(r,\ r + \theta''(0, \delta c))$, and the set of active rows is $\sigma(\pi^*)$. (By analogy, the proof of (2) shows that the set of active columns does not change.) Let $\theta^* = \min\{\theta', \theta''\} > 0$. Then, the optimal partition does not change throughout $(r,\ r + \theta^*(\delta b, \delta c))$, and its active sets are the rows in $\sigma(\pi^*)$ and the columns in $\sigma(x^*)$.

(Note: We cannot use the solutions in (1) and (2) directly because (x, π) need not be complementary, in which case it is not a solution for $r + \theta h$. This proof can be viewed as first moving to $r + \theta(\delta b, 0)$, where $\theta < \theta^*$ and the optimal partition is defined by $\sigma(x^*)$ and $\sigma(\pi)$, then changing c by $\theta\delta c$ to move to $r + \theta h$, where the optimal partition is defined by $\sigma(x^*)$ and $\sigma(\pi^*)$. Equivalently, we can move first to $r + \theta(0, \delta c)$, with optimal partition defined by $\sigma(x)$ and $\sigma(\pi^*)$, then move to $r + \theta h$ to obtain the same result. This argument is similar to the one used by Roos [13] for a different result.)

Finally, to show that $z(r + \theta h)$ has the asserted quadratic form, we shall use the defining properties of generalized inverses. Let B correspond to the optimal partition throughout $(r,\ r + \theta^* h)$. Then, $x_B(\theta) = B^+(b_N + \theta\delta b_N) + (I - B^+B)v(\theta)$, where B^+ is any generalized inverse of B, and $v(\theta)$ is any vector in $R^{|\sigma(x)|}$. The defining property of B^+ is that $BB^+B = B$, and a fundamental property is that the equation has a solution if and only if $BB^+(b_N + \theta\delta b_N) = b_N + \theta\delta b_N$. Since this applies to $\theta = 0$, we must have $BB^+b_N = b_N$, which then implies we must also have $BB^+\delta b_N = \delta b_N$. Similarly, the dual equations are $\pi_N(\theta)B = c_B + \theta\delta c_B$, so we must have $\pi_N(\theta) = (c_B + \theta\delta c_B)B^+ + u(\theta)(I - BB^+)$, where $u(\theta)$ is any vector in $R^{|\sigma(\pi)|}$.

Then,

$$\begin{aligned} z(r+\theta h) &= (c_B+\theta\delta c_B)[B^+(b_N+\theta\delta b_N)+(I-B^+B)v(\theta)] \\ &= c_BB^+b_N+\theta(\delta c_BB^+b_N+c_BB^+\delta b_N)+\theta^2(\delta c_BB^+\delta b_N), \end{aligned}$$

where the terms with $v(\theta)$ are zero because $c_B+\theta\delta c_B=(c_B+\theta\delta c_B)B^+B$, so

$$\begin{aligned} (c_B+\theta\delta c_B)(I-B^+B)v(\theta) &= (c_B+\theta\delta c_B)B^+B(I-B^+B)v(\theta) \\ &= (c_B+\theta\delta c_B)B^+(B-BB^+B)v(\theta) \\ &= 0. \end{aligned}$$

(The last equation follows from $B=BB^+B$.) □

Example. min $x_1+3x_2+2x_3 : x\geq 0, x_1+x_2\geq 1, x_2+x_3\geq 1$.

A strictly complementary optimal solution is $x=(\frac{1}{2},\frac{1}{2},\frac{1}{2})$ and $\pi=(1,2)$, so the optimal partition has $\sigma(x)=\{1,2,3\}$ and $\sigma(\pi)=\{1,2\}$, which gives the optimality regions:

$$\begin{aligned} &P^*(r)=\{x : x\geq 0,\ x_1+x_2=1,\ x_2+x_3=1\}=\{(1-x_2,x_2,1-x_2) : 0\leq x_2\leq 1\}, \\ &D^*(r)=\{\pi : \pi\geq 0,\ \pi_1=1,\ \pi_1+\pi_2=3,\ \pi_2=2\}=\{(1,2)\}. \end{aligned}$$

For $\delta b=(-1,0)$ and $\delta c=(-1,0,0)$, the two differential linear programs and their duals are as follows:

$$\begin{aligned} &\delta P : \min\{-x_1 : x\in P^*(r)\}, && \max\{\xi_1'+\xi_2' : \xi_1'\leq -1, \xi_1'+\xi_2'\leq 0, \xi_2'\leq 0\}, \\ &\delta D : \max\{-\pi_1 : \pi\in D^*(r)\}, && \min\{\xi_1+3\xi_2+2\xi_3 : \xi_1+\xi_2\geq -1, \xi_2+\xi_3\geq 0\}. \end{aligned}$$

A strictly complementary solution for δP and its dual is $x^*=(1,0,1)$ and $\xi'=(-1,0)$, so the optimal partition has $\sigma(x^*)=\{1,3\}$. A strictly complementary solution for δD and its dual is $\pi^*=(1,2)$ and $\xi=(-2,1,-1)$, so its optimal partition has $\sigma(\pi^*)=\{1,2\}$. As given in Theorem 3.6 on optimal partition perturbation, $\sigma(x)=\sigma(x^*)$ from δP, and $\sigma(\pi)=\sigma(\pi^*)$ from δD for the optimal partition in $(r,\ r+\theta h)$. Let us verify this.

The perturbed linear program is the following primal-dual pair:

$$\begin{aligned} &\min\ (1-\theta)x_1+3x_2+2x_3 : x\geq 0, && \max\ \pi_1(1-\theta)+\pi_2 : \pi\geq 0, \\ &\quad x_1+x_2\geq 1-\theta,\ x_2+x_3\geq 1, && \quad \pi_1\leq 1-\theta,\ \pi_1+\pi_2\leq 3, \\ & && \quad \pi_2\leq 2. \end{aligned}$$

For $\theta\in(0,1)$ a strictly complementary optimal solution is $x=(1-\theta,0,1)$ (so $s=0$) and $\pi=(1-\theta,2)$ (so $d=(0,\theta,0)$). Indeed, $\sigma(x)=\{1,3\}$ and $\sigma(\pi)=\{1,2)$.

4. Relation to basic compatibility. Now we develop a range theory for the optimal partition analogous to the *range of basic compatibility* [4]. Here the optimal partition can change initially but must then remain invariant. Let $\Upsilon(h)$ denote the greatest value of θ for which the optimal partition does not change throughout $(r,\ r+\theta h)$ for $h\in\mathcal{H}$. Note that the line segment is open, so the optimal partition need not be the same at the endpoints. In particular, the partition might have to change at r (i.e., $\tau(h)=0$); otherwise, $\Upsilon(h)=\tau(h)$. The optimal partition perturbation theorem (Theorem 3.6) tells us that $\Upsilon(h)>0$ when h is admissible, in which case $z(r+\theta h)$ has constant functional form for $\theta\in(0,\Upsilon(h))$. When h is decoupled (i.e., $\delta b=0$ or $\delta c=0$), $(0,\Upsilon(h))$ is a linearity interval of $z(r+\theta h)-z(r)$.

The following lemma says that this bounds each basic range of compatibility, which establishes the optimal partition range theorem (Theorem 4.2).

LEMMA 4.1. *Suppose h is an admissible direction for which $\mathcal{B}$ is a compatible basis with range $\rho = \rho(\mathcal{B}; h)$. Then, the optimal partition does not change throughout $(r,\ r+\rho h)$.*

Proof. From the optimal partition perturbation theorem (Theorem 3.6), there exists $\theta > 0$ such that the optimal partition does not change in $(r,\ r+\theta h)$. Let θ^* be the supremum value of θ for which this is true. If $\theta^* \geq \rho$, we are done, so suppose $\theta^* < \rho$. Let (x^0, π^0) be any strictly complementary solution for $r + \frac{1}{2}\theta^* h$, so that $\sigma(x^0)$ and $\sigma(\pi^0)$ determine the optimal partition throughout $(r,\ r+\theta^* h)$. We shall reach a contradiction by constructing (x', π') that is optimal for $r + \theta h$, where $\theta^* < \theta < \rho$, and $\sigma(x') = \sigma(x^0), \sigma(s') = \sigma(s^0), \sigma(\pi') = \sigma(\pi^0)$, and $\sigma(d') = \sigma(d^0)$.

Define $\alpha = (\rho - \theta)/\rho$, so $1 - \alpha = \theta/\rho$ and $\alpha \in (0, 1)$. We shall form a convex combination of the strictly complementary solution and the basic solution response values, which we shall prove is feasible and has the same support sets as the strictly complementary solution. Suppose the basic solution for r, $(\overline{x}, \overline{\pi})$, changes by $(\Delta x, \Delta \pi)$, for $r + \rho h$. Then, define the following convex combination:

$$x = \alpha x^0 + (1-\alpha)(\overline{x} + \Delta x) \quad \text{and} \quad \pi = \alpha \pi^0 + (1-\alpha)(\overline{\pi} + \Delta \pi).$$

Clearly, $(x, \pi) \geq 0$. Further, we have $\Delta x_{\mathcal{B}} = \rho \mathcal{B}^{-1} \delta b_{\mathcal{N}}$ and $\Delta x_{\mathcal{N}} = 0$, so the primal equations are given by

$$\begin{aligned}
\mathcal{B}\, x_{\mathcal{B}} + \mathcal{N}\, x_{\mathcal{N}} &= \mathcal{B}[\alpha x^0_{\mathcal{B}} + (1-\alpha)(\overline{x}_{\mathcal{B}} + \rho \mathcal{B}^{-1} \delta b_{\mathcal{N}})] + \alpha \mathcal{N} x^0_{\mathcal{N}} \\
&= \alpha[\mathcal{B} x^0_{\mathcal{B}} + \mathcal{N} x^0_{\mathcal{N}}] + (1-\alpha)[\mathcal{B}\overline{x}_{\mathcal{B}} + \rho \delta b_{\mathcal{N}}] \\
&\geq \alpha b_{\mathcal{N}} + (1-\alpha) b_{\mathcal{N}} + \theta \delta b_{\mathcal{N}} \\
&= b_{\mathcal{N}} + \theta \delta b_{\mathcal{N}},
\end{aligned}$$

$$\begin{aligned}
\mathcal{B}^* x_{\mathcal{B}} + \mathcal{N}^* x_{\mathcal{N}} &= \mathcal{B}^*[\alpha x^0_{\mathcal{B}} + (1-\alpha)(\overline{x}_{\mathcal{B}} + \rho \mathcal{B}^{-1} \delta b_{\mathcal{N}})] + \alpha \mathcal{N}^* x^0_{\mathcal{N}} \\
&= \alpha[\mathcal{B}^* x^0_{\mathcal{B}} + \mathcal{N}^* x^0_{\mathcal{N}}] + (1-\alpha)\mathcal{B}^*[\overline{x}_{\mathcal{B}} + \Delta x_{\mathcal{B}}] \\
&\geq \alpha b_{\mathcal{B}} + (1-\alpha)(b_{\mathcal{B}} + \rho \delta b_{\mathcal{B}}) \\
&= b_{\mathcal{B}} + \theta \delta b_{\mathcal{B}}.
\end{aligned}$$

Thus, $Ax \geq b + \theta \delta b$, which proves x is feasible in the primal. Similarly, $\Delta \pi_{\mathcal{N}} = \rho \delta c_{\mathcal{B}} \mathcal{B}^{-1}$ and $\Delta \pi_{\mathcal{B}} = 0$, so the dual equations are given by

$$\begin{aligned}
\pi_{\mathcal{N}} \mathcal{B} + \pi_{\mathcal{B}} \mathcal{B}^* &= [\alpha \pi^0_{\mathcal{N}} + (1-\alpha)(\overline{\pi}_{\mathcal{N}} + \rho \delta c_{\mathcal{B}} \mathcal{B}^{-1})]\mathcal{B} + \alpha \pi^0_{\mathcal{B}} \mathcal{B}^* \\
&= \alpha[\pi^0_{\mathcal{N}} \mathcal{B} + \pi^0_{\mathcal{B}} \mathcal{B}^*] + (1-\alpha)[\overline{\pi}_{\mathcal{N}} \mathcal{B} + \rho \delta c_{\mathcal{B}}] \\
&\leq \alpha c_{\mathcal{B}} + (1-\alpha)(c_{\mathcal{B}} + \rho \delta c_{\mathcal{B}}) \\
&= c_{\mathcal{B}} + \theta \delta c_{\mathcal{B}},
\end{aligned}$$

$$\begin{aligned}
\pi_{\mathcal{N}} \mathcal{N} + \pi_{\mathcal{B}} \mathcal{N}^* &= [\alpha \pi^0_{\mathcal{N}} + (1-\alpha)(\overline{\pi}_{\mathcal{N}} + \rho \delta c_{\mathcal{B}} \mathcal{B}^{-1})]\mathcal{N} + \alpha \pi^0_{\mathcal{B}} \mathcal{N}^* \\
&= \alpha[\pi^0_{\mathcal{N}} \mathcal{N} + \pi^0_{\mathcal{B}} \mathcal{N}^*] + (1-\alpha)[\overline{\pi}_{\mathcal{N}} + \Delta \pi_{\mathcal{N}}]\mathcal{N} \\
&\leq \alpha c_{\mathcal{N}} + (1-\alpha)(c_{\mathcal{N}} + \rho \delta c_{\mathcal{N}}) \\
&= c_{\mathcal{N}} + \theta c_{\mathcal{N}}.
\end{aligned}$$

Thus, $\pi A \le c + \theta \delta c$, which proves π is feasible in the dual.

We have proven that (x, π) satisfies the primal-dual conditions for $r + \theta h$. We now prove its support sets are the same as those of (x^0, π^0). Let β_j denote the jth row of $\mathcal{B}^{-1}$. For a nonbasic activity (j), $x_j = \alpha x_j^0$, so $j \in \sigma(x)$ if and only if $j \in \sigma(x^0)$. For a basic activity (j), $x_j = \alpha x_j^0 + (1-\alpha)(\overline{x}_j + \rho \beta_j \delta b_{\mathcal{N}})$. For $j \in \sigma(x^0)$, we have $x_j > 0$ because $\overline{x}_j + \rho \beta_j \delta b_{\mathcal{N}} \ge 0$, so $\sigma(x^0) \subseteq \sigma(x)$. Now suppose $j \in \sigma(x)$, so

$$0 < x_j = \alpha x_j^0 + (1-\alpha)\overline{x}_j + \theta \beta_j \delta b_{\mathcal{N}}.$$

We shall prove that $x_j^0 = 0$ leads to a contradiction. Upon so doing, we will have proven $\sigma(x) \subseteq \sigma(x^0)$, thus proving $\sigma(x) = \sigma(x^0)$.

The contradiction comes from the meaning of the optimal partition: every optimal solution, say, $x^*(\lambda)$, for $r + \lambda h$ $(\lambda \in (0, \theta^*))$, must have $x_j^*(\lambda) = 0 \, \forall j \in \sigma(x^0)$. One such optimal solution is the basic one: $\overline{x}_j + \lambda \beta_j \delta b_{\mathcal{N}} = 0$. Since this must hold for all $\lambda \in (0, \theta^*)$, we must have $\overline{x}_j = 0$ and $\beta_j \delta b_N = 0$, so we reach the contradiction: $x_j = 0$. Hence, $\sigma(x) = \sigma(x^0)$. The remaining support set equalities follow in a similar manner. □

The opposite inequality does not hold. The optimal partition can be invariant on $(r,\, r + \theta^* h)$, but the optimal bases at r may have a range far less than θ^*. For example, consider the following linear program:[1]

$$\min x_2 : (x_1, x_2) \ge 0,\ 1 \le x_1 + x_2 \le 4,\ -1 \le x_1 - x_2 \le 2,\ \theta \le x_2 \le 2.$$

For $\theta \in [0, 2]$, the strictly complementary solution is $(\frac{3}{2}, \theta)$, and the optimal bases correspond to two extreme points, starting with $(1, 0)$ and $(2, 0)$ at $\theta = 0$. No matter which compatible basis is used, $\rho^*(h) = 1$, stopped by the turning point at $x_2 = 1$ when $\theta = 1$. Thus, the optimal partition does not change throughout $[r,\, r + 2h]$, but there is no basis that is optimal at r and at $r + \theta h$ for $\theta > 1$.

THEOREM 4.2 (optimal partition range). $\Upsilon(h) \ge \rho^*(h)$.

Proof. This is immediate from Lemma 4.1. □

Theorem 4.2 says that the range of the perturbation for which the (possibly new) partition remains the same is at least as great as the maximum of the ranges of basic compatibility, taken over all optimal bases. Thus, the associated interval for which $z(r + \theta h) - z(r)$ has constant functional form in θ is determined by when the optimal partition changes, which could be strictly greater than the basic spectrum. This generalizes the linear case (where h is decoupled).

Using the previous example, there are three optimal bases, as follows, with compatibility conditions following the semicolons:

$$\begin{array}{lllll}
\mathcal{B}^1 = [A_{\bullet 1}\ A_{\bullet 2}]: & x^1 = (0,1,0), & \pi^1 = (1,2); & \delta b_1 - \delta b_2 \ge 0, & \delta c_1 - \delta c_2 + \delta c_3 \ge 0. \\
\mathcal{B}^2 = [A_{\bullet 1}\ A_{\bullet 3}]: & x^2 = (1,0,1), & \pi^2 = (1,2); & & \delta c_1 - \delta c_2 + \delta c_3 \le 0. \\
\mathcal{B}^3 = [A_{\bullet 2}\ A_{\bullet 3}]: & x^3 = (0,1,0), & \pi^3 = (1,2); & -\delta b_1 + \delta b_2 \ge 0, & \delta c_1 - \delta c_2 + \delta c_3 \ge 0.
\end{array}$$

For $\delta b = (-1, 0)$ and $\delta c = (-1, 0, 0)$, only $\mathcal{B}^2$ is compatible, and its range of compatibility is $\rho(\mathcal{B}^2; h) = \rho^*(h) = 1$. Thus, the basic compatibility theorem of [4] tells us that $z(r + \theta h) - z(r)$ has constant functional form if we decrease b_1 and c_1, both at

[1] The author thanks Tamás Terlaky for pointing this out and Kees Roos for the example.

	Purchase			Generate					
	PCL	POL	PUR	GCL	GOL	GUR			
COST	18	15	20	.8	.6	.4	=	min	
BCL	1			−1			≥	0	balance coal
BOL		1			−1		≥	0	balance oil
BUR			1			−1	≥	0	balance uranium
LNU						−1	≥	−10	limit nuclear generation
DEL				.33	.3	.4	≥	10	demand electricity

FIG. 1. *Electricity generation example.*

unit rate. In particular, we have the following quadratic function for $\theta \in [0, 1]$:

$$\begin{aligned} z(r+\theta h) - z(r) &= (c_{\mathcal{B}} + \theta\delta c_{\mathcal{B}})[\mathcal{B}^2]^{-1}(b_{\mathcal{N}} + \theta\delta b_{\mathcal{N}}) - c_{\mathcal{B}}[\mathcal{B}^2]^{-1}b_{\mathcal{N}} \\ &= [1-\theta,\ 2]\begin{bmatrix} 1-\theta \\ 1 \end{bmatrix} - 3 \\ &= -2\theta + \theta^2. \end{aligned}$$

The interior point approach gave us the same result, but in a different manner.

From one of the main results of the basic compatibility theory [4], we have the following.

COROLLARY 4.3. *h is admissible if and only if $\Upsilon(h) > 0$.*

Proof. By definition, $\Upsilon(h) > 0 \rightarrow h \in \mathcal{H}$, and the converse follows from Theorem 4.2 and $\mathcal{H} = \{h : \rho^*(h) > 0\}$ [4]. □

This corollary says that the set of admissible directions equals the set of directions for which the optimal partition is invariant on the associated open line segment. We now consider another example [4] to help understand economic interpretations, introduce the *optimal partition transition graph*, and illustrate a form of activity analysis built on how the optimal partition changes rather than on how optimal bases change.

There are three fuels from which to generate electricity: coal, oil, and uranium. Define six activities, as follows:

PCL: purchase coal, GCL: generate electricity from coal,
POL: purchase oil, GOL: generate electricity from oil,
PUR: purchase uranium, GUR: generate electricity from uranium.

Figure 1 shows the linear program. The objective is to minimize cost, shown as the first row, while meeting the required electricity demand, shown as the last row. Rows BCL, BOL, and BUR balance the associated fuels: what is purchased must be at least as great as what is used for generation. Row LNU limits the generation from uranium: GUR $\leq$ 10.

Generation from uranium is the least costly (per unit of electricity generated, including the cost of uranium), so its level is as high as possible, limited to 10 units, which generates 4 units of electricity. The other 6 units are generated from oil, and none is generated from coal. Thus, the levels of PCL and GCL are zero in every optimal solution; however, PCL is in one optimal basis (compatible with increasing the right-hand side) and GCL is in another (compatible with decreasing the right-hand side).

Figure 2 shows the active submatrix, where the optimal partition has $\sigma(x) =$

$$\begin{array}{cc} & \begin{array}{cccc} \sigma(x): \text{POL} & \text{PUR} & \text{GOL} & \text{GUR} \end{array} \\ B = & \left[\begin{array}{cccc} & & & \\ 1 & & -1 & \\ & 1 & & -1 \\ & & & -1 \\ & & .3 & .4 \end{array}\right] \end{array} \begin{array}{c} \sigma(\pi) \\ \text{BCL} \\ \text{BOL} \\ \text{BUR} \\ \text{LNU} \\ \text{DEL} \end{array} \quad \begin{array}{cc} & \begin{array}{cc} \sigma(d): \text{PCL} & \text{GCL} \end{array} \\ N = & \left[\begin{array}{cc} 1 & -1 \\ & \\ & \\ & \\ & .33 \end{array}\right] \end{array}$$

FIG. 2. *Optimal partition for the electricity generation example.*

{POL, PUR, GOL, GUR} (activities to generate electricity from oil and uranium) and $\sigma(\pi)$ equal to all rows (B^* and N^* are null).

For $b_{\text{BCL}} > 0$, we increase the right-hand side of the coal balance row, which corresponds to a *stockpile requirement*. The theory of basic compatibility says that the coal purchase activity (PCL) needs to be in the basis to provide the appropriate response: buy coal. As $b_{\text{BCL}} < 0$, we are providing free coal, making the cost of electricity generation consist of only the operation and maintenance cost. This is \$.80 per unit of coal, which is \$2.42 per unit of electricity (\$.80 ÷ .33). Thus, the generation activity (GCL) needs to be in the basis to provide the appropriate response: displace oil-fired generation with coal-fired generation. The displacement continues until all oil-fired generation is displaced, which occurs at $b_{\text{BCL}} = -18.18$. A view of these is with the *basis transition graph*, shown in Figure 3, that is a part of the theory of basic compatibility, which we now extend.

Let $\delta b = -\mathrm{e}_1$ (i.e., decrease the right-hand side of row BCL). An interior point approach first considers the differential linear program:

$$\max\{-\pi_1 : \pi \in D^*(r)\} = -\min\{\pi_1 : \pi = (p, 15, 20, .4, 52), 16.36 \le p \le 18\} = -16.36.$$

This gives us the new optimal partition for $r - \theta \mathrm{e}_1$ with θ sufficiently small. (Our goal is to obtain the greatest value of θ, which defines $\Upsilon(-\mathrm{e}_1)$.)

The new optimal partition adds activity GCL to the set of active columns, so the following equations must hold as θ is increased:

$$Bx_B = \left[\begin{array}{ccccc} & & & & -1 \\ 1 & & -1 & & \\ & 1 & & -1 & \\ & & & -1 & \\ & & .3 & .4 & .33 \end{array}\right] \left[\begin{array}{c} x_{POL} \\ x_{PUR} \\ x_{GOL} \\ x_{GUR} \\ x_{GCL} \end{array}\right] = \left[\begin{array}{r} -\theta \\ 0 \\ 0 \\ -10 \\ 10 \end{array}\right].$$

This gives the following primal conditions that limit θ:

$$\begin{aligned} \Upsilon(-\mathrm{e}_1) = \max\{\, \theta : x \ge 0, & \\ -x_{GCL} &= -\theta \\ x_{POL} - x_{GOL} &= 0 \\ x_{PUR} - x_{GUR} &= 0 \\ -x_{GUR} &= -10 \\ .33\, x_{GCL} + .3\, x_{GOL} + .4\, x_{GUR} &= 10 \,\}. \end{aligned}$$

This reduces to $\Upsilon(-\mathrm{e}_1) = \max\{\theta : .33\theta \le 6\} = 18.18$. While this equals the range we obtained from the basis-driven approach, the reasoning is different. At $b_{BCL} = -18.18$, the optimal partition changes again to deactivate oil-fired generation—i.e., exclude activities GOL and POL from the set of active columns. (POL must remain

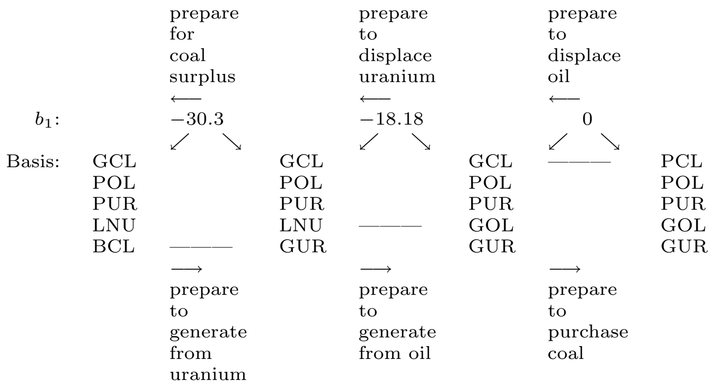


FIG. 3. *Basis transition graph.*

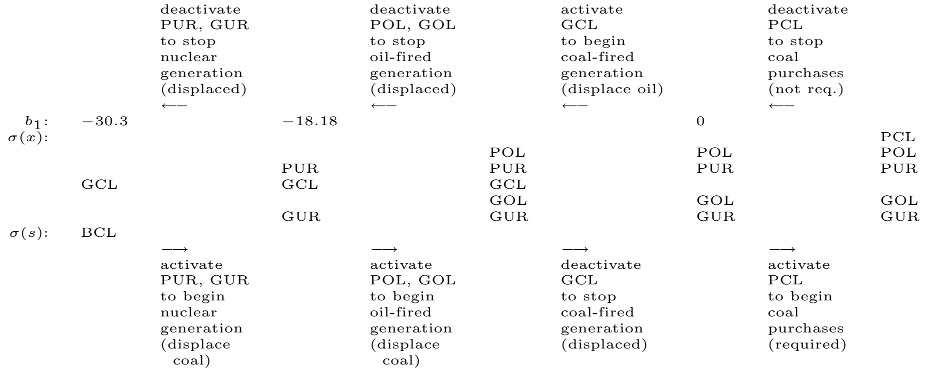

deactivate PUR, GUR to stop nuclear generation (displaced)
deactivate POL, GOL to stop oil-fired generation (displaced)
activate GCL to begin coal-fired generation (displace oil)
deactivate PCL to stop coal purchases (not req.)
b_1:
$\sigma(x)$:
$\sigma(s)$:
−30.3
GCL
BCL
−18.18
PUR GCL GUR
POL PUR GCL GOL GUR
0
POL PUR GOL GUR
PCL POL PUR GOL GUR
activate PUR, GUR to begin nuclear generation (displace coal)
activate POL, GOL to begin oil-fired generation (displace coal)
deactivate GCL to stop coal-fired generation (displaced)
activate PCL to begin coal purchases (required)

FIG. 4. *Optimal partition transition graph.*

basic in the theory of basic compatibility, even though its level is zero in every optimal solution, in order to have the correct price of oil, π_{BOL}.) Analogous to basic compatibility, the optimal partition changes due to an *event* that makes something change status: from inactive to active, or vice versa.

Whereas Figure 3 shows the basis transition graph that was introduced in [4] for varying the amount of coal, Figure 4 introduces a *partition transition graph.* Notice that in the basis transition graph, events occur at the threshold, choosing the event that is compatible with the particular variation (left or right transition). By contrast, in the optimal partition transition graph, events occur just on one side of each threshold. At $b_1 = 0$, it is *after* $\theta > 0$ that coal purchases begin (i.e., activity PCL is activated by entering $\sigma(x)$). Similarly, it is *after* $b_1 < 0$ that coal-fired generation begins (i.e., activity GCL is activated). As we continue to move to the left, the optimal partition remains invariant on the open interval, $(-18.18, 0)$. At the threshold, all of the oil is displaced by coal, so the optimal partition changes at $r - 18.18\mathrm{e}_1$. It is just *before* this change that the event occurs: deactivate POL and GOL. Then, the optimal partition is invariant on the half-open interval: $\theta \in [18.18, 30.3) \implies \sigma(x) = \{\text{PUR, GCL, GUR}\}$ for $r - \theta \mathrm{e}_1$.

This view of events that activities are *activated* or *deactivated* just before or after the threshold where the optimal partition changes complements the basic view that describes which basis is a compatible one in terms of events that *prepare* for the movement away from the threshold. Of course, phrases like "just before" and "just after" are not mathematical, but the idea is to gain insight from the solution, and this distinction in the two kinds of transition graphs does provide an added vantage point, based on the underlying events.

5. Summary. Here is a summary of the main points:

- The new optimal partition is obtained by solving two differential linear programs, one over the primal optimality region, the other over the dual. The new set of active columns equals that of the primal differential linear program; the new set of active rows equals that of the dual differential linear program.
- The interval for which the objective value has constant functional form, obtained from the range of the (possibly new) optimal partition, contains the interval obtained from the range of compatible bases. Further, this containment can be strict.
- The optimal partition transition graph, which shows threshold events when the optimal partition changes, provides another visualization of the underlying economics.

Acknowledgments. The author gratefully acknowledges helpful comments from Allen Holder, Kees Roos, and Tamás Terlaky. Also, this article benefited from the reviews of Karla Hoffman and an anonymous referee.

REFERENCES

[1] I. Adler and R. Monteiro, *A geometric view of parametric linear programming*, Algorithmica, 8 (1992), pp. 161–176.

[2] T. Gal, *Postoptimal Analyses, Parametric Programming, and Related Topics*, 2nd ed., Walter de Gruyter, Berlin, Germany, 1995.

[3] A. Goldman and A. Tucker, *Theory of linear programming*, in Linear Inequalities and Related Systems, H. Kuhn and A. Tucker, eds., Ann. of Math. Stud. 38, Princeton University Press, Princeton, NJ, 1956, pp. 53–97.

[4] H. Greenberg, *An analysis of degeneracy*, Naval Res. Logist., 33 (1986), pp. 635–655.

[5] H. Greenberg, *The use of the optimal partition in a linear programming solution for postoptimal analysis*, Oper. Res. Lett., 15 (1994).

[6] H. Greenberg, *Mathematical Programming Glossary*, http://www.cudenver.edu/~hgreenbe/glossary/glossary.html, 1996-9.

[7] H. Greenberg, *Myths and Counterexamples in Mathematical Programming: LP-*2, http://www.cudenver.edu/~hgreenbe/myths/myths.html, 1996-9.

[8] H. Greenberg, *Linear programming* 1: *Basic principles*, in Recent Advances in Sensitivity Analysis and Parametric Programming, T. Gal and H. Greenberg, eds., Kluwer Academic Publishers, Boston, MA, 1997.

[9] H. J. Greenberg, A. G. Holder, K. Roos, and T. Terlaky, *On the dimension of the set of Rim perturbations for optimal partition invariance*, SIAM J. Optim., 9 (1999), pp. 207–216.

[10] B. Jansen, C. Roos, and T. Terlaky, *An Interior Point Approach to Postoptimal and Parametric Analysis in Linear Programming*, Report 92-21, Faculty of Technical Mathematics and Informatics/Computer Science, Delft University of Technology, Delft, The Netherlands, 1992.

[11] H. Mills, *Marginal values of matrix games and linear programs*, in Linear Inequalities and Related Systems, H. Kuhn and A. Tucker, eds., Ann. of Math. Stud. 38, Princeton University Press, Princeton, NJ, 1956, pp. 183–193.

[12] R. D. C. Monteiro and S. Mehrotra, *A general parametric analysis approach and its implication to sensitivity analysis in interior point methods*, Math. Programming, 47 (1996), pp. 65–82.

[13] C. Roos, *Interior point approach to linear programming: Theory, algorithms & parametric analysis*, in Topics in Engineering Mathematics, A. van der Burgh and J. Simonis, eds., Kluwer Academic Publishers, Norwell, MA, 1992, pp. 181–216.

[14] C. Roos, T. Terlaky, and J.-P. Vial, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley, New York, 1996.

[15] A. Williams, *Marginal values in linear programming*, J. Soc. Indust. Appl. Math., 11 (1963), pp. 82–94.

[16] S. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, PA, 1997.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 443–461

SOLVING LARGE-SCALE SPARSE SEMIDEFINITE PROGRAMS FOR COMBINATORIAL OPTIMIZATION[*]

STEVEN J. BENSON[†], YINYU YE[‡], AND XIONG ZHANG[§]

Abstract. We present a dual-scaling interior-point algorithm and show how it exploits the structure and sparsity of some large-scale problems. We solve the positive semidefinite relaxation of combinatorial and quadratic optimization problems subject to boolean constraints. We report the first computational results of interior-point algorithms for approximating maximum cut semidefinite programs with dimension up to 3,000.

Key words. semidefinite programming, dual potential reduction algorithm, maximum cut problem

AMS subject classifications. 90C33, 90C27, 52A20

PII. S1052623497328008

1. Introduction.

Recently, there were several theoretical results on the effectiveness of approximating combinatorial and nonconvex quadratic optimization problems by using semidefinite programming (see, e.g., Goemans and Williamson [11], Nesterov [24], and Ye [33]). These results raise the hope that some hard optimization problems can be tackled by solving large-scale semidefinite relaxation programs. The positive semidefinite relaxation was considered early on by Lovász [18] and Shor [29], and the field received further contributions from many other researchers (e.g., see Lovász and Shrijver [19], Alizadeh [2], Sherali and Adams [28], and references therein).

The approximate solution to the quadratic optimization problem is obtained by solving a semidefinite relaxation, i.e., a semidefinite program (SDP) of the form

$$\text{(SDP)} \qquad \begin{array}{ll} \text{minimize} & C \bullet X \\ \text{subject to} & A_i \bullet X = b_i, \quad i = 1, \ldots, m, \\ & X \succeq 0. \end{array} \tag{1}$$

Here, the given matrices $C, A_i \in \mathcal{S}^n$, the set of n-dimensional symmetric matrices; vector $b \in \mathcal{R}^m$; and unknown $X \in \mathcal{S}^n$. Furthermore, the A_i's are linearly independent, meaning that $\sum_{i=1}^m y_i A_i = 0$ implies $y_1 = \cdots = y_m = 0$; $C \bullet X = \operatorname{tr} C^T X = \sum_{jk} C_{jk} X_{jk}$; and $X \succeq 0$ means that X is positive semidefinite.

In this paper, one additional assumption will be made: the constraint matrices have a rank-1 form, $A_i = a_i a_i^T$, $a_i \in \mathcal{R}^n$. This structure arises in many large-scale problems and results in considerable simplifications.

[*] Received by the editors September 26, 1997; accepted for publication (in revised form) September 18, 1998; published electronically February 10, 2000. This work is partially supported by NSF grants DMI-9522507 and DMS-9703490.

http://www.siam.org/journals/siopt/10-2/32800.html

[†] Applied Mathematics and Computational Sciences, The University of Iowa, Iowa City, IA 52242 (benson@mcs.anl.gov). This author is currently visiting the Computational Optimization Lab, Department of Management Sciences, The University of Iowa, Iowa City, IA 52242.

[‡] Department of Management Sciences, The University of Iowa, Iowa City, IA 52242 (yyye@dollar.biz.uiowa.edu).

[§] School of Mechanical Engineering, Huazhong University of Science and Technology, Wuhan, Hubei, China (xzhang@dollar.biz.uiowa.edu).

The dual of (SDP) can be written as

$$\text{(DSDP)} \qquad \begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & \sum_{i=1}^{m} y_i A_i + S = C, \quad S \succeq 0, \end{array} \tag{2}$$

where y_i, $i = 1, \ldots, m$, are scalar variables.

We have the following well-known duality theorem [25].

THEOREM 1 (strong duality). *Provided that* (SDP) *and* (DSDP) *are both feasible and there is a strictly interior point to either* (SDP) *or* (DSDP)*, there is no duality gap.*

Thus, if both (SDP) and (DSDP) are well behaved or a primal and dual optimal solution pair (X^*) and (y^*, S^*) exists, then $C \bullet X^* = b^T y^*$.

A well-behaved pair of semidefinite programs can be solved in polynomial time. There are actually several interior-point polynomial algorithms. One is the primal-scaling algorithm (Nesterov and Nemirovskii [25], Alizadeh [2], Vandenberghe and Boyd [31], and Ye [34]), which is the analogue of the primal path-following and potential reduction algorithm for linear programming. This algorithm uses only X to generate the iterate direction. In other words,

$$\begin{pmatrix} X^{k+1} \\ S^{k+1} \end{pmatrix} = F_p(X^k),$$

where F_p is the primal algorithm iterative mapping.

Another is the dual-scaling algorithm (Vandenberghe and Boyd [31], Anstreicher and Fampa [4], and Ye [34]), which is the analogue of the dual path-following and potential reduction algorithm for linear programming. The dual-scaling algorithm uses only S to generate the new iterate:

$$\begin{pmatrix} X^{k+1} \\ S^{k+1} \end{pmatrix} = F_d(S^k),$$

where F_d is the dual algorithm iterative mapping.

The third is the primal-dual scaling algorithm, which uses both X and S to generate the new iterate (see Todd [30] and references therein):

$$\begin{pmatrix} X^{k+1} \\ S^{k+1} \end{pmatrix} = F_{pd}(X^k, S^k),$$

where F_{pd} is the primal-dual algorithm iterative mapping.

All these algorithms generate primal and dual iterates simultaneously and possess $O(\sqrt{n} \ln(1/\epsilon))$ iteration complexity to yield the duality gap accuracy ϵ. Other scaling algorithms have been proposed in the past. For example, a semidefinite program equivalent to Dikin's affine-scaling algorithm could be very fast. However, this algorithm may not even converge. Muramatsu [22] and Muramatsu and Vanderbei [23] showed an example in which these affine scaling algorithms will not converge to an optimal answer.

There are also quite a few computational results and implementations of these interior algorithms; see Anstreicher and Fampa [4], Alizadeh, Haeberly, and Overton [3], Fujie and Kojima [9], Fujisawa, Kojima, and Nakata [8], Helmberg et al. [13], Karisch, Rendl, and Clausen [14], Vandenberghe and Boyd [31], Wolkowicz and Zhao [32], and Zhao et al. [35], [36]. To the best of our knowledge, the largest problem that could be solved was at $n = 900$ from their reports. (After the initial version of this paper was submitted, one more implementation came out: Fujisawa et al. [10] reported that they could solve a maximum cut semidefinite program with $n = 1,250$ using a powerful workstation.)

The practical winner in solving semidefinite programs was Helmberg and Rendl [12], an implementation of a non-interior-point algorithm called the bundle method. They reported the solutions of a set of dual semidefinite programs with n up to $3,000$. The bundle method enables them to take advantage of the sparsity structure of these problems. The (minor) weakness of their method is that the method does not simultaneously solve the primal problem, it cannot guarantee or verify the optimality accuracy at its termination, and it is not a polynomial time algorithm.

Therefore, an open question is how to exploit the sparsity structure by polynomial interior-point algorithms so that they can solve large-scale problems in practice. In this paper we try to answer this question. We show that many large-scale semidefinite programs that have arisen from combinatorial and quadratic optimization have features that make the dual-scaling interior-point algorithm the most suitable choice:

(1) The computational cost of each iteration in the dual algorithm is less than the cost of a primal-dual iteration. Although primal-dual algorithms may possess superlinear convergence, the approximation problems under consideration require less accuracy than some other applications. Therefore, the superlinear convergence exhibited by primal-dual algorithms may not be utilized in our applications. The dual-scaling algorithm has been shown to perform equally well when only a lower precision answer is required; see, e.g., Adler et al. [1] and Vandenberghe and Boyd [31].

(2) In most combinatorial applications we need only a lower bound for the optimal objective value of (SDP). Solving (DSDP) alone would be sufficient to provide such a lower bound. Thus, we may not need any X at all. Even if an optimal primal solution is necessary, our dual-scaling algorithm can generate an optimal X at the termination of the algorithm with little additional cost.

(3) For large-scale problems, S tends to be very sparse and structured since it is the linear combination of C and the A_i's. This sparsity allows considerable savings in both memory and computation time. The primal matrix, X, may be much less sparse and have a structure unknown beforehand. Consequently, primal and primal-dual algorithms may not fully exploit the sparseness and structure of the data.

These problems include the semidefinite relaxations of the graph-partition problem, the box-constrained quadratic optimization problem, the 0–1 integer set covering problem, and other problems. We will use the maximum cut problem to illustrate our points later, where we report our computational result, using a PC, on solving the maximum cut semidefinite relaxations of the Helmberg and Rendl set of graph problems for n up to $3,000$.

2. Dual-scaling algorithm.

The dual-scaling algorithm, which is a modification of the dual-scaling linear programming algorithm, reduces the Tanabe–Todd–Ye primal-dual potential function

$$\Psi(X, S) = \rho \ln(X \bullet S) - \ln \det X - \ln \det S.$$

The first term decreases the duality gap, while the second and third terms keep X and S in the interior of the positive semidefinite matrix cone. When $\rho > n$, the infimum of the potential function occurs at an optimal solution. Also note that, using the arithmetic-geometric mean inequality, we have (see, e.g., [34])

$$n \ln(X \bullet S) - \ln\det X - \ln\det S \geq n \ln n.$$

The algorithm, along with other semidefinite programming algorithms, is described in Ye [34], and we shall use notation defined there.

Let operator $\mathcal{A}(X) : \mathcal{S}^n \to \mathcal{R}^m$ be defined as

$$\mathcal{A}(X) = \begin{pmatrix} A_1 \bullet X \\ A_2 \bullet X \\ \vdots \\ A_m \bullet X \end{pmatrix}.$$

Since $\mathcal{A}(X)^T y = \sum_{i=1}^m y_i (A_i \bullet X) = (\sum_{i=1}^m y_i A_i) \bullet X$, the adjoint operator $\mathcal{A}^T : \mathcal{R}^m \to \mathcal{S}^n$ is

$$\mathcal{A}^T(y) = \sum_{i=1}^m y_i A_i.$$

Let $\bar{z} = C \bullet X$ for some feasible X and consider the dual potential function

$$\psi(y, \bar{z}) = \rho \ln\left(\bar{z} - b^T y\right) - \ln\det S.$$

Its gradient is

$$\nabla\psi(y, \bar{z}) = -\frac{\rho}{\bar{z} - b^T y} b + \mathcal{A}\left(S^{-1}\right). \tag{3}$$

To estimate the reduction in the potential function from a current iterate $(y^k, \bar{z}^k)$ to the next, we will use a lemma from linear programming that can be found in [34] and is essentially due to Karmarkar [15].

Lemma 1. *Let $X \in \mathcal{S}^n$ and $\|X - I\|_\infty < 1$. Then*

$$\ln\det(X) \geq \operatorname{tr}(X - I) - \frac{\|X - I\|}{2(1 - \|X - I\|_\infty)},$$

where I denotes the identity matrix, $\|\cdot\|$ denotes the Frobenius norm,

$$\|A\|_\infty = \max_{i=1,\ldots,n} \left\{|\lambda_i(A)|\right\} \leq \|A\|,$$

and $\lambda_i(A)$ is the ith eigenvalue of $A \in \mathcal{S}^n$.

Proof. We have $0 < \lambda_i := \lambda_i(X) < 2$ for all $i = 1, \ldots, n$, since $\|X - I\|_\infty < 1$. Moreover,

$$\begin{aligned} \ln\lambda_i &= \ln(1 + \lambda_i - 1) \\ &= (\lambda_i - 1) - \frac{(\lambda_i - 1)^2}{2} + \frac{(\lambda_i - 1)^3}{3} - \frac{(\lambda_i - 1)^4}{4} + \cdots \\ &\geq (\lambda_i - 1) - \frac{(\lambda_i - 1)^2}{2}(1 + |\lambda_i - 1| + |\lambda_i - 1|^2 + \cdots) \\ &= (\lambda_i - 1) - \frac{(\lambda_i - 1)^2}{2(1 - |\lambda_i - 1|)} \geq (\lambda_i - 1) - \frac{(\lambda_i - 1)^2}{2(1 - \|X - I\|_\infty)}. \end{aligned}$$

Summing the inequality over i, we have the desired result. □

For any given y and $S = C - \mathcal{A}^T(y)$ such that $S \succ 0$ and $\|(S^k)^{-.5}(\mathcal{A}^T(y - y^k))(S^k)^{-.5}\| < 1$, using the above lemma, the concavity of the first term in the potential function, and the fact that

$$\left(S^k\right)^{-.5} S \left(S^k\right)^{-.5} - I = \left(S^k\right)^{-.5}\left(S - S^k\right)\left(S^k\right)^{-.5} = -\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5},$$

we establish an overestimator for the potential reduction:

$$\begin{aligned}
&\psi\left(y, \bar{z}^k\right) - \psi\left(y^k, \bar{z}^k\right)\\
&\quad = \rho \ln\left(\bar{z}^k - b^T y\right) - \rho \ln\left(\bar{z}^k - b^T y^k\right) - \ln\det\left(\left(S^k\right)^{-.5} S \left(S^k\right)^{-.5}\right)\\
&\quad \le \rho \ln\left(\bar{z}^k - b^T y\right) - \rho \ln\left(\bar{z}^k - b^T y^k\right) - \operatorname{tr}\left(\left(S^k\right)^{-.5} S \left(S^k\right)^{-.5} - I\right)\\
&\qquad + \frac{\left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|}{2\left(1 - \left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|_\infty\right)}\\
&\quad = \rho \ln\left(\bar{z}^k - b^T y\right) - \rho \ln\left(\bar{z}^k - b^T y^k\right) + \mathcal{A}\left(\left(S^k\right)^{-1}\right)^T\left(y - y^k\right)\\
&\qquad + \frac{\left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|}{2\left(1 - \left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|_\infty\right)}\\
&\quad \le \nabla\psi\left(y^k, \bar{z}^k\right)^T\left(y - y^k\right) + \frac{\left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|}{2\left(1 - \left\|\left(S^k\right)^{-.5}\left(\mathcal{A}^T\left(y - y^k\right)\right)\left(S^k\right)^{-.5}\right\|_\infty\right)}.
\end{aligned} \tag{4}$$

Therefore, beginning with a strictly feasible dual point (y^k, S^k) and upper bound $\bar{z}^k$, each iteration solves the following problem:

$$\begin{array}{ll}
\text{minimize} & \nabla\psi^T\left(y^k, \bar{z}^k\right)\left(y - y^k\right)\\
\text{subject to} & \|(S^k)^{-.5}\left(\mathcal{A}^T(y - y^k)\right)(S^k)^{-.5}\| \le \alpha,
\end{array} \tag{5}$$

where α is a positive constant less than 1. For simplicity, in what follows we let

$$\Delta^k = \bar{z}^k - b^T y^k.$$

The first order Karusch–Kuhn–Tucker conditions state that the minimum point, y^{k+1}, of this convex problem satisfies

$$\begin{aligned}
&M^k\left(y^{k+1} - y^k\right) + \beta\nabla\psi\left(y^k, \bar{z}^k\right)\\
&\qquad = M^k\left(y^{k+1} - y^k\right) + \beta\Big(-\frac{\rho}{\bar{z}^k - b^T y^k} b + \mathcal{A}\left(\left(S^k\right)^{-1}\right)\Big) = 0
\end{aligned} \tag{6}$$

for a positive value of β, where

$$M^k = \begin{pmatrix}
A_1\left(S^k\right)^{-1} \bullet \left(S^k\right)^{-1} A_1 & \cdots & A_1\left(S^k\right)^{-1} \bullet \left(S^k\right)^{-1} A_m\\
\vdots & \ddots & \vdots\\
A_m\left(S^k\right)^{-1} \bullet \left(S^k\right)^{-1} A_1 & \cdots & A_m\left(S^k\right)^{-1} \bullet \left(S^k\right)^{-1} A_m
\end{pmatrix}$$

and

$$\mathcal{A}\big(\big(S^k\big)^{-1}\big) = \begin{pmatrix} A_1 \bullet \big(S^k\big)^{-1} \\ \vdots \\ A_m \bullet \big(S^k\big)^{-1} \end{pmatrix}.$$

The matrix M^k is a Gram matrix and is positive definite when $S^k \succ 0$ and the A_i's are linearly independent. In this paper, it will sometimes be referred to as M.

Using the ellipsoidal constraint, the minimal solution, y^{k+1}, of (5) is given by

$$y^{k+1} - y^k = \beta d\big(\bar{z}^k\big)_y, \tag{7}$$

where

$$\begin{aligned} d\big(\bar{z}^k\big)_y &= -\big(M^k\big)^{-1}\nabla\psi\big(y^k, \bar{z}^k\big), \\ \beta &= \frac{\alpha}{\sqrt{-\nabla\psi^T\big(y^k, \bar{z}^k\big)d\big(\bar{z}^k\big)_y}}. \end{aligned} \tag{8}$$

Unlike linear programming, positive semidefinite programming requires a significant amount of time to compute the system of equations that determines the step direction. For arbitrary symmetric matrices A_i, Monteiro and Zanjácomo [20] demonstrated an efficient implementation of several primal-dual step directions. The Alizadeh–Heaberly–Overton direction [3] can be computed in $5nm^3 + n^2m^2 + O(\max\{m, n\}^3)$ operations. The direction used in [13], [16], and [21] uses $2nm^3 + n^2m^2 + O(\max\{m, n\}^3)$ operations, and the direction used in [26] uses $nm^3 + n^2m^2/2 + O(\max\{m, n\}^3)$ operations. The complexity of computing the matrix is a full order of magnitude higher than any other step of the algorithm. Fujisawa, Kojima, and Nakata [8] explored another technique for computing primal-dual step directions that exploit the sparsity of the data matrices. However, it is our belief that only the dual-scaling algorithm can fully exploit the structure and sparsity of many problems, as explained below.

Generally, $M_{ij}^k = A_i(S^k)^{-1} \bullet (S^k)^{-1}A_j$. When $A_i = a_i a_i^T$, the Gram matrix can be rewritten in the form

$$M^k = \begin{pmatrix} \big(a_1^T\big(S^k\big)^{-1}a_1\big)^2 & \cdots & \big(a_1^T\big(S^k\big)^{-1}a_m\big)^2 \\ \vdots & \ddots & \vdots \\ \big(a_m^T\big(S^k\big)^{-1}a_1\big)^2 & \cdots & \big(a_m^T\big(S^k\big)^{-1}a_m\big)^2 \end{pmatrix} \quad \text{and} \tag{9}$$

$$\mathcal{A}\big(\big(S^k\big)^{-1}\big) = \begin{pmatrix} a_1^T\big(S^k\big)^{-1}a_1 \\ \vdots \\ a_m^T\big(S^k\big)^{-1}a_m \end{pmatrix}.$$

This matrix can be computed very quickly without computing, or saving, $(S^k)^{-1}$. Instead, S^k can be factored, and then we can use the following algorithm.

ALGORITHM M. To compute M^k and $\mathcal{A}((S^k)^{-1})$, factor $S^k = L^k(L^k)^T$ and do the following:

```
For $i = 1 : m$;
    Solve $L^k w_i = a_i$;
    $\mathcal{A}((S^k)^{-1})_i = w_i^T w_i$  and  $M_{ii}^k = (\mathcal{A}((S^k)^{-1})_i)^2$;
    For $j = 1 : i-1$;  $M_{ij}^k = (w_i^T w_j)^2$;  end;
end.
```

Solving each of the m systems of equations uses $n^2 + O(n)$ floating point operations. Since there are $m(m+1)/2$ vector multiplications, Algorithm M uses $nm^2 + n^2 m + O(nm)$ operations after factoring S^k. Note that these operations can be significantly reduced if S^k is structured and sparse. In applications like the maximum cut problem, discussed in section 3, the matrix S^k is indeed very sparse, whereas its inverse is usually dense, so working with S^k is faster than working with its inverse. Using matrices of the form $A_i = a_i a_i^T$ also reduces the complexity of primal-dual algorithms by a factor of n, but even the quickest direction to compute takes about twice as long as our dual-scaling direction. Furthermore, they all need to handle dense X.

Algorithm M needs to store all vectors $w_1, \ldots, w_m$, and they are generally dense. To save storage and exploit the sparsity of $a_i, \ldots, a_m$, an alternative algorithm is as follows.

ALGORITHM M′. To compute M^k and $\mathcal{A}((S^k)^{-1})$, factor $S^k = L^k(L^k)^T$ and do the following:

```
For $i = 1 : m$;
    Solve $S^k w_i = a_i$;
    $\mathcal{A}((S^k)^{-1})_i = w_i^T a_i$  and  $M_{ii}^k = (\mathcal{A}((S^k)^{-1})_i)^2$;
    For $j = i+1 : m$;  $M_{ij}^k = (w_i^T a_j)^2$;  end;
end.
```

Algorithm M′ does not need to store w_j but uses one more back-solve for w_i.

To find a feasible primal point X, we solve the least squares problem

$$\begin{array}{ll} \text{minimize} & \left\| \left(S^k\right)^{.5} X \left(S^k\right)^{.5} - \frac{\Delta^k}{\rho} I \right\| \\ \text{subject to} & \mathcal{A}(X) = b. \end{array} \tag{10}$$

This problem looks for a matrix $X(\bar{z}^k)$ near the central path. Larger values of ρ generally give a lower objective value but provide a solution matrix that is not positive definite more frequently. The answer to (10) is a by-product of computing (8), given explicitly by

$$X\left(\bar{z}^k\right) = \frac{\Delta^k}{\rho} \left(S^k\right)^{-1} \left(\mathcal{A}^T\left(d\left(\bar{z}^k\right)_y\right) + S^k\right) \left(S^k\right)^{-1}. \tag{11}$$

Creating the primal matrix may be costly. However, the evaluation of the primal objective value $C \bullet X(\bar{z}^k)$ requires drastically less work:

$$\begin{aligned} C \bullet X\left(\bar{z}^k\right) &= b^T y^k + X\left(\bar{z}^k\right) \bullet S^k \\ &= b^T y^k + \operatorname{tr}\left(\frac{\Delta^k}{\rho} \left(S^k\right)^{-1} \left(\mathcal{A}^T\left(d\left(\bar{z}^k\right)_y\right) + S^k\right) \left(S^k\right)^{-1} S^k\right) \\ &= b^T y^k + \frac{\Delta^k}{\rho} \operatorname{tr}\left(\left(S^k\right)^{-1} \mathcal{A}^T\left(d\left(\bar{z}^k\right)_y\right) + I\right) \\ &= b^T y^k + \frac{\Delta^k}{\rho} \left(d\left(\bar{z}^k\right)_y^T \mathcal{A}\left(\left(S^k\right)^{-1}\right) + n\right). \end{aligned}$$

Since the vectors $\mathcal{A}((S^k)^{-1})$ and $d(\bar{z}^k)_y$ were previously found in calculating the dual step direction, the cost of computing a primal objective value is the cost of a vector dot product! The matrix $X(\bar{z}^k)$ never gets computed during the iterative process, saving time and memory. On the other hand, primal-dual methods require far more resources to compute the primal variables X.

Defining

$$P(\bar{z}^k) = \frac{\rho}{\Delta^k}\left(S^k\right)^{.5} X\left(\bar{z}^k\right)\left(S^k\right)^{.5} - I, \tag{12}$$

we have the following lemma.

LEMMA 2. *Let* $\mu^k = \frac{\Delta^k}{n} = \frac{\bar{z}^k - b^T y^k}{n}$, $\mu = \frac{X(\bar{z}^k)\bullet S^k}{n} = \frac{C\bullet X(\bar{z}^k)-b^T y^k}{n}$, $\rho \geq n + \sqrt{n}$, *and* $\alpha < 1$. *If*

$$\|P(\bar{z}^k)\| < \min\left(\alpha\sqrt{\frac{n}{n+\alpha^2}}, 1-\alpha\right), \tag{13}$$

then the following three inequalities hold:

(1) $X(\bar{z}^k) \succ 0$;
(2) $\|(S^k)^{.5}X(\bar{z}^k)(S^k)^{.5} - \mu I\| \leq \alpha\mu$;
(3) $\mu \leq (1 - .5\alpha/\sqrt{n})\mu^k$.

Proof. The proofs are by contradiction. If the first inequality is false, then $(S^k)^{.5}X(\bar{z}^k)(S^k)^{.5}$ has at least one nonpositive eigenvalue, which by (12) implies that $\|P(\bar{z}^k)\| \geq 1$.

If the second does not hold, then

$$\begin{aligned}
\left\|P(\bar{z}^k)\right\|^2 &= \left\|\frac{\rho}{n\mu^k}\left(S^k\right)^{.5} X\left(\bar{z}^k\right)\left(S^k\right)^{.5} - I\right\|^2 \\
&= \left\|\frac{\rho}{n\mu^k}\left(S^k\right)^{.5} X\left(\bar{z}^k\right)\left(S^k\right)^{.5} - \frac{\rho\mu}{n\mu^k}I + \frac{\rho\mu}{n\mu^k}I - I\right\|^2 \\
&= \left\|\frac{\rho}{n\mu^k}\left(S^k\right)^{.5} X\left(\bar{z}^k\right)\left(S^k\right)^{.5} - \frac{\rho\mu}{n\mu^k}I\right\|^2 + \left\|\frac{\rho\mu}{n\mu^k}I - I\right\|^2 \\
&> \left(\frac{\rho\mu}{n\mu^k}\right)^2\alpha^2 + \left(\frac{\rho\mu}{n\mu^k} - 1\right)^2 n \\
&\geq \alpha^2\left(\frac{n}{n+\alpha^2}\right),
\end{aligned}$$

where the last inequality is true because the quadratic term has a minimum at $\frac{\rho\mu}{n\mu^k} = \frac{n}{n+\alpha^2}$.

If the third inequality does not hold, then

$$\frac{\rho\mu}{n\mu^k} > \left(1 + \frac{1}{\sqrt{n}}\right)\left(1 - \frac{.5\alpha}{\sqrt{n}}\right) \geq 1,$$

which leads to

$$\begin{aligned} \|P(\bar{z}^k)\|^2 &\ge \left(\frac{\rho\mu}{n\mu^k}-1\right)^2 n \\ &\ge \left(\left(1+\frac{1}{\sqrt{n}}\right)\left(1-\frac{\alpha}{2\sqrt{n}}\right)-1\right)^2 n \\ &= \left(1-\frac{\alpha}{2}-\frac{\alpha}{2\sqrt{n}}\right)^2 \\ &\ge (1-\alpha)^2. \qquad \square \end{aligned}$$

Focusing on the expression $P(\bar{z}^k)$, it can be rewritten as

$$\begin{aligned} P(\bar{z}^k) &= \frac{\rho}{\Delta^k}\left(S^k\right)^{.5}\left(\frac{\Delta^k}{\rho}\left(S^k\right)^{-1}\left(\mathcal{A}^T\left(d(\bar{z}^k)_y\right)+S^k\right)\left(S^k\right)^{-1}\right)\left(S^k\right)^{.5}-I \\ &= \left(S^k\right)^{-.5}\mathcal{A}^T\left(d(\bar{z}^k)_y\right)\left(S^k\right)^{-.5} \\ &= \left(S^k\right)^{-.5}\mathcal{A}^T\left(\frac{y^{k+1}-y^k}{\beta}\right)\left(S^k\right)^{-.5}, \end{aligned}$$

which by (7) and (8) makes

$$\nabla\psi^T\left(y^k,\bar{z}^k\right)d\left(\bar{z}^k\right)_y = -\left\|P\left(\bar{z}^k\right)\right\|^2 \tag{14}$$

and

$$\nabla\psi^T\left(y^k,\bar{z}^k\right)\left(y^{k+1}-y^k\right) = -\alpha\left\|P\left(\bar{z}^k\right)\right\|. \tag{15}$$

Updating the dual variables according to

$$y^{k+1} = y^k + \beta d(\bar{z})_y = y^k + \frac{\alpha}{\|P(\bar{z}^{k+1})\|}d(\bar{z})_y \quad \text{and} \quad S^{k+1} = C - \mathcal{A}^T\left(y^{k+1}\right) \tag{16}$$

ensures the positive definiteness of S^{k+1} when $\alpha < 1$, which implies that they are feasible. Using (15) and (4), the reduction in the potential function satisfies the inequality

$$\psi\left(y^{k+1},\bar{z}^k\right) - \psi\left(y^k,\bar{z}^k\right) \le -\alpha\left\|P\left(\bar{z}^k\right)\right\| + \frac{\alpha^2}{2(1-\alpha)}. \tag{17}$$

The theoretical algorithm can be stated as follows.

Dual algorithm. Given an upper bound $\bar{z}^0$ and a dual point (y^0, S^0) such that $S^0 = C - \mathcal{A}^T y^0 \succ 0$, set $k = 0$, $\rho > n + \sqrt{n}$, $\alpha \in (0,1)$, and do the following:

```
while $\bar{z}^k - b^T y^k \ge \epsilon$ do
begin
  1. Compute $\mathcal{A}((S^k)^{-1})$ and the Gram matrix $M^k$ (9) using Algorithm M or M$'$.
  2. Solve (8) for the dual step direction $d(\bar{z}^k)_y$.
  3. Calculate $\|P(\bar{z}^k)\|$ using (14).
  4. If (13) is true, then $X^{k+1} = X(\bar{z}^k)$, $\bar{z}^{k+1} = C \bullet X^{k+1}$, and $(y^{k+1}, S^{k+1}) = (y^k, S^k)$;
     else $y^{k+1} = y^k + \frac{\alpha}{\|P(\bar{z}^k)\|}d(\bar{z}^{k+1})_y$, $S^{k+1} = C - \mathcal{A}^T(y^{k+1})$, $X^{k+1} = X^k$, and $\bar{z}^{k+1} = \bar{z}^k$.
     endif
```

```
    5. $k := k + 1$.
end
```

We can derive the following potential reduction theorem based on Lemma 2.

THEOREM 2.

$$\Psi\big(X^{k+1}, S^{k+1}\big) \leq \Psi\big(X^k, S^k\big) - \delta,$$

where $\delta > 1/50$ for a suitable α.

Proof.

$$\begin{aligned}\Psi\big(X^{k+1},&S^{k+1}\big) - \Psi\big(X^k, S^k\big)\\ &= \big(\Psi\big(X^{k+1}, S^{k+1}\big) - \Psi\big(X^{k+1}, S^k\big)\big) + \big(\Psi(X^{k+1}, S^k) - \Psi(X^k, S^k)\big).\end{aligned}$$

In each iteration, one of the differences is zero. If $\|P(\bar{z}^k)\|$ does not satisfy (13), the dual variables get updated and (17) shows sufficient improvement in the potential function when $\alpha = 0.4$.

On the other hand, if the primal matrix gets updated, then using Lemma 1 and the first two parts of Lemma 2,

$$\begin{aligned} n\ln\big(X^{k+1} \bullet S^k\big) &- \ln\det\big(X^{k+1}\big) - \ln\det\big(S^k\big)\\ &= n\ln\big(X^{k+1} \bullet S^k\big) - \ln\det\big(X^{k+1}S^k\big)\\ &= n\ln\big(X^{k+1} \bullet S^k/\mu\big) - \ln\det\big(X^{k+1}S^k/\mu\big)\\ &= n\ln n - \ln\det\big((S^k)^{.5}X^{k+1}(S^k)^{.5}/\mu\big)\\ &\leq n\ln n + \frac{\|(S^k)^{.5}X^{k+1}(S^k)^{.5}/\mu - I\|}{2\left(1 - \|(S^k)^{.5}X^{k+1}(S^k)^{.5}/\mu - I\|_\infty\right)}\\ &\leq n\ln n + \frac{\alpha^2}{2(1-\alpha)}\\ &\leq n\ln\big(X^k \bullet S^k\big) - \ln\det\big(X^k\big) - \ln\det\big(S^k\big) + \frac{\alpha^2}{2(1-\alpha)}.\end{aligned}$$

Additionally, by the third part of Lemma 2,

$$\sqrt{n}\left(\ln\big(X^{k+1} \bullet S^k\big) - \ln\big(X^k \bullet S^k\big)\right) = \sqrt{n}\ln\frac{\mu}{\mu^k} \leq -\frac{\alpha}{2}.$$

Adding the two inequalities gives

$$\Psi\big(X^{k+1}, S^k\big) \leq \Psi\big(X^k, S^k\big) - \frac{\alpha}{2} + \frac{\alpha^2}{2(1-\alpha)}.$$

By choosing $\alpha = 0.4$ again, we have the desired result. □

This theorem leads to the following corollary.

COROLLARY 1. *Let $\rho \geq n + \sqrt{n}$ and $\Psi(X^0, S^0) \leq (\rho - n)\ln(X^0 \bullet S^0)$. Then, the algorithm terminates in at most $O((\rho - n)\ln(X^0 \bullet S^0/\epsilon))$ iterations.*

Proof. In $O((\rho - n)\ln(X^0 \bullet S^0/\epsilon))$ iterations,

$$\Psi\big(X^k, S^k\big) \leq (\rho - n)\ln(\epsilon).$$

Also,

$$(\rho - n)\ln\big(C \bullet X^k - b^Ty^k\big) = (\rho - n)\ln\big(X^k \bullet S^k\big) \leq \Psi\big(X^k, S^k\big) - n\ln n \leq \Psi\big(X^k, S^k\big).$$

Combining the two inequalities,

$$C \bullet X^k - b^T y^k = X^k \bullet S^k < \epsilon. \qquad \square$$

Again, from (11) we see that the algorithm can generate an X^k as a by-product. However, it is not needed in generating the iterate direction, and it is only explicitly used for proving convergence and complexity.

The computation cost in each iteration of the algorithm can be summarized as follows. First, updating S or $S+\mathcal{A}^T(d(\bar{z}^k))$ uses matrix additions and mn^2 operations, and factoring it uses $O(n^3)$ operations. Second, creating the Gram matrix uses $nm^2 + 2n^2m + O(nm)$ operations, and factoring and solving the system of equations uses $O(m^3)$ operations. Finally, dot products for $\bar{z}^{k+1}$ and $\|P(\bar{z}^k)\|$ and the calculation of y^{k+1} use only $O(m)$ operations. These give the total $O(m^3+nm^2+n^2m+n^3)$ floating point operations. Note that the procedure uses only the Cholesky factorization.

In contrast, each iteration of primal-dual methods requires several additional computations. First, the various Schur complement matrices used to compute the step directions cost significantly more to compute than the matrices used in this dual-scaling algorithm. Second, primal-dual algorithms must compute a primal step direction. This step direction involves the product of three matrices, which can be very costly. Third, the primal-dual algorithms do use line searches in both the primal and dual problems. Such a search requires additional dense matrix factorizations.

3. Maximum cut problem. The maximum cut problem asks to partition the vertices of a graph into two sets that maximize the sum of the weighted edges connecting vertices in one set with vertices in the other. The positive semidefinite relaxation of the maximum cut problem can be expressed as (see, e.g., [11], [27])

$$\text{(MAX-CUT)} \qquad \begin{array}{ll} \text{minimize} & C \bullet X \\ \text{subject to} & \operatorname{diag}(X) = e, \\ & \quad X \succeq 0. \end{array} \tag{18}$$

The operator diag$(\cdot)$ takes the diagonal of a matrix and makes it a vector. In other words, $A_i = e_i e_i^T$, $i = 1, \ldots, n$, where e_i is the vector with 1 for the ith component and 0 for all others.

The dual program can be expressed as

$$\text{(DMAX)} \qquad \begin{array}{ll} \text{maximize} & e^T y \\ \text{subject to} & \operatorname{Diag}(y) + S = C, \quad S \succeq 0, \end{array} \tag{19}$$

The operator Diag$(\cdot)$ forms a diagonal matrix from a vector.

Many examples of the maximum cut problem have a very sparse matrix C. Since S is a linear combination of C and a diagonal matrix, it possesses the same sparse structure of C that remains constant for all iterations. This sparsity can be exploited by reordering S to reduce fill-in during the Cholesky factorization. A good reordering will drastically speed up the factorization and the many forward and back substitutions required to compute the Gram matrices.

Applying the dual-scaling algorithm to this relaxation,

$$\nabla \psi(y^k, \bar{z}^k) = -\frac{\rho}{\Delta^k} e + \operatorname{diag}\left(\left(S^k\right)^{-1}\right)$$

and

$$M^k = \left(S^k\right)^{-1} \circ \left(S^k\right)^{-1}, \tag{20}$$

where $\circ$ represents the Hadamard product and $\Delta^k = \bar{z}^k - e^T y^k$. That is, $M^k_{ij} = ((S^k)^{-1}_{ij})^2$. When the graph represented by C is connected, M^k is generally dense—even when C is sparse.

The direction $d(\bar{z}^k)_y$ of (8) comprises two parts,

$$dy_1 = \left(M^k\right)^{-1} e, \tag{21}$$

$$dy_2 = \left(M^k\right)^{-1} \operatorname{diag}\left(\left(S^k\right)^{-1}\right), \tag{22}$$

so that

$$d\left(\bar{z}^k\right)_y = \frac{\rho}{\Delta^k} dy_1 - dy_2. \tag{23}$$

Since the dual direction depends upon the upper bound $\bar{z}^k$, splitting the direction into these two parts allows the algorithm to take advantage of a possibly improved upper bound.

To determine the stepsize and measure the improvement in the potential function, we again compute

$$\left\|P\left(\bar{z}^k\right)\right\| = \sqrt{-\nabla\psi^T\left(y^k, \bar{z}^k\right)^T d\left(\bar{z}^k\right)_y}. \tag{24}$$

If $\|P(\bar{z}^k)\|$ is sufficiently small, Lemma 2 guarantees an improved primal solution, $X(\bar{z}^k)$ with $C \bullet X(\bar{z}^k) < \bar{z}^k$, where from (11),

$$X\left(\bar{z}^k\right) = \frac{\Delta^k}{\rho}\left(S^k\right)^{-1}\left(\operatorname{Diag}\left(d\left(\bar{z}^k\right)_y\right) + S^k\right)\left(S^k\right)^{-1}.$$

Frequently, an improved primal objective value $\bar{z}$ can be found for even larger values of $\|P(\bar{z}^k)\|$. We may first compute

$$\bar{z} := C \bullet X\left(\bar{z}^k\right) = e^T y^k + \frac{\Delta^k}{\rho}\left(\operatorname{diag}\left(\left(S^k\right)^{-1}\right)^T d\left(\bar{z}^k\right)_y + n\right). \tag{25}$$

If $\bar{z} < \bar{z}^k$, then we go on to check if $X(\bar{z}^k) \succ 0$. But from the above expression, $X(\bar{z}^k) \succ 0$ if and only if

$$\left(\operatorname{Diag}\left(d\left(\bar{z}^k\right)_y\right) + S^k\right) \succ 0. \tag{26}$$

To check if $\operatorname{Diag}(d(\bar{z}^k)_y) + S^k \succ 0$, we use the Cholesky factorization and simply check if its pivots are all positive. We stop the factorization process as soon as we encounter a negative or zero pivot and conclude that the matrix is not positive definite. Note that $\operatorname{Diag}(d(\bar{z}^k)_y) + S^k$ has the same sparse structure as S^k or C, allowing it to be stored in the same data structure. If $\operatorname{Diag}(d(\bar{z}^k)_y) + S^k \succ 0$, we set $\bar{z}^{k+1} = \bar{z} < \bar{z}^k$. Otherwise, $\bar{z}^{k+1} = \bar{z}^k$.

An improved upper bound $\bar{z}^{k+1}$ results in a smaller $\Delta^k := \bar{z}^{k+1} - e^T y^k$ and will modify the dual step direction calculated in (23), which is why the step direction was divided into two parts. Finally, the dual variables will be updated by

$$y^{k+1} = y^k + \frac{\alpha}{\|P(\bar{z}^{k+1})\|} d\big(\bar{z}^{k+1}\big)_y \quad \text{and} \quad S^{k+1} = C - \operatorname{Diag}\big(y^{k+1}\big).$$

If $\alpha < 1$, $S(\alpha) = C - \operatorname{Diag}(y^k + \frac{\alpha}{\|P(\bar{z}^{k+1})\|} d(\bar{z}^{k+1})_y) \succ 0$. Larger values of α increase the stepsize, which can speed up the convergence of the algorithm. Larger stepsizes, however, can also step outside the cone of positive semidefinite matrices. If a larger step is used, a Cholesky factorization can check the positive definiteness of $S(\alpha)$. Note that this factorization is needed in the next iteration anyway. Since the matrix $S(\alpha)$ is sparse and well ordered, an unsuccessful attempt to increase the stepsize costs very little. In general, these factorizations cost far less than a factorization of the dense M^k, but allow large stepsizes to significantly reduce the number of iterations required to achieve the desired accuracy.

We now state the specialized dual-scaling algorithm for solving the maximum cut semidefinite program.

DUAL ALGORITHM. Reorder C to reduce fill-in during Cholesky factorization. Set $\bar{z}^0 = C \bullet I$ and choose y^0 such that $S^0 = C - \operatorname{Diag}(y^0) \succ 0$. Set $k = 0$, $\alpha = .99$, and do the following:

```
while $\frac{\bar{z}^k - e^T y^k}{|e^T y^k| + 1} \geq \epsilon$ do
begin
  1. Compute diag($(S^k)^{-1}$) and the matrix $M^k$ (20) using Algorithm M or M′.
  2. Solve (21), (22), and (23) for the dual step direction.
  3. Use (25) to compute a new upper bound $\bar{z}$.
  4. If $\bar{z} < \bar{z}^k$ and $(\operatorname{Diag}(d(\bar{z}^k)_y) + S^k) \succ 0$,
     then let $\bar{z}^{k+1} = \bar{z}$ and recompute $d(\bar{z}^{k+1})_y$ using (23);
     else let $\bar{z}^{k+1} = \bar{z}^k$. endif
  5. Compute $\|P(\bar{z}^{k+1})\|$ using (24).
  6. Select $\beta \geq \alpha / \|P(\bar{z}^{k+1})\|$, so that $y^{k+1} = y^k + \beta d(\bar{z}^{k+1})_y$ and $S^{k+1} = C -$
     $\operatorname{Diag}(y^{k+1}) \succ 0$.
  7. $k := k + 1$.
end
```

4. Computational results. We implemented the dual-scaling algorithm in ANSI C and ran the program on a PC with 233 MHz, 64 MB RAM, and 256 K cache memory. (The code and its user guide are available for public download at http://dollar.biz.uiowa.edu/col/.)

To accelerate convergence of the algorithm, the implementation used more aggressive stepsizes. It used values of α equal to $.99, 1.5, 3$, and 6. Initially, we set $\alpha = 3$. When the value of α was successful for three consecutive iterates, we tried the next larger value. If we stepped out of the feasible region, we tried the next smaller value of α. We found that larger stepsizes were frequently used, and this strategy yields a significant improvement in the total number of iterations.

In addition, we initialized the value of ρ to be $5n$. Larger values of ρ more aggressively seek the optimal solution, but are also more likely to yield infeasible points. After a couple of iterates, ρ was dynamically selected using the following criteria:

$$\rho = 1.6n * \sqrt{(rgap^{k-1} \,/\, rgap^k)},$$

where $rgap^{k-1}$ and $rgap^k$ are the relative duality gaps at the previous and current iteration

$$rgap^k := \frac{\bar{z}^k - b^T y^k}{1 + |b^T y^k|}.$$

We let the initial point

$$X^0 = I \quad \text{and} \quad y_i^0 = C_{ii} - \sum_{j \neq i} |C_{ij}| - 1, \quad i = 1, \ldots, n,$$

which by Gerschgorin's theorem guarantees $S^0 \succ 0$ (see Atkinson [5]). This value generally provides a reasonable starting point. We have used the minimum degree ordering algorithm to reorder C.

We have stopped the iteration process when the relative duality gap

$$rgap^k = \frac{\bar{z}^k - b^T y^k}{1 + |b^T y^k|} \leq 10^{-6}.$$

Most combinatorial applications ask for a reasonable bound to be found very quickly. Therefore, the precision required in the semidefinite program is far less than that required by other applications. In addition, the original maximum cut problem has only simple, binary variables. For these problems, we believe that a precision of 10^{-4} is sufficient, so we have recorded the number of iterations and seconds needed to compute that level of precision.

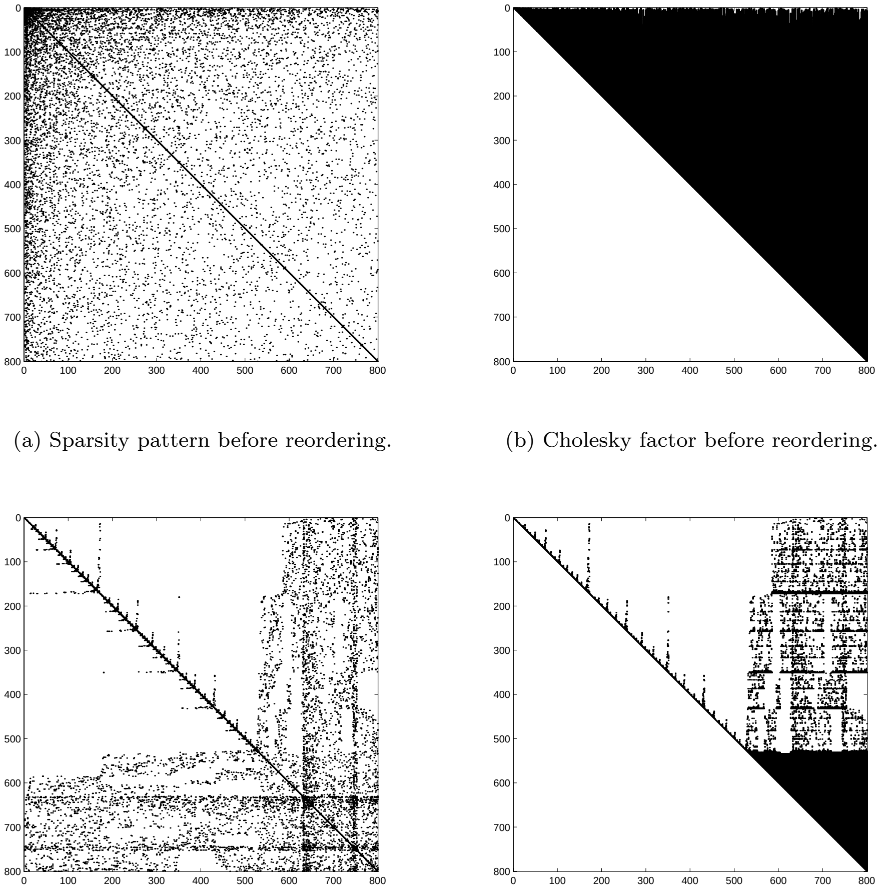
Our experiments used a machine-independent graph generator, called rudy, created by G. Rinaldi. We tested the maximum cut semidefinite program on the G set of graphs used by Helmberg and Rendl [12]. This set of problems becomes a standard test set for graph optimization. These maximum cut problems range in size from $n = 800$ to $n = 3,000$. Many of these problems, like G1–G10, G22–G31, and G43–G47 have a randomly created structure. Problems G11–G13, G32–G34, and G48–G50 come from a two-dimensional toroidal grid, while the others represent planar type graphs. (Helmberg and Rendl [12] actually solved G53–G54 semidefinite programs for another graph problem, the ϑ-function [18], instead of the maximum cut problem.)

Table 1
Seconds used for the three most expensive computations.

Name	Sparsity of S_{chol}	Factor S (sec.)	Compute M (sec.)	Factor M (sec.)
G1	73.6%	1.412	12.856	1.983
G11	4.2%	0.010	1.272	1.863
G14	14.3%	0.140	3.105	1.863
G22	47.8%	11.076	129.917	32.046
G32	1.6%	0.030	9.864	30.744
G35	11.8%	1.352	41.113	30.764

Table 1 shows the cost of key steps of the algorithm for six different problems. It shows the seconds required to factor S, create M, and factor M. The sparsity statistic in the second column gives the percentage of nonzero entries in the factor after reordering.

This table shows that when S is sparse, the factorization of M dominates the computation time. Since M is generally dense, regardless of the sparsity of S, its



(a) Sparsity pattern before reordering.

(b) Cholesky factor before reordering.

(c) Sparsity pattern after reordering.

(d) Cholesky factor after reordering.

FIG. 1. *Sparsity patterns and Cholesky factors of* $G14$.

computation time is constant for problems of equal size. For more dense problems, the creation of M dominates the computation time. This is not surprising since it uses $3n^3$ floating point operations, while the Cholesky factorization uses a sixth of that amount. Most large-scale applications, however, will contain a certain sparse structure, and the table shows how this dual-scaling algorithm exploits that structure to save computation time.

The table also emphasizes the importance of a good ordering of the matrix in the beginning of the algorithm. The reordering of matrices has been studied for years [6], but to illustrate its importance, we include a few figures. Figure 1 shows the structure of S and its Cholesky factor in the 800×800 example G14.

The objective matrix G14 has about 1.58% nonzero entries. Figure 1(a) shows the sparsity structure of the matrix before the minimum degree ordering and Figure 1(c) presents the structure after the minimum degree ordering. The dual solution S has the same sparse structure. Figures 1(b) and 1(d) show the sparsity patterns of the

Table 2
Performance on solving the G-set semidefinite programs.

						$rgap = 10^{-4}$		$rgap = 10^{-6}$	
Name	Dim	Spars	*Pobj*	*Dobj*	*Rgap*	Iter	Time	Iter	Time
G1	800	6.12%	−4.833276e+04	−4.833279e+04	5.452359e−07	20	616.08	24	741.15
G2	800	6.12%	−4.835767e+04	−4.835772e+04	9.502824e−07	19	592.69	23	719.25
G3	800	6.12%	−4.833732e+04	−4.833733e+04	3.401351e−07	19	589.56	24	746.54
G4	800	6.12%	−4.844576e+04	−4.844581e+04	9.824817e−07	19	595.43	23	723.28
G5	800	6.12%	−4.839953e+04	−4.839955e+04	3.552550e−07	19	594.71	24	752.24
G6	800	6.12%	−1.062463e+04	−1.062464e+04	8.541375e−07	21	646.12	25	769.41
G7	800	6.12%	−9.957042e+03	−9.957051e+03	8.420776e−07	21	654.40	25	779.28
G8	800	6.12%	−1.002773e+04	−1.002773e+04	9.678249e−07	21	655.63	24	780.46
G9	800	6.12%	−1.011492e+04	−1.011493e+04	7.775437e−07	21	654.09	25	778.95
G10	800	6.12%	−9.940246e+03	−9.940253e+03	7.465415e−07	20	620.63	24	742.82
G11	800	0.63%	−2.516658e+03	−2.516659e+03	3.719002e−07	18	64.40	23	82.05
G12	800	0.63%	−2.495497e+03	−2.495498e+03	5.198500e−07	19	71.03	24	89.50
G13	800	0.63%	−2.588544e+03	−2.588546e+03	9.570427e−07	20	76.70	24	91.88
G14	800	1.59%	−1.276625e+04	−1.276627e+04	9.986387e−07	29	166.43	33	189.11
G15	800	1.58%	−1.268623e+04	−1.268623e+04	3.450954e−07	33	188.68	39	222.87
G16	800	1.59%	−1.270007e+04	−1.270007e+04	4.674935e−07	27	154.31	31	177.01
G17	800	1.58%	−1.268530e+04	−1.268531e+04	4.548353e−07	26	149.60	30	172.44
G18	800	1.59%	−4.664038e+03	−4.664040e+03	4.708559e−07	47	276.82	51	299.92
G19	800	1.58%	−4.328040e+03	−4.328042e+03	3.345754e−07	33	193.66	38	221.98
G20	800	1.59%	−4.445567e+03	−4.445570e+03	7.261350e−07	33	193.80	37	216.54
G21	800	1.58%	−4.417133e+03	−4.417136e+03	8.094406e−07	38	223.30	42	246.10
G22	2000	1.05%	−5.654376e+04	−5.654378e+04	3.957778e−07	23	8215.71	28	9997.62
G23	2000	1.05%	−5.658202e+04	−5.658204e+04	3.910436e−07	23	8146.30	28	9920.25
G24	2000	1.05%	−5.656340e+04	−5.656342e+04	4.981024e−07	23	8323.22	27	9759.06
G25	2000	1.05%	−5.657696e+04	−5.657698e+04	3.876964e−07	23	8306.01	28	10106.27
G26	2000	1.05%	−5.653145e+04	−5.653148e+04	5.027876e−07	23	8323.72	27	9756.52
G27	2000	1.05%	−1.656663e+04	−1.656664e+04	6.729729e−07	25	8851.74	29	10268.68
G28	2000	1.05%	−1.640315e+04	−1.640316e+04	7.335225e−07	25	8862.04	29	10278.55
G29	2000	1.05%	−1.683555e+04	−1.683556e+04	6.531697e−07	25	9034.53	29	10483.11
G30	2000	1.05%	−1.686152e+04	−1.686153e+04	6.413532e−07	26	9386.62	30	10843.05
G31	2000	1.05%	−1.646672e+04	−1.646673e+04	6.898466e−07	25	9047.75	29	10493.99
G32	2000	0.25%	−6.270553e+03	−6.270559e+03	9.633737e−07	23	1070.39	27	1255.70
G33	2000	0.25%	−6.177246e+03	−6.177250e+03	6.333926e−07	25	1175.24	29	1362.61
G34	2000	0.25%	−6.186747e+03	−6.186750e+03	4.510949e−07	24	1182.73	28	1381.23
G35	2000	0.64%	−3.205895e+04	−3.205896e+04	3.752328e−07	46	5167.17	51	5716.93
G36	2000	0.64%	−3.202383e+04	−3.202386e+04	9.011525e−07	38	4381.39	42	4841.52
G37	2000	0.64%	−3.207448e+04	−3.207449e+04	3.820329e−07	42	4836.22	47	5400.13
G38	2000	0.64%	−3.205987e+04	−3.205990e+04	8.761401e−07	47	5392.57	52	5952.48
G39	2000	0.64%	−1.151058e+04	−1.151059e+04	7.376637e−07	59	6615.01	63	7056.50
G40	2000	0.64%	−1.145915e+04	−1.145916e+04	3.728182e−07	52	6123.85	57	6703.57
G41	2000	0.64%	−1.146087e+04	−1.146087e+04	4.158517e−07	58	6629.09	63	7194.15
G42	2000	0.64%	−1.178500e+04	−1.178501e+04	7.308258e−07	45	5049.62	49	5495.55
G43	1000	2.10%	−2.812887e+04	−2.812889e+04	7.416858e−07	18	767.17	22	939.50
G44	1000	2.10%	−2.811152e+04	−2.811154e+04	7.511560e−07	18	770.22	22	939.07
G45	1000	2.10%	−2.809911e+04	−2.809913e+04	4.530243e−07	21	900.51	25	1075.49
G46	1000	2.10%	−2.811972e+04	−2.811973e+04	3.978937e−07	18	782.46	23	999.66
G47	1000	2.10%	−2.814662e+04	−2.814664e+04	7.609519e−07	18	751.57	22	920.39
G48	3000	0.17%	−2.399999e+04	−2.400000e+04	3.699426e−07	14	2878.85	19	3861.30
G49	3000	0.17%	−2.399999e+04	−2.400000e+04	3.718535e−07	14	2873.69	19	3826.62
G50	3000	0.17%	−2.395268e+04	−2.395269e+04	3.543263e−07	14	2389.73	19	3192.86
G51	1000	1.28%	−1.602501e+04	−1.602502e+04	7.759416e−07	29	341.45	33	388.06
G52	1000	1.28%	−1.603854e+04	−1.603856e+04	7.427919e−07	36	441.94	41	489.28
G53	1000	1.28%	−1.603886e+04	−1.603887e+04	8.122974e−07	26	306.85	30	353.86
G54	1000	1.28%	−1.602476e+04	−1.602478e+04	7.421491e−07	29	340.43	33	387.51

Cholesky factors before and after the minimum degree ordering. The Cholesky factor of the unordered matrix is very dense. Comparing the running times shows that the reordering matrix would reduce the running time approximately by a factor of 4.

Table 2 shows the performance of the code on solving the G-set maximum cut semidefinite programs for stopping tolerances of $rgap \leq 10^{-4}$ and $rgap \leq 10^{-6}$. *PriObj*, *DualObj*, and *rgap* are the primal and dual objective values and the relative duality gap at termination. Also shown is the dimension and the percentage of nonzero entries in the objective matrix, as well as the time (in seconds) and number of iterations required by the program.

Most of the previous numerical tests [7], [17], [32], [35], [36] were conducted on smaller problem data sets where the dimension n was only a few hundred or less, so that no available computation result could be compared to ours. After our results were reported, a study of using a primal-dual algorithm for solving relative larger problems, including the maximum cut problem, was conducted by Fujisawa et al. [10]. They tested solving sparse maximum cut semidefinite programs with dimension up to 1,250. On a sparse problem with dimension of 1,000, they required 63,130 seconds; on a problem of dimension 1,250, they used 111,615 seconds. Their computations were performed on a DEC AlphaServer 8,400 with a processing speed of 437 MHz and 8 GB memory, which is far superior to the PC machine used in our test.

As we mentioned, Helmberg and Rendl [12] used a spectral bundle method to solve the same set of G1–G42 maximum cut problems. Their computations were performed on an UltraSPARC station with 64 MB memory. One advantage of the spectral bundle method is that it uses considerably less memory since it does not create or store a matrix as large as M. On problems with a randomly created structure, the bundle method appears slightly faster than ours. In these problems, the Cholesky factor of the slack matrix is relatively dense, despite the sparsity of the objective matrix. For the toroidal and planar graphs (such as G14), the dual matrix has a much better structure. In these problems, a minimum degree ordering kept the Cholesky factor sparse and the back and forward substitutions quick. In problems with a more structured objective matrix, the dual semidefinite programming algorithm outperformed the bundle method.

Finally, our implementation of the dual-scaling algorithm appears to be the first algorithm to converge to an optimal point in polynomial time, to use the characteristics inherent in many large-scale problems to its advantage, and to verify the optimality by solving both the primal and dual problems simultaneously. Its success with even relatively dense examples shows that the algorithm is generally efficient, while the improved performance on more sparse examples shows how it exploits the structure of most large-scale problems.

Acknowledgments. The authors would like to thank the two referees for their valuable comments and suggestions on improving the presentation of the paper.

REFERENCES

[1] I. Adler, N. K. Karmarkar, M. G. C. Resende, and G. Veiga, *An implementation of Karmarkar's algorithm for linear programming*, Math. Programming, 44 (1989), pp. 297–335; *errata*, Math. Programming, 50 (1991), p. 415.

[2] F. Alizadeh, *Combinatorial Optimization with Interior Point Methods and Semi-Definite Matrices*, Ph.D. thesis, University of Minnesota, Minneapolis, 1991.

[3] F. Alizadeh, J. P. A. Haeberly, and M. L. Overton, *Primal-Dual Interior Point Methods*

for Semidefinite Programming, Technical Report 659, Computer Science Dept., Courant Institute of Mathematical Sciences, New York University, 1994.

[4] K. M. Anstreicher and M. Fampa, *A Long-Step Path Following Algorithm for Semidefinite Programming Problems*, Working Paper, Dept. of Management Science, The University of Iowa, Iowa City, 1996.

[5] K. E. Atkinson, *An Introduction to Numerical Analysis*, 2nd ed., John Wiley, New York, 1989.

[6] I. S. Duff, *Sparse numerical linear algebra: Direct methods and preconditioning*, in The State of the Art in Numerical Analysis, Clarendon Press, Oxford, UK, 1997, pp. 27–62.

[7] J. Falkner, F. Rendl, and H. Wolkowicz, *A computational study of graph partitioning*, Math. Programming, 66 (1994), pp. 211–240.

[8] K. Fujisawa, M. Kojima, and K. Nakata, *Exploiting Sparsity in Primal-Dual Interior Point Methods for Semidefinite Programming*, Research Report B-324, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, 1997.

[9] T. Fujie and M. Kojima, *Semidefinite programming relaxation for nonconvex quadratic programs*, J. Global Optim., 10 (1997), pp. 367–380.

[10] K. Fujisawa, M. Fukuda, M. Kojima, and K. Nakata, *Numerical Evaluation of SDPA (SemiDefinite Programming Algorithm)*, Research Report B-330, Dept. of Mathematical and Computing Sciences, Tokyo Institute of Technology, 1997.

[11] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for Maximum Cut and Satisfiability problems using semidefinite programming*, J. Assoc. Comput. Mach., 42 (1995), pp. 1115–1145.

[12] C. Helmberg and F. Rendl, *A Spectral Bundle Method for Semidefinite Programming*, ZIB Preprint SC 97-37, Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustrasse 7, D-14195 Berlin, Germany, 1997.

[13] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, *An interior point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[14] S. E. Karisch, F. Rendl, and J. Clausen, *Solving Graph Bisection Problems with Semidefinite Programming*, Technical Report DIKU-TR-97/9, Dept. of Computer Science, University of Copenhagen, 1997.

[15] N. Karmarkar, *A new polynomial-time algorithm for linear programming*, Combinatorica 4 (1984), pp. 373–395.

[16] M. Kojima, S. Shindoh, and S. Hara, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.

[17] C. Lin and R. Saigal, *On Solving Large Scale Semidefinite Programming Problems—A Case Study of Quadratic Assignment Problem*, Technical Report, Dept. of Industrial and Operations Engineering, University of Michigan, Ann Arbor, 1997.

[18] L. Lovász, *On the Shannon capacity of a graph*, IEEE Trans. Inform. Theory, IT-25 (1979), pp. 1–7.

[19] L. Lovász and A. Shrijver, *Cones of matrices and set-functions and* 0–1 *optimization*, SIAM J. Optim., 1 (1991), pp. 166–190.

[20] R. D. C. Monteiro and P. R. Zanjácomo, *Implementation of Primal-Dual Methods for Semidefinite Programming Based on Monteiro and Tsuchiya Directions and Their Variants*, Technical Report, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, 1997.

[21] R. D. C. Monteiro and Y. Zhang, *A Unified Analysis for a Class of Path-Following Primal-Dual Interior-Point Algorithms for Semidefinite Programming*, Math. Programming, 81 (1998), pp. 281–299.

[22] M. Muramatsu, *Affine Scaling Algorithm Fails for Semidefinite Programming*, Technical Report, Department of Mechanical Engineering, Sophia University, Tokyo, 1996.

[23] M. Muramatsu and R. Vanderbei, *Primal-Dual Affine Scaling Algorithms Fail for Semidefinite Programming*, Technical Report, SOR, Princeton University, Princeton, NJ, 1997.

[24] Yu. E. Nesterov, *Quality of Semidefinite Relaxation for Nonconvex Quadratic Optimization*, CORE Discussion Paper 9719, Louvain, Belgium, 1997.

[25] Yu. E. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Methods in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, PA, 1994.

[26] Y. E. Nesterov and M. Todd, *Primal-Dual Interior Point Methods for Self-Scaled Cones*, SIAM J. Optim., 8 (1998), p. 324–364.

[27] S. Polijak, F. Rendl, and H. Wolkowicz, *A recipe for semidefinite relaxation for* 0–1 *quadratic programming*, J. Global Optim., 7 (1995), pp. 51–73.

[28] H. D. Sherali and W. P. Adams, *Computational advances using the reformulation-linearization technique (rlt) to solve discrete and continuous nonconvex problems*, Optima,

49 (1996), pp. 1–6.

[29] N. Z. Shor, *Quadratic optimization problems*, Soviet J. Comput. Systems Sci., 25 (1987), pp. 1–11; originally published in Tekhnicheskaya Kibernetika, 1 (1987), pp. 128–139.

[30] M. J. Todd, *On Search Directions in Interior-Point Methods for Semidefinite Programming*, Technical Report 1205, School of Operations Research and Industrial Engineering, Cornell University, Ithaca, NY, 1997.

[31] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[32] H. Wolkowicz and Q. Zhao, *Semidefinite programming for the graph partitioning problem*, Discrete Applied Math., to appear.

[33] Y. Ye, *Approximating Quadratic Programming with Bound and Quadratic Constraints*, Math. Programming, 84 (1999), pp. 219–226.

[34] Y. Ye, *Interior Point Algorithms: Theory and Analysis*, Wiley-Intersci. Ser. Discrete Math. Optim., John Wiley, New York, 1997.

[35] Q. Zhao, *Semidefinite Programming for Assignment and Partitioning Problems*, Ph.D. thesis, University of Waterloo, ON, Canada, 1996.

[36] Q. Zhao, S. E. Karisch, F. Rendl, and H. Wolkowicz, *Semidefinite programming relaxations for the quadratic assignment problems*, J. Combin. Optim., 2 (1998), pp. 71–109.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 462–502

PRIMAL-DUAL INTERIOR-POINT METHODS FOR SEMIDEFINITE PROGRAMMING IN FINITE PRECISION[*]

MING GU[†]

Abstract. Recently, a number of primal-dual interior-point methods for semidefinite programming have been developed. To reduce the number of floating point operations, each iteration of these methods typically performs block Gaussian elimination with block pivots that are close to singular near the optimal solution. As a result, these methods often exhibit complex numerical properties in practice.

We consider numerical issues related to some of these methods. Our error analysis indicates that these methods could be numerically stable if certain coefficient matrices associated with the iterations are well-conditioned, but are unstable otherwise. With this result, we explain why one particular method, the one introduced by Alizadeh, Haeberly, and Overton is in general more stable than others. We also explain why the so-called least squares variation, introduced for some of these methods, does not yield more numerical accuracy in general. Finally, we present results from our numerical experiments to support our analysis.

Key words. semidefinite programming, primal-dual, interior-point method, error analysis

AMS subject classifications. 49M15, 65K05, 90C3, 15A06, 65F05, 65G05

PII. S105262349731950X

1. Introduction.

The semidefinite programming problem (SDP) is the following convex optimization problem:

$$\text{(1.1a)} \qquad \begin{array}{ll} \min_{X\in\mathbf{S}^n} & C\bullet X \\ \text{subject to} & A_k\bullet X = b_k, \quad k=1,\dots,m, \\ & X\succeq 0, \end{array}$$

where $\mathbf{S}^n$ is the vector space of real symmetric n by n matrices; $A\bullet B$ is an inner product satisfying

$$A\bullet B \stackrel{\text{def}}{=} \mathbf{tr}(A^TB) = \sum_{i,j=1}^{n} A_{ij}B_{ij} \quad \text{for} \quad A,B\in\mathbf{R}^{n\times n}\ ;$$

$C\in\mathbf{S}^n$; and $A_k\in\mathbf{S}^n$ for $k=1,\dots,m$. By $X\succeq 0$ we mean that X is positive semidefinite. The *dual* problem to (1.1a) is of the form

$$\text{(1.1b)} \qquad \begin{array}{ll} \max_{y\in\mathbf{R}^m, Z\in\mathbf{S}^n} & b^Ty \\ \text{subject to} & \sum_{k=1}^m y_kA_k + Z = C, \\ & Z\succeq 0, \end{array}$$

where $b=(b_1,\dots,b_m)^T\in\mathbf{R}^m$. The dual problem is itself an SDP.

The SDP arises in many areas of science and engineering and includes the linear programming problem (LP) as a special case (see Vandenberghe and Boyd [30]). The

[*] Received by the editors April 7, 1997; accepted for publication (in revised form) January 22, 1999; published electronically February 18, 2000. This work was supported in part by NSF Career Award CCR-9702866 and by an Alfred Sloan research fellowship.

http://www.siam.org/journals/siopt/10-2/31950.html

[†] Department of Mathematics, University of California, Los Angeles, CA 90095-1555 (mgu@math.ucla.edu).

recent book by Boyd et al. [7] and survey articles by Alizadeh [2], Lewis and Overton [18], and Vandenberghe and Boyd [30] contain many applications to system and control theory, combinatorial optimization, and eigenvalue optimization.

Let **svec** be an isometry identifying $\mathbf{S}^n$ with $\mathbf{R}^{n(n+1)/2}$, so that $K \bullet L = (\mathbf{svec}(K))^T \cdot \mathbf{svec}(L)$ for all K, $L \in \mathbf{S}^n$; and let **smat** be the inverse of **svec**. The optimality conditions for problem (1.1) are

$$\mathcal{A}\,\mathbf{svec}(X) = b, \tag{1.2a}$$

$$\mathbf{smat}\left(\mathcal{A}^T\, y\right) + Z = C, \tag{1.2b}$$

$$X\, Z = 0, \tag{1.2c}$$

where X and $Z \succeq 0$; $\mathcal{A} = (\mathbf{svec}(A_1), \ldots, \mathbf{svec}(A_m))^T$; and $y = (y_1, \ldots, y_m)^T$. Throughout this paper, we assume that rank$\,(\mathcal{A}) = m$ and that equations (1.2) have a *unique* solution (X^*, Z^*, y^*) such that X^* and $Z^* \succeq 0$. Hence, (X^*, Z^*, y^*) is a feasible solution to (1.1) that further satisfies the complementarity condition (1.2c).

1.1. Interior-point methods for the SDP. Interior-point methods for the SDP were originally proposed by Alizadeh [1] and Nesterov and Nemirovskii [25]. Most of the interior-point methods for SDP are path-following methods, meaning that they generate a sequence of iterates approximating the so-called *central path* and converging to the primal and dual solutions. For SDP, the points on the central path satisfy (1.2a) and (1.2b) and the following condition relaxed from (1.2c):

$$XZ - \mu I = 0. \tag{1.3}$$

It is well known that under certain conditions the solution to (1.2a), (1.2b), and (1.3) is unique and converges to the optimal solution of (1.1) as μ goes to 0 (see Nesterov and Todd [26]). However, directly applying Newton's method to (1.2a), (1.2b), and (1.3) usually results in nonsymmetric search directions (see Helmberg et al. [15], and Kojima, Shindoh, and Hara [17]). Several methods have been introduced in the literature to ensure a symmetric search direction. For example, Zhang [34] defines a symmetrization operator

$$\mathbf{H}_P(M) \overset{\text{def}}{=} \frac{1}{2}\left(P\ M\ P^{-1} + \left(P\ M\ P^{-1}\right)^T\right)$$

for any given nonsingular matrix P, and shows that (1.3) is equivalent to

$$\mathbf{H}_P(X\ Z) - \mu\, I = 0 \tag{1.4}$$

for symmetric X and Z. Applying Newton's method to (1.2a), (1.2b), and (1.4) results in a family of symmetric search directions parameterized by P, usually referred to as the Monteiro–Zhang (MZ) family.

The MZ family includes a number of important symmetric search directions that were introduced earlier. The AHO method introduced by Alizadeh, Haeberly, and Overton [5] is based on a direction that corresponds to $P = I$. Taking $P^T P = X^{-1}$ and $P^T P = Z$ results in the two directions suggested by Monteiro [20]. These directions are also equivalent to two special directions of the family of directions introduced by Kojima, Shindoh, and Hara [17]; and the direction corresponding to $P^T P = Z$ was also suggested by Helmberg et al. [15]. We refer to methods based on these two directions as the HKM methods to reflect the history of their discovery.[1]

[1] They are called the H.K.M. directions in [29].

The NT method, suggested by Nesterov and Todd [26, 27], corresponds to a search direction defined by any P that satisfies

$$P^T\,P = R^{-1}\,\left(R\,Z\,R^T\right)^{\frac{1}{2}}\,R^{-T}\,, \tag{1.5}$$

where $R \in \mathbf{R}^{n\times n}$ is any matrix such that $R^T\,R = X$. Another family of symmetric search directions has been introduced recently by Monteiro and Tsuchiya [22].

Polynomial complexity has been established for primal-dual path-following algorithms based on any direction in these three families. See Kojima, Shindoh, and Hara [17], Monteiro [20, 21], Monteiro and Tsuchiya [22, 23], Monteiro and Zhang [24], and Zhang [34].

1.2. Computational issues. Since we are primarily concerned with computational issues in this paper, from now on we will not make clear distinctions between interior-point methods and their search directions.

The reason interior-point methods attract so much attention is because they have remarkable computational promise. Alizadeh, Haeberly, and Overton [5] and Todd, Toh, and Tütüncü [29] implemented and compared the AHO method, the NT method, and the HKM method corresponding to $P^T\,P = Z$. A number of SDP solvers are now available in the public domain (see Alizadeh et al. [3], Borchers [6], and Fujisawa, Kojima, and Nakata [12]).

These implementations reveal a number of computational issues for SDP that are surprisingly complex. It is observed that for these interior-point methods, some implementations were capable of yielding solutions in relatively good agreement with the true optimal solution, whereas others, being slightly different but mathematically equivalent in exact arithmetic, yielded very limited accuracy in the computed solution and sometimes even failed to converge. It is also observed that the AHO method of [5] appeared to be the most accurate among the methods tested [5, 29].

The search directions of these methods are usually solved via a Schur complement equation obtained from block Gaussian elimination (see section 2.1). The Schur complement is nonsymmetric for the AHO method. Todd, Toh, and Tütüncü [29] showed that for a subfamily of search directions in the MZ family, the Schur complement is symmetric positive definite, and the Schur complement equation can be expressed as the normal equation of a linear least squares (LS) problem and thus can be solved instead as an LS problem. Monteiro and Zhang [24] gave a parameterization of this subfamily. Throughout this paper we refer to this subfamily as the TTT family. It includes the two HKM directions and the NT direction. Zhang [34] also discussed the LS approach for some members of the TTT family. Although their numerical results indicated that the two approaches seemed to be comparable in terms of accuracy, Todd, Toh, and Tütüncü [29] argued that the LS approach could perform much better than the Schur complement approach in certain cases since the condition number of the coefficient matrix involved in the LS problem is the square root of that of the Schur complement.

Computational issues have been discussed earlier for other interior-point methods in optimization. For example, Ponceleón [28] analyzed linear systems arising from barrier methods for quadratic programming. Forsgren, Gill, and Shinnerl [11] analyzed linear systems arising from interior methods for constrained optimization. S. Wright [33, 32] analyzed interior-point methods for LP and linear complementarity problems. M. Wright [31] analyzed ill-conditioning and computational error in interior-point methods for nonlinear programming [31].

1.3. Main results. We analyze the accuracy of the AHO method and methods based on directions in the TTT family in finite precision arithmetic. We explain why some implementations of these methods are more accurate than others and why the LS approach in general does not perform better than the Schur complement approach. Most importantly, we show that, with the Schur complement approach, methods based on the AHO direction and the TTT family of directions can be numerically stable if certain coefficient matrices associated with the search direction are well-conditioned but are unstable otherwise. We present results from our numerical experiments that support this analysis.

Our error analysis is on the accuracy in the computed search direction for *one* step of the interior-point methods at a point (X, Z, y) that is "close" to the optimal solution of (1.1). We do not discuss the iteration complexity of these methods in finite precision. S. Wright [33, 32] took a somewhat similar approach in his finite precision analysis of interior-point methods for LP and linear complementarity problems.

This paper is organized as follows. In section 2 we discuss the Schur complement equation and the parameterization of the TTT family. In section 3 we discuss the AHO method and analyze it in finite precision. In section 4 we discuss methods of the TTT family, relate them to the NT and HKM methods, and analyze them in finite precision. In section 5 we present results from our numerical experiments that support our analysis. Finally, in section 6 we discuss some extensions and future work.

1.4. Notation and conventions. Throughout this paper, the symmetrized Kronecker product of G and K is a square matrix of order $n(n+1)/2$; its action on $\mathbf{svec}(H)$, where $H \in \mathbf{S}^n$, is given by

$$(G \otimes_s K)\ \mathbf{svec}(H) \overset{\text{def}}{=} \frac{1}{2}\mathbf{svec}\left(K\ H\ G^T + G\ H\ K^T\right). \tag{1.6}$$

The appendices in [5] and [29] contain some frequently used properties of the symmetrized Kronecker products in the context of SDP. They include

$$G \otimes_s K = K \otimes_s G\ , \quad (G \otimes_s K)^T = G^T \otimes_s K^T, \quad \text{and} \quad (G \otimes_s G)^{-1} = G^{-1} \otimes_s G^{-1},$$
$$(G \otimes_s K)\,(H \otimes_s H) = (G\,H) \otimes_s (K\,H)\,, \ \text{ and } \ (H \otimes_s H)\ (G \otimes_s K) = (H\,G) \otimes_s (H\,K)\,.$$

We will need the following vector from time to time:

$$\mathbf{e} \overset{\text{def}}{=} \mathbf{svec}\begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{pmatrix} \in \mathbf{R}^{n(n+1)/2}\ .$$

A *flop* is a floating point operation $\alpha \circ \beta$, where α and β are floating point numbers and $\circ$ is one of $+, -, \times$, and $\div$. In our error analysis, we take the usual model of arithmetic:

$$\mathbf{fl}(\alpha \circ \beta) = (\alpha \circ \beta)\ (1 + \xi)\ , \tag{1.7}$$

where $\mathbf{fl}(\alpha \circ \beta)$ is the floating point result of the operation $\circ$ and $|\xi| \le \epsilon$, with ϵ being the machine precision. For simplicity, we ignore the possibility of overflow and underflow.

The norm used is the 2-norm. Let α and β be numbers. We write $\alpha = O(\beta)$ if $|\alpha| \le c\ |\beta|$ for some positive *constant* c that is "moderate" and independent of β. We say that a matrix or a vector is $O(\alpha)$ if its norm is $O(\alpha)$. In such cases, the constant hidden in $O(\alpha)$ usually is a moderate multiple (such as 10 or 100) of a low-degree polynomial in the matrix dimensions. We write $\alpha = \Omega(\beta)$ if $\alpha = O(\beta)$ and $\beta = O(\alpha)$.

For any matrix X, $|X|$ is the matrix with entries $(|X|)_{ij} = |X_{ij}|$, and $|X| \le |Y|$ means that $|X_{ij}| \le |Y_{ij}|$ holds for all i and j. $\sigma_{\max}(X)$ and $\sigma_{\min}(X)$ are the largest and smallest singular values of X, respectively, and $\kappa(X) = \sigma_{\max}(X)/\sigma_{\min}(X) \ge 1$ is its condition number. For any *symmetric* matrix X, $\lambda_{\max}(X)$ and $\lambda_{\min}(X)$ are its largest and smallest eigenvalues, respectively. When we say a matrix is positive definite, we implicitly assume that it is symmetric.

2. The Schur complement and the TTT family.

Primal-dual methods are Newton-like methods applied to optimality equations (1.2). At a given point (X, Z, y), where X and Z are positive definite, the search direction (dX, dZ, dy) satisfies

$$\mathcal{A}\,\mathbf{svec}(dX) = r_p, \tag{2.1a}$$

$$\mathcal{A}^T\,dy + \mathbf{svec}(dZ) = r_d\,, \tag{2.1b}$$

where $r_p = b - \mathcal{A}\mathbf{svec}(X)$ and $r_d = \mathbf{svec}\left(C - Z - \mathbf{smat}\left(\mathcal{A}^T \cdot y\right)\right)$.

For the MZ family, equation (1.4) is linearized to give

$$\mathbf{H}_P\left(dX\;Z + X\;dZ\right) = \mu I - \mathbf{H}_P\left(X\;Z\right). \tag{2.1c}$$

2.1. The Schur complement.

To solve for the search direction (dX, dZ, dy), we write equations (1.1) in a single 3×3 block equation (cf. Todd, Toh, and Tütüncü [29] and Zhang [34]):

$$\mathcal{J}\,d\mathcal{X} = \mathcal{R}\,,\; \mathcal{J} = \begin{pmatrix} \mathcal{E} & \mathcal{F} & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ \mathcal{A} & 0 & 0 \end{pmatrix},\; d\mathcal{X} = \begin{pmatrix} \mathbf{svec}(dX) \\ \mathbf{svec}(dZ) \\ dy \end{pmatrix},\; \mathcal{R} = \begin{pmatrix} r_c \\ r_d \\ r_p \end{pmatrix}, \tag{2.2}$$

where $\mathcal{I}$ is the identity matrix of appropriate dimension and

$$\mathcal{E} = \left(P^{-T}\;Z\right) \otimes_s P\,, \quad \mathcal{F} = (P\;X) \otimes_s P^{-T}, \quad \text{and} \quad r_c = \mathbf{svec}\left(\mu I - \mathbf{H}_P\left(X\;Z\right)\right).$$

A straightforward way to compute the search direction $d\mathcal{X}$ is to solve (2.2) as a dense linear system of equations. However, this approach is too expensive for large SDPs. To compute $d\mathcal{X}$ more efficiently by taking advantage of the block structure in (2.2), we perform a block LU factorization on (2.2) to get

$$\begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{A}\,\mathcal{E}^{-1} & -\mathcal{A}\,\mathcal{E}^{-1}\,\mathcal{F} & \mathcal{I} \end{pmatrix} \begin{pmatrix} \mathcal{E} & \mathcal{F} & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ 0 & 0 & \mathcal{M} \end{pmatrix} d\mathcal{X} = \mathcal{R}\,, \tag{2.3}$$

where

$$\mathcal{M} = \mathcal{A}\,\mathcal{E}^{-1}\,\mathcal{F}\,\mathcal{A}^T$$

is the Schur complement. Todd, Toh, and Tütüncü [29] showed that $\mathcal{E}$ is nonsingular under the assumption that both X and Z are positive definite. Applying forward block substitution to (2.3) gives

$$\begin{pmatrix} \mathcal{E} & \mathcal{F} & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ 0 & 0 & \mathcal{M} \end{pmatrix} \begin{pmatrix} \mathbf{svec}(dX) \\ \mathbf{svec}(dZ) \\ dy \end{pmatrix} = \begin{pmatrix} r_c \\ r_d \\ r_p + \mathcal{A}\,\mathcal{E}^{-1}\,\left(\mathcal{F}\,r_d - r_c\right) \end{pmatrix}, \tag{2.4}$$

and applying block backward substitution to (2.4) gives (cf. Zhang [34])

$$\mathcal{M}\; dy = r_p + \mathcal{A}\,\mathcal{E}^{-1}\;(\mathcal{F}\, r_d - r_c)\,, \tag{2.5a}$$

$$dZ = \mathbf{smat}\left(r_d - \mathcal{A}^T\; dy\right), \tag{2.5b}$$

$$dX = \mathbf{smat}\left(\mathcal{E}^{-1}\;(r_c - \mathcal{F}\,\mathbf{svec}(dZ))\right)\;. \tag{2.5c}$$

Following the literature, we now briefly discuss how to solve (2.5) efficiently, under the assumption that P is a general matrix and no information about its possible relation to (X, Z, y) is known. The matrix-vector products $\mathcal{F}\,u$ for $u = r_d$ and $u = \mathbf{svec}(dZ)$ on the right-hand sides of (2.5) are

$$\mathcal{F}\; u = \frac{1}{2}\mathbf{svec}\left((P\;X)\;\;\mathbf{smat}(u)\;P^{-1} + P^{-T}\;\mathbf{smat}(u)\;\;(P\;X)^T\right). \tag{2.6}$$

Note that $\mathcal{E}^{-1}$ appears in $\mathcal{M}$ and on the right-hand sides of (2.5). Since $\mathcal{E}$ is an $n(n+1)/2$ by $n(n+1)/2$ matrix, explicitly computing $\mathcal{E}^{-1}$ can be very expensive. However, the expressions $\mathcal{E}^{-1}\;(\mathcal{F}\,r_d - r_c)$ and $\mathcal{E}^{-1}\;(r_c - \mathcal{F}\,\mathbf{svec}(dZ))$ in (2.5) can be computed through two linear systems of equations of the form

$$\mathcal{E}\;u = v \tag{2.7a}$$

with right-hand sides $v = \mathcal{F}\,r_d - r_c$ and $v = r_c - \mathcal{F}\,\mathbf{svec}(dZ)$, respectively. The Schur complement matrix $\mathcal{M}$ can be computed in a similar way: we first explicitly compute the $n(n+1)/2$ by m matrix $\mathcal{F}\,\mathcal{A}^T$, and then compute $\mathcal{E}^{-1}\,\mathcal{F}\,\mathcal{A}^T$ by solving m linear systems of equations of the form (2.7a). To solve (2.7a), we first rewrite it in matrix form as

$$P\;U\;Z\;P^{-1} + P^{-T}\;Z\;U\;P^T = 2\;V, \tag{2.7b}$$

where $U = \mathbf{smat}(u)$ and $V = \mathbf{smat}(v)$ (see (1.6) and (2.2)). By setting

$$\widetilde{U} = P\;U\;P^T \quad\text{and}\quad \widetilde{Z} = P^{-T}\;Z\;P^{-1},$$

we can rewrite (2.7b) as

$$\widetilde{U}\;\widetilde{Z} + \widetilde{Z}\;\widetilde{U} = 2\;V.$$

This last equation is a Lyapunov equation with a positive definite coefficient matrix $\widetilde{Z}$. Hence the solution $\widetilde{U}$ can be efficiently computed via the eigendecomposition of $\widetilde{Z}$ (see sections 3 and 4) and U can be computed from $\widetilde{U}$ as $U = P^{-1}\,\widetilde{U}\,P^{-T}$. Most of the work in solving (2.5) is in the formation and factorization of $\mathcal{M}$.

For the AHO method in section 3 and the methods in the MZ family in section 4, additional information about P and its relation to the current iterate (X, Z, y) is known. We will discuss more efficient ways to get the solutions to (2.5) for these methods in sections 3 and 4, respectively.

2.2. The TTT family. The Schur complement matrix $\mathcal{M}$ is not symmetric in general. Todd, Toh, and Tütüncü [29] considered the family of search directions for which $\mathcal{E}^{-1}\mathcal{F}$ is symmetric, and Monteiro and Zhang [24] provided a parameterization of this family. We refer to it as the TTT family. Let

$$X = R^T\;R \quad\text{and}\quad Z = H^T\;H \tag{2.8}$$

be decompositions of X and Z, respectively. They can be computed via the Cholesky factorizations or the eigendecompositions of X and Z. Let

$$R\,H^T = W\,\Sigma\,V^T \tag{2.9}$$

be the SVD of $R\,H^T$. Assume that $R\,H^T$ has k distinct singular values $\sigma_1 < \cdots < \sigma_k$ and that W and V are chosen such that $\Sigma = \operatorname{diag}(\sigma_1 I_1, \ldots, \sigma_k I_k)$ is a block diagonal matrix with distinct diagonal blocks.

According to Monteiro and Zhang [24], $\mathcal{E}^{-1}\mathcal{F}$ is symmetric if and only if there exists a nonsingular block diagonal matrix $B = \operatorname{diag}(B_1, \ldots, B_k)$, where the dimension of B_j is that of I_j for $j = 1, \ldots, k$, such that

$$P = S\;B\;\widetilde{H}\;, \quad \text{where } S \in \mathbf{R}^{n\times n} \text{ is orthogonal and } \widetilde{H} = V^T\,H, \tag{2.10a}$$

$$= S\;\widetilde{B}\;\widetilde{R}^{-T}, \quad \text{where } \widetilde{R} = W^T\,R \text{ and } \widetilde{B} = B\,\Sigma. \tag{2.10b}$$

Equation (2.10b) is equivalent to (2.10a) because (2.9) implies

$$\widetilde{R}\;\widetilde{H}^T = \Sigma\;. \tag{2.11}$$

Note that the orthogonal matrix S in (2.10) leaves the search direction (2.2) invariant. Furthermore, with some basic linear algebra, it is easy to show that the block diagonal matrix B can always be made diagonal with a proper choice of the singular vector matrices of $R\,H^T$ in (2.9).

The two HKM search directions [15, 17, 20] defined by $P^T\,P = X^{-1}$ and $P^T\,P = Z$ are members in the TTT family with $B = \Sigma^{-1}$ and $B = I$, respectively; and the NT direction [26, 27] (see (1.5)) is a member in the TTT family with $B = \Sigma^{-\frac{1}{2}}$. Let P satisfy (2.10a). Then it is straightforward to verify that

$$\mathbf{H}_P\,(X\;Z) = S\;\Sigma^2\;S^T. \tag{2.12}$$

The matrix $\mathcal{M}$ is positive definite and (2.2) has a unique solution for any member of the TTT family [29].

3. Analysis of the AHO method.

3.1. The AHO method.

The AHO method of [5] is a special case of (2.2) with $P = I$:

$$\mathcal{J}\;d\mathcal{X} = \mathcal{R}, \qquad \text{where} \quad \mathcal{J} = \begin{pmatrix} \mathcal{E} & \mathcal{F} & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ \mathcal{A} & 0 & 0 \end{pmatrix} \quad \text{and} \quad \mathcal{R} = \begin{pmatrix} r_c \\ r_d \\ r_p \end{pmatrix}, \tag{3.1}$$

$$\text{with} \quad \mathcal{E} = Z \otimes_s I, \quad \mathcal{F} = X \otimes_s I, \quad \text{and} \quad r_c = \mathbf{svec}\left(\mu I - \frac{X\;Z + Z\;X}{2}\right).$$

The matrix-vector product (2.6) is

$$\mathcal{F}\;u = \frac{1}{2}\mathbf{svec}\left(X\;\mathbf{smat}(u) + \mathbf{smat}(u)\;X\right).$$

Hence $\mathcal{F}\,u$ can be computed with just one matrix-matrix product, which costs about $2n^3$ flops (see Golub and Van Loan [13, Chap. 1]). The matrix form of (2.7) is simply

$$U\;Z + Z\;U = 2\;V, \tag{3.2}$$

which is already a Lyapunov equation. To solve this equation, eigendecompose Z to get $Z = Q\,\Lambda\,Q^T$, where $Q \in \mathbf{R}^{n\times n}$ is orthogonal and $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n) \succeq 0$ is diagonal. In practice, this computation requires about $9n^3$ flops or fewer (see Demmel [10, Chap. 5] and Golub and Van Loan [13, Chap. 8]). The solution to (3.2) is

$$U = Q\,\bar{U}\,Q^T, \quad \text{where} \quad \bar{U} = \left(\frac{2\bar{V}_{ij}}{\lambda_i + \lambda_j}\right) \quad \text{with} \quad \bar{V} = Q^T\,V\,Q. \tag{3.3}$$

The cost for computing $\bar{V}$ from V is about $3n^3$ flops, taking into account symmetry in $\bar{V}$; the cost for computing $\bar{U}$ from $\bar{V}$ is about n^2 flops; and the cost for computing U from $\bar{U}$ is about $3n^3$ flops.

There are $m+2$ equations of the form (3.2) in (2.5), all of which can be solved via the same eigendecomposition of Z. The total cost for eigendecomposing Z and solving these equations is about $6mn^3$ flops. Adding up the costs for computing $\mathcal{F}\,\mathcal{A}^T$ and computing $\mathcal{M}$ from $\mathcal{E}^{-1}\,\mathcal{F}\,\mathcal{A}^T$, the total cost for computing $\mathcal{M}$ is about $m^2n^2 + 8mn^3$ flops. If we assume that Gaussian elimination with partial pivoting, which is usually stable and costs about $2/3\ m^3$ flops, is used to factorize $\mathcal{M}$, then the total cost for solving (2.5) is about $2/3\ m^3 + m^2n^2 + 8mn^3$ flops. Algorithm 3.1 describes the AHO method.

Algorithm 3.1. AHO Method.

1. Choose $0 \le \sigma < 1$ and determine (dX, dZ, dy) from (3.1), using $\mu = \frac{X\bullet Z}{n}\sigma$.
2. Choose steplengths α and β and update the iterates by

$$(X, Z, y) \leftarrow (X + \alpha\,dX, Z + \beta\,dZ, y + \beta\,dy)\,.$$

The steplength rule is given by choosing a parameter $\tau \in (0,1)$ and defining, via the matrices R and H in factorizations (2.8),

$$\alpha = \begin{cases} 1 & \text{if} \quad \lambda_{\min}\left(R^{-T} dX\,R^{-1}\right) \ge 0, \\ \min\left(1, -\dfrac{\tau}{\lambda_{\min}\left(R^{-T} dX\,R^{-1}\right)}\right) & \text{otherwise;} \end{cases} \tag{3.4a}$$

$$\beta = \begin{cases} 1 & \text{if} \quad \lambda_{\min}\left(H^{-T} dZ\,H^{-1}\right) \ge 0, \\ \min\left(1, -\dfrac{\tau}{\lambda_{\min}\left(H^{-T} dZ\,H^{-1}\right)}\right) & \text{otherwise.} \end{cases} \tag{3.4b}$$

The computation of α and β involves the computation of the factorizations (2.8) and the eigenvalues of $R^{-T}\,dX\,R^{-1}$ and $H^{-T}\,dZ\,H^{-1}$. The total cost for this computation is about $24n^3$ or less (see Golub and Van Loan [13, Chap. 8]). Hence Algorithm 3.1 costs about $2/3\ m^3 + m^2n^2 + 8mn^3$ flops per step for $m \gg 1$.

3.2. Preliminary analysis. To prepare for our analysis of the AHO method in finite precision, in this section we analyze the round-off errors in the solution to (3.2). Assume that the eigendecomposition of Z is computed as

$$Z = \widehat{Q}\,\widehat{\Lambda}\,\widehat{Q}^T + O\left(\epsilon\|Z\|\right), \tag{3.5}$$

where $\widehat{Q}$ is a *nearly* orthogonal matrix satisfying equation $\widehat{Q}^T\,\widehat{Q} = I + O(\epsilon)$, and $\widehat{\Lambda} = \mathrm{diag}(\widehat{\lambda}_1, \ldots, \widehat{\lambda}_n)$ is a diagonal matrix. Then there exists an *exactly* orthogonal matrix $Q^\dagger$ such that $\widehat{Q} = Q^\dagger + O\left(\epsilon\right)$ (see Chandrasekaran and Ipsen [8]). Hence

$$Z = Q^\dagger\,\widehat{\Lambda}\,\left(Q^\dagger\right)^T + O\left(\epsilon\|Z\|\right) \overset{\text{def}}{=} Z^\dagger + O\left(\epsilon\|Z\|\right). \tag{3.6}$$

Note that $Z^\dagger = Q^\dagger \widehat{\Lambda} \left(Q^\dagger\right)^T$ is an exact eigendecomposition. We further let

$$\mathcal{E}^\dagger = Z^\dagger \otimes_s I = \mathcal{E} + O\left(\epsilon \cdot \|Z\|\right) \quad \text{and} \quad \mathcal{M}^\dagger = \mathcal{A}\ \left(\mathcal{E}^\dagger\right)^{-1}\ \mathcal{F}\,\mathcal{A}^T. \tag{3.7}$$

Lemma 3.1 is the basis of our error analysis in section 3.3. We leave its proof to the appendix.

Lemma 3.1. *Assume* (2.7a) *for* $\mathcal{E} = Z \otimes_s I$ *is solved as in* (3.3). *Then*

$$\mathbf{fl}\left(\mathcal{E}^{-1}\ v\right) = \mathbf{svec}\left(\mathbf{fl}\left(U\right)\right) = \left(\mathcal{I} + \Delta_2\right) \cdot \left(\mathcal{E}^\dagger\right)^{-1}\ \left(\mathcal{I} + \Delta_3\right)\ v, \tag{3.8}$$

where $\Delta_2 = O(\epsilon)$ *and* $\Delta_3 = O(\epsilon)$ *are* $n(n+1)/2$ *by* $n(n+1)/2$ *perturbation matrices.*

It is important to note that the matrix $\mathcal{E}^\dagger$ does *not* depend on v. For different right-hand sides v, the corresponding numerical solutions in (3.8) will in general have different perturbation matrices Δ_2 and Δ_3, but always the same $\mathcal{E}^\dagger$.

3.3. Error analysis for the AHO method. In our analysis, we will need some standard results in perturbation theory and error analysis in matrix analysis. Let A be a matrix and x be a vector. Then the round-off errors in the matrix-vector product $A\,x$ satisfy (see Higham [16, Chap. 3])

$$\mathbf{fl}\left(A\ x\right) = \left(A + \delta A\right)\ x, \quad \text{where} \quad |\delta A| \le O(\epsilon)|A|\ . \tag{3.9}$$

A linear system of equations $Ax = b$ is solved backward stably if the computed solution $\widehat{x}$ satisfies $\left(A + \delta A\right)\ \widehat{x} = b$ for $\delta A = O\left(\epsilon\|A\|\right)$. We make the following assumptions.

Assumption 3.1. The matrices A_k have been scaled so that $\|\mathcal{A}\| = \Omega(1)$.

Assumption 3.2. $\sigma_{\min}(\mathcal{J})$ is much larger than $\epsilon\|\mathcal{J}\|$.

Assumption 3.3. The Schur complement $\mathcal{M}$ is explicitly computed and (2.5a) is then solved by a backward stable method.

Assumption 3.4. The current iterate (X, Z, y) is near (X^*, Z^*, y^*) and

$$\|b\| \le O\left(\|\mathcal{A}\|\ \|X\|\right) \quad \text{and} \quad \|C\| \le O\left(\|Z\| + \|\mathcal{A}\|\ \|y\|\right).$$

We note that Assumption 3.2 implies both primal and dual nondegeneracy and strict complementarity. See [5, 14].

In Algorithm 3.1, $d\mathcal{X}$ is computed using (2.5). Round-off errors are in general made at every step of the computation. Let

$$\mathcal{X} = \begin{pmatrix} \mathbf{svec}(X) \\ \mathbf{svec}(Z) \\ y \end{pmatrix}, \quad \widehat{d\mathcal{X}} = \begin{pmatrix} \mathbf{svec}(\widehat{dX}) \\ \mathbf{svec}(\widehat{dZ}) \\ \widehat{dy} \end{pmatrix}, \quad \text{and} \quad \widehat{\mathcal{R}} = \begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \widehat{r}_p \end{pmatrix}.$$

The computation of $\mathcal{R}$ involves a number of simple matrix-matrix and matrix-vector products as well as matrix and vector additions. By standard error analysis (see Golub and Van Loan [13, Chap. 2]) and Assumption 3.4, we have

$$\widehat{\mathcal{R}} = \mathcal{R} + \begin{pmatrix} O\left(\epsilon\|X\|\ \|Z\|\right) \\ O\left(\epsilon \cdot \|Z\| + \epsilon\|\mathcal{A}\|\ \|y\|\right) \\ O\left(\epsilon\|\mathcal{A}\|\ \|X\|\right) \end{pmatrix} = \mathcal{R} + O\left(\epsilon\|\mathcal{J}\|\ \|\mathcal{X}\|\right). \tag{3.10}$$

Theorem 3.2 below is the main result of this section. It puts round-off errors in solving (2.5) into a backward error in $\mathcal{J}$.

THEOREM 3.2. *The computed solution* $(\widehat{dX}, \widehat{dZ}, \widehat{dy})$ *to* (3.1) *by procedure* (2.5) *satisfies*

$$(\mathcal{J} + \delta\mathcal{J})\ \widehat{d\mathcal{X}} = \widehat{\mathcal{R}}\ , \tag{3.11}$$

where $\widehat{\mathcal{R}}$ *satisfies* (3.10) *and* $\delta\mathcal{J}$ *is an* $(n(n+1)+m)$ *by* $(n(n+1)+m)$ *perturbation matrix satisfying*

$$\delta\mathcal{J} = O\left(\epsilon\|\mathcal{J}\|\right) + O\left(\epsilon\left(1+\|\mathcal{A}\|\right)^2\ \left(\|\mathcal{E}\| + \|\mathcal{F}\|\right)\ \|\left(\mathcal{E}^\dagger\right)^{-1}\|\right). \tag{3.12}$$

Proof. We first consider round-off errors on the right-hand side of (2.4). Since $\mathcal{F}\,\widehat{r}_d$ is a matrix-vector product, by (3.9) we have

$$\mathbf{fl}\left(\mathcal{F}\,\widehat{r}_d\right) = \left(\mathcal{F} + \delta_1\mathcal{F}\right)\ \widehat{r}_d, \quad \text{where} \quad \delta_1\mathcal{F} = O\left(\epsilon\|\mathcal{F}\|\right).$$

By our model of arithmetic (1.7), every component in $\mathbf{fl}\left(\mathcal{F}\,\widehat{r}_d\right) - \widehat{r}_c$ is computed to high relative accuracy. So there exists an $n(n+1)/2$ by $n(n+1)/2$ diagonal perturbation matrix $\Delta_4 = O(\epsilon)$ such that

$$\mathbf{fl}\left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right) = \left(\mathcal{I} + \Delta_4\right)\ \left(\left(\mathcal{F} + \delta_1\mathcal{F}\right)\ \widehat{r}_d - \widehat{r}_c\right).$$

According to (3.8), there exist $n(n+1)/2$ by $n(n+1)/2$ perturbation matrices Δ_5 and Δ_6 such that

$$\begin{aligned}\mathbf{fl}\left(\mathcal{E}^{-1}\left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right) &= \left(\mathcal{I} + \Delta_6\right)\ \left(\mathcal{E}^\dagger\right)^{-1}\ \left(\mathcal{I} + \Delta_5\right)\ \mathbf{fl}\left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\\ &= \left(\mathcal{I} + \Delta_6\right)\ \left(\mathcal{E}^\dagger\right)^{-1}\ \left(\mathcal{I} + \Delta_5\right)\ \left(\mathcal{I} + \Delta_4\right)\ \left(\left(\mathcal{F} + \delta_1\mathcal{F}\right)\ \widehat{r}_d - \widehat{r}_c\right).\end{aligned}$$

Putting this together, we can write

$$\begin{aligned}&\mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\,\mathcal{E}^{-1}\ \left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right)\\ &\quad= \left(\mathcal{I} + \Delta_7\right)\ \left(\widehat{r}_p + \mathbf{fl}\left(\mathcal{A}\,\mathcal{E}^{-1}\ \left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right)\right)\\ &\quad= \left(\mathcal{I} + \Delta_7\right)\ \left(\widehat{r}_p + \left(\mathcal{A} + \delta_1\mathcal{A}\right)\ \mathbf{fl}\left(\mathcal{E}^{-1}\left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right)\right)\\ &\quad= \left(\mathcal{I} + \Delta_7\right)\left(\widehat{r}_p + \left(\mathcal{A} + \delta_1\mathcal{A}\right)\left(\mathcal{I} + \Delta_6\right)\left(\mathcal{E}^\dagger\right)^{-1}\left(\mathcal{I} + \Delta_8\right)\left(\left(\mathcal{F} + \delta_1\mathcal{F}\right)\widehat{r}_d - \widehat{r}_c\right)\right), \qquad (3.13)\end{aligned}$$

where $\Delta_7 = O(\epsilon) \in \mathbf{R}^{m\times m}$ is a diagonal perturbation matrix, $\delta_1\mathcal{A} = O(\epsilon\|\mathcal{A}\|)$ is an m by $n(n+1)/2$ perturbation matrix, and $\Delta_8 = \left(\mathcal{I} + \Delta_5\right)\left(\mathcal{I} + \Delta_4\right) - \mathcal{I} = O(\epsilon)$.

Similar to (3.13), the (i,j) entry of the computed Schur complement $\mathcal{M}$ can be written as

$$\begin{aligned}&\mathbf{svec}\left(A_i + \delta_{i,j}A_i\right)^T\ \left(I + \Delta_{i,j}\right)\ \left(\mathcal{E}^\dagger\right)^{-1}\ \left(I + \bar{\Delta}_{i,j}\right)\ \left(\mathcal{F} + \delta_{i,j}\mathcal{F}\right)\ \mathbf{svec}(A_j)\\ &\quad= \mathbf{svec}\left(A_i\right)^T\ \left(\mathcal{E}^\dagger\right)^{-1}\ \mathcal{F}\,\mathbf{svec}\left(A_j\right) + O\left(\epsilon\|\mathcal{A}\|^2\ \|\left(\mathcal{E}^\dagger\right)^{-1}\|\ \|\mathcal{F}\|\right)\\ &\quad= \left(\mathcal{M}^\dagger\right)_{i,j} + O\left(\epsilon\|\mathcal{A}\|^2\ \|\left(\mathcal{E}^\dagger\right)^{-1}\|\ \|\mathcal{F}\|\right).\end{aligned}$$

In other words, the computed $\mathcal{M}$ can be written as $\mathcal{M}^\dagger + O(\epsilon\|\mathcal{A}\|^2\ \|(\mathcal{E}^\dagger)^{-1}\|\ \|\mathcal{F}\|)$. By Assumption 3.3, the backward errors committed during the solution of (2.5a) are bounded by $O(\epsilon\cdot\|\mathcal{M}\|)$, which is also bounded by $O(\epsilon\|\mathcal{A}\|^2\ \|(\mathcal{E}^\dagger)^{-1}\|\ \|\mathcal{F}\|)$. Putting all these errors together, we have

$$\left(\mathcal{M}^\dagger + \delta\mathcal{M}^\dagger\right)\ \widehat{dy} = \mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\,\mathcal{E}^{-1}\ \left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right), \tag{3.14}$$

where $\delta \mathcal{M}^\dagger = O\left(\epsilon \|\mathcal{A}\|^2 \, \| \left(\mathcal{E}^\dagger\right)^{-1} \| \, \|\mathcal{F}\|\right)$.

With similar analysis, the round-off errors in (2.5b) and (2.5c) can be written as

$$\begin{aligned}
\widehat{dZ} &= \mathbf{smat}\left((\mathcal{I} + \Delta_9) \, \left(\widehat{r}_d - (\mathcal{A} + \delta_2 \mathcal{A})^T \, \widehat{dy} \right) \right), \\
\widehat{dX} &= \mathbf{smat}\left((\mathcal{I} + \Delta_{11}) \, \left(\mathcal{E}^\dagger\right)^{-1} \, (\mathcal{I} + \Delta_{10}) \, \left(\widehat{r}_c - (\mathcal{F} + \delta_2 \mathcal{F}) \, \mathbf{svec}\left(\widehat{dZ}\right) \right) \right).
\end{aligned}$$

We now rewrite these equations in a form similar to (2.4) to get

$$\begin{pmatrix} \mathcal{E}^\dagger + \delta \mathcal{E}^\dagger & \mathcal{F} + \delta_2 \mathcal{F} & 0 \\ 0 & (\mathcal{I} + \Delta_9)^{-1} & (\mathcal{A} + \delta_2 \mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger + \delta \mathcal{M}^\dagger \end{pmatrix} \widehat{d\mathcal{X}} = \begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \mathbf{fl}\left(\widehat{r}_p + \mathcal{A} \, \mathcal{E}^{-1} \, (\mathcal{F} \, \widehat{r}_d - \widehat{r}_c) \right) \end{pmatrix}, \tag{3.15}$$

where (see (3.7))

$$\delta \mathcal{E}^\dagger = (\mathcal{I} + \Delta_{10} I)^{-1} \cdot \mathcal{E}^\dagger \, (I + \Delta_{11} I)^{-1} - \mathcal{E}^\dagger = O\left(\epsilon \|Z\|\right).$$

With (3.13), we rewrite (3.15) in a form similar to (2.3):

$$\begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{L}_{3,1} & \mathcal{L}_{3,2} & (\mathcal{I} + \Delta_7)^{-1} \end{pmatrix} \begin{pmatrix} \mathcal{E}^\dagger + \delta \mathcal{E}^\dagger & \mathcal{F} + \delta_2 \mathcal{F} & 0 \\ 0 & (\mathcal{I} + \Delta_9)^{-1} & (\mathcal{A} + \delta_2 \mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger + \delta \mathcal{M}^\dagger \end{pmatrix} \widehat{d\mathcal{X}}$$

$$= \begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \widehat{r}_p \end{pmatrix}, \tag{3.16}$$

where

$$\begin{aligned}
\mathcal{L}_{3,1} &= (\mathcal{A} + \delta_1 \mathcal{A}) \, (\mathcal{I} + \Delta_6) \, \left(\mathcal{E}^\dagger\right)^{-1} \, (\mathcal{I} + \Delta_8) = \mathcal{A} \, \left(\mathcal{E}^\dagger\right)^{-1} + O\left(\epsilon \|\mathcal{A}\| \, \| \left(\mathcal{E}^\dagger\right)^{-1} \| \right), \\
\mathcal{L}_{3,2} &= -\mathcal{L}_{3,1} \left(\mathcal{F} + \delta_1 \mathcal{F}\right) = -\mathcal{A} \, \left(\mathcal{E}^\dagger\right)^{-1} \, \mathcal{F} + O\left(\epsilon \|\mathcal{A}\| \, \| \left(\mathcal{E}^\dagger\right)^{-1} \| \, \|\mathcal{F}\|\right).
\end{aligned}$$

Comparing (3.16) with (3.11), we see that the backward error matrix $\delta \mathcal{J}$ satisfies

$$\begin{aligned}
\delta \mathcal{J} &= \begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{L}_{3,1} & \mathcal{L}_{3,2} & (\mathcal{I} + \Delta_7)^{-1} \end{pmatrix} \begin{pmatrix} \mathcal{E}^\dagger + \delta \mathcal{E}^\dagger & \mathcal{F} + \delta_2 \mathcal{F} & 0 \\ 0 & (\mathcal{I} + \Delta_9)^{-1} & (\mathcal{A} + \delta_2 \mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger + \delta \mathcal{M}^\dagger \end{pmatrix} - \mathcal{J} \\
&= \begin{pmatrix} \mathcal{E}^\dagger + \delta \mathcal{E}^\dagger - \mathcal{E} & \delta_2 \mathcal{F} & 0 \\ 0 & (\mathcal{I} + \Delta_9)^{-1} - \mathcal{I} & \delta_2 \mathcal{A}^T \\ \mathcal{L}_{3,1} \, \mathcal{E}^\dagger - \mathcal{A} & \mathcal{L}_{3,1} \, \left(\mathcal{F} - \mathcal{F} \, (\mathcal{I} + \Delta_9)^{-1} \right) & (\mathcal{I} + \Delta_7)^{-1} \, \mathcal{M}^\dagger + \mathcal{L}_{3,2} \, \mathcal{A}^T \end{pmatrix} \\
&\quad + \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \mathcal{L}_{3,1} \delta \mathcal{E}^\dagger & \mathcal{L}_{3,1} \, \left(\delta_2 \mathcal{F} - \delta_1 \mathcal{F} \, (\mathcal{I} + \Delta_9)^{-1} \right) & (\mathcal{I} + \Delta_7)^{-1} \, \delta \mathcal{M}^\dagger + \mathcal{L}_{3,2} \delta_2 \mathcal{A}^T \end{pmatrix},
\end{aligned}$$

which is bounded by (3.12). □

The first term in (3.12), which includes backward errors in the first two block rows of $\mathcal{J}$, is a small perturbation to $\mathcal{J}$, but the second term, which includes backward errors in the last block row, could be very large. To interpret Theorem 3.2, we need the following result from standard perturbation theory (see, for example, Demmel [10, Chap. 2]):

$$\frac{\left\|\widehat{d\mathcal{X}} - d\mathcal{X}\right\|}{\|d\mathcal{X}\|} \leq \frac{\kappa\left(\mathcal{J}\right)}{1-\kappa\left(\mathcal{J}\right)\,\dfrac{\|\delta\mathcal{J}\|}{\|\mathcal{J}\|}}\,\left(\frac{\|\delta\mathcal{J}\|}{\|\mathcal{J}\|}+\frac{\left\|\widehat{\mathcal{R}}-\mathcal{R}\right\|}{\|\mathcal{R}\|}\right). \tag{3.17}$$

Since Algorithm 3.1 is an iterative method, it usually is not necessary for $d\mathcal{X}$ to be computed very accurately for the method to make progress. However, if the round-off errors in $\mathcal{J}$ are so large that $\|\delta\mathcal{J}\| = \Omega\left(\sigma_{\min}(\mathcal{J})\right)$, then the right-hand side of (3.17) becomes at least $\Omega(1)$ or even undefined, implying that the computed search direction $\widehat{d\mathcal{X}}$ could be *completely* different from $d\mathcal{X}$, making it unlikely that Algorithm 3.1 will make any further progress. It follows that Algorithm 3.1 could stop making further progress if $\|\delta\mathcal{J}\| = \Omega\left(\sigma_{\min}(\mathcal{J})\right)$, or

$$\epsilon\left(1+\|\mathcal{A}\|\right)^2\,\left(\|\mathcal{E}\|+\|\mathcal{F}\|\right)\,\|\left(\mathcal{E}^\dagger\right)^{-1}\| = \Omega\left(\sigma_{\min}(\mathcal{J})\right),$$

which simplifies to

$$\frac{\lambda_{\min}(Z)}{\|Z\|+\|X\|} = \frac{\lambda_{\min}(Z)}{\|\mathcal{E}\|+\|\mathcal{F}\|} = O\left(\epsilon\cdot\kappa(\mathcal{J})\right)\ , \tag{3.18}$$

where we have used the fact that $\|\mathcal{J}\| = O(1)$ and that

$$\lambda_{\min}(Z) = \lambda_{\min}(Z^\dagger) + O(\epsilon\cdot\|Z\|) = \left\|\left(\mathcal{E}^\dagger\right)^{-1}\right\|^{-1} + O(\epsilon\|Z\|).$$

The optimal solution Z^* is in general singular (see (1.2)). Hence one can only expect Algorithm 3.1 to converge to a numerical solution (X, Z, y) which satisfies (3.18). This is not a severe restriction on numerical accuracy if $\mathcal{J}$ is well-conditioned. However, if $\mathcal{J}$ is ill-conditioned, then (3.18) indicates that Algorithm 3.1 could stop making progress well before some eigenvalues of Z become sufficiently small, making it numerically unstable. Since (3.10) indicates that the right-hand side of (2.2) is always computed very accurately, the backward errors in $\mathcal{J}$ appear to be the only source of potential numerical instability in Algorithm 3.1.

We have only analyzed Algorithm 3.1 in section 3.3. Mehrotra's predictor-corrector (PC) rule [19] is a very powerful technique to accelerate convergence and has been extended to Algorithm 3.1 by Alizadeh, Haeberly, and Overton [5]. The PC rule requires the solution of two linear systems of equations with the same coefficient matrix $\mathcal{J}$ in (2.2). Repeating the arguments in section 3.3, it is easy to see that the PC rule might stop making progress as soon as it reaches an iterate (X, Z, y) that satisfies (3.18), and hence it could be numerically unstable if $\mathcal{J}$ is ill-conditioned.

The potential numerical instability of Algorithm 3.1 is due to the block LU factorization procedure discussed in section 2.1. As our numerical results in section 5.3 indicate, this instability is not present if the search direction is computed by solving equation (2.2) as a dense linear system of equations. Similar observations were also made by Alizadeh, Haeberly, and Overton [4].

Finally, we caution that the above analysis merely identifies situations in which Algorithm 3.1 *could* be numerically unstable. It does *not* assert instability in these situations nor does it guarantee progress of Algorithm 3.1 in other situations. Despite this weakness, it is clear that this analysis does provide important new insight into understanding the numerical stability of Algorithm 3.1 in finite precision arithmetic. In section 5 we will present results from our numerical experiments that support the analysis in section 3.3.

3.4. Error analysis for a variation of the AHO method. Several mathematically equivalent formulas are possible for computing the search direction. For example, the expression $\mathcal{F} r_d - r_c$ in (2.5a) can be written equivalently as

$$\mathcal{F}\, r_d - r_c = \mathbf{svec}\left(X\ \left(C - \mathbf{smat}(\mathcal{A}^T\, y)\right) + \left(C - \mathbf{smat}(\mathcal{A}^T\, y)\right)\ X - 2\mu I\right). \tag{3.19}$$

However, Alizadeh, Haeberly, and Overton [5] observed numerical instability leading to significant loss of primal feasibility near the exact solution with (3.19). Todd, Toh, and Tütüncü [29] also observed that some mathematically equivalent formulas for computing the search direction appear to be much less numerically stable than others in the case of the NT method.

In the following, we briefly explain why (3.19) leads to instability. In finite precision computation, define

$$\eta = \mathbf{fl}\left(\mathbf{svec}\left(X\ \left(C - \mathbf{smat}(\mathcal{A}^T\, y)\right) + \left(C - \mathbf{smat}(\mathcal{A}^T\, y)\right)\ X - 2\,\mu\, I\right)\right) - \left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right).$$

Equation (3.13) becomes

$$\begin{aligned}
&\mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\,\mathcal{E}^{-1}\ (\mathcal{F}\,\widehat{r}_d - \widehat{r}_c)\right) \\
&\qquad = (\mathcal{I} + \Delta_7)\left(\widehat{r}_p + (\mathcal{A} + \delta_1\mathcal{A})\,(\mathcal{I} + \Delta_6)\left(\mathcal{E}^\dagger\right)^{-1}(\mathcal{I} + \Delta_8)\,(\mathcal{F}\widehat{r}_d - \widehat{r}_c + \eta)\right) \\
&\qquad = (\mathcal{I} + \Delta_7)\left(\widehat{r}_p + \mathcal{L}_{3,1}\eta + \mathcal{L}_{3,1}\left(\mathcal{F}\widehat{r}_d - \widehat{r}_c\right)\right),
\end{aligned} \tag{3.20}$$

where

$$\mathcal{L}_{3,1} = (\mathcal{A} + \delta_1\mathcal{A})\,(\mathcal{I} + \Delta_6)\left(\mathcal{E}^\dagger\right)^{-1}(\mathcal{I} + \Delta_8).$$

Equation (3.15) is still valid in this case, except that $\mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\,\mathcal{E}^{-1}\ (\mathcal{F}\,\widehat{r}_d - \widehat{r}_c)\right)$ now satisfies (3.20) instead of (3.13). Hence (3.19) amounts to a replacement of mathematically equivalent but numerically different right-hand sides in the middle of a block Gaussian elimination procedure. When (3.19) is used, (3.16) becomes

$$\begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{L}_{3,1} & -\mathcal{L}_{3,1}\,\mathcal{F} & (\mathcal{I} + \Delta_7)^{-1} \end{pmatrix} \begin{pmatrix} \mathcal{E}^\dagger + \delta\mathcal{E}^\dagger & \mathcal{F} + \delta_2\mathcal{F} & 0 \\ 0 & (\mathcal{I} + \Delta_9)^{-1} & (\mathcal{A} + \delta_2\mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger + \delta\mathcal{M}^\dagger \end{pmatrix} \widehat{d\mathcal{X}}$$
$$= \begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \widehat{r}_p + \mathcal{L}_{3,1}\,\eta \end{pmatrix}.$$

On the other hand, similar to (3.10), we have

$$\|\eta\| = O\left(\epsilon \cdot \|\mathcal{J}\|\,\|\mathcal{X}\|\right) \tag{3.21a}$$

and hence

$$\mathcal{L}_{3,1}\,\eta = O\left(\epsilon \cdot \|\mathcal{A}\|\,\|\left(\mathcal{E}^\dagger\right)^{-1}\|\,\|\mathcal{J}\|\,\|\mathcal{X}\|\right). \tag{3.21b}$$

As before, the backward errors in the coefficient matrix of the equation above are bounded by (3.12). However, the round-off errors on the right-hand side could become huge as the iterates converge. For example, assume that the current iterate (X, Z, y) is sufficiently close to (X^*, Z^*, y^*) so that

$$\lambda_{\min}(Z) \leq O\left(\sqrt{\epsilon} \cdot \|\mathcal{X}\|\right) \quad \text{and} \quad \|\mathcal{R}\| \leq O\left(\sqrt{\epsilon} \cdot \|\mathcal{J}\| \, \|\mathcal{X}\|\right).$$

It follows from (3.21) that there might be *no* significant digits at all in the right-hand side vector $\widehat{r}_p + \mathcal{L}_{3,1}\,\eta$, and $\|\widehat{r}_p + \mathcal{L}_{3,1}\,\eta\|$ could be significantly larger than $\|\widehat{r}_c\|$ and $\|\widehat{r}_d\|$. Hence the computed search direction could be completely in error. It follows that the AHO method with (3.19) could stop making progress when $\|\mathcal{R}\| = O\left(\sqrt{\epsilon}\|\mathcal{J}\|\,\|\mathcal{X}\|\right)$, even if $\mathcal{J}$ is well-conditioned.

Similar analysis holds for the NT method. As we will show in section 4, the NT method, when implemented according to a similar block Gaussian elimination procedure, is reasonably accurate in general. On the other hand, if mathematically equivalent but numerically different formulas are used to replace computed quantities during the computation, as is done for the AHO method in (3.20), then the resulting method could be highly unstable. The same argument holds for all other methods in the TTT family as well.

4. Analysis of the TTT methods.

4.1. The TTT methods. A search direction in the TTT family is a search direction defined by (2.2) with P satisfying one of the two mathematically equivalent equations in (2.10). We assume that a proper choice of the singular vector matrices of $R\,H^T$ in (2.9) has been made so that B is a diagonal matrix. Arrange the singular values as $0 < \sigma_1 \leq \sigma_2 \leq \cdots \leq \sigma_n$ and let

$$B = \operatorname{diag}(\beta_1, \ldots, \beta_n) \quad \text{and} \quad \widetilde{B} = B\,\Sigma = \operatorname{diag}\left(\beta_1\,\sigma_1, \ldots, \beta_n\,\sigma_n\right). \tag{4.1}$$

We will base our development on the assumption that P is chosen using expression (2.10a). It follows from (2.2) and (2.12) that

$$\mathcal{E} = \left(S\,B^{-1}\,\widetilde{H}\right) \otimes_s \left(S\,B\,\widetilde{H}\right) \quad \text{and} \quad \mathcal{F} = \left(S\,B\,\widetilde{H}\,X\right) \otimes_s \left(S\,B^{-1}\,\widetilde{H}^{-T}\right). \tag{4.2a}$$

With (2.10b) and (2.11), these expressions can be rewritten as

$$\mathcal{E} = \left(S\,B^{-1}\,\Sigma\,\widetilde{R}^{-T}\right) \otimes_s \left(S\,B\,\Sigma\,\widetilde{R}^{-T}\right) \quad \text{and} \quad \mathcal{F} = \left(S\,\widetilde{B}\,\widetilde{R}\right) \otimes_s \left(S\,\widetilde{B}^{-1}\,\widetilde{R}\right). \tag{4.2b}$$

Similarly, (2.2), (2.10), and (2.12) imply that

$$r_c = \mathbf{svec}\left(\mu\,I - S\,\mathbf{H}_B\left(\widetilde{H}\,X\,\widetilde{H}^T\right)\,S^T\right) = \mathbf{svec}\left(S\,\mathbf{smat}\left(\widetilde{r}_c\right)\,S^T\right), \tag{4.3}$$

where

$$\widetilde{r}_c \overset{\text{def}}{=} \mathbf{svec}\left(\mu\,I - \Sigma^2\right).$$

With (4.2b), the matrix-vector product (2.6) can be written as

$$\begin{aligned} \mathcal{F}\,u &= \frac{1}{2}\mathbf{svec}\left(S\,\left(\widetilde{B}\,\widetilde{R}\,\mathbf{smat}(u)\,\widetilde{R}^T\,\widetilde{B}^{-1} + \widetilde{B}^{-1}\,\widetilde{R}\,\mathbf{smat}(u)\,\widetilde{R}^T\,\widetilde{B}\right)\,S^T\right) \\ &= \frac{1}{2}\mathbf{svec}\left(S\,\left(D_{\mathcal{F}} \odot \left(\widetilde{R}\,\mathbf{smat}(u)\,\widetilde{R}^T\right)\right)\,S^T\right)\,, \end{aligned} \tag{4.4}$$

where $X \odot Y = (X_{i,j}Y_{i,j})$ is the *Hadamard product* and

$$D_{\mathcal{F}} = \left(\frac{\beta_i\,\sigma_i}{\beta_j\,\sigma_j} + \frac{\beta_j\,\sigma_j}{\beta_i\,\sigma_i}\right) = \left(\frac{\beta_i^2\,\sigma_i^2+\beta_j^2\,\sigma_j^2}{\beta_i\,\beta_j\,\sigma_i\,\sigma_j}\right).$$

We now solve (2.7). With (4.2b), the left-hand side of (2.7b) can be rewritten as

$$\begin{aligned} & S\left(B^{-1}\Sigma\,\widetilde{R}^{-T}U\,\widetilde{R}^{-1}\Sigma\,B + B\,\Sigma\,\widetilde{R}^{-T}U\,\widetilde{R}^{-1}\Sigma\,B^{-1}\right)S^T \\ &= S\left(\left(\frac{\beta_j\,\sigma_i\,\sigma_j}{\beta_i}+\frac{\beta_i\,\sigma_i\,\sigma_j}{\beta_j}\right)\cdot\left(\widetilde{R}^{-T}U\,\widetilde{R}^{-1}\right)_{ij}\right)S^T.\end{aligned}$$

Hence the solution to (2.7) is

$$\mathcal{E}^{-1}v = 2\mathbf{svec}\left(\widetilde{R}^T\left(D_{\mathcal{E}}\odot\left(S^T\mathbf{smat}(v)S\right)\right)\widetilde{R}\right)\ , \qquad D_{\mathcal{E}} = \left(\frac{\beta_i\beta_j}{\sigma_i\sigma_j\left(\beta_i^2+\beta_j^2\right)}\right). \tag{4.5}$$

With (4.4) and (4.5), we can rewrite the Schur complement matrix $\mathcal{M}$ as

$$\begin{aligned} \mathcal{M} &= \left(\mathbf{svec}(A_i)^T\,\mathcal{E}^{-1}\,\mathcal{F}\,\mathbf{svec}(A_j)\right) \\ &= 2\,\left(\mathbf{svec}(A_i)^T\mathbf{svec}\left(\widetilde{R}^T\ \left(D_{\mathcal{E}}\odot\left(S^T\,\mathbf{smat}\left(\mathcal{F}\,\mathbf{svec}(A_j)\right)\ S\right)\right)\ \widetilde{R}\right)\right) \\ &= \left(\mathbf{svec}(A_i)^T\mathbf{svec}\left(\widetilde{R}^T\ \left(D_{\mathcal{E}}\odot\left(S^T\left(S\left(D_{\mathcal{F}}\odot\left(\widetilde{R}\,A_j\,\widetilde{R}^T\right)\right)S^T\right)S\right)\ \widetilde{R}\right)\right)\right) \\ &= \left(\mathbf{svec}(A_i)^T\mathbf{svec}\left(\widetilde{R}^T\ \left((D_{\mathcal{E}}\odot D_{\mathcal{F}})\odot\left(\widetilde{R}\,A_j\,\widetilde{R}^T\right)\right)\ \widetilde{R}\right)\right) \\ &= \left(\left(\widetilde{R}\,A_i\,\widetilde{R}^T\right)\bullet\left((D_{\mathcal{E}}\odot D_{\mathcal{F}})\odot\left(\widetilde{R}\,A_j\,\widetilde{R}^T\right)\right)\right) = \widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}\,\widetilde{\mathcal{A}}^T, \end{aligned} \tag{4.6}$$

where $\widetilde{\mathcal{A}} = (\mathbf{svec}(\widetilde{R}\ A_1\ \widetilde{R}^T),\ldots,\mathbf{svec}(\widetilde{R}\ A_m\ \widetilde{R}^T))^T$ and $\mathcal{D}_{\mathcal{M}}$ is an $n(n+1)/2$ by $n(n+1)/2$ diagonal matrix that satisfies

$$\mathcal{D}_{\mathcal{M}}\ \mathbf{e} = \mathbf{svec}\left(D_{\mathcal{E}}\odot D_{\mathcal{F}}\right) = \mathbf{svec}\left(\frac{\beta_i^2\,\sigma_i^2+\beta_j^2\,\sigma_j^2}{\sigma_i^2\,\sigma_j^2\,\left(\beta_i^2+\beta_j^2\right)}\right)$$

for the vector $\mathbf{e}$ in section 1.4. Note that the matrix $\mathcal{D}_{\mathcal{M}}$ is the only part in $\mathcal{M}$ that is affected by B. For *any* choice of B, the entries of the matrix $D_{\mathcal{E}}\odot D_{\mathcal{F}}$ are always bounded. In fact,

$$\frac{1}{\sigma_i^2+\sigma_j^2} \le \frac{\beta_i^2\,\sigma_i^2+\beta_j^2\,\sigma_j^2}{\sigma_i^2\,\sigma_j^2\,\left(\beta_i^2+\beta_j^2\right)} \le \frac{1}{\sigma_i^2}+\frac{1}{\sigma_j^2}. \tag{4.7}$$

Now we use (4.4) and (4.5) to simplify the right-hand side of (2.5). By (4.3) and (4.5),

$$\begin{aligned} \mathcal{E}^{-1}\ r_c &= 2\ \mathbf{svec}\left(\widetilde{R}^T\ \left(D_{\mathcal{E}}\odot\left(S^T\left(\mu\ I - S\ \Sigma^2\ S^T\right)S\right)\right)\ \widetilde{R}\right) \\ &= \mathbf{svec}\left(\widetilde{R}^T\ \left(D_{\mathcal{E}}\odot\mathbf{smat}\left(\widetilde{r}_c\right)\right)\ \widetilde{R}\right). \end{aligned}$$

Combining this relation with (4.4) and (4.5), and with some algebra similar to that used to obtain (4.6),

$$\begin{aligned} & \mathcal{E}^{-1}\ \left(\mathcal{F}\ r_d - r_c\right) \\ &= \mathbf{svec}\left(\widetilde{R}^T\ \left((D_{\mathcal{E}}\odot D_{\mathcal{F}})\odot\left(\widetilde{R}\,\mathbf{smat}(r_d)\ \widetilde{R}^T\right) - D_{\mathcal{E}}\odot\mathbf{smat}\left(\widetilde{r}_c\right)\right)\ \widetilde{R}\right). \end{aligned} \tag{4.8}$$

However, for numerical stability reasons we will compute $\mathcal{E}^{-1}\,(r_c - \mathcal{F}\,\mathbf{svec}(dZ))$ differently. With (4.2a), the left-hand side of (2.7b) can be rewritten as

$$S\left(B^{-1}\,\widetilde{H}\,U\,\widetilde{H}^T\,B + B\,\widetilde{H}\,U\,\widetilde{H}^T\,B^{-1}\right)S^T = S\left(\left(\frac{\beta_j}{\beta_i}+\frac{\beta_i}{\beta_j}\right)\cdot\left(\widetilde{H}\,U\,\widetilde{H}^T\right)_{ij}\right)S^T.$$

Hence the solution to (2.7) can also be written as

$$\mathcal{E}^{-1}\,v = 2\,\mathbf{svec}\left(\widetilde{H}^{-1}\,\left(\bar{D}_{\mathcal{E}}\odot\left(S^T\,\mathbf{smat}(v)\,S\right)\right)\,\widetilde{H}^{-T}\right)\text{ where }\bar{D}_{\mathcal{E}} = \left(\frac{\beta_i\,\beta_j}{\beta_i^2+\beta_j^2}\right).$$

Combined with (4.3) and (4.4), and after some algebra, we obtain

$$\begin{aligned}&\mathcal{E}^{-1}\,\left(r_c - \mathcal{F}\,\mathbf{svec}(dZ)\right)\\ &= \mathbf{svec}\left(\widetilde{H}^{-1}\,\left(\bar{D}_{\mathcal{E}}\odot\mathbf{smat}\left(\widetilde{r}_c\right) - \left(\bar{D}_{\mathcal{E}}\odot D_{\mathcal{F}}\right)\odot\left(\widetilde{R}\,dZ\,\widetilde{R}^T\right)\right)\,\widetilde{H}^{-T}\right).\end{aligned}\tag{4.9}$$

As we have seen throughout section 4.1, due to relation (2.11), $\mathcal{E}$, $\mathcal{F}$, and $\mathcal{M}$ can be expressed in several different but mathematically equivalent ways, each of which may lead to a different numerical solution to (2.5). We have chosen to solve (2.5) via the SVD (2.9) so as to keep the symmetry of $\mathcal{M}$ explicit and to avoid the explicit inversion of $\widetilde{H}$ and $\widetilde{R}$ everywhere except in (4.9); this allows us to derive a relatively favorable error analysis. Our approach is somewhat different from those of Monteiro and Zhang [24] and Todd, Toh, and Tütüncü [29]. Algorithm 4.1 below is a more formal description of the method described in this section. We will postpone some details on how to compute expressions in (4.8) and (4.9) to section 4.3. We will also discuss a new choice of B in section 5.1.

Algorithm 4.1. TTT Methods.

1. Choose a matrix B in (2.10a).
2. Choose $0 \leq \sigma < 1$ and determine (dX, dZ, dy) from (2.5a), (2.5b), (4.6), (4.8), and (4.9), using
$$\mu = \frac{X\bullet Z}{n}\sigma = \frac{\mathbf{tr}\left(\Sigma^2\right)}{n}\sigma.$$
3. Choose steplengths α and β using (3.4) and update the iterates by
$$(X, Z, y) \leftarrow (X+\alpha\,dX, Z+\beta\,dZ, y+\beta\,dy).$$

The main cost of Algorithm 4.1 is in the computation and factorization of $\mathcal{M}$. To compute $\widetilde{\mathcal{A}}$ in (4.6), we need to explicitly compute the matrices $\widetilde{R}\,A_i\,\widetilde{R}^T$ for $i = 1,\ldots,m$, which costs about $3mn^3$ flops (see section 4.3). Since $\mathcal{M}$ is symmetric, computing $\mathcal{M}$ from $\widetilde{\mathcal{A}}$ costs about $1/2\ m^2n^2$ flops. The Cholesky factorization of $\mathcal{M}$ costs about $1/3\ m^3$ flops. Adding it all up, we see that Algorithm 4.1 costs about $1/3\,m^3 + 1/2\,m^2n^2 + 3mn^3$ flops per step, roughly half of the per-step cost of Algorithm 3.1. Like Algorithm 3.1, the PC rule can also be extended to Algorithm 4.1 (see Monteiro and Zhang [24] and Todd, Toh, and Tütüncü [29]).

4.2. Variations of the TTT methods.

Todd, Toh, and Tütüncü [29] showed that the Schur complement equation (2.5a) can be expressed as the normal equation of a linear least squares problem. In fact, let $\widetilde{\mathcal{D}}$ be an $n(n+1)/2$ by $n(n+1)/2$ diagonal matrix that satisfies

$$\widetilde{\mathcal{D}}\,\mathbf{e} = \mathbf{svec}\left(\frac{2\,\beta_i\,\beta_j}{\sqrt{\left(\beta_i^2\,\sigma_i^2 + \beta_j^2\,\sigma_j^2\right)\,\left(\beta_i^2 + \beta_j^2\right)}}\right)$$

for the vector $\mathbf{e}$ in section 1.4. With (4.5) and some algebra similar to that used to obtain (4.6), we can write $\mathcal{A}\,\mathcal{E}^{-1} = \widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\,\widetilde{\mathcal{D}}$. Let $X_r \in \mathbf{S}^n$ be a symmetric matrix such that

$$\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\,\mathbf{svec}(X_r) = r_p\,. \tag{4.10}$$

Then the Schur complement equation (2.5a) can be rewritten as

$$\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)\cdot\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)^T\cdot dy = \left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)\cdot\left(\mathbf{svec}(X_r) + \widetilde{\mathcal{D}}\,\left(\mathcal{F}\,r_d - r_c\right)\right),$$

which is the *normal equation* for the LS problem

$$\min_{dy}\left\|\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)^T dy - \left(\mathbf{svec}(X_r) + \widetilde{\mathcal{D}}\,\left(\mathcal{F}\,r_d - r_c\right)\right)\right\|\,. \tag{4.11}$$

Hence dy is the solution to the LS problem (4.11) and can be computed by standard methods for solving LS problems, which are both efficient and backward stable (see, for example, Golub and Van Loan [13, Chap. 5]).

Similar to Algorithm 4.1, the main cost of the LS approach is in explicitly computing and factorizing the coefficient matrix $\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}$. As in Algorithm 4.1, the cost for computing $\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}$ is about $3mn^3$ flops. If the least squares problem (4.11) is solved by computing the QR factorization of $\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}$, then the cost of this factorization is about $m^2n^2 - 2/3\;m^3$ flops. Hence the total per-step cost of the least squares approach is about $m^2n^2 + 3mn^3 - 2/3\;m^3$ flops, roughly twice the per-step cost of Algorithm 4.1 if $n \ll m \ll n^2$.

Since reliable methods for computing the SVD are only available for dense matrices, a potential drawback with Algorithm 4.1 is that the SVD computation in (2.9) could be very inefficient if matrices R and H were highly sparse or structured. Similar problems arise even if this SVD is replaced by an eigendecomposition of $R\,Z\,R^T$ (see [24]). Zhang [34] pointed out that in the special case $B = I$ (or $P^T\,P = Z$), (2.5) can be solved without the SVD:

$$\begin{aligned}\mathcal{M}\,dy &= r_p + \mathcal{A}\,\left(\left(X \otimes_s Z^{-1}\right)\,r_d - \mathbf{svec}\left(\mu\,Z^{-1} - X\right)\right),\\ dZ &= \mathbf{smat}\left(r_d - \mathcal{A}^T\,dy\right),\\ dX &= \mu\,Z^{-1} - X - \left(X \otimes_s Z^{-1}\right)\,\mathbf{svec}(dZ)\,.\end{aligned}$$

See Zhang [34] for details.

4.3. Preliminary analysis. To motivate our error analysis of Algorithm 4.1, in section 4.3 we examine (2.1c) in *exact arithmetic.* Since

$$P\,dX\,\left(Z\,P^{-1}\right) = S\,B\,\widetilde{H}\,dX\,\widetilde{H}^T\,B^{-1}\,S^T \text{ and } (P\,X)\;dZ\,P^{-1} = S\,\widetilde{B}\,\widetilde{R}\,dZ\,\widetilde{R}^T\,\widetilde{B}^{-1}\,S^T,$$

and since $r_c = \mathbf{svec}\left(\mu\, I - S\, \Sigma^2\, S^T\right)$, (2.1c) simplifies to

$$\mathbf{H}_B\left(\widetilde{H}\; dX\; \widetilde{H}^T\right) + \mathbf{H}_{\widetilde{B}}\left(\widetilde{R}\; dZ\; \widetilde{R}^T\right) = \mu I - \Sigma^2.$$

With (4.1), this can be further written as

$$\frac{1}{2}\left(\left(\frac{\beta_i}{\beta_j} + \frac{\beta_j}{\beta_i}\right)\cdot\left(\widetilde{H}\, dX\, \widetilde{H}^T\right)_{i,j}\right) + \frac{1}{2}\left(\left(\frac{\beta_i\,\sigma_i}{\beta_j\,\sigma_j} + \frac{\beta_j\,\sigma_j}{\beta_i\,\sigma_i}\right)\cdot\left(\widetilde{R}\, dZ\, \widetilde{R}^T\right)_{i,j}\right) = \mu I - \Sigma^2. \tag{4.12}$$

As we discussed at the end of section 2, the two HKM search directions [15, 17, 20] defined by $P^T P = X^{-1}$ and $P^T P = Z$ correspond to the choices $\beta_i = 1/\sigma_i$ and $\beta_i = 1$, respectively, and the NT direction [26, 27] corresponds to the choice $\beta_i = 1/\sqrt{\sigma_i}$. In general, however, the β_i's can be any positive numbers for the TTT family, making the ratios involving β_i's potentially huge. In addition, the matrices $\widetilde{H}$ and $\widetilde{R}$ themselves could be badly scaled as well. To see this, assume for the moment that X and Z commute so that we can write their eigendecompositions as $X = Q\,\Lambda_X\,Q^T$ and $Z = Q\,\Lambda_Z\,Q^T$, where Q is an orthogonal matrix and both Λ_X and Λ_Z are positive diagonal matrices. Equation (2.8) implies that there exist orthogonal matrices W_X and W_Z such that

$$R = W_X\,\Lambda_X^{\frac{1}{2}}\,Q^T \quad \text{and} \quad H = W_Z\,\Lambda_Z^{\frac{1}{2}}\,Q^T\ , \quad \text{or} \quad R\,H^T = W_X\ \left(\Lambda_X\,\Lambda_Z\right)^{\frac{1}{2}}\ W_Z^T.$$

By the definitions of the SVD in (2.9) and matrices $\widetilde{R}$ and $\widetilde{H}$ in (2.10), we get

$$\widetilde{R} = W_X^T\,R = \Lambda_X^{\frac{1}{2}}\,Q^T \quad \text{and} \quad \widetilde{H} = W_Z^T\,H = \Lambda_Z^{\frac{1}{2}}\,Q^T.$$

In other words, if X and Z commute, then $\widetilde{R}$ and $\widetilde{H}$ are row-scaled by their singular values. For X and Z that are close to the optimal solution (X^*, Z^*), some of these singular values will be very small. In practice, X and Z do not commute but become closer to commuting as they converge to (X^*, Z^*), making $\widetilde{R}$ and $\widetilde{H}$ potentially badly scaled. To fully understand the scaling problem in (4.12), we rewrite $\widetilde{R}$ and $\widetilde{H}$ in scaled forms as

$$\widetilde{R} = \Psi\,\bar{R} \quad \text{and} \quad \widetilde{H} = \Phi\,\bar{H}\ , \tag{4.13}$$

where $\Psi = \operatorname{diag}(\psi_1, \ldots, \psi_n)$ and $\Phi = \operatorname{diag}(\phi_1, \ldots, \phi_n)$ are chosen so that rows of $\bar{R}$ and $\bar{H}$ all have 2-norm 1. Some of the ϕ_i's and ψ_i's could be very small, especially when the iterates are close to the optimal solution. Equation (4.12) now becomes

$$\frac{1}{2}\left(\left(\frac{\beta_i}{\beta_j} + \frac{\beta_j}{\beta_i}\right)\,\phi_i\,\phi_j\,\left(\bar{H}\,dX\,\bar{H}^T\right)_{i,j}\right) + \frac{1}{2}\cdot\left(\left(\frac{\beta_i\,\sigma_i}{\beta_j\,\sigma_j} + \frac{\beta_j\,\sigma_j}{\beta_i\,\sigma_i}\right)\,\psi_i\,\psi_j\,\left(\bar{R}\,dZ\,\bar{R}^T\right)_{i,j}\right)$$
$$= \mu I - \Sigma^2. \tag{4.14}$$

We note that the ratios involving β_i's could be huge, while some of the factors involving ϕ_i's and ψ_i's could be very small. Consequently, among the $n(n+1)/2$ scalar equations in (4.14), some might have huge coefficients whereas others might *only* have small ones. This bad scaling could cause the matrix $\mathcal{J}$ in (2.2) to be arbitrarily ill-conditioned, even when $\mathcal{J}$ with the choice $P = I$ is well-conditioned. In exact arithmetic analysis, this bad scaling problem can be avoided by dividing the (i, j) entry in the matrix equation by

$$\left(\frac{\beta_i}{\beta_j} + \frac{\beta_j}{\beta_i}\right)\,\phi_i\,\phi_j + \left(\frac{\beta_i\,\sigma_i}{\beta_j\,\sigma_j} + \frac{\beta_j\,\sigma_j}{\beta_i\,\sigma_i}\right)\,\psi_i\,\psi_j.$$

The situation is more complex, however, in finite arithmetic analysis. See section 4.4.

We now discuss the round-off errors in the following operations required in Algorithm 4.1:

$$\mathcal{D}\ (U \otimes_s U)\ \mathbf{svec}(A) = \mathbf{svec}\left(D \odot \left(U\,A\,U^T\right)\right),$$
$$(U \otimes_s U)^{-1}\ \mathbf{svec}(A) = \mathbf{svec}\left(U^{-1}\,A\,U^{-T}\right),$$

where A and $D \in \mathbf{S}^n$, and where $\mathcal{D}$ is an $n(n+1)/2$ by $n(n+1)/2$ diagonal matrix such that $\mathcal{D}\,\mathbf{e} = \mathbf{svec}(D)$. We summarize their computations in Algorithms 4.2 and 4.3 and their error analysis in Lemma 4.1. We leave the proof of Lemma 4.1 to the appendix.

Algorithm 4.2. Computing $\mathcal{V} = \mathcal{D}\ (U \otimes_s U)\ \mathbf{svec}(A)$.

1. Compute the matrix-matrix product $U\,A$.
2. Compute the (i,j) and (j,i) entries of $U\,A\,U^T$ as the sum $\sum_{k=1} \left(\mathbf{fl}(U\,A)\right)_{ik}\ U_{jk}$.
3. Compute $\mathbf{fl}\left(D \odot \mathbf{fl}\left(U\,A\,U^T\right)\right)$.

Algorithm 4.3. Computing $\mathcal{W} = (U \otimes_s U)^{-1}\ \mathbf{svec}(A)$.

1. Factorize U with an efficient and backward stable method such as QR factorization.
2. Let $A = (a_1, \ldots, a_n)$. Compute $U^{-1}\,A$ by solving n linear systems of equations $U\,v_i = a_i$.
3. Let $\mathbf{fl}\left(U^{-1}\,A\right) = (\widetilde{v}_1, \ldots, \widetilde{v}_n)^T$. Compute $U^{-1}\,A\,U^{-T}$ by solving n linear systems of equations $\widetilde{w}_i^T\,U^T = \widetilde{v}_i^T$ and symmetrizing $(\widetilde{w}_1, \ldots, \widetilde{w}_n)$.

Algorithm 4.3 is not very efficient since it does not take advantage of the symmetry in $U^{-1}\,A\,U^{-T}$. This extra cost can be avoided with a more involved algorithm and is small compared to other costs in Algorithm 4.1. To achieve good accuracy in computing (4.9) for Algorithm 4.1, we rewrite it using (4.13) as

$$\mathcal{E}^{-1}\left(r_c - \mathcal{F}\mathbf{svec}(dZ)\right)$$
$$= \mathbf{svec}\left(\bar{H}^{-1}\Phi^{-1}\left(\bar{D}_{\mathcal{E}} \odot \mathbf{smat}\left(\widetilde{r}_c\right) - \left(\bar{D}_{\mathcal{E}} \odot D_{\mathcal{F}}\right) \odot \left(\widetilde{R}\,dZ\,\widetilde{R}^T\right)\right)\Phi^{-1}\bar{H}^{-T}\right), \tag{4.15}$$

and compute (4.15) using Algorithm 4.3 with $U = \bar{H}$.

Lemma 4.1. *Let* $\widehat{\mathcal{V}}$ *and* $\widehat{\mathcal{W}}$ *be the computed counterparts of* $\mathcal{V}$ *and* $\mathcal{W}$ *in Algorithms* 4.2 *and* 4.3*, respectively, and assume that* $\kappa(U)$ *is much smaller than* $1/\sqrt{\epsilon}$ *in Algorithm* 4.3*. Then there exist* $n(n+1)/2$ *by* $n(n+1)/2$ *matrices* Θ_1 *and* Θ_2 *such that*

$$\widehat{\mathcal{V}} = \mathcal{D}\ (U \otimes_s U + \Theta_1)\ \mathbf{svec}(A) \quad \textit{and} \quad (U \otimes_s U + \Theta_2) \cdot \widehat{\mathcal{W}} = \mathbf{svec}(A),$$

where $|\Theta_1| \leq O(\epsilon)\left(|U| \otimes_s |U|\right)$ *and* $\|\Theta_2\| \leq O\left(\epsilon \cdot \|U\|^2\right)$.

4.4. Error analysis for the TTT methods. The error analysis for the TTT methods is much more complicated than that for the AHO method, due to the potentially bad scaling of the complementarity equation (2.1c) for the TTT methods. To shorten the presentation, we will summarize some pieces of analysis into lemmas and a theorem and leave some of their proofs to the appendix.

We begin by examining the round-off errors in the decompositions (2.8) and the SVD (2.9). Assume that they are computed backward stably as

$$\widehat{R}^T\,\widehat{R} = X + O(\epsilon\|X\|),\ \widehat{H}^T\,\widehat{H} = Z + O(\epsilon\|Z\|),\ \text{and}\ \widehat{R}\,\widehat{H}^T = \widehat{W}\,\widehat{\Sigma}\,\widehat{V}^T + O\left(\epsilon\|\widehat{R}\|\,\|\widehat{H}\|\right),$$

where $\widehat{W}$ and $\widehat{V}$ are nearly orthogonal matrices satisfying $\widehat{W}^T\ \widehat{W} = I + O(\epsilon)$ and $\widehat{V}^T\ \widehat{V} = I + O(\epsilon)$, respectively, and $\widehat{\Sigma} = \operatorname{diag}(\widehat{\sigma}_1, \ldots, \widehat{\sigma}_n)$. Let $\widetilde{R}$ and $\widetilde{H}$ be computed as

$$\mathbf{fl}(\widetilde{R}) = \widehat{W}^T\ \widehat{R} + O(\epsilon\|\widetilde{R}\|) \quad \text{and} \quad \mathbf{fl}(\widetilde{H}) = \widehat{V}^T\ \widehat{H} + O(\epsilon\|\widetilde{H}\|).$$

We define

$$X^\dagger \stackrel{\text{def}}{=} \left(\mathbf{fl}(\widetilde{R})\right)^T\ \mathbf{fl}(\widetilde{R}) = X + O(\epsilon\|X\|) \quad \text{and} \quad Z^\dagger \stackrel{\text{def}}{=} \left(\mathbf{fl}(\widetilde{H})\right)^T\ \mathbf{fl}(\widetilde{H}) = Z + O(\epsilon\|Z\|).$$

To make the notation less cluttered, in the remainder of this section, we will drop the symbol $\mathbf{fl}$ in $\mathbf{fl}(\widetilde{R})$ and $\mathbf{fl}(\widetilde{H})$ and replace them by $\widetilde{R}$ and $\widetilde{H}$, respectively. Combining the above equations, we get

$$(4.16)\ X^\dagger = \widetilde{R}^T\,\widetilde{R}, \quad Z^\dagger = \widetilde{H}^T\,\widetilde{H}, \quad \text{and} \quad \widetilde{R}\,\widetilde{H}^T = \widehat{\Sigma} + O\left(\epsilon\|\widetilde{R}\|\,\|\widetilde{H}\|\right) \stackrel{\text{def}}{=} \widehat{\Sigma} + E.$$

In our analysis, we will think of the search direction defined by (2.2) as a direction defined at the point $(X^\dagger, Z^\dagger, y)$, instead of (X, Z, y). These two points are identical in exact arithmetic and differ slightly in finite precision. However, this minor difference will make our analysis much simpler. Since the round-off error matrix E in (4.16) is in general nonzero, the expressions in (2.10) for P and the expressions in (4.2) for $\mathcal{E}$ and $\mathcal{F}$, while mathematically equivalent in exact arithmetic, are *inconsistent* in finite precision arithmetic. As in section 4.1, we base our analysis on the assumption that P is chosen using (2.10a): $P = S\ B\ \widetilde{H}$. We also set $S = I$ since it is never involved in any computation (see section 4.1). Under this choice of P, the search direction defined by (2.2) at the point $(X^\dagger, Z^\dagger, y)$ satisfies (cf. (4.2a))

$$(4.17)\ \ \mathcal{J}\ d\mathcal{X} = \mathcal{R}, \qquad \text{where} \quad \mathcal{J} = \begin{pmatrix} \mathcal{E} & \mathcal{F} & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ \mathcal{A} & 0 & 0 \end{pmatrix} \quad \text{and} \quad \mathcal{R} = \begin{pmatrix} r_c \\ r_d \\ r_p \end{pmatrix},$$

with $\mathcal{E} = (B^{-1}\,\widetilde{H}) \otimes_s (B\,\widetilde{H})$, $\mathcal{F} = (B\,\widetilde{H}\,X^\dagger) \otimes_s (B^{-1}\,\widetilde{H}^{-T})$, $r_p = b - \mathcal{A}\,\mathbf{svec}(X^\dagger)$, $r_c = \mathbf{svec}(\mu\,I - \mathbf{H}_B(\widetilde{H}\,X^\dagger\,\widetilde{H}^T))$, and $r_d = \mathbf{svec}(C - Z^\dagger - \mathbf{smat}(\mathcal{A}^T\,y))$.

We could try to write the round-off errors during the computation of the search direction as perturbations to (4.17), in a form similar to (3.11). However, the coefficient matrix $\mathcal{J}$ in (4.17) is in general badly scaled and hence ill-conditioned. To make our error analysis more meaningful, we need to rescale the rows of $\mathcal{J}$ to make it as balanced as round-off errors in $\mathcal{E}$ and $\mathcal{F}$ permit, and then examine the error bounds in the rescaled version of (4.17). Rescaling is a technique often used in error analysis for linear systems solutions to reveal the effective condition number. For discussions of this technique and related literature, see Demmel [10, Chap. 2] and Golub and Van Loan [13, Chap. 3].

In this section, in addition to Assumptions 3.1–3.3, we make the following assumptions.

Assumption 4.1. The error matrix E in (4.16) satisfies $\|E\| \leq \min_{i=1}^n \widehat{\sigma}_i/2$.

Assumption 4.2. The matrix $\bar{H}$ defined in (4.13) satisfies $\kappa(\bar{H}) \ll 1/\sqrt{\epsilon}$.

We start our analysis by revealing the bad scaling in the matrix $\mathcal{J}$ in (4.17). Rewrite $\mathcal{E}$ and $\mathcal{F}$ according to (4.13) and (4.16),

$$\begin{aligned}\mathcal{E} &= \left(\left(B^{-1}\,\Phi\right)\otimes_s\left(B\,\Phi\right)\right)\left(\bar{H}\otimes_s\bar{H}\right),\\ \mathcal{F} &= \left(B\,\left(\widehat{\Sigma}+E\right)^T\,\widetilde{R}\right)\otimes_s\left(B^{-1}\,\left(\widehat{\Sigma}+E\right)^{-1}\widetilde{R}\right)\\ &= \left(\left(B\,\widehat{\Sigma}\right)\left(I+E\,\widehat{\Sigma}^{-1}\right)^T\,\widetilde{R}\right)\otimes_s\left(\left(B\,\widehat{\Sigma}\right)^{-1}\left(I+E\,\widehat{\Sigma}^{-1}\right)^{-1}\,\widetilde{R}\right).\end{aligned}$$

Since the matrix $I+E\,\widehat{\Sigma}^{-1}$ is in general dense and has 2-norm $\Omega(1)$ (see (4.16) and Assumption 4.1), we can choose the diagonal scaling matrices for $\mathcal{E}$ and $\mathcal{F}$ to be

$$\mathcal{S}_{\mathcal{E}} \overset{\text{def}}{=} \left(B^{-1}\,\Phi\right)\otimes_s\left(B\,\Phi\right) \quad\text{and}\quad \mathcal{S}_{\mathcal{F}} \overset{\text{def}}{=} (\phi+\psi)^2\left(B\,\widehat{\Sigma}\right)\otimes_s\left(B\,\widehat{\Sigma}\right)^{-1},$$

respectively, where $\phi=\max_{i=1}^n\phi_i=\Omega(\|\widetilde{H}\|)$ and $\psi=\max_{i=1}^n\psi_i=\Omega(\|\widetilde{R}\|)$. The scaled $\mathcal{E}$ matrix $\mathcal{S}_{\mathcal{E}}^{-1}\,\mathcal{E}=\bar{H}\otimes_s\bar{H}$ is well row-scaled due to (4.13) and has 2-norm $\Omega(1)$. The scaled $\mathcal{F}$ matrix $\mathcal{S}_{\mathcal{F}}^{-1}\,\mathcal{F}$ has 2-norm $O(1)$, but could still be badly row-scaled for some E. We have chosen the factor $(\phi+\psi)^2$ instead of ψ^2 in front of $\mathcal{S}_{\mathcal{F}}$ to make our analysis simpler. To see the diagonal entries of $\mathcal{S}_{\mathcal{E}}$ and $\mathcal{S}_{\mathcal{F}}$ more clearly, we apply $\mathcal{S}_{\mathcal{E}}$ and $\mathcal{S}_{\mathcal{F}}$ to the vector $\mathbf{e}$ in section 1.4:

$$\mathcal{S}_{\mathcal{E}}\,\mathbf{e}=\frac{1}{2}\cdot\mathbf{svec}\left(\left(\frac{\beta_i}{\beta_j}+\frac{\beta_j}{\beta_i}\right)\phi_i\,\phi_j\right) \quad\text{and}\quad \mathcal{S}_{\mathcal{F}}\,\mathbf{e}=\frac{(\phi+\psi)^2}{2}\cdot\mathbf{svec}\left(\frac{\beta_i\,\widehat{\sigma}_i}{\beta_j\,\widehat{\sigma}_j}+\frac{\beta_j\,\widehat{\sigma}_j}{\beta_i\,\widehat{\sigma}_i}\right).$$

Comparing with (4.14), which is the complementarity equation in exact arithmetic, we see that the scaling factors for $\mathcal{E}$ in finite precision arithmetic are similar to those for $\mathcal{E}$ in exact arithmetic. On the other hand, some of the ψ_i's can be much smaller than $\phi+\psi$, so the scaling factors for $\mathcal{F}$ in finite precision arithmetic can be drastically larger than those for $\mathcal{F}$ in exact arithmetic, potentially causing $\mathcal{J}_{\mathcal{S}}$ to be ill-conditioned even when $\mathcal{J}$ with the choice $P=I$ is well-conditioned. This ill-conditioning of $\mathcal{J}_{\mathcal{S}}$ is largely caused by some of the ψ_i's becoming very small near the optimal solution. The HKM search directions and the NT direction all share this problem (see section 5). On the other hand, we observed from numerical experiments that this ill-conditioning problem does *not* become worse even when one makes $\mathcal{J}$ arbitarily ill-conditioned with very bad choices of B.

We now rescale $\mathcal{J}$ in (4.17) with $\mathcal{S}_{\mathcal{E}}$ and $\mathcal{S}_{\mathcal{F}}$ to get

$$\mathcal{J}_{\mathcal{S}}\,d\mathcal{X}=\mathcal{R}_{\mathcal{S}},\ \text{where}\ \mathcal{J}_{\mathcal{S}}=\begin{pmatrix}\mathcal{S}^{-1}\,\mathcal{E} & \mathcal{S}^{-1}\,\mathcal{F} & 0\\ 0 & \mathcal{I} & \mathcal{A}^T\\ \mathcal{A} & 0 & 0\end{pmatrix}\ \text{and}\ \mathcal{R}_{\mathcal{S}}=\begin{pmatrix}\mathcal{S}^{-1}\,r_c\\ r_d\\ r_p\end{pmatrix},$$

with $\mathcal{S}=\mathcal{S}_{\mathcal{E}}+\mathcal{S}_{\mathcal{F}}$. Let $\widehat{\mathcal{R}}_{\mathcal{S}}$ be the vector $\widehat{\mathcal{R}}$ with $\widehat{r}_c$ replaced by $\mathcal{S}^{-1}\,\widehat{r}_c$. Scale (3.11) to get

$$(4.18)\qquad \left(\mathcal{J}_{\mathcal{S}}+\delta\mathcal{J}_{\mathcal{S}}\right)\,\widehat{d\mathcal{X}}=\widehat{\mathcal{R}}_{\mathcal{S}}.$$

We point out that $\mathcal{S}$ is introduced as part of our error analysis to reveal the effective condition number for the linear system of equations (4.17), and is *not* part of Algorithm 4.1. Backward error bounds of the form (3.11) would be too pessimistic since $\kappa(\mathcal{J})$ can diverge to ∞ very quickly.

We now consider round-off errors in $\mathcal{R}$. Although numerically $\mathcal{R}$ is evaluated at the point (X,Z,y), instead of $(X^\dagger,Z^\dagger,y)$, the difference between them is minor. Equation (3.10) still holds for r_d and r_p:

$$(4.19)\quad \hat{r}_d=r_d+O\left(\epsilon\cdot\|Z\|+\epsilon\cdot\|\mathcal{A}\|\,\|y\|\right)\quad\text{and}\quad \hat{r}_p=r_p+O\left(\epsilon\cdot\|\mathcal{A}\|\,\|X\|\right).$$

Since the round-off errors in $\widehat{r}_c$ are more complicated, we summarize the results here and leave the analysis to the appendix.

LEMMA 4.2. *In Algorithm* 4.1, *we have*

$$\left\|\mathcal{S}^{-1}\ (\widehat{r}_c - r_c)\right\| = O\left(\epsilon\|\widetilde{R}\|\,\|\widetilde{H}\|\right)$$

for the HKM direction $P^T\,P = Z$ and the NT direction, and

$$\left\|\mathcal{S}^{-1}\ (\widehat{r}_c - r_c)\right\| = O\left(\epsilon\kappa\left(\widehat{\Sigma}\right)\ \|\widetilde{R}\|\,\|\widetilde{H}\|\right)$$

in general. In all cases,

$$\|\widehat{\mathcal{R}}_{\mathcal{S}} - \mathcal{R}_{\mathcal{S}}\| = \quad O\left(\epsilon\cdot\kappa\left(\widehat{\Sigma}\right)\ \|\mathcal{X}\|\right). \tag{4.20}$$

The factor $\kappa(\widehat{\Sigma})$ disappears for the HKM direction $P^T\,P = Z$ and the NT direction. Since Algorithm 4.1 usually generates iterates that are not far away from the central path, the factor $\kappa(\widehat{\Sigma})$ is in general not very large in practice.

We now analyze the round-off errors in computing the right-hand sides of (2.5). To this end, define

$$\mathcal{E}^\dagger = \left(B\,\widehat{\Sigma}\,\widetilde{R}^{-T}\right)\otimes_s\left(B^{-1}\,\widehat{\Sigma}\,\widetilde{R}^{-T}\right), \qquad \mathcal{F}^\dagger = \left(B\,\widehat{\Sigma}\,\widetilde{R}\right)\otimes_s\left(B^{-1}\,\widehat{\Sigma}^{-1}\,\widetilde{R}\right), \tag{4.21}$$

and $\mathcal{M}^\dagger = \mathcal{A}\ \left(\mathcal{E}^\dagger\right)^{-1}\ \mathcal{F}^\dagger\,\mathcal{A}^T$.

Although $\mathcal{E}^\dagger = \mathcal{E}$, $\mathcal{F}^\dagger = \mathcal{F}$, and $\mathcal{M} = \mathcal{M}^\dagger$ in exact arithmetic, these relations do not in general hold in finite arithmetic. Let

$$\widehat{D}_{\mathcal{F}} = \left(\frac{\beta_i^2\ \widehat{\sigma}_i^2 + \beta_j^2\ \widehat{\sigma}_j^2}{\beta_i\ \beta_j\ \widehat{\sigma}_i\ \widehat{\sigma}_j}\right), \quad \widehat{D}_{\mathcal{E}} = \left(\frac{\beta_i\ \beta_j}{\widehat{\sigma}_i\ \widehat{\sigma}_j\ \left(\beta_i^2 + \beta_j^2\right)}\right), \quad \text{and} \quad \widehat{D}_{\mathcal{M}} = \widehat{D}_{\mathcal{F}} \odot \widehat{D}_{\mathcal{E}},$$

and define diagonal matrices $\widehat{\mathcal{D}}_{\mathcal{E}}$, $\widehat{\mathcal{D}}_{\mathcal{F}}$, and $\widehat{\mathcal{D}}_{\mathcal{M}}$ such that

$$\widehat{\mathcal{D}}_{\mathcal{F}}\,\mathbf{e} = \mathbf{svec}\left(\widehat{D}_{\mathcal{F}}\right), \quad \widehat{\mathcal{D}}_{\mathcal{E}}\,\mathbf{e} = \mathbf{svec}\left(\widehat{D}_{\mathcal{E}}\right), \quad \text{and} \quad \widehat{\mathcal{D}}_{\mathcal{M}}\,\mathbf{e} = \mathbf{svec}\left(\widehat{D}_{\mathcal{M}}\right).$$

LEMMA 4.3. *There exist perturbation matrices*

$$\delta_1\mathcal{A} = O(\epsilon\|\mathcal{A}\|)\,,\ |\Theta_1| \le O(\epsilon)\left(\left|\widetilde{R}^T\right|\otimes_s\left|\widetilde{R}^T\right|\right)\widehat{\mathcal{D}}_{\mathcal{E}},\ \left|\delta_1\mathcal{F}^\dagger\right| \le O(\epsilon)\widehat{\mathcal{D}}_{\mathcal{F}}\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right)$$

and diagonal perturbation matrix $\Delta_1 = O(\epsilon)$, all of appropriate dimensions, such that

$$\begin{aligned}&\mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\ \left(\mathcal{E}^\dagger\right)^{-1}\ \left(\mathcal{F}^\dagger\ \widehat{r}_d - \widehat{r}_c\right)\right)\\ &\qquad = (\mathcal{I} + \Delta_1)\ \left(\widehat{r}_p + (\mathcal{A} + \delta_1\mathcal{A})\ \left(\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_1\right)\ \left(\left(\mathcal{F}^\dagger + \delta_1\mathcal{F}^\dagger\right)\ \widehat{r}_d - \widehat{r}_c\right)\right).\end{aligned} \tag{4.22}$$

Proof. The matrix-vector product $\mathbf{fl}\left(\mathcal{F}^\dagger\,r_d\right)$ has the form in Algorithm 4.2 with $\mathcal{D} = \widehat{\mathcal{D}}_{\mathcal{F}}$ and $U = \widetilde{R}$. According to Lemma 4.1, the round-off errors satisfy

$$\mathbf{fl}\left(\mathcal{F}^\dagger\ \widehat{r}_d\right) = \left(\mathcal{F}^\dagger + \delta_1\mathcal{F}^\dagger\right)\ \widehat{r}_d, \quad \text{where} \quad \left|\delta_1\mathcal{F}^\dagger\right| \le O(\epsilon)\cdot\widehat{\mathcal{D}}_{\mathcal{F}}\cdot\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right).$$

As in section 3.3, there exists an $n(n+1)/2$ by $n(n+1)/2$ diagonal perturbation matrix $\Delta_0 = O(\epsilon)$ such that

$$\mathbf{fl}\left(\mathcal{F}^\dagger\,\widehat{r}_d - \widehat{r}_c\right) = (\mathcal{I} + \Delta_0)\ \left(\left(\mathcal{F}^\dagger + \delta_1\mathcal{F}^\dagger\right)\ \widehat{r}_d - \widehat{r}_c\right).$$

The application of $\left(\mathcal{E}^\dagger\right)^{-1}$ to $\mathcal{F}^\dagger\,\widehat{r}_d - \widehat{r}_c$ in (4.8) can be performed by applying $(\widetilde{R}^T \otimes_s \widetilde{R}^T)\,\widehat{\mathcal{D}}_{\mathcal{E}}$ to $\mathbf{fl}\left(\mathcal{F}^\dagger\,\widehat{r}_d - \widehat{r}_c\right)$. Similar to Lemma 4.1, the round-off errors satisfy

$$\begin{aligned}\mathbf{fl}\left(\left(\mathcal{E}^\dagger\right)^{-1}\left(\mathcal{F}^\dagger\widehat{r}_d - \widehat{r}_c\right)\right) &= \left(\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_0\right)\mathbf{fl}\left(\mathcal{F}^\dagger\widehat{r}_d - \widehat{r}_c\right), \\ &\qquad |\Theta_0| \le O(\epsilon)\left(\left|\widetilde{R}^T\right| \otimes_s \left|\widetilde{R}^T\right|\right)\widehat{\mathcal{D}}_{\mathcal{E}}, \\ &= \left(\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_1\right)\ \left(\left(\mathcal{F}^\dagger + \delta_1\mathcal{F}^\dagger\right)\ \widehat{r}_d - \widehat{r}_c\right),\end{aligned}$$

where $\Theta_1 \overset{\text{def}}{=} (\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_0)(\mathcal{I} + \Delta_0) - \left(\mathcal{E}^\dagger\right)^{-1}$ satisfies $|\Theta_1| \le O(\epsilon)(|\widetilde{R}^T| \otimes_s |\widetilde{R}^T|)\widehat{\mathcal{D}}_{\mathcal{E}}$. With these relations, we can write

$$\mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\left(\mathcal{E}^\dagger\right)^{-1}\left(\mathcal{F}^\dagger\,\widehat{r}_d - \widehat{r}_c\right)\right) = (\mathcal{I} + \Delta_1)\left(\widehat{r}_p + (\mathcal{A} + \delta_1\mathcal{A})\,\mathbf{fl}\left(\left(\mathcal{E}^\dagger\right)^{-1}\left(\mathcal{F}^\dagger\,\widehat{r}_d - \widehat{r}_c\right)\right)\right),$$

which is (4.22). □

The round-off errors in solving (2.5c) are analyzed by Lemma 4.4 below. We leave its proof to the appendix.

LEMMA 4.4. *The round-off errors in solving* (2.5c) *satisfy*

$$\left(\mathcal{M}^\dagger + \delta\mathcal{M}^\dagger\right)\ \widehat{dy} = \mathbf{fl}\left(\widehat{r}_p + \mathcal{A}\,\mathcal{E}^{-1}\ \left(\mathcal{F}\,\widehat{r}_d - \widehat{r}_c\right)\right),$$

where $\delta\mathcal{M}^\dagger = O(\epsilon\cdot\|\mathcal{A}\|^2\,\|X^\dagger\|\,\|\left(Z^\dagger\right)^{-1}\|)$.

A remarkable feature of Lemma 4.4 is that the upper bound on $\delta\mathcal{M}^\dagger$ does not depend on B. Hence the Schur complement equation is solved to the same accuracy no matter how badly the complementarity equation (2.1c) is scaled (see (4.14)). As in section 3.3, the round-off errors in (2.5b) satisfy

$$\widehat{dZ} = \mathbf{smat}\left((\mathcal{I} + \Delta_3)\ \left(\widehat{r}_d - (\mathcal{A} + \delta_2\mathcal{A})^T\ \widehat{dy}\right)\right),$$

where $\delta_1\mathcal{A} = O(\epsilon\|\mathcal{A}\|)$ and $\Delta_2 = O(\epsilon)$ are perturbation matrices, with Δ_2 being diagonal.

Now we consider the round-off errors in solving (2.5c) using (4.15). Similar to (4.22),

$$\mathbf{fl}\left(\widehat{r}_c - \mathcal{F}^\dagger\,\mathbf{svec}\left(\widehat{dZ}\right)\right) = (\mathcal{I} + \Delta_4)\ \left(\widehat{r}_c - \left(\mathcal{F}^\dagger + \delta_2\mathcal{F}^\dagger\right)\ \mathbf{svec}\left(\widehat{dZ}\right)\right),$$

where $\Delta_4 = O(\epsilon) \in \mathbf{R}^{m\times m}$ is diagonal and $|\delta_2\mathcal{F}^\dagger| \le O(\epsilon)\cdot\widehat{\mathcal{D}}_{\mathcal{F}}\cdot(|\widetilde{R}| \otimes_s |\widetilde{R}|)$. By Assumption 4.2 and Lemma 4.1, we write the round-off errors in the solution of (2.5c) as

$$\left(B^{-1}\,\Phi \otimes_s B\,\Phi\right)\ \left(\bar{H} \otimes_s \bar{H} + \Theta_2\right)\ \widehat{dX} = (\mathcal{I} + \Delta_4)\ \left(\widehat{r}_c - \left(\mathcal{F}^\dagger + \delta_2\mathcal{F}^\dagger\right)\ \mathbf{svec}\left(\widehat{dZ}\right)\right),$$

where $\|\Theta_2\| = O\left(\epsilon\|\bar{H}\|^2\right) = O(\epsilon)$. Since Δ_4 is a diagonal matrix, this last equation becomes

$$(\mathcal{E} + \delta\mathcal{E})\,\widehat{dX} = \widehat{r}_c - \left(\mathcal{F}^\dagger + \delta_2\mathcal{F}^\dagger\right)\mathbf{svec}\left(\widehat{dZ}\right), \tag{4.23}$$

where

$$\delta\mathcal{E} \overset{\text{def}}{=} (\mathcal{I}+\Delta_4)^{-1}\left(B^{-1}\Phi\otimes_s B\Phi\right)\left(\bar{H}\otimes_s\bar{H}+\Theta_2\right)-\mathcal{E}=\left(B^{-1}\,\Phi\otimes_s B\,\Phi\right)\Theta_3,$$

with

$$\Theta_3 \overset{\text{def}}{=} (\mathcal{I}+\Delta_4)^{-1}\left(\bar{H}\otimes_s\bar{H}+\Theta_2\right)-\bar{H}\otimes_s\bar{H}=O(\epsilon).$$

Putting Lemma 4.4 and all these relations together, we get an equation similar to (2.4):

$$\begin{pmatrix} \mathcal{E}+\delta\mathcal{E} & \mathcal{F}^\dagger+\delta_2\mathcal{F}^\dagger & 0 \\ 0 & (\mathcal{I}+\Delta_3)^{-1} & (\mathcal{A}+\delta_2\mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger+\delta\mathcal{M}^\dagger \end{pmatrix}\widehat{d\mathcal{X}}=\begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \mathbf{fl}\left(\widehat{r}_p+\mathcal{A}\left(\mathcal{E}^\dagger\right)^{-1}\left(\mathcal{F}^\dagger\,\widehat{r}_d-\widehat{r}_c\right)\right) \end{pmatrix},$$

Combining this with (4.22), we obtain an equation similar to (2.3) and (3.16):

$$\begin{pmatrix} \mathcal{I} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{L}_{3,1} & \mathcal{L}_{3,2} & (\mathcal{I}+\Delta_2)^{-1} \end{pmatrix}\begin{pmatrix} \mathcal{E}+\delta\mathcal{E} & \mathcal{F}^\dagger+\delta_2\mathcal{F}^\dagger & 0 \\ 0 & (\mathcal{I}+\Delta_3)^{-1} & (\mathcal{A}+\delta_2\mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger+\delta\mathcal{M}^\dagger \end{pmatrix}\widehat{d\mathcal{X}}=\begin{pmatrix} \widehat{r}_c \\ \widehat{r}_d \\ \widehat{r}_p \end{pmatrix},$$

where $\mathcal{L}_{3,1}=(\mathcal{A}+\delta_1\mathcal{A})(\left(\mathcal{E}^\dagger\right)^{-1}+\Theta_1)$ and $\mathcal{L}_{3,2}=-\mathcal{L}_{3,1}(\mathcal{F}^\dagger+\delta_1\mathcal{F}^\dagger)$.

Scaling the first row by $\mathcal{S}^{-1}$, we arrive at (4.18) with the backward error matrix $\delta\mathcal{J}_\mathcal{S}$ satisfying

$$\begin{aligned} \delta\mathcal{J}_\mathcal{S} &= \begin{pmatrix} \mathcal{S}^{-1} & 0 & 0 \\ 0 & \mathcal{I} & 0 \\ \mathcal{L}_{3,1} & \mathcal{L}_{3,2} & (\mathcal{I}+\Delta_2)^{-1} \end{pmatrix}\begin{pmatrix} \mathcal{E}+\delta\mathcal{E} & \mathcal{F}^\dagger+\delta_2\mathcal{F}^\dagger & 0 \\ 0 & (\mathcal{I}+\Delta_3)^{-1} & (\mathcal{A}+\delta_2\mathcal{A})^T \\ 0 & 0 & \mathcal{M}^\dagger+\delta\mathcal{M}^\dagger \end{pmatrix}-\mathcal{J}_\mathcal{S} \\ &= \begin{pmatrix} \mathcal{S}^{-1}\,\delta\mathcal{E} & \mathcal{S}^{-1}\left(\mathcal{F}^\dagger-\mathcal{F}+\delta_2\mathcal{F}^\dagger\right) & 0 \\ 0 & (\mathcal{I}+\Delta_3)^{-1}-\mathcal{I} & \delta_2\mathcal{A}^T \\ \mathcal{L}_{3,1}\,\mathcal{E}-\mathcal{A} & \mathcal{L}_{3,1}\left(\mathcal{F}^\dagger-\mathcal{F}^\dagger\,(\mathcal{I}+\Delta_3)^{-1}\right) & (\mathcal{I}+\Delta_2)^{-1}\,\mathcal{M}^\dagger+\mathcal{L}_{3,2}\,\mathcal{A}^T \end{pmatrix} \\ &+ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \mathcal{L}_{3,1}\delta\mathcal{E} & \mathcal{L}_{3,1}\left(\delta_2\mathcal{F}^\dagger-\delta_1\mathcal{F}^\dagger\,(\mathcal{I}+\Delta_3)^{-1}\right) & (\mathcal{I}+\Delta_2)^{-1}\,\delta\mathcal{M}^\dagger+\mathcal{L}_{3,2}\delta_2\mathcal{A}^T \end{pmatrix}. \end{aligned} \tag{4.24}$$

Theorem 4.5 below is the main result of this section. We leave its proof to the appendix.

THEOREM 4.5. *Define* $\rho=(\|X^\dagger\|+\|Z^\dagger\|)(\|\left(X^\dagger\right)^{-1}\|+\|\left(Z^\dagger\right)^{-1}\|)$. *Then the numerical solution* $\widehat{d\mathcal{X}}$ *computed via Algorithm* 4.1 *satisfies* (4.18) *with*

$$\delta\mathcal{J}_\mathcal{S}=\begin{cases} O\left(\epsilon\rho\right) & \textit{for HKM direction } P^T\,P=Z \textit{ and NT direction;} \\ O\left(\epsilon\sqrt{\rho}\,\left(\kappa\left(\widehat{\Sigma}\right)+\sqrt{\rho}\right)\right) & \textit{in general.} \end{cases}$$

The round-off errors on the right-hand side of (4.18) *satisfy Lemma* 4.2.

As we argued after Lemma 4.2, the factor $\kappa(\widehat{\Sigma})$ is usually not very large in practice. For the sake of argument in what follows, we assume that it is less than $\sqrt{\rho}$. Now the bound in Theorem 4.5 looks like (3.12). With arguments similar to those in section 3.3, we conclude that Algorithm 4.1 could stop making further progress as soon as it reached an iterate (X, Z, y) that satisfies

$$\frac{\min\left(\lambda_{\min}(Z), \lambda_{\min}(X)\right)}{\max\left(\|Z\|, \|X\|\right)} = O\left(\epsilon\kappa\left(\mathcal{J}_{\mathcal{S}}\right)\right), \tag{4.25}$$

and Algorithm 4.1 could be numerically unstable if $\mathcal{J}_{\mathcal{S}}$ were ill-conditioned. As with the AHO method, by repeating the arguments in section 4.4, it is easy to see that the PC rule applied to Algorithm 4.1 could also be numerically unstable if $\mathcal{J}_{\mathcal{S}}$ were ill-conditioned.

If $\kappa(\widehat{\Sigma}) \gg 1$, then the error bound in (4.20) on the scaled right-hand side of (2.2) will be large. We can eliminate the factor $\kappa(\widehat{\Sigma})$ in the error bound by choosing a scaling matrix $\mathcal{S}$ with larger diagonal entries, thereby making $\mathcal{J}_{\mathcal{S}}$ potentially worse scaled and therefore worse conditioned.

At first sight, (4.25) seems to suggest that the TTT methods could be as accurate as the AHO method. However, our numerical results in section 5.3 show that the matrix $\mathcal{J}_{\mathcal{S}}$ for the HKM methods and NT method is in general much worse conditioned than the matrix $\mathcal{J}$ for the AHO method, indicating that these methods are in general *less* accurate. In section 5.1 we discuss a choice of B that appears to make $\mathcal{J}_{\mathcal{S}}$ better conditioned than other choices.

The above analysis was on Algorithm 4.1 only. Since the NT method [26, 27] as implemented in [29] is not identical to Algorithm 4.1, our results do not directly apply to it. However, the difference between these variations does not appear to be fundamental. It is very likely that the NT method in [26, 27, 29] suffers from the same numerical instability problems Algorithm 4.1 faces. The same argument holds for the HKM direction $P^T P = X^{-1}$.

While the HKM direction $B = I$ can be computed without the SVD, the matrix Z^{-1} is still needed in the formation of $\mathcal{M}$ (see section 4.2). Hence we would expect the upper bound on the round-off errors in solving (2.5c) for this direction to be at least comparable with that in Lemma 4.4. Consequently, we could expect this variation to be numerically unstable if $\mathcal{J}_{\mathcal{S}}$ is ill-conditioned.

Finally, we note that unlike the AHO method, the potential numerical instability of Algorithm 4.1 in general remains even if the search direction is computed by solving (2.2) as a dense linear system of equations (see section 5.3).

4.5. Error analysis for the LS variation of the TTT methods.

Now we discuss the round-off errors for the LS variation of the TTT methods discussed in section 4.2. In addition to Assumptions 3.1–4.1, we further assume the following.

Assumption 4.3. Problem (4.11) is solved via a backward stable method.

As in section 4.1, we will think of the search direction defined by (2.2) as a direction defined at the point $(X^\dagger, Z^\dagger, y)$ in (4.16), instead of the point (X, Z, y). Hence the search direction satisfies (4.17).

Let $\mathcal{R}$ be computed as before and let $\widehat{X}_r$ be the computed version of X_r in (4.10). Define $\mathcal{E}^\dagger$, $\mathcal{F}^\dagger$, and $\mathcal{M}^\dagger$ as in (4.21) and let the coefficient matrix and the right-hand side vector of the LS problem (4.11) be computed as $\mathbf{fl}(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}})$ and $\mathbf{fl}(\mathbf{svec}(\widehat{X}_r) + \widetilde{\mathcal{D}}\,(\mathcal{F}\, r_d - r_c))$, respectively. With analysis similar to that in (4.22), we write

$$\begin{aligned} &\mathbf{fl}\left(\mathbf{svec}(\widehat{X}_r) + \widetilde{\mathcal{D}}\ (\mathcal{F}\, r_d - r_c)\right) \\ &\qquad = (\mathcal{I} + \Delta_2)\ \left(\mathbf{svec}(\widehat{X}_r) + (\mathcal{I} + \Delta_1)\ \widetilde{\mathcal{D}}\ \left(\left(\mathcal{F}^\dagger + \delta_1\mathcal{F}^\dagger\right)\ \widehat{r}_d - \widehat{r}_c\right)\right), \end{aligned} \tag{4.26}$$

where Δ_1 and Δ_2 are diagonal perturbation matrices and $\delta_1\mathcal{F}^\dagger$ is a perturbation to $\mathcal{F}^\dagger$. Furthermore, with an analysis similar to that in the proof of Lemma 4.4, we can write

$$\text{(4.27)}\quad \mathbf{fl}\left(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right) = \widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}} + \delta\mathcal{M}^{\frac{1}{2}}, \quad \text{where} \quad \delta\mathcal{M}^{\frac{1}{2}} = O\left(\|\mathcal{A}\|\ \|\widetilde{R}\|\ \|\widetilde{H}^{-1}\|\right).$$

By standard error analysis (see Higham [16, Chap. 19]), the computed solution $\widehat{dy}$ is the *exact* solution to a slightly perturbed LS problem

$$\min_{dy}\left\|\left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)^T dy - \mathbf{fl}\left(\mathbf{svec}(\widehat{X}_r)+\widetilde{\mathcal{D}}\ (\mathcal{F}\,r_d - r_c)\right)\right\|,$$

where the m by $n(n+1)/2$ matrix $\Theta = O(\epsilon\cdot\|\mathbf{fl}(\widetilde{\mathcal{A}}\ \mathcal{D}_{\mathcal{M}}^{\frac{1}{2}})\|)$ is a perturbation to $\mathbf{fl}(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}})$. Stating this result in an equivalent way, $\widehat{dy}$ is the *exact* solution to the normal equation of this perturbed LS problem:

$$\begin{aligned}&\left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)\ \left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)^T \widehat{dy}\\ &= \left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)\cdot\mathbf{fl}\left(\mathbf{svec}(\widehat{X}_r)+\widetilde{\mathcal{D}}\ (\mathcal{F}\,r_d - r_c)\right).\end{aligned}$$

In light of (4.27), this equation can be rewritten in the form of Lemma 4.4 as

$$\begin{aligned}\left(\mathcal{M}^\dagger+\delta\mathcal{M}^\dagger\right)\ \widehat{dy} &= \left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)\mathbf{fl}\left(\mathbf{svec}(\widehat{X}_r)+\widetilde{\mathcal{D}}\ (\mathcal{F}\,r_d - r_c)\right),\\ \text{where}\quad \delta\mathcal{M}^\dagger &\overset{\text{def}}{=} \left(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}+\delta\mathcal{M}^{\frac{1}{2}}+\Theta\right)\ \left(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}+\delta\mathcal{M}^{\frac{1}{2}}+\Theta\right)^T-\mathcal{M}^\dagger\\ &= \widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\ \left(\delta\mathcal{M}^{\frac{1}{2}}+\Theta\right)^T+\left(\delta\mathcal{M}^{\frac{1}{2}}+\Theta\right)\ \left(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}+\delta\mathcal{M}^{\frac{1}{2}}+\Theta\right)^T\\ &= O\left(\epsilon\,\|\mathcal{A}\|^2\ \|X^\dagger\|\ \left\|\left(Z^\dagger\right)^{-1}\right\|\right).\end{aligned}$$

Comparing with Lemma 4.4, the error bounds for $\delta\mathcal{M}^\dagger$ in both cases are identical. Although in the LS approach $\delta\mathcal{M}^\dagger$ has a special form, it does not seem to make $\|\delta\mathcal{M}^\dagger\|$ smaller.

Assume that (2.5b) and (2.5c) in the least squares approach are solved as in Algorithm 4.1. We get an equation similar to (2.4),

$$\begin{pmatrix}\mathcal{E}+\delta\mathcal{E} & \mathcal{F}^\dagger+\delta_2\mathcal{F}^\dagger & 0\\ 0 & (\mathcal{I}+\Delta_3)^{-1} & (\mathcal{A}+\delta_2\mathcal{A})^T\\ 0 & 0 & \mathcal{M}^\dagger+\delta\mathcal{M}^\dagger\end{pmatrix}\widehat{d\mathcal{X}} = \begin{pmatrix}\widehat{r}_c\\ \widehat{r}_d\\ \mathbf{fl}\left(\mathbf{svec}(\widehat{X}_r)+\widetilde{\mathcal{D}}\ (\mathcal{F}\,r_d - r_c)\right)\end{pmatrix},$$

which can be combined with (4.26) to give an equation similar to (2.3):

$$\begin{pmatrix}\mathcal{I} & 0 & 0\\ 0 & \mathcal{I} & 0\\ \mathcal{L}_{3,1} & \mathcal{L}_{3,2} & \mathcal{I}\end{pmatrix}\begin{pmatrix}\mathcal{E}+\delta\mathcal{E} & \mathcal{F}^\dagger+\delta_2\mathcal{F}^\dagger & 0\\ 0 & (\mathcal{I}+\Delta_3)^{-1} & (\mathcal{A}+\delta_2\mathcal{A})^T\\ 0 & 0 & \mathcal{M}^\dagger+\delta\mathcal{M}^\dagger\end{pmatrix}\widehat{d\mathcal{X}} = \begin{pmatrix}\widehat{r}_c\\ \widehat{r}_d\\ \widetilde{r}_p\end{pmatrix},$$

where

$$\mathcal{L}_{3,1} = \left(\mathbf{fl}\left(\widetilde{\mathcal{A}}\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}\right)+\Theta\right)(\mathcal{I}+\Delta_2)\,(\mathcal{I}+\Delta_1)\,\widetilde{\mathcal{D}},\quad \mathcal{L}_{3,2} = -\mathcal{L}_{3,1}\left(\mathcal{F}^\dagger+\delta_1\mathcal{F}^\dagger\right),$$

and $\widetilde{r}_p = (\mathbf{fl}(\widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}^{\frac{1}{2}}) + \Theta)\,(\mathcal{I} + \Delta_2)\;\mathbf{svec}(\widehat{X}_r)$. As in section 4.1, we can now scale the above equation by $\mathcal{S}$ and bound the round-off errors in both $\mathcal{J}_{\mathcal{S}}$ and $\mathcal{R}_{\mathcal{S}}$. Since the upper bounds for $\delta\mathcal{M}^{\dagger}$ are the same in both cases, the upper bound on the backward errors in the $(3,3)$ block of $\mathcal{J}_{\mathcal{S}}$ for the LS problem will be about the same as that of $\mathcal{J}$ for Algorithm 4.1, regardless of how small the backward errors in other blocks of $\mathcal{J}_{\mathcal{S}}$ might be. The upper bound for $\delta\mathcal{M}^{\dagger}$ is roughly the upper bound in Theorem 4.5 if we assume that $\kappa(\widehat{\Sigma}) = O(\sqrt{\rho})$ and that $\|\left(X^{\dagger}\right)^{-1}\| = \Omega(\|\left(Z^{\dagger}\right)^{-1}\|)$. Hence it appears that the LS approach in general is no more accurate than Algorithm 4.1.

5. Numerical experiments.

In this section we first discuss a new choice of search direction in the TTT family. We then discuss how to measure the amount of accuracy in a numerical solution to problem (1.1). Finally, we present results from our numerical experiments that support our analysis for the AHO method and the TTT methods.

5.1. A new search direction in the TTT family.

Our error analysis of the TTT methods indicates that one factor that potentially limits the amount of accuracy in the numerical solution is the scaled condition number $\kappa(\mathcal{J}_{\mathcal{S}})$ (see (4.25)). To achieve maximum accuracy in the numerical solution, we would like to find a direction in the TTT family that minimizes $\kappa(\mathcal{J}_{\mathcal{S}})$.

However, such a direction appears to be very hard to find. Instead, we note that the source of potential bad scaling in (4.14) is the ill-conditioning of the matrix P in (2.10a). This motivates us to choose a direction in the TTT family that minimizes $\kappa(P)$. As in section 2.2, write $B = \mathrm{diag}(B_1, \ldots, B_k)$, where the dimension of B_j is the multiplicity of the singular value σ_j of $R\,H^T$. Partition the matrix $\widetilde{H}$ in (2.10a) accordingly as $\widetilde{H} = (\widetilde{H}_1, \ldots, \widetilde{H}_k)^T$. The following result of Demmel suggests a particular choice of B that is at most a factor of $\sqrt{k}$ away from optimal.

Lemma 5.1 (see Demmel [9]). *Define $\bar{B} = \mathrm{diag}\left(\bar{B}_1, \ldots, \bar{B}_k\right)$, where $\bar{B}_j$ is chosen so that $\bar{B}_j\,\widetilde{H}_j^T$ is row orthonormal, i.e., $\bar{B}_j\,\widetilde{H}_j^T\,\widetilde{H}_j\,\bar{B}_j^T = I_j$ for $j = 1, \ldots, k$. Then*

$$\kappa\left(\bar{B}\,\widetilde{H}\right) \le \sqrt{k}\;\min\left\{\kappa\left(B\,\widetilde{H}\right) \mid \quad \textit{where} \quad B = \mathrm{diag}(B_1, \ldots, B_k)\right\}.$$

We compared the TTT method with $B = \bar{B}$ to the AHO method and other TTT methods in our numerical experiments. In our implementation, we ignored the possibility of multiple singular values in $R\,H^T$ and instead scaled $\widetilde{H}$ as in (4.13) and chose $\bar{B} = \Phi^{-1}$. This choice of $\bar{B}$ corresponds to the matrix $\bar{B}$ in Lemma 5.1 with $k = n$.

5.2. Measuring accuracy in a numerical solution.

Some recent numerical studies of interior-point methods on SDPs measured the amount of accuracy in a numerical solution by computing $\|r_p\|$, $\|r_d\|$ and $\mathbf{tr}(X\,Z)$, the *duality gap*. However, since the matrix $X\,Z$ need not be symmetric, a small duality gap does not necessarily imply a small $\|X\,Z\|$. In this paper, we measure the accuracy in (X, Z, y) by computing the *residual*

$$\widetilde{\mathcal{R}} = \begin{pmatrix} \mathbf{svec}\left(X\;Z + Z\;X\right)/2 \\ \mathbf{svec}\left(Z + \mathbf{smat}\left(\mathcal{A}^T\,y\right) - C\right) \\ b - \mathcal{A}\,\mathbf{svec}(X) \end{pmatrix}.$$

To relate $\widetilde{\mathcal{R}}$ to the amount of accuracy in (X, Z, y), we note that $X^*\,Z^* = 0$ and hence

$$\begin{aligned} X\,Z + Z\,X &= X\,Z + Z\,X - X^*\,Z^* - Z^*\,X^* \\ &= (X - X^*)\;Z + X^*\;(Z - Z^*) + (Z - Z^*)\;X + Z^*\;(X - X^*)\,. \end{aligned}$$

Since the left-hand side is symmetric, we symmetrize the right-hand side to get

$$\begin{aligned} &X\,Z + Z\,X \\ &= (X - X^*)\,\frac{Z + Z^*}{2} + \frac{Z + Z^*}{2}\,(X - X^*) + \frac{X + X^*}{2}\,(Z - Z^*) + (Z - Z^*)\,\frac{X + X^*}{2}. \end{aligned}$$

This, and the fact that (X^*, Z^*, y^*) is the exact solution to (1.2), imply

$$(\mathcal{J} + \mathcal{J}^*)\,(\mathcal{X} - \mathcal{X}^*) = 2\,\widetilde{\mathcal{R}},$$

where

$$\mathcal{X} = \begin{pmatrix} \mathbf{svec}(X) \\ \mathbf{svec}(Z) \\ y \end{pmatrix}, \quad \mathcal{X}^* = \begin{pmatrix} \mathbf{svec}(X^*) \\ \mathbf{svec}(Z^*) \\ y^* \end{pmatrix},$$

and

$$\mathcal{J} = \begin{pmatrix} Z \otimes_s I & X \otimes_s I & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ \mathcal{A} & 0 & 0 \end{pmatrix}, \qquad \mathcal{J}^* = \begin{pmatrix} Z^* \otimes_s I & X^* \otimes_s I & 0 \\ 0 & \mathcal{I} & \mathcal{A}^T \\ \mathcal{A} & 0 & 0 \end{pmatrix}.$$

Note that $\mathcal{J}$ is the coefficient matrix in (3.1). Writing this equation in the form of (3.17) and assuming that (X, Z) is sufficiently close to (X^*, Z^*), we get

$$\text{(5.1)} \qquad \frac{\|\mathcal{X} - \mathcal{X}^*\|}{\|\mathcal{X}\|} \le \kappa\,(\mathcal{J} + \mathcal{J}^*)\,\frac{2\,\left\|\widetilde{\mathcal{R}}\right\|}{\|\mathcal{J} + \mathcal{J}^*\|\;\|\mathcal{X}\|} \approx \kappa\,(\mathcal{J}) \cdot \frac{\left\|\widetilde{\mathcal{R}}\right\|}{\|\mathcal{J}\|\;\|\mathcal{X}\|}.$$

We call the ratio after $\kappa\,(\mathcal{J})$ in the last expression the *normalized residual*. We expect a stable numerical method to reduce the normalized residual to the order of machine precision, independent of how big $\kappa\,(\mathcal{J})$ might be. The above equation suggests that the smaller the normalized residual, the more accurate the numerical solution. The quantity $\kappa\,(\mathcal{J})$ appears to play the role of the condition number for the SDP. However, for an SDP with a very large condition number, a small normalized residual does not necessarily imply a small error in the optimal solution.

5.3. Numerical results. We have implemented the AHO method and the TTT methods in MATLAB and have performed a number of numerical experiments. We summarize some of the numerical results below. The computations were done on an Ultra Sparc Station in double precision ($\epsilon \approx 2{\times}10^{-16}$). We tested the following methods:

- the AHO method;
- the NT method by choosing $B = \Sigma^{-\frac{1}{2}}$ in Algorithm 4.1;
- the HKM method with $P^T\,P = Z$, without the SVD, as discussed in section 4.2;
- the method discussed in section 5.1. We will call it the New method.

The NT method in our experiments is not identical to the NT method in [26, 27, 29]. However, as we argued at the end of section 4.4, we expect both variations to suffer from similar numerical instability problems.

For comparison, we also implemented the above four methods by solving the corresponding equation (2.2) with a backward stable dense linear equation solver, with proper rescaling whenever necessary. In all cases, we set the initial guess to be $X = Z = I$ and $y = 0$. We chose $\sigma = 0.25$ and $\tau = 0.98$, and switched to the Mehrotra predictor-corrector versions as soon as[2]

$$\frac{\|r_p\|}{\|\mathcal{A}\| \ \|X\|_F} + \frac{\|r_d\|}{\|Z\|_F + \|\mathcal{A}\| \ \|y\|} \leq 10^{-4}.$$

For any given r, m, and n with $r(r+1)/2 \leq m \leq rn - r(r-1)/2$, we chose the following two types of test problems:

- Type-I SDPs. We generate the following quantities randomly:
  - — an n by n orthogonal matrix Q^*; the m by $n(n+1)/2$ matrix $\mathcal{A} = (\mathbf{svec}(A_1), \ldots, \mathbf{svec}(A_m))^T$, and the m-vector y^*.
  - — positive diagonal matrices Λ_1^* and Λ_2^* with dimensions r by r and $(n-r)$ by $(n-r)$, respectively.

  We then define the SDP by setting

  $$X^* = Q^* \ \mathrm{diag}(\Lambda_1^*, 0) \ (Q^*)^T, \ Z^* = Q^* \ \mathrm{diag}(0, \Lambda_2^*) \ (Q^*)^T, \ b = \mathcal{A}\,\mathbf{svec}\,(X^*)$$

  and $C = Z^* + \mathcal{A}^T y^*$. It is straightforward to verify that (X^*, Z^*, y^*) is a solution to (1.2). Type-I SDPs tend to be well-conditioned.
- Type-II SDPs. We generate the symmetric matrices $A_1, \ldots, A_m$ as

  $$A_k = Q^* \begin{pmatrix} U_k & L_k^T \\ L_k & V_k \end{pmatrix} (Q^*)^T,$$

  where $U_k \in \mathbf{S}^r$ and $V_k \in \mathbf{S}^{n-r}$ are random symmetric matrices, and L_k is an $(n-r)$ by r matrix such that $\|L_k\| \approx 10^{-4} \ll \|A_k\| = \Omega(1)$. The rest of the SDP is generated as in Type I. With the analysis given in Alizadeh, Haeberly, and Overton [5], it can be shown that Type-II SDPs generally are ill-conditioned.

Our analysis in section 4.4 indicates that the amount of accuracy in the numerical solution computed by the TTT methods is related to $\kappa(\mathcal{J}_{\mathcal{S}})$. But our choice of $\mathcal{S}$ suggested in section 4.4 may not be optimal. In the numerical experiments we computed the effective condition number as $\kappa_e(\mathcal{J}) = \kappa(\mathcal{D}\,\mathcal{J})$, where $\mathcal{D}$ is a diagonal matrix chosen so that the rows of $\mathcal{D}\,\mathcal{J}$ all have 2-norm 1. Since $\mathcal{J}$ is an $(m + n(n+1))$ by $(m + n(n+1))$ matrix, by Lemma 5.1, $\kappa_e(\mathcal{J})$ is at most a factor of $\sqrt{m + n(n+1)}$ away from the optimal diagonal scaling. Since we were mainly concerned with numerical stability, we stopped executing a method only when further reduction in the normalized residual did not appear possible. Only the smallest normalized residual and the number of iterations to achieve it are reported.

Tables 5.1–5.4 summarize our results. The effective condition numbers reported in these tables are $\kappa(\mathcal{J})$ for the AHO method and $\kappa_e(\mathcal{J})$ for the others.

Table 5.1 shows that for the Type-I SDPs tested, the AHO method was able to reduce the normalized residual to close to ϵ, and its corresponding $\kappa(\mathcal{J})$ was modest. On the other hand, the NT method could only reduce the normalized residual to about 10^{-10}, and its corresponding $\kappa_e(\mathcal{J})$ was much larger. The HKM and New methods were more accurate than the NT method, but less accurate than the AHO

[2] Here we followed the suggestion from a technical report version of [5].

Table 5.1
*Type-*I *SDPs, with block LU factorization.*

	Nos. of iterations			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	13	19	28	15
$(6, 20, 24)$	12	18	26	16

	Normalized residuals			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	7.2×10^{-16}	1.5×10^{-10}	3.3×10^{-13}	4.8×10^{-14}
$(6, 20, 24)$	5.1×10^{-15}	1.9×10^{-10}	9.9×10^{-12}	2.3×10^{-14}

	Effective condition numbers			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	1.5×10^{2}	3.5×10^{8}	4.5×10^{5}	1.1×10^{5}
$(6, 20, 24)$	5.7×10^{3}	1.7×10^{9}	1.9×10^{8}	3.6×10^{4}

Table 5.2
*Type-*I *SDPs, without block LU factorization.*

	Nos. of iterations			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	14	19	23	18
$(6, 20, 24)$	13	18	20	19

	Normalized residuals			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	9.3×10^{-17}	1.2×10^{-10}	3.2×10^{-11}	5.0×10^{-16}
$(6, 20, 24)$	1.7×10^{-16}	1.9×10^{-10}	4.4×10^{-12}	1.2×10^{-16}

	Effective condition numbers			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	1.5×10^{2}	1.8×10^{8}	4.8×10^{6}	5.1×10^{2}
$(6, 20, 24)$	5.7×10^{3}	4.2×10^{9}	5.3×10^{6}	5.7×10^{3}

method. Among the three TTT methods, the New method had the smallest $\kappa_e(\mathcal{J})$ and took the least number of iterations.

We also solved the problems in Table 5.1 using these four methods by solving (2.2) as a dense linear system. The results are summarized in Table 5.2. Due to the effects of finite precision arithmetic, the iterates generated without block LU factorization are in

Table 5.3
Type-II *SDPs, with block LU factorization.*

	Nos. of iterations			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	11	16	16	15
$(6, 20, 24)$	11	14	18	26

	Normalized residuals			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	2.3×10^{-9}	1.5×10^{-8}	3.6×10^{-8}	4.8×10^{-8}
$(6, 20, 24)$	1.6×10^{-10}	1.3×10^{-8}	1.1×10^{-11}	1.8×10^{-8}

	Effective condition numbers			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	3.6×10^{10}	2.2×10^{12}	7.0×10^{11}	7.8×10^{11}
$(6, 20, 24)$	2.5×10^{11}	8.9×10^{11}	3.2×10^{12}	5.3×10^{11}

Table 5.4
Type-II *SDPs, without block LU factorization.*

	Nos. of iterations			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	19	23	35	24
$(6, 20, 24)$	18	22	21	27

	Normalized residuals			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	6.5×10^{-17}	1.4×10^{-10}	1.6×10^{-12}	3.7×10^{-16}
$(6, 20, 24)$	1.1×10^{-16}	3.3×10^{-11}	9.4×10^{-12}	1.3×10^{-16}

	Effective condition numbers			
(r, n, m)	AHO	NT	HKM	New
$(3, 10, 9)$	4.3×10^{11}	4.7×10^{13}	1.9×10^{11}	4.4×10^{11}
$(6, 20, 24)$	3.7×10^{11}	3.5×10^{13}	4.2×10^{11}	3.4×10^{11}

general different than those with block LU. Curiously, this difference sometimes leads to significant differences in $\kappa_e(\mathcal{J})$ for all the methods tested except AHO. Perhaps this is an indication that $\kappa_e(\mathcal{J})$ is highly sensitive to where the current iterate is. The NT method and the HKM method still failed to reduce the normalized residual to full machine precision. Since the normalized residuals were still quite small, it is unlikely that this failure is due to nonconvergence of these methods. This suggests that the numerical instability problem with these two methods is *inherent* and there might be no way to overcome this problem for these two methods.

Table 5.3 shows that for the Type-II SDPs tested, *every one* of these methods failed to reduce the normalized residual to $O(\epsilon)$, and the corresponding $\kappa\left(\mathcal{J}\right)$ or $\kappa_e\left(\mathcal{J}\right)$ was very large for all methods. Table 5.3 supports our conclusion that the AHO method and the TTT methods could be numerically unstable if the $\mathcal{J}$ matrices have large effective condition numbers.

As in Table 5.2, we also solved the problems in Table 5.3 using these four methods by solving (2.2) as dense linear systems of equations. We summarize the results in Table 5.4. As in Table 5.2, both the AHO and the New methods were able to reduce the normalized residual to full machine precision, but the NT and the HKM methods still failed to do so.

6. Conclusions and future work.

In this paper, we analyzed the AHO method and the TTT family of methods in finite precision. Our results indicate that the AHO method and the TTT methods could be numerically stable if an effective condition number associated with the coefficient matrix in (2.2) is small, but unstable otherwise. We also discussed a number of other computational issues related to these methods.

Our numerical experiments indicate that the reason the AHO method appears to be more accurate than the TTT methods is that the condition number for the AHO method is smaller. Further study is needed to better understand this phenomenon. A related issue is how to choose a direction in the TTT family to achieve best convergence and maximum numerical accuracy.

Appendix A. Proofs of some technical results in sections 3 and 4.

Proof of Lemma 3.1. In (3.9), if the matrix A is close to an orthogonal matrix, $A = A^{\dagger} + O(\epsilon)$, where $A^{\dagger}$ is exactly orthogonal, then (3.9) can be rewritten as

$$\mathbf{fl}\left(A\ x\right) = A^{\dagger}\ \left(\left(I+\Delta\right)\ x\right) = \left(I+\bar{\Delta}\right)\ \left(A^{\dagger}\ x\right), \tag{A.1}$$

where $\Delta = (A^{\dagger})^{-1}\ (A+\delta A) - I = O(\epsilon)$ and $\bar{\Delta} = (A+\delta A)\ (A^{\dagger})^{-1} - I = O(\epsilon)$. Since

$$\|\mathbf{smat}(v)\|_F = \left\|Q^{\dagger}\ \mathbf{smat}(v)\ \left(Q^{\dagger}\right)^T\right\|_F = \left\|\left(Q^{\dagger}\right)^T\ \mathbf{smat}(v)\ Q^{\dagger}\right\|_F$$

for all $v \in \mathbf{S}^n$, it follows that

$$v \to \mathbf{svec}\left(Q^{\dagger}\ \mathbf{smat}(v)\ \left(Q^{\dagger}\right)^T\right) \quad \text{and} \quad v \to \mathbf{svec}\left(\left(Q^{\dagger}\right)^T\ \mathbf{smat}(v)\ Q^{\dagger}\right)$$

are orthogonal linear transformations on $\mathbf{R}^{n(n+1)/2}$. With (3.5), the matrix $\bar{V}$ in (3.3) is computed as $\mathbf{fl}(\bar{V}) = \mathbf{fl}(\widehat{Q}^T\ V\ \widehat{Q})$. This can be viewed as a matrix-vector product with a nearly orthogonal matrix. Similar to (A.1), there exists an $n(n+1)/2$ by $n(n+1)/2$ perturbation matrix $\Delta_1 = O(\epsilon)$ such that

$$\mathbf{fl}\left(\bar{V}\right) = \left(Q^{\dagger}\right)^T\ \mathbf{smat}\left(\left(\mathcal{I}+\Delta_1\right)\ v\right)\ Q^{\dagger}. \tag{A.2}$$

The matrix $\bar{U}$ in (3.3) is computed from $\mathbf{fl}\left(\bar{V}\right)$ and $\widehat{\Lambda}$ as

$$\mathbf{fl}\left(\bar{U}\right) = \left(\mathbf{fl}\left(\frac{2\ \left(\mathbf{fl}\left(\bar{V}\right)\right)_{i,j}}{\widehat{\lambda}_i + \widehat{\lambda}_j}\right)\right).$$

By our model of arithmetic (1.7), every entry in $\bar{U}$ is computed to full relative accuracy from $\mathbf{fl}\left(\bar{V}\right)$ and $\widehat{\Lambda}$. In other words, $\mathbf{fl}\left(\bar{U}\right)$ satisfies the equation

$$\mathbf{fl}\left(\bar{U}\right)\ \widehat{\Lambda}+\widehat{\Lambda}\ \mathbf{fl}\left(\bar{U}\right)=2\ \mathbf{fl}\left(\bar{V}\right)+\delta\bar{V}, \tag{A.3}$$

where the perturbation matrix $\delta\bar{V}\in\mathbf{S}^n$ satisfies

$$\left|\left(\delta\bar{V}\right)_{i,j}\right|\le O(\epsilon)\cdot\left|\left(\mathbf{fl}\left(\bar{V}\right)\right)_{i,j}\right|=O(\epsilon\|v\|).$$

Furthermore, the solution to (3.2) is computed from $\mathbf{fl}(\bar{U})$ as $\mathbf{fl}(U)=\mathbf{fl}(\widehat{Q}\ \mathbf{fl}(\bar{U})\ \widehat{Q}^T)$. Similar to (A.1) and (A.2), $\mathbf{fl}\left(U\right)$ satisfies

$$\mathbf{fl}\left(U\right)=\mathbf{smat}\left(\left(\mathcal{I}+\Delta_2\right)\cdot\mathbf{svec}\left(Q^\dagger\ \mathbf{fl}\left(\bar{U}\right)\ \left(Q^\dagger\right)^T\right)\right), \tag{A.4}$$

where $\Delta_2=O(\epsilon)$ is an $n(n+1)/2$ by $n(n+1)/2$ perturbation matrix.

To put all this together, we note that relations (3.6), (A.2), and (A.3) imply

$$\begin{aligned}
&\mathbf{smat}\left(\mathcal{E}^\dagger\ \mathbf{svec}\left(Q^\dagger\ \mathbf{fl}\left(\bar{U}\right)\ \left(Q^\dagger\right)^T\right)\right)\\
&=\frac{\left(Q^\dagger\mathbf{fl}\left(\bar{U}\right)\left(Q^\dagger\right)^T\right)\ Z^\dagger+Z^\dagger\left(Q^\dagger\mathbf{fl}\left(\bar{U}\right)\ \left(Q^\dagger\right)^T\right)}{2}\\
&=\frac{\left(Q^\dagger\left(\mathbf{fl}\left(\bar{U}\right)\widehat{\Lambda}+\widehat{\Lambda}\mathbf{fl}\left(\bar{U}\right)\right)\left(Q^\dagger\right)^T\right)}{2}\\
&=Q^\dagger\ \mathbf{fl}\left(\bar{V}\right)\ \left(Q^\dagger\right)^T+\frac{Q^\dagger\ \mathbf{fl}\left(\delta\bar{V}\right)\ \left(Q^\dagger\right)^T}{2}\\
&=\mathbf{smat}\left(\left(\mathcal{I}+\Delta_1\right)\ v\right)+\frac{Q^\dagger\ \mathbf{fl}\left(\delta\bar{V}\right)\ \left(Q^\dagger\right)^T}{2}\\
&=\mathbf{smat}\left(\left(\mathcal{I}+\Delta_3\right)\ v\right),\ \text{ where }\ \Delta_3\overset{\text{def}}{=}\Delta_1+\frac{\mathbf{svec}\left(Q^\dagger\ \delta\bar{V}\ \left(Q^\dagger\right)^T\right)v^T}{2\ \|v\|_2^2}.
\end{aligned}$$

According to (A.2) and (A.3), we have $\Delta_3=O(\epsilon)$ for all nonzero vectors v. Hence

$$\mathbf{svec}\left(Q^\dagger\ \mathbf{fl}\left(\bar{U}\right)\ \left(Q^\dagger\right)^T\right)=\left(\mathcal{E}^\dagger\right)^{-1}\ \left(\left(\mathcal{I}+\Delta_3\right)\ v\right).$$

Combining this with (A.4) yields the equation in Lemma 3.1. □

For Lemma 4.1 we need some notation. The Kronecker product of two $n\times n$ matrices G and K is $G\otimes K=(g_{ij}\ K)$. By [29], there exists an $n(n+1)/2$ by n^2 row orthogonal matrix $\mathcal{Q}$ such that

$$G\otimes_s K=\frac{1}{2}\mathcal{Q}\ \left(G\otimes K+K\otimes G\right)\mathcal{Q}^T\quad\text{for all } G \text{ and } K\in\mathbf{R}^{n\times n}. \tag{A.5}$$

Let $\mathbf{vec}(G)$ be the n^2-dimensional vector obtained by stacking all the columns of G. Then

$$\mathcal{Q}\ \mathcal{Q}^T=\mathcal{I}\quad\text{and}\quad\mathbf{svec}(H)=\mathcal{Q}\ \mathbf{vec}(H)\quad\text{and}\quad\mathcal{Q}^T\ \mathcal{Q}\ \mathbf{vec}(H)=\mathbf{vec}(H) \tag{A.6}$$

for all $H \in \mathbf{S}^n$. Let $\mathcal{P}$ be the permutation such that $\mathcal{P}\,\mathbf{vec}(G) = \mathbf{vec}(G^T)$ for $G \in \mathbf{R}^{n\times n}$. It is easy to verify that

$$\mathcal{P} = \mathcal{P}^T, \quad \mathcal{P}\ (A \otimes A)\ \mathcal{P} = A \otimes A, \quad \text{and} \quad \mathcal{P}\,\mathbf{vec}(H) = \mathbf{vec}(H) \tag{A.7}$$

for all $H \in \mathbf{S}^n$. We also need the following result concerning round-off errors in a dot product (see Higham [16, Chap. 3]):

$$\mathbf{fl}\left(x^T y\right) = x^T(y + \delta y), \quad \text{where} \quad |\delta y| \le O(\epsilon)|y|. \tag{A.8}$$

Proof of Lemma 4.1. Let $A = (a_1, \ldots, a_n)$. According to (3.9), the ith column of $U\,A$ is computed as $(U + \delta_i U)\,a_i$, where $|\delta_i U| \le O(\epsilon)|U|$. Hence

$$\mathbf{fl}\,(U\ A) = ((U + \delta_1 U)\ a_1, \ldots, (U + \delta_n U)\ a_n)\,.$$

By Algorithm 4.2 and (A.8), both the (i,j) and (j,i) entries of $U\,A\,U^T$ are computed as

$$\left(\mathbf{fl}\left(U\,A\,U^T\right)\right)_{ij} = \sum_{k=1} (\mathbf{fl}(U\ A))_{ik}\ (U_{jk} + \delta_{ij}U_{jk})\,, \quad \text{where} \quad |\delta_{ij}U_{jk}| \le O(\epsilon)\,|U_{jk}|\,.$$

Now we use the above round-off error quantities to define a linear transformation

$$\begin{aligned} \Theta\ \mathbf{svec}\left(\bar{A}\right) &\overset{\text{def}}{=} \mathbf{svec}\left(\sum_{k=1} ((U + \delta_1 U)\ \bar{a}_1, \ldots, (U + \delta_n U)\ \bar{a}_n)_{ik}\ (U_{jk} + \delta_{ij}U_{jk})\right) \\ &\quad - \mathbf{svec}\left(U\ \bar{A}\ U^T\right) \end{aligned} \tag{A.9}$$

for any $\bar{A} = (\bar{a}_1, \ldots, \bar{a}_n) \in \mathbf{S}^n$. The $n(n+1)/2$ by $n(n+1)/2$ matrix Θ is defined by (A.9) and satisfies

$$\mathbf{fl}\,((U \otimes_s U)\ \mathbf{svec}\,(A)) = (U \otimes_s U + \Theta)\ \mathbf{svec}\,(A)\,.$$

To bound Θ, we choose $\bar{A} \ge 0$ and rewrite (A.9) as

$$\begin{aligned} \Theta\,\mathbf{svec}\left(\bar{A}\right) = \mathbf{svec}\Bigg(&\sum_{k=1} ((U + \delta_1 U)\,\bar{a}_1, \ldots, (U + \delta_n U)\,\bar{a}_n)_{ik}\ \delta_{ij}U_{jk} \\ &+ \sum_{k=1} (\delta_1 U\,\bar{a}_1, \cdots, \delta_n U\,\bar{a}_n)_{ik}\ U_{jk}\Bigg)\,. \end{aligned}$$

Taking absolute value entrywise, and using the upper bounds on the round-off error quantities,

$$\begin{aligned} \left|\Theta\ \mathbf{svec}\left(\bar{A}\right)\right| &\le O(\epsilon)\cdot\mathbf{svec}\left(\sum_{k=1} (|U|\ \bar{a}_1, \ldots, |U|\ \bar{a}_n)_{ik}\ |U_{jk}|\right) \\ &= O(\epsilon)\,(|U| \otimes_s |U|)\ \mathbf{svec}\left(\bar{A}\right). \end{aligned}$$

Since the last relation holds for all $\bar{A} \ge 0$, we conclude that $|\Theta| \le O(\epsilon)\cdot(|U| \otimes_s |U|)$.

The last step of Algorithm 4.2 is to apply $\mathcal{D}$ to $\mathbf{fl}\,((U \otimes_s U)\ \mathbf{svec}\,(A))$. By our model of arithmetic (1.7), there exists a diagonal perturbation matrix Δ_1 such that

$$\begin{aligned} \widehat{\mathcal{V}} &= \mathbf{fl}\,(\mathcal{D}\,\mathbf{fl}\,((U \otimes_s U)\ \mathbf{svec}\,(A))) = \mathcal{D}\ (\mathcal{I} + \Delta_1)\ \mathbf{fl}\,((U \otimes_s U)\ \mathbf{svec}\,(A)) \\ &= \mathcal{D}\ \ ((U \otimes_s U + \Theta_1)\ \mathbf{svec}\,(A))\,, \end{aligned}$$

where $\Theta_1 \overset{\text{def}}{=} (\mathcal{I}+\Delta_1)\ (U \otimes_s U + \Theta) - (U \otimes_s U)$ satisfies $|\Theta_1| \le O(\epsilon)\cdot(|U| \otimes_s |U|)$.

To prove the remaining part of Lemma 4.1, define and partition

$$\widehat{V} = \mathbf{fl}\left(U^{-1}A\right) = (\widehat{v}_1,\ldots,\widehat{v}_n) = \begin{pmatrix} \widetilde{v}_1^T \\ \vdots \\ \widetilde{v}_n^T \end{pmatrix} \text{ and } \widehat{W} = \mathbf{fl}\left(\mathbf{fl}\left(U^{-1}A\right)\, U^{-T}\right) = \begin{pmatrix} \widetilde{w}_1^T \\ \vdots \\ \widetilde{w}_n^T \end{pmatrix}.$$

It follows from Algorithm 4.3 that there exist round-off error matrices $\delta_i U$ with $\|\delta_i U\| = O\left(\epsilon\|U\|\right)$ such that $(U+\delta_i U)\,\widehat{v}_i = a_i$, and $\bar{\delta}_i U$ with $\|\bar{\delta}_i U\| = O\left(\epsilon\|U\|\right)$ such that $\widetilde{w}_i^T\left(U+\bar{\delta}_i U\right) = \widetilde{v}_i^T$ for all i. Putting all these relations together and simplifying, we get

$$(U\otimes U+\Delta_2)\,\mathbf{vec}\left(\widehat{W}\right) = \mathbf{vec}(A), \quad \text{where } \|\Delta_2\| = O\left(\epsilon\cdot\|U\|^2\right) \in \mathbf{R}^{n^2\times n^2}. \tag{A.10}$$

To convert this equation into the form in Lemma 4.1, we apply $(\mathcal{I}+\mathcal{P})/2$ to it and simplify to get

$$(U\otimes U)\ \frac{(\mathcal{I}+\mathcal{P})\ \mathbf{vec}\left(\widehat{W}\right)}{2} = \mathbf{vec}(A) - \frac{\mathcal{I}+\mathcal{P}}{2}\,\Delta_2\,\mathbf{vec}\left(\widehat{W}\right), \tag{A.11}$$

where we have used (A.7) and the fact that $A \in \mathbf{S}^n$. By definition and by relation (A.6),

$$\frac{(\mathcal{I}+\mathcal{P})\ \mathbf{vec}\left(\widehat{W}\right)}{2} = \mathbf{vec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right) = \mathcal{Q}^T\ \mathbf{svec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right).$$

Applying $\mathcal{Q}$ to (A.11), and simplifying the resulting equation with this relation and (A.5) and (A.6),

$$(U\otimes_s U)\ \mathbf{svec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right) = \mathbf{svec}(A) - \mathcal{Q}\,\frac{\mathcal{I}+\mathcal{P}}{2}\,\Delta_2\,\mathbf{vec}\left(\widehat{W}\right),$$

which can be further rewritten as

$$((U\otimes_s U)+\Delta_3)\ \mathbf{svec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right) = \mathbf{svec}(A),$$

where

$$\Delta_3 \overset{\text{def}}{=} \frac{\mathcal{Q}\,\dfrac{\mathcal{I}+\mathcal{P}}{2}\,\Delta_2\,\mathbf{vec}\left(\widehat{W}\right)\cdot\mathbf{svec}\left(\dfrac{\widehat{W}+\widehat{W}^T}{2}\right)^T}{\left\|\mathbf{svec}\left(\dfrac{\widehat{W}+\widehat{W}^T}{2}\right)\right\|^2}.$$

To derive an upper bound on Δ_3, we define $W^* = U^{-1}\,A\,U^{-T} \in \mathbf{S}^n$. By assumption in Lemma 4.1, $\kappa(U) \ll 1/\sqrt{\epsilon}$. It follows that $\kappa\left(U\otimes_s U\right) \ll 1/\epsilon$. Since the backward error in (A.10) is of the order $O\left(\epsilon\cdot\|U\|^2\right)$, it follows from standard perturbation theory (cf. (3.17) and see Demmel [10, Chap. 2]) that

$$\frac{\left\|\mathbf{vec}\left(\widehat{W}\right) - \mathbf{vec}\left(W^*\right)\right\|}{\left\|\mathbf{vec}\left(\widehat{W}\right)\right\|} \ll O(1).$$

Consequently,

$$\begin{aligned}\left\|\mathbf{svec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right)\right\| &= \left\|\mathbf{svec}\left(\widehat{W}+\frac{\left(\widehat{W}-W^*\right)^T-\left(\widehat{W}-W^*\right)}{2}\right)\right\| \\ &\geq \left\|\mathbf{vec}\left(\widehat{W}\right)\right\| - \left\|\mathbf{vec}\left(\frac{\left(\widehat{W}-W^*\right)^T-\left(\widehat{W}-W^*\right)}{2}\right)\right\| \\ &= \Omega\left(\left\|\mathbf{vec}\left(\widehat{W}\right)\right\|\right).\end{aligned}$$

Plugging this into the definition of Δ_3, we have

$$\|\Delta_3\| = O\left(\|\Delta_2\|\right) = O\left(\epsilon\cdot\|U\|^2\right).$$

To complete the proof, we note that $\widehat{\mathcal{W}}$ in Lemma 4.1 is obtained by symmetrizing $\widehat{W}$ in finite precision. By our model of arithmetic (1.7), there exists a diagonal perturbation matrix $\Delta_4 = O(\epsilon)$ such that

$$\widehat{\mathcal{W}} = (I+\Delta_4)\ \mathbf{svec}\left(\frac{\widehat{W}+\widehat{W}^T}{2}\right).$$

Hence $\widehat{\mathcal{W}}$ satisfies the equation in Lemma 4.1 with

$$\Theta_2 \overset{\text{def}}{=} \left((U\otimes_s U)+\Delta_3\right)(I+\Delta_4)^{-1} - U\otimes_s U = O\left(\epsilon\|U\|^2\right). \qquad \square$$

Proof of Lemma 4.2. According to Algorithm 4.1 and (4.16),

$$\begin{aligned}\mu &= \sigma\,\frac{X^\dagger\bullet Z^\dagger}{n} = \sigma\,\frac{\left(\widetilde{R}\,\widetilde{H}^T\right)\bullet\left(\widetilde{R}\,\widetilde{H}^T\right)}{n} = \sigma\,\frac{\left(\widehat{\Sigma}+E\right)^T\left(\widehat{\Sigma}+E\right)}{n}, \\ \mathbf{fl}(\mu) &= \mathbf{fl}\left(\sigma\,\frac{\mathbf{tr}\left(\widehat{\Sigma}^2\right)}{n}\right) = \mu+\sigma\,\frac{\mathbf{tr}\left(\widehat{\Sigma}^2-\left(\widehat{\Sigma}+E\right)^T\left(\widehat{\Sigma}+E\right)\right)}{n} + O\left(\epsilon\cdot\|\widehat{\Sigma}\|^2\right) \\ &= \mu + O\left(\epsilon\|\widehat{\Sigma}\|\,\|\widetilde{R}\|\,\|\widetilde{H}\|\right),\end{aligned}$$

where we have used Assumption 4.1. With $\mathbf{fl}(\mu)$, r_c in (4.3) can be computed as

$$\mathbf{smat}\left(\hat{r}_c\right) = \mathbf{fl}\left(\mathbf{fl}(\mu)\ I - \widehat{\Sigma}^2\right) = \mu\ I - \widehat{\Sigma}^2 + E_c,$$

where both $\hat{r}_c$ and E_c are diagonal matrices, with $|E_c| \leq O(\epsilon\cdot\|\widehat{\Sigma}\|\,\|\widetilde{R}\|\,\|\widetilde{H}\|)\,I$. It now follows from (4.16) and (4.17) that

$$\begin{aligned}\mathbf{smat}\left(r_c\right) &= \mu\,I - \mathbf{H}_B\left(\widetilde{H}\ \widetilde{R}^T\ \widetilde{R}\ \widetilde{H}^T\right) = \mu\,I - \mathbf{H}_B\left(\left(\widehat{\Sigma}+E\right)^T\left(\widehat{\Sigma}+E\right)\right) \\ &= B\left(\mu\,I-\widehat{\Sigma}^2\right)\ B^{-1} - \mathbf{H}_B\left(E^T\,\widehat{\Sigma}+\widehat{\Sigma}\,E+E^T\,E\right) \\ &= \mathbf{smat}\left(\hat{r}_c\right) - E_c - \mathbf{H}_B\left(E^T\,\widehat{\Sigma}+\widehat{\Sigma}\,E+E^T\,E\right) \\ &= \mathbf{smat}\left(\hat{r}_c\right) - \mathbf{H}_B\left(\widetilde{E}_c\right), \quad \text{where} \quad \widetilde{E}_c \overset{\text{def}}{=} E_c + E^T\,\widehat{\Sigma}+\widehat{\Sigma}\,E+E^T\,E.\end{aligned}$$

It follows from Assumption 4.1 and the upper bound on E_c that $\widetilde{E}_c = O(\epsilon \|\widehat{\Sigma}\|\, \|\widetilde{R}\|\, \|\widetilde{H}\|)$. Thus,

$$
\begin{aligned}
\left\|\mathcal{S}^{-1}\ (\widehat{r}_c - r_c)\right\| &= \left\|\mathbf{smat}\left(\mathcal{S}^{-1}\ \mathbf{svec}\left(\mathbf{H}_B\left(\widetilde{E}_c\right)\right)\right)\right\|_F \\
&= \left\|\left(\frac{2\left(\mathbf{H}_B\left(\widetilde{E}_c\right)\right)_{ij}}{\left(\frac{\beta_i}{\beta_j}+\frac{\beta_j}{\beta_i}\right)\ \phi_i\ \phi_j + (\phi+\psi)^2\ \left(\frac{\beta_i\,\widehat{\sigma}_i}{\beta_j\,\widehat{\sigma}_j}+\frac{\beta_j\,\widehat{\sigma}_j}{\beta_i\,\widehat{\sigma}_i}\right)}\right)\right\|_F \\
&\leq \left\|\left(\frac{\frac{\beta_i}{\beta_j}+\frac{\beta_j}{\beta_i}}{\frac{\beta_i\,\widehat{\sigma}_i}{\beta_j\,\widehat{\sigma}_j}+\frac{\beta_j\,\widehat{\sigma}_j}{\beta_i\,\widehat{\sigma}_i}}\ \frac{O\left(\epsilon\cdot\|\widehat{\Sigma}\|\,\|\widetilde{R}\|\,\|\widetilde{H}\|\right)}{(\phi+\psi)^2}\right)\right\|_F \\
&= \left\|\left(\frac{\widehat{\sigma}_i\,\widehat{\sigma}_j\,\left(\beta_i^2+\beta_j^2\right)}{\beta_i^2\,\widehat{\sigma}_i^2+\beta_j^2\,\widehat{\sigma}_j^2}\cdot\frac{O\left(\epsilon\|\widehat{\Sigma}\|\,\|\widetilde{R}\|\,\|\widetilde{H}\|\right)}{(\phi+\psi)^2}\right)\right\|_F. \qquad \text{(A.12)}
\end{aligned}
$$

The HKM search direction [15, 17, 20] $P^T P = Z$ and the NT direction [26, 27] correspond to $B = I$ and $B = \widehat{\Sigma}^{-\frac{1}{2}}$, respectively. Since $(\phi+\psi)^2 = O(\|\widehat{\Sigma}\|)$, the expression on the right-hand side of (A.12) is bounded by $O(\epsilon\cdot\|\widetilde{R}\|\,\|\widetilde{H}\|)$ for these two choices of B. In general, we use (4.7) to bound the right-hand side of (A.12) by

$$
\left\|\left(\frac{\left(\widehat{\sigma}_i^2+\widehat{\sigma}_j^2\right)}{\widehat{\sigma}_i\,\widehat{\sigma}_j}\,\frac{O\left(\epsilon\|\widehat{\Sigma}\|\,\|\widetilde{R}\|\,\|\widetilde{H}\|\right)}{(\phi+\psi)^2}\right)\right\|_F = O\left(\epsilon\kappa\left(\widehat{\Sigma}\right)\ \|\widetilde{R}\|\,\|\widetilde{H}\|\right).
$$

Equation (4.20) follows from this relation and (4.19). □

Proof of Lemma 4.4. We first consider the round-off errors in computing $\widetilde{\mathcal{A}}$ in (4.6). It follows from Lemma 4.1 that

$$
\begin{aligned}
\left|\mathbf{fl}\left(\widetilde{\mathcal{A}}\right)-\widetilde{\mathcal{A}}\right| &\leq O(\epsilon)\left(\mathbf{svec}\left(\left|\widetilde{R}\right|\ |A_1|\ \left|\widetilde{R}\right|^T\right),\ldots,\mathbf{svec}\left(\left|\widetilde{R}\right|\ |A_m|\ \left|\widetilde{R}\right|^T\right)\right)^T \\
&= O(\epsilon)\cdot|\mathcal{A}|\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right)^T.
\end{aligned}
$$

Hence the round-off errors in the computed Schur complement $\mathcal{M}$ in (4.6) are bounded by

$$
\begin{aligned}
&O(\epsilon)\,|\mathcal{A}|\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right)^T\,\widehat{\mathcal{D}}_{\mathcal{M}}\cdot\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right)\cdot|\mathcal{A}|^T \\
&\quad= O(\epsilon)\left(\left(\left|\widetilde{R}\right|\ |A_i|\ \left|\widetilde{R}\right|^T\right)\left(\left(\frac{\beta_i^2\ \widehat{\sigma}_i^2+\beta_j^2\ \widehat{\sigma}_j^2}{\widehat{\sigma}_i^2\ \widehat{\sigma}_j^2\ \left(\beta_i^2+\beta_j^2\right)}\right)\odot\left(\left|\widetilde{R}\right|\ |A_j|\ \left|\widetilde{R}\right|^T\right)\right)\right) \\
&\quad\leq O(\epsilon)\left(\left(\left|\widetilde{R}\right|\ |A_i|\ \left|\widetilde{R}\right|^T\right)\left(\left(\frac{1}{\widehat{\sigma}_j^2}+\frac{1}{\widehat{\sigma}_i^2}\right)\odot\left(\left|\widetilde{R}\right|\ |A_j|\ \left|\widetilde{R}\right|^T\right)\right)\right) \\
&\quad= O(\epsilon)\left(\left(\left|\widetilde{R}\right|\ |A_i|\ \left|\widetilde{R}\right|^T\ \widehat{\Sigma}^{-1}\right)\bullet\left(\left|\widetilde{R}\right|\ |A_j|\ \left|\widetilde{R}\right|^T\ \widehat{\Sigma}^{-1}\right)\right) \\
&\qquad+O(\epsilon)\left(\left(\widehat{\Sigma}^{-1}\ \left|\widetilde{R}\right|\ |A_i|\ \left|\widetilde{R}\right|^T\right)\bullet\left(\widehat{\Sigma}^{-1}\ \left|\widetilde{R}\right|\ |A_j|\ \left|\widetilde{R}\right|^T\right)\right),
\end{aligned}
$$

where we have used (4.7). By Assumption 4.1, we have

$$\widehat{\Sigma}^{-1}\ \left|\widetilde{R}\right| = \left|\left(I+\widehat{\Sigma}^{-1}\ E\right)\widetilde{H}^{-T}\right| = \Omega\left(\left\|\widetilde{H}^{-1}\right\|\right).$$

With this estimate, the round-off errors in the computed Schur complement can now be bounded by

$$\begin{aligned} O\left(\epsilon\,\|\mathcal{A}\|^2\ \left\|\widetilde{R}\right\|^2\ \left\|\widehat{\Sigma}^{-1}\ \widetilde{R}\right\|^2\right) &= O\left(\epsilon\,\|\mathcal{A}\|^2\ \left\|\widetilde{R}\right\|^2\ \left\|\widetilde{H}^{-1}\right\|^2\right) \\ &= O\left(\epsilon\,\|\mathcal{A}\|^2\ \|X^\dagger\|\ \left\|\left(Z^\dagger\right)^{-1}\right\|\right). \end{aligned}$$

In other words, the computed $\mathcal{M}$ can be written as $\mathcal{M}^\dagger + O(\epsilon\|\mathcal{A}\|^2\ \|X^\dagger\|\ \|\left(Z^\dagger\right)^{-1}\|)$. In addition, it is clear from this analysis that

$$\left\|\mathcal{M}^\dagger\right\| \leq \|\mathcal{A}\|^2\ \left\|X^\dagger\right\|\ \left\|\left(Z^\dagger\right)^{-1}\right\|\ .$$

By Assumption 3.3, the backward errors committed by the backward solver to solve (2.5c) after $\mathcal{M}$ is computed are bounded by $O\left(\epsilon\|\mathcal{M}^\dagger\|\right)$. Putting all these errors together, we arrive at Lemma 4.4. □

Proof of Theorem 4.5. It is obvious that terms in the second row of $\delta\mathcal{J}_{\mathcal{S}}$ in (4.24) are bounded by $(\epsilon\cdot(1+\|\mathcal{A}\|))$. In the following we will derive upper bounds on the terms in the second and third rows that do not depend on B.

We first consider the third row of $\delta\mathcal{J}_{\mathcal{S}}$. With arguments similar to those in the proof of Lemma 4.4, we can bound all the terms in the $(3,3)$ and $(3,2)$ blocks of $\delta\mathcal{J}_{\mathcal{S}}$ by $O(\epsilon\|\mathcal{A}\|^2\ \|X^\dagger\|\ \|\left(Z^\dagger\right)^{-1}\|)$ and $O(\epsilon\|\mathcal{A}\|\ \|X^\dagger\|\ \|\left(Z^\dagger\right)^{-1}\|)$, respectively. To bound the round-off errors in the $(3,1)$ block, rewrite

$$\begin{aligned} \mathcal{L}_{3,1}\,\mathcal{E} - \mathcal{A} + \mathcal{L}_{3,1}\,\delta\mathcal{E} &= (\mathcal{A}+\delta_1\mathcal{A})\left(\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_1\right)\mathcal{E} - \mathcal{A} + \mathcal{L}_{3,1}\,\delta\mathcal{E} \\ &= \mathcal{A}\ \left(\left(\mathcal{E}^\dagger\right)^{-1}\mathcal{E} - \mathcal{I}\right) + \mathcal{A}\,\Theta_1\,\mathcal{E} + \delta_1\mathcal{A}\ \left(\left(\mathcal{E}^\dagger\right)^{-1} + \Theta_1\right)\ \mathcal{E} + \mathcal{L}_{3,1}\,\delta\mathcal{E}. \end{aligned} \tag{A.13}$$

By definitions (4.16), (4.17), and (4.21), we have

$$\begin{aligned} \left(\mathcal{E}^\dagger\right)^{-1}\ \mathcal{E} &= \left(\left(B\,\widehat{\Sigma}\,\widetilde{R}^{-T}\right)\otimes_s\left(B^{-1}\,\widehat{\Sigma}\,\widetilde{R}^{-T}\right)\right)^{-1}\ \left(\left(B^{-1}\,\widetilde{H}\right)\otimes_s\left(B\,\widetilde{H}\right)\right) \\ &= \left(\left(B\otimes_s B^{-1}\right)\left(\widehat{\Sigma}\otimes_s\widehat{\Sigma}\right)\left(\widetilde{R}^{-T}\otimes_s\widetilde{R}^{-T}\right)\right)^{-1}\ \left(\left(B^{-1}\otimes_s B\right)\cdot\left(\widetilde{H}\otimes_s\widetilde{H}\right)\right) \\ &= \left(\widetilde{R}^T\otimes_s\widetilde{R}^T\right)\left(\widehat{\Sigma}^{-1}\otimes_s\widehat{\Sigma}^{-1}\right)\left(\widetilde{H}\otimes_s\widetilde{H}\right) = \left(\widehat{\Sigma}^{-1}\ \widetilde{R}^T\ \widetilde{H}\right)\otimes_s\left(\widehat{\Sigma}^{-1}\ \widetilde{R}^T\ \widetilde{H}\right) \\ &= \left(\widehat{\Sigma}^{-1}\ \left(\widehat{\Sigma}+E\right)^T\right)\otimes_s\left(\widehat{\Sigma}^{-1}\ \left(\widehat{\Sigma}+E\right)^T\right) \\ &= \mathcal{I} + \left(\widehat{\Sigma}^{-1}\ E^T\right)\otimes_s\left(\widehat{\Sigma}^{-1}\ \left(\widehat{\Sigma}+E\right)^T\right) + I\otimes_s\left(\widehat{\Sigma}^{-1}\ E^T\right)\ . \end{aligned}$$

By Assumption 4.1 and with the last expression, we bound the first term in (A.13) as

$$\begin{aligned} \left\|\mathcal{A}\ \left(\left(\mathcal{E}^\dagger\right)^{-1}\ \mathcal{E}-\mathcal{I}\right)\right\| &\leq O\left(\|\mathcal{A}\|\ \left\|\widehat{\Sigma}^{-1}\right\|\ \|E\|\right) = O\left(\epsilon\|\mathcal{A}\|\ \|\widetilde{R}\|\ \|\widetilde{H}\|\ \left\|\widehat{\Sigma}^{-1}\right\|\right) \\ &\leq O\left(\epsilon\|\mathcal{A}\|\ \left(\|X^\dagger\|\ \|\left(X^\dagger\right)^{-1}\|\ \|Z^\dagger\|\ \|\left(Z^\dagger\right)^{-1}\|\right)^{\frac{1}{2}}\right)\ . \end{aligned}$$

By definition (4.17) and Lemma 4.1, all the other terms in (A.13) are bounded by

$$\begin{aligned} &O\left(\epsilon\right)|\mathcal{A}| \cdot \left(\left|\widetilde{R}\right|^T \otimes_s \left|\widetilde{R}\right|^T\right) \left(B\,\widehat{\Sigma} \otimes_s B^{-1}\,\widehat{\Sigma}\right)^{-1} \left(\left(B^{-1} \otimes_s B\right)|\Theta|\right) \\ &= O\left(\epsilon\right) \cdot |\mathcal{A}| \left(\left|\widehat{\Sigma}^{-1}\,\widetilde{R}\right|^T \otimes_s \left|\widehat{\Sigma}^{-1}\,\widetilde{R}\right|^T\right)|\Theta| \quad \text{for some} \quad \Theta = \Omega\left(\|\widetilde{H}\|^2\right) . \end{aligned}$$

As in Lemma 4.4, this bound can be simplified to $O(\epsilon\|\mathcal{A}\|\,\|Z^\dagger\|\,\|\left(Z^\dagger\right)^{-1}\|)$. Adding it all up, the terms in the third row of $\delta\mathcal{J}_{\mathcal{S}}$ are bounded by

$$\begin{aligned} &O\left(\epsilon\left(1+\|\mathcal{A}\|\right)^2\,\left(\|X^\dagger\|+\|Z^\dagger\|\right)\,\left(\|\left(X^\dagger\right)^{-1}\|+\|\left(Z^\dagger\right)^{-1}\|\right)\right) \\ &= O\left(\epsilon\left(\|X^\dagger\|+\|Z^\dagger\|\right)\,\left(\|\left(X^\dagger\right)^{-1}\|+\|\left(Z^\dagger\right)^{-1}\|\right)\right) . \end{aligned} \tag{A.14}$$

Now we consider the terms in the first row of $\delta\mathcal{J}_{\mathcal{S}}$. A bound on $\delta\mathcal{E}$ is given in (4.23). Since

$$\mathcal{S} = \mathcal{S}_{\mathcal{E}} + \mathcal{S}_{\mathcal{F}} \geq \mathcal{S}_{\mathcal{E}} = \left(B^{-1}\,\Phi \otimes_s B\,\Phi\right),$$

it follows from (4.23) that the term in the $(1,1)$ block of $\delta\mathcal{J}_{\mathcal{S}}$ satisfies

$$\left\|\mathcal{S}^{-1}\,\delta\mathcal{E}\right\| = \left\|\mathcal{S}^{-1}\,\mathcal{S}_{\mathcal{S}}\,\Theta_3\right\| \leq \|\Theta_3\| = O(\epsilon).$$

For the $(1,2)$ block, we use definitions (4.16), (4.17), and (4.21) to get

$$\begin{aligned} \mathcal{F}^\dagger - \mathcal{F} &= \left(B\,\widehat{\Sigma}\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widehat{\Sigma}^{-1}\,\widetilde{R}\right) - \left(B\,\widetilde{H}\,X^\dagger\right) \otimes_s \left(B^{-1}\,\widetilde{H}^{-T}\right) \\ &= \left(B\,\widehat{\Sigma}\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widehat{\Sigma}^{-1}\,\left(\widehat{\Sigma}+E\right)\,\widetilde{H}^{-T}\right) \\ &\quad - \left(B\,\left(\widehat{\Sigma}+E\right)^T\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widetilde{H}^{-T}\right) \\ &= \left(B\,\widehat{\Sigma}\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widehat{\Sigma}^{-1}\,E\;\,\widetilde{H}^{-T}\right) - \left(B\,E^T\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widetilde{H}^{-T}\right) . \end{aligned} \tag{A.15}$$

Since $\mathcal{S} \geq \mathcal{S}_{\mathcal{F}} = (\phi+\psi)^2\left(B\,\widehat{\Sigma}\right) \otimes_s \left(B\,\widehat{\Sigma}\right)^{-1}$, we can scale and bound the first term in (A.15) as

$$\begin{aligned} &\left\|\mathcal{S}^{-1}\left(B\,\widehat{\Sigma}\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widehat{\Sigma}^{-1}\,E\;\,\widetilde{H}^{-T}\right)\right\| \leq \frac{\|\widetilde{R}\|_F\,\|E\|_F\,\|\widetilde{H}^{-T}\|_F}{(\phi+\psi)^2} \\ &\leq O\left(\epsilon\left(\|X^\dagger\|+\|Z^\dagger\|\right)^{\frac{1}{2}}\,\left(\|\left(X^\dagger\right)^{-1}\|+\|\left(Z^\dagger\right)^{-1}\|\right)^{\frac{1}{2}}\right) , \end{aligned}$$

where we have used (4.16) and the fact that $\phi = \Omega(\|\widetilde{H}\|)$ and $\psi = \Omega(\|\widetilde{R}\|)$. We also chose to write the bound in a form similar to (A.14). For the second term in (A.15), we have

$$\left\|\mathcal{S}^{-1}\left(B\,E^T\,\widetilde{R}\right) \otimes_s \left(B^{-1}\,\widetilde{H}^{-T}\right)\right\| \leq \left\|\mathcal{S}_{\mathcal{F}}^{-1}\left(B \otimes_s B^{-1}\right)\right\|\,\|E\|_F\,\|\widetilde{R}\|_F\,\|\widetilde{H}^{-T}\|_F. \tag{A.16}$$

To see the diagonal entries of $\mathcal{S}_{\mathcal{F}}^{-1}\left(B \otimes_s B^{-1}\right)$ more clearly, we apply it to the vector $\mathbf{e}$ in section 1.4:

$$\mathcal{S}_{\mathcal{F}}^{-1}\left(B \otimes_s B^{-1}\right)\;\mathbf{e} = \frac{1}{(\phi+\psi)^2}\mathbf{svec}\left(\frac{\dfrac{\beta_i}{\beta_j}+\dfrac{\beta_j}{\beta_i}}{\dfrac{\beta_i\,\widehat{\sigma}_i}{\beta_j\,\widehat{\sigma}_j}+\dfrac{\beta_j\,\widehat{\sigma}_j}{\beta_i\,\widehat{\sigma}_i}}\right).$$

Similar to (A.12), the entries in the last matrix are bounded by $1/(\phi+\psi)^2$ for the HKM search direction $P^T\,P = Z$ and the NT direction, and by $\kappa\left(\widehat{\Sigma}\right)/(\phi+\psi)^2$ in general. Combining this with (4.16) and (A.16), we obtain a bound on the second term in (A.15) similar to that on the first term,

$$\begin{aligned}&\left\|\mathcal{S}^{-1}\left(B\,E^T\,\widetilde{R}\right)\otimes_s\left(B^{-1}\,\widetilde{H}^{-T}\right)\right\|\\ &\le O\left(\epsilon\cdot\kappa\left(\widehat{\Sigma}\right)\,\left(\|X^\dagger\|+\|Z^\dagger\|\right)^{\frac{1}{2}}\,\left(\|\left(X^\dagger\right)^{-1}\|+\|\left(Z^\dagger\right)^{-1}\|\right)^{\frac{1}{2}}\right).\end{aligned}$$

The last term of $\delta\mathcal{J}_{\mathcal{S}}$ to be bounded is $\mathcal{S}^{-1}\delta_2\mathcal{F}^\dagger$. It follows from Lemma 4.1 that

$$\left|\delta_2\mathcal{F}^\dagger\right| \le O(\epsilon)\left(B\,\widehat{\Sigma}\,\left|\widetilde{R}\right|\right)\otimes_s\left(B^{-1}\,\widehat{\Sigma}^{-1}\,\left|\widetilde{R}\right|\right) = O(\epsilon)\mathcal{S}_{\mathcal{F}}\,\left(\left|\widetilde{R}\right|\otimes_s\left|\widetilde{R}\right|\right).$$

Hence an analysis similar to the above yields $\left\|\mathcal{S}^{-1}\delta_2\mathcal{F}^\dagger\right\| = O(\epsilon)$. Adding up bounds for all three rows of $\delta\mathcal{J}_{\mathcal{S}}$, we arrive at the equation in Theorem 4.5. □

Acknowledgments. The author thanks Professor Michael Overton of the Courant Institute for attracting his attention to the subject of this paper, for many helpful discussions, and for making available to him a MATLAB code of the AHO method and other methods. He also thanks Michael and the anonymous referees for very helpful suggestions that greatly improved the presentation of this paper.

REFERENCES

[1] F. ALIZADEH, *Combinatorial Optimization with Interior Point Methods and Semidefinite Matrices*, Ph.D. thesis, University of Minnesota, Minneapolis, 1991.

[2] F. ALIZADEH, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.

[3] F. ALIZADEH, J.-P. HAEBERLY, M. V. NAYAKKANKUPPAM, M. L. OVERTON, AND S. SCHMIETA, *SDPpack User's Guide (Version* 0.9 *Beta)*, Technical Report 737, Department of Computer Science, New York University, June 1997; also available online from http://www.cs.nyu.edu/cs/faculty/overton/sdppack/sdppack.html.

[4] F. ALIZADEH, J.-P. HAEBERLY, AND M. OVERTON, *personal communication with M. Overton*, 1997.

[5] F. ALIZADEH, J.-P. A. HAEBERLY, AND M. L. OVERTON, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., (1998), pp. 746–768.

[6] B. BORCHERS, *CSDP, A C Library for Semidefinite Programming*, http://www.nmt.edu/˜borchers/csdp.html, 1997.

[7] S. BOYD, EL GHAOUI, E. FERON, AND V. BALAKRISHNAN, *Linear Matrix Inequalities in System and Control Theory*, SIAM Stud. Appl. Math. 15, SIAM, Philadelphia, 1994.

[8] S. CHANDRASEKARAN AND I. IPSEN, *Backward errors for eigenvalue and singular value decompositions*, Numer. Math., 68 (1994), pp. 215–223.

[9] J. DEMMEL, *The condition number of equivalence transformations that block diagonalize matrix pencils*, SIAM J. Numer. Anal., 20 (1983), pp. 599–610.

[10] J. W. DEMMEL, *Numerical Linear Algebra*, SIAM, Philadelphia, 1997.

[11] A. FORSGREN, P. E. GILL, AND J. R. SHINNERL, *Stability of symmetric ill-conditioned systems arising in interior methods for constrained optimization*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 187–211.

[12] K. FUJISAWA, M. KOJIMA, AND K. NAKATA, *SDPA User's Manual*, ftp.is.titech.ac.jp/pub/OpRes/articles/b308.ps.Z, 1998.

[13] G. GOLUB AND C. VAN LOAN, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, MD, 1996.

[14] J.-P. HAEBERLY, *Remarks on Nondegeneracy in Mixed Semidefinite-Quadratic Programming.* Unpublished memorandum, available from http://corky.fordham.edu/haeberly/papers/sqldegen.ps.gz.

[15] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[16] N. J. Higham, *Accuracy and Stability of Numerical Algorithms*, SIAM, Philadelphia, 1996.

[17] M. Kojima, S. Shindoh, and S. Hara, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.

[18] A. S. Lewis and M. L. Overton, *Eigenvalue optimization*, Acta Numerica, 5 (1996), pp. 149–190.

[19] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

[20] R. D. C. Monteiro, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.

[21] R. D. C. Monteiro, *Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions*, SIAM J. Optim., 8 (1998), pp. 797–812.

[22] R. D. C. Monteiro and T. Tsuchiya, *Polynomial convergence of a new family of primal-dual algorithms for semidefinite programming*, SIAM J. Optim, 9 (1999), pp. 551–577.

[23] R. D. C. Monteiro and T. Tsuchiya, *Polynomiality of primal-dual algorithms for semidefinite linear complementarity problems based on the Kojima-Shindoh-Hara family of directions*, Math. Programming, 84 (1999), pp. 39–53.

[24] R. D. C. Monteiro and Y. Zhang, *A unified analysis for a class of path-following primal-dual interior-point algorithms for semidefinite programming*, Math. Programming, 81 (1998), pp. 281–299.

[25] Yu. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[26] Yu. E. Nesterov and M. J. Todd, *Self-scaled barriers and interior-point methods in convex programming*, Technical Report 1091, School of Operations Research and Industrial Engineering, Cornell University, 1994; Math. Oper. Res., to appear.

[27] Y. E. Nesterov and M. J. Todd, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

[28] D. B. Ponceleón, *Barrier Methods for Large-Scale Quadratic Programming*, Ph.D. thesis, Stanford University, Stanford, CA, 1990.

[29] M. J. Todd, K. C. Toh, and R. H. Tütüncü, *On the Nesterov–Todd direction in semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 769–796.

[30] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[31] M. H. Wright, *Ill-conditioning and computational error in interior methods for nonlinear programming*, SIAM J. Optim., 9 (1999), pp. 84–111.

[32] S. J. Wright, *Stability of linear equations solvers in interior-point methods*, SIAM J. Matrix Anal. Appl., 16 (1994), pp. 1287–1307.

[33] S. J. Wright, *Stability of augmented system factorizations in interior-point methods*, SIAM J. Matrix Anal. Appl., 18 (1997), pp. 191–222.

[34] Y. Zhang, *On extending some primal-dual interior-point algorithms from linear programming to semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 365–386.

SIAM J. Optim.
Vol. 10, No. 2, pp. 503–533

NEWTON'S METHOD FOR A CLASS OF OPTIMAL SHAPE DESIGN PROBLEMS*

MANFRED LAUMEN†

Abstract. A class of optimal shape design problems is considered where a part of the boundary of the domain represents the free parameter. The variable domain is parametrized by a class of functions in such a way that the optimal design problem results in an optimal control problem on a fixed domain. The functions for the parametrization of the domain are used as controls, and the corresponding states are then given by the solution of an elliptic boundary value problem on a fixed domain.

Discretizing this control problem normally leads to a large-scale optimization problem, where the corresponding solution methods are characterized by the requirement of solving many boundary value problems. In spite of this interesting numerical challenge, until now little work has been done to derive more efficient algorithms by taking advantage of the specific structure of this kind of problem.

In this report, Newton's method in function space is derived, resulting in an efficient algorithm for the discretized optimization problems. By using the specific structure of these optimal shape design problems, an efficient implementation of the numerical algorithm is introduced. The properties of this algorithm are compared with those of the gradient method using illustrative numerical examples.

Key words. Newton's method, optimal shape design, optimal control

AMS subject classifications. 65K10, 49K20, 49M15, 49M05

PII. S1052623496302877

1. Introduction.

Shape optimization is described by finding the geometry of a structure that is optimal in the sense of a given minimized cost function with respect to certain constraints. The study of this subject is located at the interface of four distinct fields: optimal control, partial differential equations, numerical analysis, and optimization. Many-faceted problems naturally arise in engineering applications with the goal of designing a specific structure in an optimal sense, or alternatively, of understanding and determining the shape of a given structure. Typical applications are the design of a nozzle [32], a thermal diffuser [13], [14], an airfoil boundary [33], or various beams and plates [22], [36], [37] with respect to specific optimality conditions.

Several methods, e.g., the speed method [37], the boundary element method [32], and the fictitious domain method [21], have been developed for such problems in the past. The computation of the solution is a time-consuming task for all of them because discretizing normally leads to a large-scale optimization problem, which requires the subsequent solution of many boundary value problems.

In spite of this interesting numerical challenge, the approaches for solving optimal shape design problems mostly deal with optimization methods based only on the first order information of the cost function; see, for example, Pironneau and Vossinis [33] and the above-mentioned citations. It is a well-known fact that first order optimization schemes require only a small number of operations in each iteration, while retaining the disadvantage of a slow local convergence rate. Contrary to this, optimization methods exploiting second order information of the cost function, such as Newton's method, converge with a fast convergence rate by accepting an increased number of operations per iteration.

*Received by the editors May 1, 1996; accepted for publication (in revised form) January 2, 1999; published electronically February 18, 2000.
http://www.siam.org/journals/siopt/10-2/30287.html

†Universität Trier, FB IV - Mathematik, 54286 Trier, Germany (laumen@uni-trier.de).

In recent years, much research has been done to analyze the advantage and disadvantage of using second order information for constructing efficient algorithms for large-scale problems. In this context, the analysis and numerical results of the multidisciplinary design (see, for example, [17], [30], [38]) and the various experiences of automatic differentiation [20] to compute the derivatives have to be mentioned. Interesting results have also been achieved by using quasi-Newton methods for solving these kinds of large-scale problems [24], [25], [27]. One key to constructing efficient methods for such problems is always the exploitation of the specific structure of the problems under consideration. The aim of this paper is to analyze a specific class of optimal shape design problems and to construct a more efficient second order method for these large-scale problems by taking advantage of the problem-inherent structure.

We consider a class of optimal shape design problems, where the feasible domains are described by

$$\tilde{\Omega} = \left\{ \tilde{x} = (\tilde{x}_1, \tilde{x}_2)^T \in \mathbb{R}^2 \mid \tilde{x}_2 \in I := (0,1) \wedge \tilde{x}_1 \in (0, u(\tilde{x}_2)) \right\}. \tag{1.1}$$

These domains are parametrized by a function $u \in \mathcal{U}_{ad}$, where

$$\mathcal{U}_{ad} = \left\{ u \in \mathcal{C}^{0,1}(I) \mid 0 < \beta_1 \leq u(x_2) \leq \beta_2 \text{ a.e.} \right\} \tag{1.2}$$

is a suitable subset of the Banach space $\mathcal{U} = \mathcal{C}^{0,1}(I)$. This space restriction reflects further constraints on the physical model and is necessary to guarantee the existence of a solution. Moreover, the given twice Fréchet-differentiable cost function $\tilde{J}$ depends on a function $\tilde{y}$, which is defined by the solution of a second order elliptic boundary value problem on $\tilde{\Omega}$. Since the boundary value problem is solved with the finite element method, the weak formulation in terms of a variational equation on the moving domain is used from the beginning. The boundaries of the domains are divided into the moving boundary part

$$\tilde{\Gamma}^M = \left\{ (\tilde{x}_1, \tilde{x}_2)^T \in \mathbb{R}^2 \mid \tilde{x}_1 = u(\tilde{x}_2) \ \forall \tilde{x}_2 \in I \right\}$$

and the fixed boundary part $\tilde{\Gamma}^F$ with $\partial\tilde{\Omega} = \tilde{\Gamma}^M \cup \tilde{\Gamma}^F$ and $\tilde{\Gamma}^M \cap \tilde{\Gamma}^F = \emptyset$. Based on this definition, the space of the state is defined by

$$\tilde{\mathcal{V}} = \left\{ \phi \in \mathcal{H}^1(\tilde{\Omega}) | \gamma_0 \phi|_{\tilde{\Gamma}_0} = 0 \right\},$$

where the trace map $\gamma_0 \in \mathcal{L}(\mathcal{H}^1(\tilde{\Omega}), \mathcal{H}^{\frac{1}{2}}(\partial\tilde{\Omega}))$ of order zero is further omitted if the meaning is obvious. Assuming that $\tilde{\Gamma}_0$ is a closed subset of $\partial\tilde{\Omega}$, it can be inferred that the space $\tilde{\mathcal{V}}$ is a closed subspace of $\mathcal{H}^1(\tilde{\Omega})$, and therefore also a Hilbert space. Evidently, the limiting case $\tilde{\Gamma}_0 = \emptyset$ reduces to $\tilde{\mathcal{V}} = \mathcal{H}^1(\tilde{\Omega})$, while the limiting case $\tilde{\Gamma}_0 = \partial\tilde{\Omega}$ reduces to $\tilde{\mathcal{V}} = \mathcal{H}^1_0(\tilde{\Omega})$.

Thus, the optimal shape design problem is defined by

$$\min_{u \in \mathcal{U}_{ad}} \tilde{J}(u, \tilde{y}, \tilde{z}), \tag{1.3}$$

with the equality constraint

$$\tilde{a}(\tilde{y}, \tilde{\eta}) = \tilde{l}(\tilde{\eta}) \quad \forall \tilde{\eta} \in \tilde{\mathcal{V}} := \tilde{\mathcal{V}}(\tilde{\Omega}),$$

where $\tilde{a}(\cdot,\cdot) : \tilde{\mathcal{V}} \times \tilde{\mathcal{V}} \to \mathbb{R}$ denotes the continuous $\tilde{\mathcal{V}}$-elliptic bilinear form

$$\tilde{a}(\tilde{y}, \tilde{\eta}) = \sum_{|i|,|j| \leq 1} \left\langle \tilde{a}_{ij} \tilde{D}^i \tilde{y}, \tilde{D}^j \tilde{\eta} \right\rangle_{\mathcal{L}^2(\tilde{\Omega})} + \left\langle \tilde{b}\tilde{y}, \tilde{\eta} \right\rangle_{\mathcal{L}^2(\Gamma_1)} \tag{1.4}$$

and the linear functional $\tilde{l}(\cdot) : \tilde{\mathcal{V}} \to \mathbb{R}$ is given by

$$\tilde{l}(\tilde{\eta}) = \sum_{|i|\leq 1} \big\langle \tilde{f}_i, \tilde{D}^i\tilde{\eta}\big\rangle_{\mathcal{L}^2(\tilde{\Omega})} + \big\langle \tilde{f}, \tilde{\eta}\big\rangle_{\mathcal{L}^2(\Gamma_1)}. \tag{1.5}$$

It will be assumed that this bilinear form $\tilde{a}$ is always $\tilde{\mathcal{V}}$-elliptic and continuous but not necessarily symmetric. In most situations, the function $\tilde{z} \in \tilde{\mathcal{Z}}$, $\tilde{\mathcal{Z}}$ Hilbert space, has to be introduced to describe a desired state of the function $\tilde{y}$, or for handling inhomogeneous Dirichlet boundary conditions. Although the explicit statement of the dependency on $\tilde{z}$ is not standard, it is included in our presentation in order to get a clear arrangement. Particular emphasis is placed on the fact that the considered class of problems deals with cost functions that can be defined on a part of or on the entire domain, as well as on a part of or on the entire boundary of the domain. Several examples of cost functions are discussed in [22]. In addition, almost all elliptic boundary value problems of the second order are included, in particular, the cases of homogeneous and inhomogeneous Dirichlet, homogeneous and inhomogeneous Neumann, Robin's, and mixed problems.

The optimal shape design problems under consideration occur, for instance, if a nozzle is designed [32], where the control u determines the shape of the symmetric structure. In this case, the flow through the nozzle is defined by a boundary value problem, and the cost function is given in such a way that a certain flow distribution has to be satisfied on the domain or on a boundary part.

Another application with a parabolic boundary value problem is illustrated by Banks and Kojima [3], [4] and Banks, Kojima, and Winfree [5], where the thermal tomography problem is stated in the optimal shape design context to estimate structural flows in materials, which could arise, e.g., from corrosion or cracks. This leads to a nondestructive evaluation method that is appropriate for assessing the structural integrity of structures. In particular, defects of fiber-reinforced composite materials, recently proposed in space structures studies, are determined, although they may not be detectable by visual inspection [5].

The algorithm presented in this paper is based on the mapping method as described by Begis and Glowinski [9] and Pironneau [32]. This method has the advantage that the theory presented also covers a class of optimal control problems, where the coefficients of the variational equation are influenced by the control. However, second order algorithms can be constructed analogously for other kinds of methods, leading to a similar structure of the algorithm.

As sketched in Figure 1, this mapping method transforms the problem defined on the moving domain $\tilde{\Omega}$ to a problem defined on a fixed one Ω. For instance, the transformation

$$T^{-1} : \begin{array}{l} \tilde{\Omega} \longrightarrow \Omega, \\ (\tilde{x}_1, \tilde{x}_2)^T \longrightarrow (x_1, x_2)^T = \left(\frac{\tilde{x}_1}{u(\tilde{x}_2)}, \tilde{x}_2\right)^T \end{array} \tag{1.6}$$

can be used to map the moving domain to the fixed reference domain $\Omega = (0,1) \times I$.

In the following section we will describe in detail how such a transformation converts the original optimal shape design problem into a distributed optimal control problem

$$\min_{u\in\mathcal{U}_{ad}} J\big(u, y, z(u)\big), \tag{1.7}$$

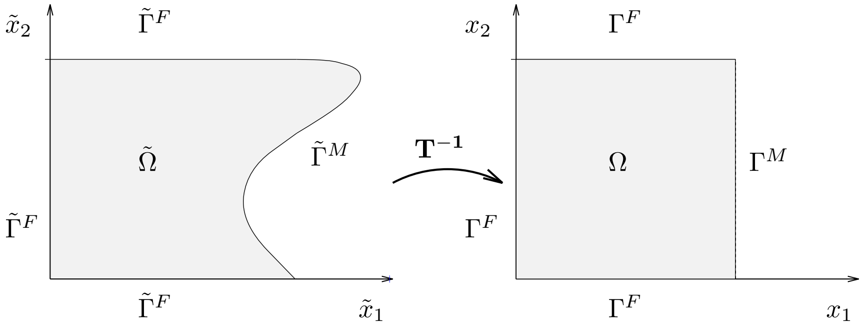


FIG. 1. *Transformation of the domain.*

where the transformed state $y \in \mathcal{V}$, $\mathcal{V} := \mathcal{V}(\Omega)$ Hilbert space, is defined by the solution of a more complicated and nonsymmetric variational equation on the fixed domain Ω depending nonlinearly on the control $u \in \mathcal{U}_{ad}$:

$$a(u; y, \eta) = l(u; \eta) \quad \forall \eta \in \mathcal{V}. \tag{1.8}$$

This is contrary to fictitious domain methods, where a simplified variational equation has to be solved by using a nondifferentiable optimization method, which prevents or complicates the use of high-order optimization schemes. For instance, Haslinger, Hoffmann, and Kočvara [21] derived a fictitious domain method for a specific class of optimal shape design problems in the finite dimensional setting that is only differentiable if higher order finite elements are used for the implementation.

In general, there exist different ways for solving the constraint minimization problem (1.7). The all-at-once (AAO) method [30] considers the control and the state as independent variables coupled by the state equation (1.8). Then the derived optimal control problem can be interpreted as a constrained minimization problem, where the variational equation represents one equality constraint. Various algorithms are developed for this kind of minimization problem in the field of optimization which are characterized by the fact that each evaluation of the cost function and its derivative does not require the solution of the current state equation. One disadvantage of this approach is the enlargement of the variable space from $\mathcal{U}_{ad}$ to $\mathcal{U}_{ad} \times \mathcal{V}$ [27], [35]. This leads to the fact that the control, the state, and the adjoint correspond to each other only at the solution point. For Newton's method these functions correspond to each other in each iteration, i.e., $y_i = S(u_i)$ and $p_i = T(u_i)$ for $i = 1, 2, \ldots$.

Another method, which Lewis [30] called the multidisciplinary feasible (MDF) formulation, is used in this paper, where the state y is treated as being dependent on the control u in order to avoid the enlargement of the variable space. Under weak assumptions the existence of a solution operator S, with $y = S(u)$, will be proven in section 3, leading to the minimization problem

$$\min_{u \in \mathcal{U}_{ad}} J\big(u, S(u), z(u)\big).$$

To handle the possible ill-posedness of the problems, a Tikhonov regularization term (see, e.g., [6], [8]) is added to the original cost function if necessary. Thus, the cost function is written as

$$\min_{u \in \mathcal{U}_{ad}} F(u) \tag{1.9}$$

with

$$F(u) := J\big(u, S(u), z(u)\big) + \frac{\varepsilon}{2}\|u - u_{\mathcal{T}}\|^2_{\mathcal{T}}, \quad \varepsilon \in \mathbb{R},$$

and as a function $u_{\mathcal{T}}$. The minimization problem is further simplified by supposing initially the nonactivity of the constraint $u \in \mathcal{U}_{ad}$; i.e, the problem (1.9) can be treated as an unconstrained minimization problem. This simplification is justified if the solution u_* is supposed to be an interior point of $\mathcal{U}_{ad}$ and if the starting point is near the solution, which is guaranteed if a nested iteration is used [28], [29]. The methods discussed here under the nonactivity assumption can be generalized to handle further constraints without difficulties, for instance, by using projections for simple constraints.

In the last section a comparison of the gradient method with and without Armijo line search to Newton's method shows the local superiority of the derived second order method that takes advantage of the specific structure of this kind of problem. Certainly, these considerations are restricted to the local behavior of this method. For an assertion on the global convergence properties, a further globalization technique, e.g., trust region or line search, has to be added to Newton's method. Since each step of Newton's method requires much computing time on a fine grid, we suggest a combination of nested iteration and a simple line search method. Due to our construction procedure of Newton's method in the infinite dimensional setting, we are in the convenient situation of being able to compare the discretized iterates with the infinite ones. A modified mesh independence principle for this specific class of problem is presented in [29].

Besides the advantage of the mesh independence principle for predicting the convergence of the computable method on the basis of the analyzed infinite dimensional convergence, there is a further important point for practical implementation. The mesh independence lays the theoretical foundation for the justification of refinement strategies and helps to design this refinement process (see, e.g., [2]). Since the focus is on the infinite dimensional solution, a fine discretization scheme has to be chosen, so that the discrete solution approximates the infinite dimensional one appropriately. However, a fine discretization also means that the finite problem consists of many variables, and therefore an increased amount of work per iteration has to be expected.

Thus, for simplicity we start with a projective gradient method on a coarse mesh by using a line search algorithm with backtracking. We carry out iterations corresponding to a certain stopping criterion and use this solution approximation, which is interpolated to the finer mesh, as a new starting point for Newton's method on this mesh. This local or global refinement process continues until the finest mesh is reached. The numerical computations, presented in [28] and in the last section, underline the efficiency of the described procedure.

In [28] a detailed discussion of the comparison of different methods, including quasi-Newton methods, is given for this kind of problems. It is shown that the SR1- and the BFGS-update is far more efficient for this class of problems than the PSB- or DFP-update. However, since in general it is not guaranteed that the Hessian is positive definite, the BFGS-update failed for some test problems. Moreover, fewer convergence assertions compared to the other updates are proven for the SR1-update. One main advantage of this report is given by the detailed analysis of the structure of this class of problem, which could also be exploited for the additive structured updates [16].

At this point a few comments have to be made on the research on optimal shape design of Goto and Fujii [18], [19]. They state Newton's method for some specific problems and present corresponding numerical results. Contrary to our approach, normal variations are used for deriving the derivatives. Their reports also show some shortcomings in the theory, as well as in the numerical implementation. The existence of their derivatives is not proven in the Fréchet sense, which is necessary for the convergence theory of Newton's method in function spaces. Therefore, no common convergence result of Newton's method can be applied. Even if the derivatives exist in the Fréchet sense, then the question of how to choose adequate spaces has to be answered. For this reason, the discretization used in these papers is not justified by a statement concerning the convergence of the computed solution to the infinite dimensional one. They also take no advantage of the problem-inherent structure for the numerical implementation. In addition, their numerical comparison between Newton's and the gradient method is not informative, since the control function is discretized with only four nodes and some implementational details are missing. Finally, they report numerical difficulties with examples where the Hessian is indefinite at the beginning. This kind of problem never occurred for the algorithm we present.

The tilde indicates a function, boundary, etc., on the moving domain $\tilde\Omega$ and it will be omitted if the symbols are defined analogously on the fixed region Ω. Furthermore, C and c, with all possible accents and indices, always represent generic constants.

2. Transformation of the problem. In this section, the transformation of the optimal shape design problem into an optimal control problem is analyzed in detail for any transformation $T = T(u)$ that is completely determined by a function $u \in \mathcal{U}_{ad}$. This transformation is illustrated by the frequently used transformation mentioned in the introduction.

The general variational equation

$$\tilde a(\tilde y, \tilde\eta) = \tilde l(\tilde\eta) \quad \forall \tilde\eta \in \tilde{\mathcal{V}}$$

on the moving domain $\tilde\Omega$ is investigated, where the not necessarily symmetric bilinear form $\tilde a(\cdot,\cdot)$ and the linear functional $\tilde l(\cdot)$ are given by (1.4) and (1.5).

The transformation $T^{-1} : \tilde\Omega \to \Omega$ of the domain is assumed to be continuous everywhere, as well as bijective and differentiable almost everywhere. Then the generalized substitution rule of Rudin [34, Theorem 7.26, p. 153] can be applied to the variational equation. Corresponding to our convention, the functions without tilde are evaluated at $x = T^{-1}(\tilde x)$. The symbol $T|_{\Gamma_1}$ denotes the restriction of T to Γ_1, and J denotes the Jacobian of the transformation. Thus, the variational equation is transformed into

$$\tilde a(y, \eta) = \tilde l(\eta) \quad \forall \eta \in \mathcal{V},$$

with

$$\tilde a(y,\eta) = \sum_{|i|,|j|\le 1} \big\langle a_{ij} |\det J_T| \tilde D^i y, \tilde D^j \eta \big\rangle_{\mathcal{L}^2(\Omega)} + \big\langle b |\det J_{T|_{\Gamma_1}}| y, \eta \big\rangle_{\mathcal{L}^2(\Gamma_1)}$$

and

$$\tilde l(\eta) = \sum_{|i|\le 1} \big\langle f_i |\det J_T|, \tilde D^i \eta \big\rangle_{\mathcal{L}^2(\Omega)} + \big\langle f |\det J_{T|_{\Gamma_1}}|, \eta \big\rangle_{\mathcal{L}^2(\Gamma_1)}.$$

Until now, the compact notation $\tilde D^i$ with the multi-index i has been used for the different derivatives with respect to $\tilde x$. However, for the transformation of this derivative operator it will be advantageous to use the partial derivative notation $\frac{\partial\eta}{\partial\tilde{x}_\mu}$ with $\mu = 1, 2$. Evaluating the Jacobian of the transformation

$$J_{T^{-1}}(x) = \tilde{J}_{T^{-1}}(T(x)) = \tilde{J}_{T^{-1}}(\tilde{x}) = \begin{pmatrix} \frac{\partial x_1}{\partial\tilde{x}_1} & \frac{\partial x_1}{\partial\tilde{x}_2} \\ \frac{\partial x_2}{\partial\tilde{x}_1} & \frac{\partial x_2}{\partial\tilde{x}_2} \end{pmatrix},$$

the chain rule

$$\frac{\partial\eta}{\partial\tilde{x}_\mu} = \sum_{\nu=1}^{2} \frac{\partial\eta}{\partial x_\nu}\frac{\partial x_\nu}{\partial\tilde{x}_\mu} \quad \text{for } \mu = 1, 2$$

can be written shortly as

$$\tilde{\nabla}\eta = \left(\frac{\partial\eta}{\partial\tilde{x}_1}, \frac{\partial\eta}{\partial\tilde{x}_2}\right)^T = J_{T^{-1}}^T \nabla\eta. \tag{2.1}$$

Hence, two cases have to be distinguished as far as the derivative operator $\tilde{D}^i$ is concerned. For $i = (0, 0)$ the simple equality $\tilde{D}^i = D^i$ holds, while otherwise $\tilde{D}^i$ is given as a linear combination of $D^{(1,0)}$ and $D^{(0,1)}$. Either way, by a suitable redefinition of the coefficient functions a_{ij}, f_i, b, and f, a structure can be derived that is similar to the original variational equation.

Assuming that the transformation $T = T(u)$ is completely determined by a function $u \in \mathcal{U}_{ad}$, we finally obtain a variational equation

$$a(u; y, \eta) = l(u; \eta) \quad \forall\eta \in \mathcal{V},$$

with

$$a(u; y, \eta) = \sum_{|i|,|j|\le 1} \left\langle a_{ij}(u)D^i y, D^j\eta\right\rangle_{\mathcal{L}^2(\Omega)} + \langle b(u)y, \eta\rangle_{\mathcal{L}^2(\Gamma_1)}$$

and

$$l(u; \eta) = \sum_{|i|\le 1} \left\langle f_i(u), D^j\eta\right\rangle_{\mathcal{L}^2(\Omega)} + \langle f(u), \eta\rangle_{\mathcal{L}^2(\Gamma_1)}.$$

This problem is described on the fixed domain Ω, where the coefficient functions now depend nonlinearly on the parameter function $u \in \mathcal{U}_{ad}$.

This transformation process is illustrated by the specific transformation (1.6), where the definition (1.2) of $\mathcal{U}_{ad}$ implies that the control u is in the space of bounded variations. Therefore, Lebesgue's differentiation theorem [7, Theorem 8, p. 85] yields that the function u is differentiable almost everywhere and that its derivative is bounded almost everywhere, too. Thus, the transformation T^{-1} satisfies the assumptions for applying the substitution rule.

For simplicity we consider the general bilinear form

$$\tilde{a}(\tilde{y}, \tilde{\eta}) = \int_{\tilde{\Omega}} \left(\tilde{\nabla}\tilde{y}\right)^T \tilde{A}\tilde{\nabla}\tilde{\eta} + \tilde{a}^T\tilde{\nabla}\tilde{y}\tilde{\eta} + \tilde{b}\tilde{y}\tilde{\eta}\, d\tilde{x} + \int_{\tilde{\Gamma}_1} \tilde{c}\tilde{y}\tilde{\eta}\, d\tilde{\Gamma}_1 \tag{2.2}$$

on the moving domain $\tilde{\Omega}$ with the matrix $\tilde{A} \in \mathbb{R}^{2\times 2}$, the vector $\tilde{a} \in \mathbb{R}^2$, and the functions $\tilde{b}$, $\tilde{c} \in \mathbb{R}$. Since the transformation of the linear functional $\tilde{l}(\tilde{\eta})$ and the cost function $\tilde{J}(\tilde{y},\tilde{z})$ is done analogously, the considerations are restricted to the bilinear form (2.2).

Taking advantage of the transformation described above and the equality (2.1), the following reformulation of the bilinear form (2.2) is obtained:

$$\begin{aligned}\tilde{a}(\tilde{y},\tilde{\eta}) &= \int_\Omega (\nabla y)^T \big(|\det J_T| J_{T^{-1}} \tilde{A} J_{T^{-1}}^T\big)\nabla\eta + \big(|\det J_T| J_{T^{-1}}\tilde{a}\big)^T \nabla y\eta \\ &\quad + \big(|\det J_T|\tilde{b}\big) y\eta\, dx + \int_{\Gamma_1} \big(|\det J_{T|_{\Gamma_1}}|\tilde{c}\big) y\eta\, d\Gamma_1 \\ &=: \int_\Omega (\nabla y)^T A(u)\nabla\eta + a(u)^T\nabla y\eta + b(u)y\eta\, dx + \int_{\Gamma_1} c(u)y\eta\, d\Gamma_1 \\ &=: a(u;y,\eta). \qquad (2.3)\end{aligned}$$

The matrix $A(u)$, the vector $a(u)$, and the remaining function $b(u)$, as well as $c(u)$, have to be evaluated for each boundary value problem by simple multiplications if the terms $\det J_T$, $\det J_{T|_{\Gamma_1}}$, and $J_{T^{-1}}$ are explicitly known.

To illustrate this transformation a specific test example is considered, which is used for the numerical experiences afterwards. Initially, the optimal shape design problem is given by

$$\min_{u\in\mathcal{U}_{ad}} \int_{\tilde{\Omega}} (\hat{y}-\hat{z})^2\, d\tilde{x},$$

with $\hat{z} = \sin\big(\frac{2\pi\tilde{x}_1}{u}\big)\sin(2\pi\tilde{x}_2)$, Ω defined by (1.1), and the equality constraint stated by a boundary value problem with inhomogeneous Dirichlet boundary conditions:

$$\begin{aligned}-\Delta\hat{y}+\hat{y} &= (8\pi^2+1)\sin(2\pi\tilde{x}_1)\sin(2\pi\tilde{x}_2) \quad \text{in } \tilde{\Omega},\\ \hat{y} = \tilde{g}_D &:= \sin(2\pi\tilde{x}_1)\sin(2\pi\tilde{x}_2) \quad \text{on } \partial\tilde{\Omega}.\end{aligned}$$

This example is constructed in such a way that $\hat{y}_* = \sin(2\pi\tilde{x}_1)\sin(2\pi\tilde{x}_2)$ is the solution of the state corresponding to the optimal domain with $u_*\equiv 1$. For stating the weak formulation of the inhomogeneous Dirichlet problem, the splitting $\hat{y} = \tilde{y}+\tilde{g}_D$ has to be introduced leading to the modified optimal shape design problem

$$\min_{u\in\mathcal{U}_{ad}} \int_{\tilde{\Omega}} (\tilde{y}-\tilde{z})^2 d\tilde{x}$$

with $\tilde{z} = \hat{z}-\tilde{g}_D$ and the corresponding variational equation

$$\tilde{a}(\tilde{y},\tilde{\eta}) = \tilde{l}(\tilde{\eta}) \quad \forall\tilde{\eta}\in\mathcal{H}_0^1(\tilde{\Omega}),$$

where the bilinear form is defined by

$$\tilde{a}(\tilde{y},\tilde{\eta}) = \int_{\tilde{\Omega}} \nabla\tilde{y}\nabla\tilde{\eta} + \tilde{y}\tilde{\eta}\, d\tilde{x}$$

and the linear function is given by

$$\tilde{l}(\tilde{\eta}) = \int_{\tilde{\Omega}} (8\pi^2+1)\sin(2\pi\tilde{x}_1)\sin(2\pi\tilde{x}_2)\eta - \tilde{a}(\tilde{g}_D,\tilde{\eta}).$$

Now, the general transformation (2.3) can be used by setting $\tilde{A} = I$, $\tilde{a} = 0$, $\tilde{b} = 1$, and $\tilde{c} = 0$. For the specific transformation (1.6), the following formulas can be computed:

$$\det J_T = u,$$

$$\det J_{T|_{\Gamma_1}} = \begin{cases} 0 & \forall x \in \{(0, x_2) \in \mathbb{R}^2 \mid x_2 \in I\}, \\ u & \forall x \in \{(x_1, 0) \in \mathbb{R}^2 \mid x_1 \in [0,1]\} \\ & \qquad \cup \{(x_1, |I|) \in \mathbb{R}^2 \mid x_1 \in [0,1]\}, \\ \sqrt{1 + u'(x_2)^2} & \forall x \in \{(1, x_2) \in \mathbb{R}^2 \mid x_2 \in (0, |I|)\}, \end{cases}$$

$$J|_{T^{-1}} = \begin{pmatrix} \frac{1}{u} & -\frac{x_1 u'}{u} \\ 0 & 1 \end{pmatrix}.$$

This leads to the nonsymmetric transformed bilinear form

$$\begin{aligned} a(u; y, \eta) &= \int_\Omega (\nabla y)^T \left(|\det J_T| J_{T^{-1}} J_{T^{-1}}^T\right) \nabla \eta + \left(|\det J_T| J_{T^{-1}} 0\right)^T \nabla y \eta \\ &\quad + (|\det J_T|)\, y\eta \, dx + \int_{\Gamma_1} \Big(|\det J_{T|_{\Gamma_1}}| 0\Big)\, y\eta \, d\Gamma_1 \\ &= \int_\Omega \left(\frac{1}{u} + \frac{x_{1^2 u'^2}}{u}\right) D^{(1,0)} y D^{(1,0)} \eta - x_1 u' D^{(0,1)} y D^{(1,0)} \eta \\ &\quad - x_1 u' D^{(1,0)} y D^{(0,1)} \eta + u D^{(0,1)} y D^{(0,1)} \eta + u y \eta \, dx. \end{aligned} \tag{2.4}$$

It is easily verified that the entire transformed optimal shape design problem is given by the nonlinear problem

$$\min_{u \in \mathcal{U}} \int_\Omega (y - z)^2 u \, dx$$

under the constraint of

$$a(u; y, \eta) = l(u; \eta) \quad \forall \eta \in \mathcal{V} := \mathcal{H}_0^1(\Omega),$$

with the bilinear form (2.4) and the linear functional

$$l(u; \eta) = \int_\Omega \big(8\pi^2 + 1\big) u \sin(2u\pi x_1) \sin(2\pi x_2) \eta \, dx - a\big(u; g_D(u), \eta\big).$$

This problem is now defined on the fixed domain Ω, where the coefficient functions depend nonlinearly on the parameter function $u \in \mathcal{U}_{ad}$.

Finally, the question arises of what kind of connection holds between the solutions $\tilde{y}$ and y and between the spaces that are described on the domains $\tilde{\Omega}$ and Ω.

THEOREM 2.1. *Let $T : \Omega \to \tilde{\Omega}$ be continuous everywhere. Furthermore, let the transformation be bijective, as well as differentiable almost everywhere, such that its derivative is also bounded almost everywhere.*

Then the norms $\|\cdot\|_{\mathcal{H}^1(\tilde{\Omega})}$ and $\|\cdot\|_{\mathcal{H}^1(\Omega)}$, as well as $\|\cdot\|_{\mathcal{H}^{1/2}(\partial\tilde{\Omega})}$ and $\|\cdot\|_{\mathcal{H}^{1/2}(\partial\Omega)}$, are equivalent. In addition, the bilinear form $\tilde{a}$ is continuous and $\tilde{\mathcal{V}}$-elliptic if and only if the bilinear form a is continuous and $\mathcal{V}$-elliptic.

Proof. The proof of this theorem is similar to that of the transformation theorem proven by Adams [1, Theorem 3.35] for a $\mathcal{C}^1$ transformation and by Wloka [39, Theorem 4.1] for a $\mathcal{C}^{0,1}$ transformation. □

3. Newton's method in function space. Based on Taylor's formula, Newton's method minimizes a quadratic model of the cost function F in each current point u. This is done by solving the linear equation

$$F''(u)(w)(v) = -F'(u)(v) \quad \forall v \in \mathcal{U} \tag{3.1}$$

and correcting the initial approximation by $u_+ = u + w$.

In the optimal control problems under consideration, the cost function $F(u)$ is implicitly defined by the state that is given as the solution of a variational equation. Thus, to state Newton's method, the dependent state, with respect to the control u, must first be analyzed in the context of Fréchet differentiability.

In addition to the variational equation (1.8) that determines the state, the adjoint variational equation

$$a(u;\eta,p) = l(u;\eta) \quad \forall \eta \in \mathcal{V} \tag{3.2}$$

is also considered with the same bilinear form $a(u;\cdot,\cdot)$, which is not assumed to be symmetric. For the sake of simplicity, the arbitrary linear functional on the right-hand side of the adjoint equation is also denoted by $l(u;\cdot)$ in the first part of this section. Afterwards, it is chosen in a convenient way differing from the right-hand side of the state equation.

Throughout this section, the following assumptions are made.

(A1) $\Omega \subset \mathbb{R}^2$ is Lipschitz continuous.

(A2) $a(u;\cdot\cdot)$ is a $\mathcal{V}$-elliptic bilinear form for all $u \in \mathcal{U}_{ad}$.

(A3) $f(u) \in \mathcal{V}'$, $a_{ij}(u) \in \mathcal{L}^\infty(\Omega)$, $b(u) \in \mathcal{L}^\infty(\Gamma_1)$.

The trace inequalities (see, e.g., [39, Theorem 6.1]) yield

$$\|y\|_{\mathcal{H}^1(\Omega)} \geq C_1 \|\gamma_0 y\|_{\mathcal{H}^{\frac{1}{2}}(\partial\Omega)} \geq C_2 \|\gamma_0 y\|_{\mathcal{L}^2(\partial\Omega)} \geq C_3 \|\gamma_0 y\|_{(\mathcal{H}^{\frac{1}{2}}(\partial\Omega))'},$$

and therefore,

$$\|y\|_{\mathcal{V}} \geq C_1 \|\gamma_0 y\|_{\mathcal{H}^{\frac{1}{2}}(\partial\Omega)} \geq C_2 \|\gamma_0 y\|_{\mathcal{L}^2(\partial\Omega)} \geq C_3 \|\gamma_0 y\|_{(\mathcal{H}^{\frac{1}{2}}(\partial\Omega))'}. \tag{3.3}$$

Hence, by using the Cauchy–Schwarz inequality, the continuity of the bilinear form $a(u;\cdot,\cdot)$ is obtained as a simple consequence of (A3):

$$\begin{aligned}
&|a(u;y,\eta)| \\
&\quad \leq \sum_{|i|,|j|\leq 1} \|a_{ij}(u)\|_{\mathcal{L}^\infty(\Omega)} \langle D^i y, D^j \eta\rangle_{\mathcal{L}^2(\Omega)} + \|b(u)\|_{\mathcal{L}^\infty(\Gamma_1)} \langle y,\eta\rangle_{\mathcal{L}^2(\Gamma_1)} \\
&\quad \leq \sum_{|i|,|j|\leq 1} \|a_{ij}(u)\|_{\mathcal{L}^\infty(\Omega)} \|D^i y\|_{\mathcal{L}^2(\Omega)} \|D^j \eta\|_{\mathcal{L}^2(\Omega)} + \|b(u)\|_{\mathcal{L}^\infty(\Gamma_1)} \|y\|_{\mathcal{L}^2(\Gamma_1)} \|\eta\|_{\mathcal{L}^2(\Gamma_1)} \\
&\quad \leq c\|y\|_{\mathcal{V}} \|\eta\|_{\mathcal{V}}.
\end{aligned}$$

In the second part of this section, some more assumptions are needed, which are ordered by strength. For example, (A7) implies the assumptions (A3) and (A4)–(A6).

(A4) f, a_{ij}, and b are once continuously Fréchet differentiable in u.

(A5) The first Fréchet derivatives of f, a_{ij}, and b are Lipschitz continuous.

(A6) f, a_{ij}, and b are twice continuously Fréchet differentiable in u.

(A7) The second Fréchet derivatives of f, a_{ij}, and b are Lipschitz continuous in u.

The first theorem characterizes the dependence on the control of the state and the adjoint by proving that y and p can be written as an operator of u.

THEOREM 3.1. *Let* (A1)–(A3) *be satisfied. For each $f(u) \in \mathcal{V}'$, there exists a unique solution $y \in \mathcal{V}$ and a unique solution $p \in \mathcal{V}$ of the two variational equations*

$$\begin{aligned} a(u; y, \eta) &= l(u; \eta) \quad \forall \eta \in \mathcal{V}, \\ a(u; \eta, p) &= l(u; \eta) \quad \forall \eta \in \mathcal{V}. \end{aligned}$$

Thus, there also exist unique solution operators $S : \mathcal{U}_{ad} \to \mathcal{V}$ with $y = S(u)$ and $T : \mathcal{U}_{ad} \to \mathcal{V}$ with $p = T(u)$.

Proof. To use the Lax–Milgram lemma, it remains to be verified that $l(u; \cdot)$ is a continuous linear functional. However, this is a simple implication of

$$\begin{aligned} |l(u; \eta)| &\leq |\langle f(u), \eta \rangle_{\mathcal{V}' \times \mathcal{V}}| \\ &\leq \|f(u)\|_{\mathcal{V}'} \|\eta\|_{\mathcal{V}} \\ &\leq c \|\eta\|_{\mathcal{V}}. \end{aligned}$$

Therefore, the Lax–Milgram lemma can be applied, resulting in the existence of a unique y and a unique p for every $u \in \mathcal{U}_{ad}$, given by the solutions of the different variational problems. Hence, unique solution operators S with $y = S(u)$ and T with $p = T(u)$, respectively, exist. □

The existence of the solution operators is not sufficient to state Newton's method. In addition, it is necessary to prove their Fréchet differentiability. This will be done in the following theorem, by comparing each variational equation with another one, whose unique solution is expected to be the derivative. Attention should be drawn to the fact that both variational equations consist of the same bilinear form, and thus, the only difference is due to the linear functional on the right-hand side. Furthermore, it should be noted that the $\mathcal{V}$-ellipticity of the bilinear form is essential for the proof of this theorem.

THEOREM 3.2. *Let* (A1)–(A4) *be satisfied. Then the solution operators $S : \mathcal{U}_{ad} \to \mathcal{V}$ and $T : \mathcal{U}_{ad} \to \mathcal{V}$ are Fréchet differentiable with $S'(u), T'(u) \in \mathcal{L}(\mathcal{U}, \mathcal{V})$. Moreover, $\hat{y}^v := S'(u)(v)$ and $\hat{p}^v := T'(u)(v)$ are the unique solutions of the variational problems*

$$a(u; \hat{y}^v, \eta) = l_u(u; \eta)(v) - a_u(u; S(u), \eta)(v) \quad \forall \eta \in \mathcal{V}, \tag{3.4}$$

$$a(u; \eta, \hat{p}^v) = l_u(u; \eta)(v) - a_u(u; \eta, T(u))(v) \quad \forall \eta \in \mathcal{V}, \tag{3.5}$$

with

$$a_u(u; y, \eta)(v) = \sum_{|i|,|j| \leq 1} \langle a'_{ij}(u)(v) D^i y, D^j \eta \rangle_{\mathcal{L}^2(\Omega)} + \langle b'(u)(v) y, \eta \rangle_{\mathcal{L}^2(\Gamma_1)}$$

and

$$l_u(u; \eta)(v) = \langle f'(u)(v), \eta \rangle_{\mathcal{V}' \times \mathcal{V}}.$$

The proof is given in the appendix.

To state Newton's method, the cost function F is assumed to be twice continuously Fréchet differentiable in u. In order to prove this, the Lipschitz continuity of the solution operators $S'(\cdot)$ and $T'(\cdot)$ will be necessary.

LEMMA 3.3. *If* (A1)–(A5) *are satisfied, then* $S'(\cdot) : \mathcal{U}_{ad} \to \mathcal{V}$ *and* $T'(\cdot) : \mathcal{U}_{ad} \to \mathcal{V}$ *are Lipschitz continuous.*

Proof. This assertion can be proven analogously to the proof of Theorem 3.2. □

It should be noted that, if (A6) is supposed to be true, the assumption (A5) is implicitly valid. In other words, the assumptions of Lemma 3.3 are satisfied if the coefficient functions are twice Fréchet differentiable, which must be assumed anyway to state Newton's method.

3.1. First derivative of the cost function. After these theoretical investigations the first Fréchet derivative of the cost function can be derived. Under suitable assumptions, the first Fréchet derivative

$$F(u) = J(u, S(u), z(u)) + \frac{\varepsilon}{2}\|u - u_{\mathcal{T}}\|_{\mathcal{T}}^2$$

is given by

$$\begin{aligned} F'(u)(v) &= J_u\big(u, S(u), z(u)\big)(v) + J_y\big(u, S(u), z(u)\big)\big(S'(u)(v)\big) \\ &\quad + J_z\big(u, S(u), z(u)\big)\big(z'(u)(v)\big) + \varepsilon\langle u - u_{\mathcal{T}}, v\rangle_{\mathcal{T}}. \end{aligned} \tag{3.6}$$

Unfortunately, the function v implicitly influences $z'(u)(v)$ and $S'(u)(v)$. In common applications the first term is explicitly given, while $S'(u)(v)$ is the solution of a variational equation. Since this implicit formulation is not convenient for the construction of our algorithm, the first two derivatives of the cost functions are equivalently reformulated. This is done by using the adjoint equation with the linear functional $l(u;\eta) := J_y(u, S(u), z(u))(\eta)$, $\eta \in \mathcal{V}$. For stating the following theorems some modified versions of the assumptions (A3)–(A5) are needed.

(B3) $J : \mathcal{U}_{ad} \times \mathcal{V} \times \mathcal{Z} \to \mathbb{R}$ is once Fréchet differentiable with $J_y(u, S(u), z(u)) \in \mathcal{V}'$.

(B4) $J : \mathcal{U}_{ad} \times \mathcal{V} \times \mathcal{Z} \to \mathbb{R}$ is twice Fréchet differentiable and $J_{yu}(u, S(u), z(u))(v) \in \mathcal{V}'$ for all $v \in \mathcal{U}$, $J_{yy}(u, S(u), z(u)) \in (\mathcal{V} \times \mathcal{V})'$, as well as $J_{yz}(u, S(u), z(u)) \in (\mathcal{V} \times \mathcal{Z})'$, are continuous with respect to u.

(B5) $J : \mathcal{U}_{ad} \times \mathcal{V} \times \mathcal{Z} \to \mathbb{R}$ is twice continuously Fréchet differentiable and $J_{yu}(u, S(u), z(u))(v) \in \mathcal{V}'$ for all $v \in \mathcal{U}$, $J_{yy}(u, S(u), z(u)) \in (\mathcal{V} \times \mathcal{V})'$, $J_{yz}(u, S(u), z(u)) \in (\mathcal{V} \times \mathcal{Z})'$ are Lipschitz continuous with respect to all components.

Attention should be drawn to the fact that, from now on, the assumptions (A1)–(A7) correspond only to the state equation. To apply the results of the previous section to the adjoint, the modified assumptions (B3)–(B5) must additionally be satisfied.

THEOREM 3.4. *Let* (A1)–(A3) *and* (B3) *be satisfied. Then there exists a unique solution* $p \in \mathcal{V}$ *of the adjoint equation*

$$a(u; \eta, p) = J_y(u, S(u), z(u))(\eta) \quad \forall \eta \in \mathcal{V} \tag{3.7}$$

and a unique solution operator $T : \mathcal{U}_{ad} \to \mathcal{V}$ *with* $p = T(u)$. *Moreover, if* (A4) *is valid and the function* $z \in \mathcal{Z}$ *is Fréchet differentiable, then the cost function* F *is also Fréchet differentiable and* $F'(u) \in \mathcal{L}(\mathcal{U}, \mathbb{R})$ *is defined by*

$$\begin{aligned} F'(u)(v) &= J_u(u, S(u), z(u))(v) + J_z(u, S(u), z(u))\,(z'(u)(v)) \\ &\quad + l_u(u; T(u))(v) - a_u(u; S(u), T(u))(v) + \varepsilon\langle u - u_{\mathcal{T}}, v\rangle_{\mathcal{T}}. \end{aligned} \tag{3.8}$$

Proof. The existence and uniqueness of the solution operators $S(u)$ and $T(u)$ are given by Theorem 3.1.

Since (A1)–(A4) are satisfied, Theorem 3.2 yields the existence of $\hat{y}^v = S'(u)(v)$ as the solution of

$$a\big(u; \hat{y}^v, \eta\big) = l_u(u; \eta)(v) - a_u(u; y, \eta)(v) \quad \forall \eta \in \mathcal{V}.$$

By setting $\eta = \hat{y}^v$ in the adjoint variational equation (3.7) and $\eta = p$ in the previous equation, the equality of both left-hand sides is obtained. Hence, the two right-hand sides evaluated at the distinct points are also equal:

$$J_y\big(u, S(u), z(u)\big)\big(S'(u)(v)\big) = l_u(u; p)(v) - a_u(u; y, p)(v).$$

This completes the proof by substituting the last equality into the first derivative (3.6) of the cost function. □

Before stating the second derivative of the cost function, the auxiliary bilinear operators $H_1(u; \cdot, \cdot) : \mathcal{V} \times \mathcal{U} \to \mathbb{R}$ and $H_{\mathcal{T}}(\cdot, \cdot) : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ are introduced:

$$\begin{aligned} H_1(u; p, v) &:= l_u(u; p)(v) - a_u\big(u; S(u), p\big)(v), \\ H_{\mathcal{T}}(u - u_{\mathcal{T}}, v) &:= \varepsilon \langle u - u_{\mathcal{T}}, v \rangle_{\mathcal{T}}. \end{aligned}$$

A notation analogous to the bilinear form $a(u; \cdot, \cdot)$ is used for the auxiliary operator $H_1(u; \cdot, \cdot)$. The functions that H_1 depends on linearly are separated by a comma, and the nonlinear dependency of u is indicated by a semicolon.

Thus, the first derivative is rewritten as

$$\begin{aligned} F'(u)(v) = {} & J_u\big(u, S(u), z(u)\big)(v) + J_z\big(u, S(u), z(u)\big)\big(z'(u)(v)\big) \\ & + H_1\big(u; T(u), v\big) + H_{\mathcal{T}}(u - u_{\mathcal{T}}, v). \end{aligned} \tag{3.9}$$

To sum up, the right-hand side of Newton's equation is computed in three steps. First the variational equations (1.8) and (3.7) are solved to obtain the state and the adjoint of the control problem. The first derivative of the cost function can then be evaluated. Since the state is needed to evaluate the linear functional of the adjoint variational equation, and since both solutions are required for the computation of the right-hand term of Newton's equation, these steps have to be done sequentially. To summarize, these steps are listed in Algorithm 3.5.

Algorithm 3.5 (first derivative in function spaces).

1. *Compute the state $y = S(u)$ as the solution of*

$$a(u; y, \eta) = l(u; \eta) \quad \forall \eta \in \mathcal{V}.$$

2. *Compute the adjoint $p = T(u)$ as the solution of*

$$a(u; \eta, p) = J_y(u, y, z(u))(\eta) \quad \forall \eta \in \mathcal{V}.$$

3. *Compute the negative first derivative as*

$$\begin{aligned} -F'(u)(v) = {} & -J_u\big(u, y, z(u)\big)(v) - J_z\big(u, y, z(u)\big)\big(z'(u)(v)\big) \\ & - H_1(u; p, v) - H_{\mathcal{T}}(u - u_{\mathcal{T}}, v). \end{aligned}$$

3.2. Second derivative of the cost function. Instead of differentiating the original formula (3.6), the second derivative of the cost function is derived in the next theorem from the adjoint equation (3.7) of Theorem 3.4.

Theorem 3.6. *Let* (A1)–(A4), (B3)–(B4) *be satisfied and the function* $z \in \mathcal{Z}$ *be once Fréchet differentiable. The derivative* $\hat{p} := \hat{p}^w := T'(u)(w)$ *of the adjoint* $p = T(u)$ *is then given by the solution of the variational problem*

$$\begin{aligned} a(u;\eta,\hat{p}) = {} & J_{yu}\big(u, S(u), z(u)\big)(\eta)(w) + J_{yy}\big(u, S(u), z(u)\big)(\eta)\big(S'(u)(w)\big) \\ & + J_{yz}\big(u, S(u), z(u)\big)(\eta)\big(z'(u)(w)\big) - a_u\big(u;\eta,T(u)\big)(w) \quad \forall \eta \in \mathcal{V}. \end{aligned} \tag{3.10}$$

If, in addition, even (A6) *is valid and the function* $z \in \mathcal{Z}$ *is twice Fréchet differentiable, then the cost function* F *is also twice Fréchet differentiable and* $F''(u) \in \mathcal{L}(\mathcal{U}, \mathcal{L}(\mathcal{U}, \mathbb{R}))$ *is defined by*

$$\begin{aligned} F''(u)(v)(w) = {} & J_{uu}\big(u, S(u), z(u)\big)(v)(w) \\ & + J_{uy}\big(u, S(u), z(u)\big)(v)\big(S'(u)(w)\big) \\ & + J_{uz}\big(u, S(u), z(u)\big)(v)\big(z'(u)(w)\big) \\ & + J_{zu}\big(u, S(u), z(u)\big)\big(z'(u)(v)\big)(w) \\ & + J_{zy}\big(u, S(u), z(u)\big)\big(z'(u)(v)\big)\big(S'(u)(w)\big) \\ & + J_{zz}\big(u, S(u), z(u)\big)\big(z'(u)(v)\big)\big(z'(u)(w)\big) \\ & + J_{z}\big(u, S(u), z(u)\big)\big(z''(u)(v)(w)\big) \\ & - a_u\big(u; S'(u)(w), T(u)\big)(v) - a_u\big(u; S(u), T'(u)(w)\big)(v) \\ & - a_{uu}\big(u; S(u), T(u)\big)(v)(w) \\ & + l_u\big(u; T'(u)(w)\big)(v) + l_{uu}\big(u; T(u)\big)(v)(w) + \varepsilon\langle v, w\rangle_{\mathcal{T}}, \end{aligned} \tag{3.11}$$

with

$$a_{uu}(u; y, p)(v)(w) = \sum_{|i|,|j| \le 1} \big\langle a''_{ij}(u)(v)(w) D^i y, D^j p \big\rangle_{\mathcal{L}^2(\Omega)} + \big\langle b''(u)(v)(w) y, p \big\rangle_{\mathcal{L}^2(\Gamma_1)}$$

and

$$l_{uu}(u; p)(v)(w) = \big\langle f''(u)(v)(w), p \big\rangle_{\mathcal{V}' \times \mathcal{V}}.$$

Proof. Since the assumptions (A1)–(A4) are valid, Theorem 3.2 yields the existence of a unique state derivative $\hat{y} := S'(u)(w)$ in the new direction w. Recalling the definition of the bilinear operator $H_1(u;\cdot,\cdot)$ from the last subsection, $\hat{y}$ is given as the solution of

$$a\big(u; \hat{y}^v, \eta\big) = H_1(u; \eta, w) \quad \forall \eta \in \mathcal{V}. \tag{3.12}$$

Moreover, the corresponding modified assumptions (B3) and (B4) for the linear functional of the adjoint are satisfied. Therefore, Theorem 3.2 can be applied to show that $\hat{p} = T'(u)(w)$ is the unique solution of the variational equation (3.10).

Hence, all operators in (3.11) are well defined and it remains to prove that the equality holds. To show this, the first derivative is considered in the representation (3.8) of the last theorem. Based on the given assumptions, this formula can simply be differentiated to prove the desired assertion. □

At this point it should be emphasized that due to the use of the adjoint equation no derivative of the solution operator $S(u)$ is needed for the first derivative of the cost function. Moreover, the last theorem shows how to compute the second derivative of the cost function without explicitly computing the second derivative of a solution operator.

To simplify the formulation of the algorithm, three more bilinear operators $H_2(u;\cdot,\cdot) : \mathcal{V} \times \mathcal{U} \to \mathbb{R}$, $H_3(u;\cdot,\cdot) : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$, and $H_4(u;\cdot,\cdot) : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ are introduced by the following definitions:

$$\begin{aligned}
H_2(u;\eta,w) &:= J_{yu}\big(u,S(u),z(u)\big)(\eta)(w)\\
&\quad + J_{yz}\big(u,S(u),z(u)\big)(\eta)\big(z'(u)(w)\big) - a_u\big(u;\eta,T(u)\big)(w),\\
H_3(u;\eta,\hat{y}) &:= J_{yy}\big(u,S(u),z(u)\big)(\eta)\,(\hat{y})\,,\\
H_4(u;v,w) &:= J_{uu}\big(u,S(u),z(u)\big)(v)(w)\\
&\quad + J_{uz}\big(u,S(u),z(u)\big)(v)\big(z'(u)(w)\big)\\
&\quad + J_{zu}\big(u,S(u),z(u)\big)\big(z'(u)(v)\big)(w)\\
&\quad + J_{zz}\big(u,S(u),z(u)\big)\big(z'(u)(v)\big)\big(z'(u)(w)\big)\\
&\quad + J_{z}\big(u,S(u),z(u)\big)\big(z''(u)(v)(w)\big)\\
&\quad - a_{uu}\big(u;S(u),T(u)\big)(v)(w) + l_{uu}\big(u;T(u)\big)(v)(w).
\end{aligned}$$

First, the variational equation (3.10) of the adjoint derivative $\hat{p} = T'(u)(w)$ is rewritten as

$$a(u;\eta,\hat{p}) = H_2(u;\eta,w) + H_3(u;\eta,\hat{y}) \quad \forall \eta \in \mathcal{V}. \tag{3.13}$$

Moreover, the second Fréchet derivative of the cost function (3.11) can be written in the shortened form

$$\begin{aligned}
F''(u)(v)(w) &= H_1(u;\hat{p},v) + H_4(u;v,w) + H_{\mathcal{T}}(v,w)\\
&\quad + J_{uy}\big(u,S(u),z(u)\big)(v)\big(S'(u)(w)\big)\\
&\quad + J_{zy}\big(u,S(u),z(u)\big)\big(z'(u)(v)\big)\big(S'(u)(w)\big)\\
&\quad - a_u\big(u;S'(u)(w),T(u)\big)(v).
\end{aligned}\tag{3.14}$$

Looking at this formulation, one recognizes that $H_2(u;\hat{y},v)$ is equal to the last three lines of (3.14) if the second partial derivatives in different directions of the functional J are interchangeable. This can be done if the assumptions of the next theorem are satisfied.

THEOREM 3.7. *Let* (A1)–(A6), (B3)–(B5) *be satisfied and* $z(u) \in \mathcal{Z}$ *be twice continuously Fréchet differentiable. Then the following assertions hold.*

1. *All partial derivatives of the functional* J *up to the second order and all the operators* a_u, a_{uu}, l_u, l_{uu}, F', *and* F'' *are continuous with respect to* u.
2. *The following equalities are true:*

$$\begin{aligned}
J_{uy}(u,y,z)(v)(\hat{y}) &= J_{yu}(u,y,z)(\hat{y})(v) &&\forall v \in \mathcal{U}, \hat{y} \in \mathcal{V},\\
J_{uz}(u,y,z)(v)(\hat{z}) &= J_{zu}(u,y,z)(\hat{z})(v) &&\forall v \in \mathcal{U}, \hat{z} \in \mathcal{Z},\\
J_{yz}(u;y,z)(\hat{y})(\hat{z}) &= J_{zy}(u;y,z)(\hat{z})(\hat{y}) &&\forall \hat{y} \in \mathcal{V}, \hat{z} \in \mathcal{Z},\\
a_{uu}(u;\cdot,\cdot)(v)(w) &= a_{uu}(u;\cdot,\cdot)(w)(v) &&\forall v,w \in \mathcal{U},\\
l_{uu}(u;\cdot,\cdot)(v)(w) &= l_{uu}(u;\cdot,\cdot)(w)(v) &&\forall v,w \in \mathcal{U},\\
F''(u)(v)(w) &= F''(u)(w)(v) &&\forall v,w \in \mathcal{U}.
\end{aligned}$$

3. *If in addition the second derivatives of the coefficient functions are Lipschitz continuous (i.e.,* (A7) *holds) and $z''(u)$, as well as all partial derivatives of J, are Lipschitz continuous, too, then the operators $S'(\cdot)$, $T'(\cdot)$, and the second derivative of the cost function are also Lipschitz continuous.*

Proof. Since this can be shown easily with Lemma 3.3, the proof is omitted here. □

The second part of the last theorem lists the required assumptions for yielding a symmetric $F''(u)$. In comparison to Theorem 3.6, which states the second derivative of the cost function, only (B5) and the continuity assumption of $z''(u)$, as well as of all partial derivatives of J, are added for this symmetry result. But these are very weak assumptions, since either way the functions have to be assumed to be Lipschitz continuous to obtain the q-quadratic convergence rate.

Finally, supposing that all the mentioned assumptions of Theorem 3.7 are fulfilled, the second derivative of the cost function (3.14) further reduces to

$$F''(u)(v)(w) = H_1(u;\hat{p},v) + H_2(u;\hat{y},v) + H_4(u;v,w) + H_{\mathcal{T}}(v,w), \tag{3.15}$$

and the computation of the second derivative can be summarized in a short description.

As in Algorithm 3.5 used to compute the negative first derivative of the cost function, the derivatives of the state (3.12) and of the adjoint (3.13) have to be computed as the solutions of variational equations with the same bilinear form. This must be done before the left-hand term of Newton's equation can be evaluated. So once again, these steps have to be done sequentially.

Algorithm 3.8 (second derivative in function space).

1. *Compute the derivative of the state $\hat{y} = S'(u)(w)$ as the solution of*

$$a(u;\hat{y},\eta) = H_1(u;\eta,w) \quad \forall \eta \in \mathcal{V}.$$

2. *Compute the derivative of the adjoint $\hat{p} = T'(u)(w)$ as the solution of*

$$a(u;\eta,\hat{p}) = H_2(u;\eta,w) + H_3(u;\eta,\hat{y}) \quad \forall \eta \in \mathcal{V}.$$

3. *Compute the second derivative of the cost function in the direction w as*

$$F''(u)(v)(w) = H_1(u;\hat{p},v) + H_2(u;\hat{y},v) + H_4(u;v,w) + H_{\mathcal{T}}(v,w).$$

The auxiliary operators introduced above not only make a shortened notation possible, but they also lead to advantageous implementational aspects. In Newton's method the operator $H_1(u;\cdot,\cdot)$ appears not only in the computation of the right-hand term (see Algorithm 3.5), but also as the linear functional $H_1(u;\cdot,w)$ in the computation of the state derivative (3.12), and directly in the computation of the second derivative as $H_1(u;\hat{p},v)$. Analogously, $H_2(u;\cdot,w)$ is a part of the linear functional for the adjoint derivative (3.13), and it is also directly used as $H_2(u;\hat{y},v)$ for the computation of the second derivative. In addition, in each iteration of a linear iterative solver, the auxiliary operators have to be evaluated in different directions. In the following, the discretization of the algorithm leads to sparse matrices H_1, H_2, H_3, H_4, and $H_{\mathcal{T}}$, which make efficient storage possible. Thus, the handling of the auxiliary operators results in the multiplication of various sparse stored matrices with different vectors.

After all these theoretical investigations, the theorem of Newton's q-quadratic convergence rate is adapted for the problems under consideration.

THEOREM 3.9. *Let* (A1)–(A7), (B3)–(B5) *be satisfied. Let $z(u) \in \mathcal{Z}$ be twice continuously Fréchet differentiable in an open neighborhood U of the solution u_*, and $z''(u)$, as well as all partial derivatives of J, be Lipschitz continuous, too. Furthermore, assume that the existence of $F''(u_*)^{-1} \in \mathcal{L}(\mathcal{L}(\mathcal{U},\mathbb{R}),\mathcal{U})$ and of the solution u_* in the interior of $\mathcal{U}_{ad}$ are guaranteed.*

Then there exists a $\delta > 0$, such that the iterative sequence $\{u_n\}$ generated by (3.1) *converges to the solution u_* for every initial value u_0 with $\|u_0 - u_*\|_{\mathcal{U}} \leq \delta$. Moreover, there is a suitable constant C depending on U and on the existing Lipschitz constant L of the second derivative of the cost function, such that*

$$\|u_n - u_*\|_{\mathcal{U}} \leq C\|u_{n-1} - u_*\|^2_{\mathcal{U}}. \tag{3.16}$$

Proof. The theorem is stated in such a way that all the assumptions required for the last theorem are fulfilled. Hence, Newton's method is well defined, $F''(u)$ is Lipschitz continuous with a constant L, and the desired assertion is derived as an application of the general theorem of Newton's q-quadratic convergence rate [40, p. 208]. □

4. Discretization of Newton's method. The derived infinite dimensional Newton's method is discretized by using the finite element method. This discretization procedure inheres the advantage that the behavior of the discretized and the infinite dimensional iterates can be compared. Besides theoretical aspects, this would also lead to an attractive way to speed up the numerical algorithm by using a kind of nested iteration. We also proved a modified mesh independence behavior of Newton's method for these optimal shape design problems; however, since these aspects would be beyond the scope of this paper it will be published separately [29].

Using the finite element method with arbitrary element functions for the discretization means the replacement of the infinite dimensional spaces $\mathcal{V}$ and $\mathcal{U}$ by the finite dimensional subspaces $\mathcal{V}^N$ and $\mathcal{U}^M$. The finite element method is used for the solution of the different variational equations in $\Omega \subset \mathbb{R}^2$, as well as for the calculation of Newton's equation in $I \subset \mathbb{R}$. To keep matters simple, a triangulation is always used for the state discretization and all explanations are based on simple linear spline basis functions. However, other kinds of partitions and element functions would not change anything essentially in the theory and could be done in a similar way.

So let $\{\phi_\tau\}_{\tau=1}^N := \{\phi_\tau(x_1,x_2)\}_{\tau=1}^N$, $\phi_\tau \in \mathcal{V}^N$, be two-dimensional spline basis functions, which are used to solve the distinct variational equations. Then, the functions defined on $\Omega \subset \mathbb{R}^2$ are given by the following linear combinations of the basis functions:

two-dimensional test function, $\quad \eta^N(x_1,x_2) = \sum_{\tau=1}^N \eta_\tau^N \phi_\tau(x_1,x_2),$

state function, $\quad y^N(x_1,x_2) = \sum_{\mu=1}^N y_\mu^N \phi_\mu(x_1,x_2),$

adjoint function, $\quad p^N(x_1,x_2) = \sum_{\nu=1}^N p_\nu^N \phi_\nu(x_1,x_2),$

derivative of the state function, $\quad \hat{y}^N(x_1,x_2) = \sum_{\hat{\mu}=1}^N \hat{y}_{\hat{\mu}}^N \phi_{\hat{\mu}}(x_1,x_2),$

derivative of the adjoint function, $\hat{p}^N(x_1, x_2) = \sum_{\hat{\nu}=1}^{N} \hat{p}_{\hat{\nu}}^N \phi_{\hat{\nu}}(x_1, x_2)$.

In addition, the discretization of the functions u, v, and $w \in \mathcal{U}$, resulting from the transformed moving part Γ^M of the boundary, is needed. Therefore, let $\{\psi_k\}_{k=1}^M := \{\psi_k(x_2)\}_{k=1}^M, \psi_k \in \mathcal{U}^M$, be the one-dimensional spline basis functions that lead to the following discretization of the one-dimensional functions mentioned above:

one-dimensional test function, $v^M(x_2) = \sum_{i=1}^{M} v_i^M \psi_i(x_2)$,

step function, $w^M(x_2) = \sum_{j=1}^{M} w_j^M \psi_j(x_2)$,

domain function, $u^M(x_2) = \sum_{k=1}^{M} u_k^M \psi_k(x_2)$.

In practice, the dimension M will always be chosen to be equal to $2^{-l}\sqrt{N} - 2^{-l} + 1$, $l \in \mathbb{N}_0$. Moreover, the elements will be defined in such a way that the overlapping parts of the supports of the one-dimensional and two-dimensional element functions, with respect to the x_2 component, is minimized. This lays the foundation for the sparsity pattern of the discretization matrices.

Throughout this report, the arguments (x_1, x_2) and (x_2) will be omitted for better readability of the equations. Nevertheless, the fact that the functions u, v, w, and ψ are defined on the one-dimensional interval I, depending only on x_2, is emphasized. Aside from that, a vector consisting of the N or M basis coefficients is always denoted with an arrow on top of the same letter as the coefficient, for example, $\vec{y}^N = (y_1^N, \dots, y_N^N)^T$.

With this notation, the infinite dimensional Newton's equation (3.1) is rewritten as the finite dimensional equation

$$F_N''\big(u^M\big)\big(v^M\big)\big(w^M\big) = -F_N'\big(u^M\big)\big(v^M\big) \quad \forall v^M \in \mathcal{U}^M,$$

which is equivalent to

$$\sum_{j=1}^{M} w_j^M F_N''\big(u^M\big)(\psi_i)(\psi_j) = -F_N'\big(u^M\big)(\psi_i) \quad \forall i = 1, \dots, M.$$

If the Hessian matrix $H \in \mathbb{R}^{M \times M}$ and the vector $d \in \mathbb{R}^M$ are denoted by

$$H = \big[F_N''\big(u^M\big)(\psi_i)(\psi_j)\big]_{i,j=1}^M$$

and

$$d = \big[-F_N'\big(u^M\big)(\psi_i)\big]_{i=1}^M,$$

then the discretized Newton's method is written in the following form.

Algorithm 4.1 (discretized Newton's method).

0. *Given* $\vec{u}^M \in \mathbb{R}^M$.
1. *Compute* $\vec{w}^M$ *as the solution of*

$$H\vec{w}^M = d.$$

2. *Set* $\vec{u}_+^M = \vec{u}^M + \vec{w}^M$.

Thus, the optimal shape design problem reduces to solving a sequence of linear systems. Discretizing and explicitly computing the Hessian matrix results in an expensive numerical method. Alternatively, an iterative method can be used to solve these linear subproblems. The symmetry of the Hessian matrix H in a region around the solution point is a trivial conclusion of Theorem 3.7. However, since F is highly nonlinear, it is not ensured that the Hessian matrix is positive definite. This case is also underlined by the observed numerical difficulties for using the BFGS-update in some examples. For this reason, the well-known CG method that requires a symmetric positive definite matrix does not work.

To overcome this drawback, the special Krylov subspace method SYMMLQ from Paige and Saunders [23], [31], which deals with an indefinite matrix, is used for the implementation. Similar to the CG method, SYMMLQ needs only the result of a matrix-vector multiplication $H\vec{w}^M$, instead of the matrix H explicitly, and further takes advantage of the matrix symmetry. Hence, it only remains to show how to compute the right-hand side vector d and the matrix-vector multiplication $H\vec{w}^M$ with an arbitrary vector $\vec{w}^M$. The required algorithms will be the discretized versions of Algorithm 3.5 for computing d and of Algorithm 3.8 for computing the matrix-vector multiplication. Hence, the negative discretized first derivative of the cost function can be computed by Algorithm 4.2, which follows.

To understand this algorithm, some auxiliary matrices resulting from the discretization of the various operators introduced in the previous section must be explained. First, the two-dimensional stiffness matrix $A \in \mathbb{R}^{N\times N}$ of the elliptic variational equations, given by

$$A = [a(u;\phi_\mu,\phi_\tau)]_{\mu,\tau=1}^N,$$

consists of the typical sparse structure that is worked out in every standard finite element book. Corresponding to the symmetry property of the underlying bilinear form the stiffness matrix is also symmetric or nonsymmetric. Various sparse storing techniques have been developed for this kind of matrix to minimize the required amount of storage. To exploit this sparsity pattern, the linear system has to be solved by an iterative method, for instance, by Krylov subspace methods with preconditioning.

Algorithm 4.2 (computation of Newton's right-hand-side vector).

1. *Compute the state* y^N *as the solution of*

$$A\vec{y}^N = l^y$$

with

$$l^y = \big[l\big(u^M;\phi_\tau\big)\big]_{\tau=1}^N.$$

2. *Compute the adjoint* p^N *as the solution of*

$$A^T\vec{p}^N = l^p$$

with

$$l^p = \big[J_y\big(u^M;y^N,z\big(u^M\big)\big)(\phi_\tau)\big]_{\tau=1}^N.$$

3. *Compute the right-hand side vector of Newton's equation*

$$d = \big[-J_u\big(u^M,y^N,z\big(u^M\big)\big)(\psi_i) - J_z\big(u^M,y^N,z\big(u^M\big)\big)\big(z'\big(u^M\big)(\psi_i)\big)\big]_{i=1}^M$$
$$- H_1\vec{p}^N - H_{\mathcal{T}}\big(\vec{u}^M - \vec{u}_{\mathcal{T}}^M\big).$$

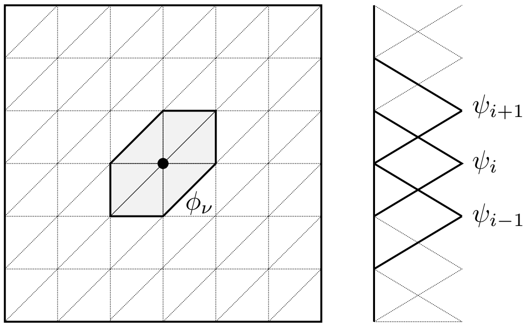


FIG. 2. *Triangulation of the two-dimensional domain* Ω *with the corresponding one-dimensional finite element functions.*

In addition, the auxiliary matrix $H_1 \in \mathbb{R}^{M\times N}$, and the matrices $H_2 \in \mathbb{R}^{M\times N}$, $H_3 \in \mathbb{R}^{N\times N}$, $H_4 \in \mathbb{R}^{M\times M}$, and $H_\mathcal{T} \in \mathbb{R}^{M\times M}$ used later, also result from the discretization of the corresponding operators that were introduced in the previous section:

$$\begin{aligned}
H_1 &:= \left[H_1\left(u^M;\phi_\nu,\psi_i\right)\right]_{\substack{i\,=\,1,\dots,M\\ \nu\,=\,1,\dots,N}},\\
H_2 &:= \left[H_2\left(u^M;\phi_\tau,\psi_j\right)\right]_{\substack{j\,=\,1,\dots,M\\ \tau\,=\,1,\dots,N}},\\
H_3 &:= \left[H_3\left(u^M;\phi_\tau,\phi_{\hat\mu}\right)\right]_{\substack{\tau\,=\,1,\dots,N\\ \hat\mu\,=\,1,\dots,N}},\\
H_4 &:= \left[H_4\left(u^M;\psi_i,\psi_j\right)\right]_{\substack{i\,=\,1,\dots,M\\ j\,=\,1,\dots,M}},\\
H_\mathcal{T} &:= \left[H_\mathcal{T}\left(\psi_i,\psi_j\right)\right]_{\substack{i\,=\,1,\dots,M\\ j\,=\,1,\dots,M}}.
\end{aligned}$$

Computing these auxiliary matrices every time they occur will increase the computing time, so that the algorithm slows down enormously. Therefore, it must be decided if it is acceptable to store these matrices explicitly at each nonlinear iteration. Since the two-dimensional stiffness matrix A, as well as the symmetric matrix H_3, result from the integration of two two-dimensional element functions, their sparsity patterns consist of the same structure. The same arguments hold for the matrices H_4 and $H_\mathcal{T}$ resulting from two one-dimensional finite element functions. Both consist of the same tridiagonal sparsity pattern as the usual symmetric one-dimensional stiffness matrix, and the matrix $H_\mathcal{T}$ is also always symmetric. In addition to this special structure, these matrices are only defined in $\mathbb{R}^{M\times M}$, where M, resulting from the one-dimensional discretization, will usually be less than or equal to $\sqrt{N}$.

The remaining matrices H_1 and H_2, which are based on one one-dimensional function and one two-dimensional basis function, also consist of a specific sparse structure. On the left-hand side of Figure 2, a regular triangulation of the domain Ω is given with an arbitrary two-dimensional finite element function ϕ_ν, where its support $S_{\phi_\nu} := \{(x_1,x_2) \mid \phi_\nu(x_1,x_2) \geq 0\}$ is hatched. On the right-hand side, the one-dimensional element functions, where $M = \sqrt{N}$, are sketched. The node $(x_1^\nu, x_2^\nu) \in \Omega$ is uniquely defined by $\phi_\nu(x_1^\nu,x_2^\nu) = 1$, and the one-dimensional node $x_2^i \in I$ is determined by $\psi_i(x_2^i) = 1$.

It is easy to see that the entries of the matrices H_1 and H_2, with respect to ϕ_ν, are always zero, unless x_2^ν is equal to x_2^i or to one of its direct neighbors, x_2^{i-1} and x_2^{i+1}.

Consequently, using the discretization given by $M = \sqrt{N}$, for each of the N two-dimensional functions ϕ_ν, at most three nonzero entries have to be stored. If the one-dimensional grid is coarser than the two-dimensional one, i.e., if $M = 2^{-l}\sqrt{N} - 2^{-l} + 1$, $l > 0$, it is easily verified that only two entries have to be stored for every ϕ_ν.

To summarize, each of these latter matrices can be stored in an $N \times 3$, possibly an $N \times 2$, matrix, if a pointer indicates where the one-dimensional neighbors of a two-dimensional node, with respect to the x_2 component, are stored. This pointer is implemented in a convenient way to evaluate matrix-vector multiplications with the transpose of the matrix, as well as with the matrix itself. Both kinds of multiplications are necessary to compute the right-hand side of Newton's equation and to compute the matrix-vector multiplication in each iteration of the linear solver SYMMLQ.

To sum up, at most $2(3N) + 5\sqrt{N} - 3$ numbers have to be stored for H_1, H_2, H_4, and $H_{\mathcal{T}}$, and two stiffness matrix storages, one for H_3 and one for the stiffness matrix A, are needed. Under these circumstances it is reasonable to store the auxiliary matrices.

Finally, the discretized version of Algorithm 3.8 for computing the matrix-vector multiplication for SYMMLQ completes the discretization. Once again, it should be emphasized that these steps must be done sequentially, and each computation of a variational equation in the infinite dimensional setting results in solving a linear system with the same stiffness matrices A and A^T, respectively.

ALGORITHM 4.3 (Hessian matrix multiplied by a vector $\vec{w}^M$).

1. *Compute the derivative of the state $\hat{y}^N$ as the solution of*

$$A\vec{\hat{y}}^N = l^{\hat{y}}$$

with

$$l^{\hat{y}} = H_1^T \vec{w}^M.$$

2. *Compute the derivative of the adjoint $\hat{p}^N$ as the solution of*

$$A^T \vec{\hat{p}}^N = l^{\hat{p}}$$

with

$$l^{\hat{p}} = H_2^T \vec{w}^M + H_4 \vec{\hat{y}}^N.$$

3. *Compute the Hessian matrix multiplied by a vector $\vec{w}^M$ as*

$$H\vec{w}^M = H_1 \vec{\hat{p}}^N + H_2 \vec{\hat{y}}^N + H_4 \vec{w}^M + H_{\mathcal{T}} \vec{w}^M.$$

At this point, attention should be drawn to the fact that the work for computing one matrix-vector multiplication $H\vec{w}^M$ is essentially determined by the operations required to solve two linear systems, one for the derivative of the state and the other one for the derivative of the adjoint.

5. Numerical results. In this section a comprehensive comparison will illustrate the superiority of Newton's method over first order numerical methods. The linear iteration for solving each linearized optimal control problem has been realized by the algorithm SYMMLQ with the stopping criterion

$$\|H\vec{w}^M - d\|_2 \leq \text{TOL}_{\text{SYMMLQ}} = 10^{-9}.$$

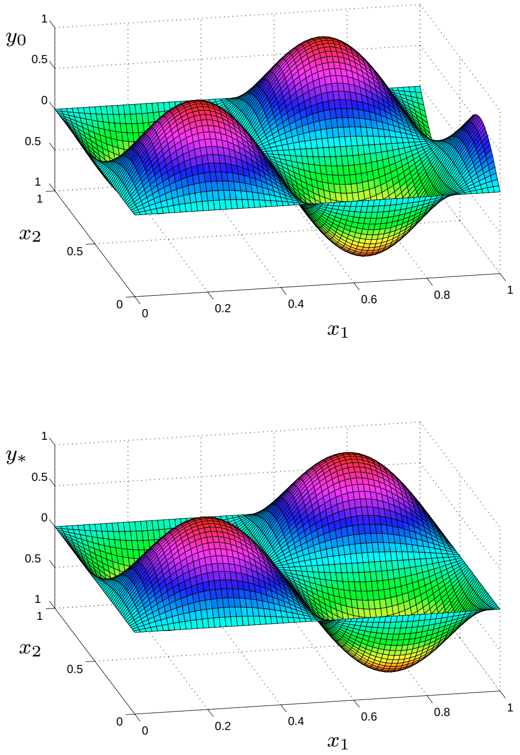


Fig. 3. *Initial and optimal states.*

Since for our example the variational equations consist of a symmetric bilinear form, the CG method with a hierarchical and a diagonal preconditioner is implemented to accelerate the expected convergence rate. This iterative method terminates if the $\mathcal{L}^2(\Omega)$ norm of the residual is less than or equal to the constant $\text{TOL}_{\text{cg}} = 10^{-11}$. These stopping criterions for the subproblems are rather small in order to exclude the influence of these errors on the observed convergence rate. However, the algorithm could be accelerated enormously by taking advantage of the inexact Newton's method concept [15].

For all numerical experiments, the Tikhonov regularization term is implemented with $u_{\mathcal{T}} \equiv 0$ and $\varepsilon = 10^{-4}$. The nonlinear iteration is stopped if the norm of the gradient F'_N is less than 10^{-8}. Furthermore, to compare the different methods, the discretization parameters are chosen to be $M = 65$ and $N = 65^2$. All presented computations are done on a SUNSparcstation 20 in double precision FORTRAN.

We tested our codes with several different examples and various parameter selections. Since it is impossible to present all of them within the limitations of this report, we restrict ourselves to the example of section 2. In this paper we are only interested in the local convergence behavior in order to support the presented theory. Therefore, globalization aspects are omitted by starting close to the solution ($u_0 \equiv 1.1$, $y_0 = \sin(2u_0\pi x_1)\sin(2\pi x_2)$). For clarity, the function y_0 and $z \equiv y_*$ are drawn in Figure 3.

Table 1
Gradient method without any line search technique.

It	Time	$\left\|w^M\right\|$	$F_N\left(u_i^M\right)$	$\left\|F_N'\left(u_i^M\right)\right\|$	$\left\|u_i^M - u_*\right\|$
0	7.87	$0.000E+00$	$0.411E-01$	$0.167E-01$	$0.100E+00$
100	718.68	$0.124E-03$	$0.137E-03$	$0.120E-03$	$0.252E-01$
500	3326.47	$0.206E-04$	$0.557E-04$	$0.187E-04$	$0.581E-02$
1000	6349.57	$0.370E-05$	$0.517E-04$	$0.336E-05$	$0.104E-02$
1500	9233.63	$0.690E-06$	$0.516E-04$	$0.626E-06$	$0.520E-03$
2000	11989.68	$0.130E-06$	$0.516E-04$	$0.117E-06$	$0.546E-03$
2500	14623.33	$0.244E-07$	$0.516E-04$	$0.220E-07$	$0.556E-03$
2737	15825.93	$0.110E-07$	$0.516E-04$	$0.997E-08$	$0.557E-03$

Table 2
Gradient method.

It	Time	$\left\|w^M\right\|$	$F_N\left(u_i^M\right)$	$\left\|F_N'\left(u_i^M\right)\right\|$	$\left\|u_i^M - u_*\right\|$
0	7.87	$0.000E+00$	$0.411E-01$	$0.167E-01$	$0.100E+00$
50	1146.71	$0.385E-03$	$0.548E-04$	$0.386E-04$	$0.492E-02$
100	2205.26	$0.582E-04$	$0.517E-04$	$0.496E-05$	$0.816E-03$
150	3216.18	$0.751E-05$	$0.516E-04$	$0.664E-06$	$0.525E-03$
200	4167.96	$0.101E-05$	$0.516E-04$	$0.932E-07$	$0.551E-03$
250	5071.99	$0.282E-06$	$0.516E-04$	$0.297E-07$	$0.557E-03$
265	5333.37	$0.892E-07$	$0.516E-04$	$0.844E-08$	$0.558E-03$

Although the differentiation of the expressions seems to be rather time consuming, the derivatives could be derived within an appropriate time frame by using symbolic differentiation, supplied by standard software. In addition, the implementation of these functions can be simplified by splitting up the computation into many subproblems, as explained by Chenais [11] and Chenais, Rousselet, and Benedict [12].

For this example, the convergence behavior of the gradient method without any line search techniques is illustrated in Table 1. The first column gives the number of the nonlinear iterations. Then, the accumulative time, measured in seconds, for all the previous iterations are specified. Right next to it, the $\mathcal{L}^2(I)$ norm of the step, the value of the cost function, the $\mathcal{L}^2(\Omega)$ norm of the gradient, and the $\mathcal{L}^2(I)$ norm of the control error $u_i - u_*$ are tabulated. Note that u_i is the finite ith iteration, whereas u_* is the solution of the infinite dimensional problem without regularization. On account of this, the q-quadratic convergence rate of these values cannot be expected.

This method apparently converges very slowly to the local solution. For example, after the thousandth iteration, the distance $u_i - u_*$ is still greater than 10^{-3}. This behavior has to be expected since this method is known to converge slowly but with fewer operations per iteration. In this example, each iteration is done within 5 to 8 seconds, and more than 6350 seconds are needed to reduce the error of the control under the bound 10^{-3}.

Table 2 documents the convergence of the implemented gradient method with the Armijo rule [26] for the specific parameters $\sigma = .1$ and $\varsigma = .5$, which have been selected

Table 3
Newton's method.

It	It/Sym	Time	$\left\|w^M\right\|$	$F_N\left(u_i^M\right)$	$\left\|F_N'\left(u_i^M\right)\right\|$	$\left\|u_i^M - u_*\right\|$
0	0	9.98	$0.000E+00$	$0.411E-01$	$0.167E-01$	$0.100E+00$
1	101	394.59	$0.988E-01$	$0.759E-04$	$0.245E-03$	$0.659E-02$
2	70	653.61	$0.543E-02$	$0.518E-04$	$0.620E-05$	$0.149E-02$
3	51	840.44	$0.110E-02$	$0.516E-04$	$0.865E-06$	$0.623E-03$
4	42	995.69	$0.163E-03$	$0.516E-04$	$0.202E-06$	$0.565E-03$
5	36	1130.51	$0.257E-04$	$0.516E-04$	$0.599E-07$	$0.559E-03$
6	32	1250.86	$0.480E-05$	$0.516E-04$	$0.230E-07$	$0.559E-03$
7	25	1346.70	$0.104E-05$	$0.516E-04$	$0.102E-07$	$0.559E-03$
8	25	1442.44	$0.349E-06$	$0.516E-04$	$0.513E-08$	$0.559E-03$

by numerical experiences. The tabulated results for this method are superior to the latter one. Already after the hundredth iteration, the difference $u_i - u_*$ is less than 10^{-3} with the same cost function value that is achieved by the method without line search technique after the thousandth iteration. Since each cost function evaluation for the line search rule needs the solution of a further variational equation to obtain the state with respect to the current control, an increased number of operations is required for this method. However, although 16 to 31 seconds are needed for each iteration, the method is far more efficient. Thus, only about a third of the time is required for a similar reduction of the control error in the hundredth iteration.

The improved performance is due to a globalization effect. The first iterations of the gradient method yield good descent directions such that the control error in the 50th iteration remains far under the bound 10^{-2}. However, 265 iterations are further necessary to satisfy the stopping criterion with respect to the gradient.

Finally, Table 3 presents the iterates of Newton's method. The second column is added here to present the required iterations of the SYMMLQ algorithm. After the eighth iteration, the stopping criterion is fulfilled, the control error does not change after the fifth iteration, and the optimal value of the control value is already achieved after the third iteration. This fast convergence rate has to be put in the perspective of the increased computing time per iteration. However, Newton's method only needs 840 seconds to remain under the 10^{-3} bound for the control error in the third iteration. This is more than 7 times faster than the gradient method without line search technique and 2.5 times faster than the method with Armijo rule.

This comparison is based only on the control error. The ratio of the performances of the distinct methods is even improved if we look at the gradient. Newton's method requires 1442 seconds to satisfy the stopping criterion, and therefore it is about 11 times faster than the gradient method without line search technique and more than 3.5 times faster than the globalized gradient method.

The discretization error apparently influences the convergence rate. Thus, although the q-quadratic convergence rate of Newton's method in the infinite dimensional setting cannot be directly observed, the fast reduction of the gradient norm and the cost function within the first two iterations points to such a convergence property. After the second iteration, the convergence rate is dominated by the large discretization error. Nevertheless, Newton's method is apparently superior to the others.

Analyzing the profile of each method shows that Newton's method requires almost all time for solving the linear systems of the variational equations with the CG method. In contrast to this, the gradient method spends as much time solving the variational equation with the CG method as it does evaluating the various functions. Therefore, using smaller stopping criteria for the subproblems will further increase the gap of efficiency between these methods.

For numerical results on nested iteration for this class of problems, we refer to the mesh independence analysis [29] and present only some results of a rudimentary implementation. Four levels, $M = 9, 17, 33, 65$, with $N = M^2$, are used, where each approximation is transferred to the finer grid by linear interpolation (see Table 4). The regularization parameter is decreased with respect to the different levels in order to handle the ill-posedness of the problem.

To illustrate the applicability of the derived algorithm, the starting approximation is chosen to be $u_0 = 1.8$, which is far away from the solution u_*. Thus, a further constraint $u \geq \beta_1$ with $\beta_1 = 0.3$ has to be added, since otherwise the control would become negative, which makes no sense for the domain. For simplicity, the projected gradient method with the projected Armijo rule [10] is implemented on the coarse grid. Other kinds of unconstrained minimization algorithms could also be used for this task. However, since an appropriate approximation is computed on the coarse grid within a few seconds, the efficiency of the entire method is not essentially influenced. The stopping criteria are kept on the coarse grid for getting a good approximation without consuming too much computing time.

After these iterations on the coarse grid, the approximation is interpolated to the finer grid to carry out five Newton iterations. Then, the approximation is interpolated to the next finer grid and once again improved by five Newton iterations. This process proceeds until the finest grid with $M = 65$ is reached in order to compare the results with the tables of the presented methods without nested iteration.

Although the starting point is now further away from the solution, this nested iteration method is only about 140 seconds slower than Newton's method with the starting point $u_0 = 1.1$. The final value $F_N(u_i^M) = 0.516E-04$ is reached already after 529 seconds while Newton's method requires more than 840 seconds. It can also be observed that the control error at each level is unchanged at the last few iterations. Therefore, the refinement strategy could even be improved by using a modified stopping criterion at each level. However, even this rudimentary nested iteration and globalization technique illustrate the efficiency of the presented method.

To sum up, it can be concluded from the computational experiments that it is possible to derive more efficient numerical methods for optimal shape design problems than the often-used gradient method. The key to constructing such attractive methods is to exploit the problem-specific structure and to take advantage of second order information about the cost function.

Appendix A. Proof of Theorem 3.2. The difference between the state and the adjoint variational equations is caused by the interchanged position of the bilinear form. Since this bilinear form enters the proof only in the Cauchy–Schwarz inequality, both assertions are similarly proven. For this reason, we restrict ourselves to the proof of the state assertion, which is divided into four parts. First, Theorem 3.1 is used to prove the existence and uniqueness of the solution $\hat{y}^v$. Then the Lipschitz continuity of the operator S, the inequality of the Fréchet differentiability definition, and afterwards, the linearity and continuity of the Fréchet derivative are proven.

1. For $a_{ij}(u) \in \mathcal{L}^\infty(\Omega)$ and $b(u) \in \mathcal{L}^\infty(\Gamma_1)$ we have seen above that the bilinear

Table 4
Nested iteration of Newton's method ($u \geq 0.3$, $u_0 = 1.8$).

It	Time$_\Sigma$	$\left\|w^M\right\|$	$F_N\left(u_i^M\right)$	$\left\|F_N'\left(u_i^M\right)\right\|$	$\left\|u_i^M - u_*\right\|$
$M = 9, \varepsilon = 10^{-1}$ (Projected gradient method)					
0	0.11	$0.000E+00$	$0.956E+00$	$0.605E-01$	$0.800E+00$
10	1.67	$0.207E+00$	$0.293E+00$	$0.135E+00$	$0.274E+00$
20	3.06	$0.279E-02$	$0.542E-01$	$0.433E-02$	$0.267E-01$
30	4.32	$0.475E-04$	$0.541E-01$	$0.656E-04$	$0.284E-01$
40	5.52	$0.363E-05$	$0.541E-01$	$0.552E-05$	$0.285E-01$
50	6.71	$0.751E-07$	$0.541E-01$	$0.107E-06$	$0.285E-01$
60	8.53	$0.235E-08$	$0.541E-01$	$0.295E-07$	$0.285E-01$
62	8.83	$0.473E-08$	$0.541E-01$	$0.919E-08$	$0.285E-01$
$M = 17, \varepsilon = 10^{-2}$ (Newton's method)					
0	9.54	$0.000E+00$	$0.706E-02$	$0.102E-01$	$0.285E-01$
1	12.01	$0.260E-01$	$0.540E-02$	$0.105E-02$	$0.473E-02$
2	14.47	$0.215E-02$	$0.539E-02$	$0.635E-04$	$0.554E-02$
3	16.81	$0.113E-03$	$0.539E-02$	$0.261E-04$	$0.549E-02$
4	18.91	$0.272E-04$	$0.539E-02$	$0.123E-04$	$0.548E-02$
5	20.90	$0.832E-05$	$0.539E-02$	$0.611E-05$	$0.547E-02$
$M = 33, \varepsilon = 10^{-3}$ (Newton's method)					
0	22.28	$0.000E+00$	$0.563E-03$	$0.752E-03$	$0.547E-02$
1	46.85	$0.392E-02$	$0.526E-03$	$0.281E-04$	$0.181E-02$
2	69.28	$0.330E-03$	$0.526E-03$	$0.738E-05$	$0.168E-02$
3	88.88	$0.515E-04$	$0.526E-03$	$0.364E-05$	$0.168E-02$
4	107.75	$0.137E-04$	$0.526E-03$	$0.213E-05$	$0.168E-02$
5	125.19	$0.499E-05$	$0.526E-03$	$0.133E-05$	$0.168E-02$
$M = 65, \varepsilon = 10^{-4}$ (Newton's method)					
0	134.56	$0.000E+00$	$0.538E-04$	$0.820E-04$	$0.168E-02$
1	360.82	$0.114E-02$	$0.517E-04$	$0.409E-05$	$0.693E-03$
2	528.86	$0.342E-03$	$0.516E-04$	$0.114E-05$	$0.547E-03$
3	678.52	$0.625E-04$	$0.516E-04$	$0.463E-06$	$0.564E-03$
4	827.88	$0.269E-04$	$0.516E-04$	$0.229E-06$	$0.557E-03$
5	945.28	$0.929E-05$	$0.516E-04$	$0.128E-06$	$0.559E-03$
6	1058.88	$0.433E-05$	$0.516E-04$	$0.795E-07$	$0.558E-03$
7	1165.30	$0.173E-05$	$0.516E-04$	$0.530E-07$	$0.559E-03$
8	1261.01	$0.816E-06$	$0.516E-04$	$0.370E-07$	$0.558E-03$
9	1353.00	$0.390E-06$	$0.516E-04$	$0.266E-07$	$0.559E-03$
10	1430.53	$0.210E-06$	$0.516E-04$	$0.196E-07$	$0.559E-03$
11	1501.06	$0.128E-06$	$0.516E-04$	$0.146E-07$	$0.559E-03$
12	1546.82	$0.827E-07$	$0.516E-04$	$0.111E-07$	$0.559E-03$
13	1581.78	$0.571E-07$	$0.516E-04$	$0.844E-08$	$0.559E-03$

form is continuous. The linearity of

$$l_u(u;\eta)(v) - a_u(u;y,\eta)(v)$$

with respect to η is easily proven, and the continuity follows since

$$\begin{aligned}
&|l_u(u;\eta)(v) - a_u(u;y,\eta)(v)| \\
&\qquad \le \sum_{|i|,|j|\le 1} \|a'_{ij}(u)(v)\|_{\mathcal{L}^\infty(\Omega)}\|D^i y\|_{\mathcal{L}^2(\Omega)}\|D^j\eta\|_{\mathcal{L}^2(\Omega)} \\
&\qquad\quad + \|b'(u)(v)\|_{\mathcal{L}^\infty(\Gamma_1)}\|y\|_{\mathcal{L}^2(\Gamma_1)}\|\eta\|_{\mathcal{L}^2(\Gamma_1)} + \|f'(u)(v)\|_{\mathcal{V}'}\|\eta\|_{\mathcal{V}} \\
&\qquad \le c\|\eta\|_{\mathcal{V}}.
\end{aligned}$$

The assumptions of Theorem 3.1 are now satisfied. Hence, there exists a unique solution $\hat{y}^v$ of the variational equation (3.4).

2. Here the Lipschitz continuity of the operator S is proven. Let $y = S(u)$ be the solution of the state equation

$$a(u;y,\eta) = l(u;\eta) \quad \forall \eta \in \mathcal{V}, \tag{A.1}$$

and for any $u + v \in \mathcal{U}_{ad}$, let the function $\bar{y} = S(u+v)$ be the solution of the perturbed variational equation

$$a(u+v;\bar{y},\eta) = l(u+v;\eta) \quad \forall \eta \in \mathcal{V}, \tag{A.2}$$

where existence and uniqueness follow from the previous theorem. The state equation (A.1) can be subtracted from the previous one, yielding

$$a(u+v;\bar{y},\eta) - a(u;y,\eta) = l(u+v;\eta) - l(u;\eta) \quad \forall \eta \in \mathcal{V}.$$

Using the linearity of the bilinear form and subtracting the term $a(u+v;y,\eta)$ from both sides, the last equality is written as

$$a(u+v;\bar{y}-y,\eta) = a(u;y,\eta) - a(u+v;y,\eta) + l(u+v;\eta) - l(u;\eta) \quad \forall \eta \in \mathcal{V}.$$

Since the coefficient functions are Fréchet differentiable, they are also Lipschitz continuous. Hence, the boundedness for a given y is obtained from

$$\begin{aligned}
&|a(u{+}v;\bar{y}-y,\eta)| \\
&\qquad \le |\sum_{|i|,|j|\le 1} \langle (a_{ij}(u) - a_{ij}(u+v))D^i y, D^j\eta\rangle_{\mathcal{L}^2(\Omega)}| \\
&\qquad\quad + |\langle (b(u) - b(u+v))y, \eta\rangle_{\mathcal{L}^2(\Gamma_1)}| \\
&\qquad\quad + |\langle f(u) - f(u+v), \eta\rangle_{\mathcal{V}'\times\mathcal{V}}| \\
&\qquad \le \sum_{|i|,|j|\le 1} \|a_{ij}(u) - a_{ij}(u+v)\|_{\mathcal{L}^\infty(\Omega)}\|D^i y\|_{\mathcal{L}^2(\Omega)}\|D^j\eta\|_{\mathcal{L}^2(\Omega)} \\
&\qquad\quad + \|b(u) - b(u+v)\|_{\mathcal{L}^\infty(\Gamma_1)}\|y\|_{\mathcal{L}^2(\Gamma_1)}\|\eta\|_{\mathcal{L}^2(\Gamma_1)} \\
&\qquad\quad + \|f(u) - f(u+v)\|_{\mathcal{V}'}\|\eta\|_{\mathcal{V}} \\
&\qquad \le c_1\|v\|_{\mathcal{U}}\|\eta\|_{\mathcal{V}}.
\end{aligned}$$

Using again the Lipschitz continuity of the coefficient functions leads to the inequality

$$\begin{aligned}|a(u;\bar{y}-y,\eta)| &\le |a(u;\bar{y}-y,\eta)-a(u+v;\bar{y}-y,\eta)|+|a(u+v;\bar{y}-y,\eta)|\\ &\le c_2\|v\|_{\mathcal{U}}\|\bar{y}-y\|_{\mathcal{V}}\|\eta\|_{\mathcal{V}}+c_1\|v\|_{\mathcal{U}}\|\eta\|_{\mathcal{V}}.\end{aligned}$$

This holds for every $\eta \in \mathcal{V}$, especially for $\eta = \bar{y}-y$, and therefore, the $\mathcal{V}$-ellipticity

$$\begin{aligned}c_e\|\bar{y}-y\|_{\mathcal{V}}^2 &\le a(u;\bar{y}-y,\bar{y}-y)\\ &\le c_2\|v\|_{\mathcal{U}}\|\bar{y}-y\|_{\mathcal{V}}^2+c_1\|v\|_{\mathcal{U}}\|\bar{y}-y\|_{\mathcal{V}}\end{aligned}$$

implies

$$(c_e-c_2\|v\|_{\mathcal{U}})\|\bar{y}-y\|_{\mathcal{V}} \le c_1\|v\|_{\mathcal{U}}.$$

For sufficiently small $\|v\|_{\mathcal{U}}$ we finally obtain

$$\begin{aligned}\|S(u+v)-S(u)\|_{\mathcal{V}} &= \|\bar{y}-y\|_{\mathcal{V}}\\ &\le \frac{c_1}{c_e-c_2\|v\|_{\mathcal{U}}}\|v\|_{\mathcal{U}}\\ &\le c\|v\|_{\mathcal{U}}.\end{aligned}$$

3. Similarly to the part above, the Fréchet differentiability will now be proven. Subtracting the variational equation (3.4) of the expected Fréchet derivative and the variational equation (A.1) of the state equation from the perturbed equation (A.2) yields

$$\begin{aligned}a\big(u;\bar{y}-y-\hat{y}^v,\eta\big) &= -a(u+v;\bar{y},\eta)+a(u;\bar{y},\eta)+a_u(u;y,\eta)(v)\\ &\quad + l(u+v;\eta)-l(u;\eta)-l_u(u;\eta)(v) \quad \forall \eta\in\mathcal{V}.\end{aligned}$$

Making use of the Fréchet differentiability of the coefficient functions and of the Lipschitz continuity of their derivatives, as well as the Lipschitz continuity of the operator S, we get

$$\begin{aligned}&\big|a\big(u;\bar{y}-y-\hat{y}^v,\eta\big)\big|\\ &\quad\le |a(u+v;\bar{y},\eta)-a(u;\bar{y},\eta)-a_u(u;\bar{y},\eta)(v)|\\ &\qquad + |a_u(u;\bar{y},\eta)(v)-a_u(u;y,\eta)(v)|+|l(u+v;\eta)-l(u;\eta)-l_u(u;\eta)(v)|\\ &\quad\le \sum_{|i|,|j|\le 1}\langle(a_{ij}(u+v)-a_{ij}(u)-a'_{ij}(u)(v))D^i\bar{y},D^j\eta\rangle_{\mathcal{L}^2(\Omega)}\\ &\qquad + |\langle(b(u+v)-b(u)-b'(u)(v))\bar{y},\eta\rangle_{\mathcal{L}^2(\Gamma_1)}|\\ &\qquad + \left|\sum_{|i|,|j|\le 1}\langle a'_{ij}(u)(v)D^i(\bar{y}-y),D^j\eta\rangle_{\mathcal{L}^2(\Omega)}\right|\\ &\qquad + |\langle b'(u)(v)(\bar{y}-y),\eta\rangle_{\mathcal{L}^2(\Gamma_1)}|\\ &\qquad + |\langle(f(u+v)-f(u)-f'(u)(v)),\eta\rangle_{\mathcal{V}'\times\mathcal{V}}|\\ &\quad\le \sum_{|i|,|j|\le 1}\|a_{ij}(u+v)-a_{ij}(u)-a'_{ij}(u)(v)\|_{\mathcal{L}^\infty(\Omega)}\|D^i\bar{y}\|_{\mathcal{L}^2(\Omega)}\|D^j\eta\|_{\mathcal{L}^2(\Omega)}\\ &\qquad + \|b(u+v)-b(u)-b'(u)(v)\|_{\mathcal{L}^\infty(\Gamma_1)}\|\bar{y}\|_{\mathcal{L}^2(\Gamma_1)}\|\eta\|_{\mathcal{L}^2(\Gamma_1)}\\ &\qquad + \sum_{|i|,|j|\le 1}\|a'_{ij}(u)(v)\|_{\mathcal{L}^\infty(\Omega)}\|D^i(\bar{y}-y)\|_{\mathcal{L}^2(\Omega)}\|D^j\eta\|_{\mathcal{L}^2(\Omega)}\end{aligned}$$

$$
\begin{aligned}
&+ \|b'(u)(v)\|_{\mathcal{L}^\infty(\Gamma_1)} \|\bar{y} - y\|_{\mathcal{L}^2(\Gamma_1)} \|\eta\|_{\mathcal{L}^2(\Gamma_1)} \\
&+ \|f(u+v) - f(u) - f'(u)(v)\|_{\mathcal{V}'} \|\eta\|_{\mathcal{V}} \\
\le\ & \tilde{\alpha}(\|v\|_{\mathcal{U}}) \|v\|_{\mathcal{U}} \|\eta\|_{\mathcal{V}}.
\end{aligned}
$$

Here $\tilde{\alpha}(r) \to 0$ as $r \to 0$. Due to the $\mathcal{V}$-ellipticity,

$$
\begin{aligned}
c_e \|\bar{y} - y - \hat{y}^v\|_{\mathcal{V}}^2 &\le a\big(u; \bar{y} - y - \hat{y}^v, \bar{y} - y - \hat{y}^v\big) \\
&\le \tilde{\alpha}(\|v\|_{\mathcal{U}}) \|v\|_{\mathcal{U}} \|\bar{y} - y - \hat{y}^v\|_{\mathcal{V}}
\end{aligned}
$$

is obtained, and therefore

$$
\begin{aligned}
\|S(u+v) - S(u) - S'(u)(v)\|_{\mathcal{V}} &= \|\bar{y} - y - \hat{y}^v\|_{\mathcal{V}} \\
&\le \alpha(\|v\|_{\mathcal{U}}) \|v\|_{\mathcal{U}},
\end{aligned}
$$

with $\alpha(r) \to 0$ as $r \to 0$, finishes the proof.

4. The continuity of $\hat{y}^v = S'(u)(v)$ with respect to v is derived by

$$
\begin{aligned}
\|\hat{y}^v\|_{\mathcal{V}} &\le \|\bar{y} - y - \hat{y}^v\|_{\mathcal{V}} + \|y - \bar{y}\|_{\mathcal{V}} \\
&\le c\|v\|_{\mathcal{U}}.
\end{aligned}
$$

Let $\hat{y}_k^v = S'(u)(v_k)$, $k = 1, 2$, be the solution of the variational equation (3.4) with respect to v_k. Multiplying each variational equation by c_k and using the linearity properties of a, a_u, l, and l_u, the equations can be added together, resulting in

$$
a\big(u; c_1\hat{y}_1^v + c_2\hat{y}_2^v, \eta\big) = l_u(u;\eta)(c_1v_1 + c_2v_2) - a_u(u; y, \eta)(c_1v_1 + c_2v_2) \quad \forall \eta \in \mathcal{V}.
$$

Since $S'(u)(c_1v_1 + c_2v_2)$ is the unique solution of this variational equation, the linearity of the operator is obtained by

$$
\begin{aligned}
S'(u)(c_1v_1 + c_2v_2) &= c_1\hat{y}_1^v + c_2\hat{y}_2^v \\
&= c_1S'(u)(v_1) + c_2S'(u)(v_2).
\end{aligned}
$$

Acknowledgments. I would like to thank Prof. E. Sachs for drawing my attention to the research into optimal shape design problems. During the time of my research I have benefited from his encouraging support and his professional knowledge of numerical optimization. Furthermore, I wish to thank Prof. J. Sokolowski for our fruitful discussions and his valuable advice concerning various aspects of optimal shape design. Finally, I am very grateful to Dr. M. Heinkenschloß, who provided his finite element code for solving the variational subproblems.

REFERENCES

[1] R. A. Adams, *Sobolev Spaces*, Academic Press, New York, 1975.

[2] E. L. Allgower and K. Böhmer, *Application of the mesh-independence principle to mesh refinement strategies*, SIAM J. Numer. Anal., 24 (1987), pp. 1335–1351.

[3] H. T. Banks and F. Kojima, *Boundary Shape Identification Problems in Two-Dimensional Domains Related to Thermal Testing of Materials*, Technical Report 181654, NASA Contractor Report, NASA Langley Research Center, Hampton, VA, 1988.

[4] H. T. Banks and F. Kojima, *Boundary shape identification problems in two-dimensional domains related to thermal testing of materials*, Quart. Appl. Math., 47 (1989), pp. 273–293.

[5] H. T. Banks, F. Kojima, and W. P. Winfree, *Boundary shape identification problems in two-dimensional domains undary estimation problems arising in thermal tomography*, Inverse Problems, 6 (1990), pp. 897–921.

[6] H. T. Banks and K. Kunisch, *Estimation Techniques for Distributed Parameter Systems*, Systems Control Found. Appl., Birkhäuser, Boston, Basel, Berlin, 1989.

[7] G. de Barra, *Measure Theory and Integration*, Ellis Horwood, Chichester, UK, 1981.

[8] J. Baumeister, *Stable Solution of Inverse Problems*, Vieweg-Verlag, Braunschweig, Wiesbaden, 1987.

[9] D. Begis and R. Glowinski, *Application de la méthode des éléments finis à l'approximation d'un problème de domain optimal*, Appl. Math. Comput., 2 (1975), pp. 130–169.

[10] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[11] D. Chenais, *Optimal design of midsurface of shells: Differentiability proof and sensitivity computation*, Appl. Math. Optim., 16 (1987), pp. 93–133.

[12] D. Chenais, B. Rousselet, and R. Benedict, *Design sensitivity for arch structures with respect to midsurface shape under static loading*, J. Optim. Theory Appl., 2 (1988), pp. 225–239.

[13] M. Delfour, G. Payre, and J.-P. Zolésio, *Optimal design of a minimum weight thermal diffuser with constraint on the output thermal power flux*, Appl. Math. Optim., 9 (1983), pp. 225–262.

[14] M. Delfour, G. Payre, and J.-P. Zolésio, *Optimal parametrized design of thermal diffusers for communication satellites*, Optim. Control Appl. Methods., 7 (1986), pp. 163–184.

[15] R. S. Dembo, S. C. Eisenstat, and T. Steihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19 (1982), pp. 400–408.

[16] J. E. Dennis and H. F. Walker, *Convergence theorems for least-change secant update methods*, SIAM J. Numer. Anal., 18 (1981), pp. 949–987.

[17] O. Ghattas and C. E. Orozco, *A parallel reduced Hessian SQP method for shape optimization*, in SIAM Proceedings of the ICASE/NASA Langley Workshop on Multidisciplinary Design Optimization, N. M. Alexandrov and M. Y. Hussaini, eds., SIAM, Philadelphia, 1997, pp. 133–152.

[18] Y. Goto and N. Fujii, *A Newton's method in a domain optimization problem*, in Control of Boundaries and Stabilization, Proceedings of the IFIP WG 7.2 Conference, Clermont Ferrand, France, Vol. 125, J. Simon, ed., Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona, Budapest, 1988.

[19] Y. Goto and N. Fujii, *Second-order numerical method for domain optimization problems*, J. Optim. Theory Appl., 67 (1990), pp. 533–550.

[20] A. Griewank and G. F. Corliss, eds., *Automatic Differentiation of Algorithms: Theory, Implementation, and Application*, SIAM Proceedings in Applied Mathematics 53, SIAM, Philadelphia, 1991.

[21] J. Haslinger, K.-H. Hoffmann, and M. Kočvara, *Control/fictitious domain method for solving optimal shape design problems*, MAN, 27 (1993), pp. 157–182.

[22] J. Haslinger and P. Neittaanmäki, *Finite Element Approximation for Optimal Shape Design: Theory and Applications*, John Wiley, Chichester, New York, Brisbane, Toronto, Singapore, 1988.

[23] M. Heinkenschloß, *Krylov Subspace Methods for the Solution of Linear Systems and Linear Least Squares Problems*, Lecture notes, Universität Trier, Germany, 1993.

[24] C. T. Kelley and E. W. Sachs, *Quasi-Newton methods and unconstrained optimal control problems*, SIAM J. Control Optim., 25 (1987), pp. 1503–1516.

[25] C. T. Kelley and S. J. Wright, *Sequential quadratic programming for certain parameter identification problems*, Numer. Anal., 29 (1992), pp. 1793–1820.

[26] P. Kosmol, *Methoden zur numerischen Behandlung nichtlinearer Gleichungen und Optimierungsaufgaben*, Teubner-Verlag, Stuttgart, 1989.

[27] F.-S. Kupfer, *An infinite-dimensional convergence theory for reduced SQP methods in Hilbert space*, SIAM J. Optim., 6 (1996), pp. 126–164.

[28] M. Laumen, *A comparison of numerical methods for optimal shape design problems*, in Optim. Methods Softw., 10 (1999), pp. 497–537.

[29] M. Laumen, *Newton's mesh independence principle for a class of optimal shape design problems*, SIAM J. Control Optim., 37 (1999), pp. 1070–1088.

[30] R. M. Lewis, *Practical aspects of variable reduction formulations and reduced basis algorithms in multidisciplinary design optimization*, in SIAM Proceedings of the ICASE/NASA Langley Workshop on Multidisciplinary Design Optimization, N. M. Alexandrov and M. Y. Hussaini, eds., SIAM, Philadelphia, 1997, pp. 173–188.

[31] C. C. Paige and M. A. Saunders, *Solution of sparse indefinite systems of linear equations*, SIAM J. Numer. Anal., 4 (1975), pp. 617–629.

[32] O. Pironneau, *Optimal Shape Design for Elliptic Systems*, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1984.

[33] O. Pironneau and A. Vossinis, *Comparison of Some Optimization Algorithms for Optimum Shape Design in Aerodynamics*, Programme 6: Calcul Scientifique, Modélisation et Logiciels Numériques 1392, INRIA, Le Chesrag, France, 1991.

[34] W. Rudin, *Functional Analysis*, 3rd ed., McGraw-Hill, New York, St. Louis, San Francisco, 1987.

[35] E. W. Sachs, *SQP-Methods in Infinite Dimensions*, Technical Report, Fachbereich IV-Mathematik, Universität Trier, Germany, 1991.

[36] J. Sokolowski and J. P. Zolesio, *Shape design sensitivity analysis of plates and plane elastic solids under unilateral constraints*, J. Optim. Theory Appl., 54 (1987), pp. 361–381.

[37] J. Sokolowski and J. P. Zolesio, *Introduction to Shape Optimization: Shape Sensitivity Analysis*, Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona, Budapest, 1992.

[38] S. Ta'asan, *Fast solvers for MDO problems*, in SIAM Proceedings of the ICASE/NASA Langley Workshop on Multidisciplinary Design Optimization, N. M. Alexandrov and M. Y. Hussaini, eds., SIAM, Philadelphia, 1997, pp. 227–R239.

[39] J. Wloka, *Partielle Differentialgleichungen*, Teubner, Stuttgart, 1982.

[40] E. Zeidler, *Nonlinear Functional Analysis and Its Applications* I: *Fixed-Point Theorems*, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1986.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 534–550

UNDERSTANDING THE GEOMETRY OF INFEASIBLE PERTURBATIONS OF A CONIC LINEAR SYSTEM*

JAVIER PEÑA[†]

Abstract. We discuss some properties of the distance to infeasibility of a conic linear system $Ax = b,\ x \in C$, where C is a closed convex cone.

Some interesting connections between the distance to infeasibility and the solution of certain optimization problems are established. Such connections provide insight into the estimation of the distance to infeasibility and the explicit computation of infeasible perturbations of a given system. We also investigate the properties of the distance to infeasibility assuming that the perturbations are restricted to have a particular structure. Finally, we extend most of our results to more general conic systems $Ax - b \in C_Y,\ x \in C_X$, where C_X and C_Y are closed, convex cones.

Key words. condition numbers, singular values, conic systems, convex programming, distance to infeasibility

AMS subject classifications. 15A12, 65F35, 90C25, 90C31

PII. S1052623497323674

1. Introduction.

The distance to infeasibility of a conic linear system, as introduced by Renegar, plays an interesting role in the study of interior-point methods (see [2, 5, 6]).

Given finite-dimensional Hilbert spaces X, Y, a linear operator $A : X \to Y$, a vector $b \in Y$, and a closed convex cone $C \subseteq X$ consider the *conic system*

$$\begin{aligned} Ax &= b, \\ x &\in C. \end{aligned} \tag{1.1}$$

We call the pair (A, b) the *data* of the conic system (1.1).

The Euclidean norms on X and Y induce a norm on the data by defining the norm of (A, b) as the operator norm of $\left[\begin{array}{cc} A & b \end{array}\right] : X \times \mathbb{R} \to Y$, i.e.,

$$\|(A,b)\| := \left\| \left[\begin{array}{cc} A & b \end{array}\right] \right\| = \max\{\|Ax + tb\| : \|x\|^2 + t^2 \le 1\}.$$

The *distance to infeasibility* of the conic system (1.1) is defined as the smallest perturbation of the data that yields an infeasible system, that is,

$$\operatorname{dist}((A,b),\mathcal{I}) := \inf\{\|(\Delta A, \Delta b)\| : (A + \Delta A, b + \Delta b) \in \mathcal{I}\},$$

where

$$\mathcal{I} := \{(A,b) :\ \text{the system (1.1) is infeasible}\}.$$

An interesting problem is, How can one compute or estimate $\operatorname{dist}((A,b),\mathcal{I})$ for a given instance (A, b)? Or even more ambitiously, how can one describe the set of

*Received by the editors June 30, 1997; accepted for publication (in revised form) March 8, 1999; published electronically February 18, 2000. This work was supported by an Olin Foundation Fellowship at Cornell University.

http://www.siam.org/journals/siopt/10-2/32367.html

[†]Center for Applied Mathematics, Cornell University, Ithaca, NY 14853. Current address: Graduate School of Industrial Administration, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213 (jfp@andrew.cmu.edu).

infeasible instances that are near a given feasible instance? This paper addresses these two questions.

Sometimes instead of $\mathcal{I}$ it will be convenient to work with the following slightly larger set of instances, which can essentially be identified with $\mathcal{I}$:

$$\bar{\mathcal{I}} := \{(A, b) : \text{ there exists } \Delta b \text{ arbitrarily small s.t. } (A, b + \Delta b) \in \mathcal{I}\}.$$

We shall refer to $\bar{\mathcal{I}}$ as the set of *essentially infeasible* data instances.

To provide motivation and intuition, in section 2 we present some insight into the geometry underlying some of our central results, illustrating how certain facts about matrix perturbation theory concerning singular values extend naturally to the more general context of conic systems.

In section 3 we concentrate on a particular type of perturbation (rank-one) of a given instance. We show that the smallest essentially infeasible perturbation of this type can be characterized as the solution of a certain convex optimization problem; in particular, our results here provide a new, simpler proof of a particularly useful characterization of $\text{dist}((A, b), \mathcal{I})$ given by Renegar (see Theorem 3.5 in [6]).

We also present a description of the set of essentially infeasible rank-one perturbations about a given feasible instance that are of minimal size in a certain sense.

Some work has already been done in order to give estimates for $\text{dist}((A, b), \mathcal{I})$. In [3] the authors describe several optimization problems whose optimal values provide bounds for $\text{dist}((A, b), \mathcal{I})$. Our work has a different nature as we focus more on characterizing infeasible perturbations rather than just estimating $\text{dist}((A, b), \mathcal{I})$.

In section 4 we deal with perturbations with restricted structure. Some of the results presented in section 3 for general perturbations are extended to the case in which only certain columns and/or rows of the matrix A are allowed to be perturbed.

Finally, in section 5 we discuss more general conic systems: $Ax - b \in C_Y, x \in C_X$, where C_X and C_Y are closed, convex cones in X and Y, respectively. By introducing slack variables, this context can be seen as a special case of the structured perturbations studied in section 4. We also present a description of the set of minimal-size, essentially infeasible rank-one perturbations that are near a given feasible instance.

Several of our results use dual cones. Let us recall this notion here: Given a convex cone C in a Hilbert space X, its *dual* cone is defined as

$$C^* := \{x \in X : \langle x, y \rangle \geq 0 \text{ for all } y \in C\}.$$

We conclude this introduction with a discussion about homogeneous systems. Such systems will be our focus of attention.

Consider a *homogeneous* conic system, that is, assume $b = 0$:

$$\begin{aligned} Ax &= 0, \\ x &\in C. \end{aligned} \tag{1.2}$$

Notice that given an infeasible perturbation of (1.2):

$$\begin{aligned} (A + \Delta A)x &= \Delta b, \\ x &\in C, \end{aligned} \tag{1.3}$$

the system (1.3) remains infeasible if we scale Δb by any arbitrarily small positive number. Hence (1.3) is infeasible for some Δb if and only if $(A + \Delta A, 0) \in \bar{\mathcal{I}}$, i.e., if *essentially* we only need to perturb A to obtain infeasible instances. In particular,

$$\text{dist}((A, 0), \mathcal{I}) = \inf\{\|\Delta A\| : (A + \Delta A, 0) \in \bar{\mathcal{I}}\}. \tag{1.4}$$

For a homogeneous system (1.2) define $\operatorname{dist}(A, \mathcal{I}) := \operatorname{dist}((A, 0), \mathcal{I})$.

Throughout our development we will assume that $\operatorname{dist}((A, b), \mathcal{I}) > 0$ and that we work with homogeneous systems. Via a *homogenization procedure* Proposition 1.1 below shows that with regard to infeasible perturbations of (1.1), we can focus on homogeneous systems without loss of generality.

Given a nonhomogeneous system

$$\begin{aligned} Ax &= b, \\ x &\in C, \end{aligned} \tag{1.5}$$

consider the corresponding homogenized version

$$\begin{aligned} Ax - tb &= 0, \\ (x, t) &\in C \times \mathbb{R}_+, \end{aligned} \tag{1.6}$$

whose data, variable space, and cone of constraints are, respectively, $\left(\begin{bmatrix} A & -b \end{bmatrix}, 0\right)$, $X \times \mathbb{R}$, and $C \times \mathbb{R}_+$. In the new (larger) data space, let $\mathcal{I}'$ and $\bar{\mathcal{I}}'$ denote the sets of infeasible and essentially infeasible instances, respectively.

Proposition 1.1. *$(A, b) \in \bar{\mathcal{I}}$ if and only if $(\begin{bmatrix} A & -b \end{bmatrix}, 0) \in \bar{\mathcal{I}}'$. In particular,*

$$\operatorname{dist}((A, b), \mathcal{I}) = \operatorname{dist}(\begin{bmatrix} A & -b \end{bmatrix}, \mathcal{I}');$$

i.e., the distance to infeasibility of the systems (1.5) *and* (1.6) *is the same.*

Proof. Suppose $(\begin{bmatrix} A & -b \end{bmatrix}, 0) \in \bar{\mathcal{I}}'$. Let Δb be such that the following system is infeasible:

$$\begin{aligned} Ax - tb &= \Delta b, \\ (x, t) &\in C \times \mathbb{R}_+. \end{aligned}$$

It then easily follows that

$$\begin{aligned} Ax &= b + \Delta b, \\ x &\in C, \end{aligned}$$

is infeasible. Since Δb can be chosen arbitrarily small, $(A, b) \in \bar{\mathcal{I}}$.

For the converse assume $\left(\begin{bmatrix} A & -b \end{bmatrix}, 0\right) \notin \bar{\mathcal{I}}'$. If $b = 0$, then it trivially follows that $(A, b) \notin \bar{\mathcal{I}}$, so let us assume $b \neq 0$. Notice that

$$Y = \{Ax - tb : (x, t) \in C \times \mathbb{R}_+\}$$

because $\left(\begin{bmatrix} A & -b \end{bmatrix}, 0\right) \notin \bar{\mathcal{I}}'$. Therefore, since Y is finite-dimensional,

$$0 \in \operatorname{int}\{Ax - tb : (x, t) \in C \times \mathbb{R}_+, \|(x, t)\| \leq 1\};$$

i.e., there exists $\epsilon > 0$ such that

$$v \in \{Ax - tb : (x, t) \in C \times \mathbb{R}_+, \|(x, t)\| \leq 1\}$$

for all $\|v\| \leq \epsilon$.

We claim that the system

$$\begin{aligned} Ax &= b + \Delta b, \\ x &\in C, \end{aligned}$$

is feasible for any Δb such that $\|\Delta b\| \leq \epsilon$, which consequently implies that $(A, b) \notin \bar{\mathcal{I}}$.

To prove the claim assume Δb satisfies $\|\Delta b\| \leq \epsilon$. For $i = 1, 2$ let $(x_i, t_i) \in C \times \mathbb{R}_+$ with $\|(x_i, t_i)\| \leq 1$ such that

$$Ax_1 - t_1 b = \Delta b \quad \text{and} \quad Ax_2 - t_2 b = \frac{\epsilon}{\|b\|} b.$$

Let $\lambda = \frac{1-t_1}{t_2 + \frac{\epsilon}{\|b\|}}$; notice that λ is well defined and nonnegative because $0 \leq t_1, t_2 \leq 1$, and $\epsilon, \|b\| > 0$. Furthermore,

$$A(x_1 + \lambda x_2) = b + \Delta b,$$

with $x_1 + \lambda x_2 \in C$.

The second part follows by applying (1.4) to $\big(\left[\begin{array}{cc} A & -b \end{array}\right], 0\big)$ and the fact that

$$\begin{aligned} \|(A, b) - (A', b')\| &= \| \left[\begin{array}{cc} A - A' & b - b' \end{array}\right] \| = \| \left[\begin{array}{cc} A - A' & b' - b \end{array}\right] \| \\ &= \| \big(\left[\begin{array}{cc} A & -b \end{array}\right], 0\big) - \big(\left[\begin{array}{cc} A' & -b' \end{array}\right], 0\big) \|. \qquad \square \end{aligned}$$

2. Geometry of infeasible perturbations.

When $C = X$ and $\dim(X) \geq \dim(Y)$, the distance to infeasibility of the conic system

$$\begin{aligned} Ax &= 0, \\ x &\in C, \end{aligned}$$

corresponds to the distance from A to the set of rank-deficient matrices. By relying on the singular value decomposition (see [4, 8]), it is easy to see that the distance to infeasibility is equal to the smallest singular value of A, which can be written as

$$\inf\{\|\beta\| : \nexists x \text{ s.t. } Ax = \beta, \|x\| \leq 1\}.$$

In general, Renegar's characterization (see Corollary 3.6 of this paper) of the distance to infeasibility states that

$$\operatorname{dist}(A, \mathcal{I}) = \inf\{\|\beta\| : \nexists x \text{ s.t. } Ax = \beta, \|x\| \leq 1, x \in C\},$$

and notice how $\operatorname{dist}(A, \mathcal{I})$ naturally extends the smallest singular value of A.

Now suppose we are interested in perturbing A so that A becomes rank-deficient. Moreover, suppose we choose a nonzero vector $\beta \in Y$ and we want to perturb A to $A + \Delta A$ so that

$$\beta \notin \{(A + \Delta A)x : x \in X\}.$$

We can, again using the singular value decomposition, easily construct such a perturbation by taking

$$\Delta A = -\frac{1}{\|\alpha_\beta\|^2} \beta \alpha_\beta^T,$$

where α_β is the minimum-norm solution to the equation system

$$A\alpha = \beta.$$

It is easy to see that this is the smallest rank-one perturbation of the form $\beta\gamma^T$ that yields a rank-deficient system.

We prove (see Propositions 3.1 and 3.2) that a very natural extension holds for conic systems: If α_β is the minimum-norm solution to

$$A\alpha = \beta, \qquad \alpha \in C,$$

then letting $\Delta A = -\frac{1}{\|\alpha_\beta\|^2}\beta\alpha_\beta^T$ we have

$$\beta \notin \{(A + \Delta A)x : x \in C\}.$$

This is also the smallest rank-one perturbation of the form $\beta\gamma^T$ that makes $Ax = 0, x \in C$, essentially infeasible (see Proposition 3.2).

The norm of the rank-one perturbation $\frac{1}{\|\alpha_\beta\|^2}\beta\alpha_\beta^T$ is exactly the length of the segment in the direction β contained in

$$\{Ax : \|x\| \leq 1, x \in C\},$$

and the distance to infeasibility $\operatorname{dist}(A, \mathcal{I})$ is precisely the length of the shortest such segment.

3. Rank-one perturbations.

We study the conic linear system

$$\begin{aligned} Ax &= 0, \\ x &\in C, \end{aligned} \tag{3.1}$$

where C is a closed convex cone.

We have the following goal in this section: Given a nonzero vector $\beta \in Y$, characterize the rank-one perturbations to A of the form $\beta\gamma^T, \gamma \in X$, that lead to an essentially infeasible system

$$\begin{aligned} (A - \beta\gamma^T)x &= 0, \\ x &\in C. \end{aligned}$$

The following program is crucial in our development:

$$\begin{aligned} (P_\beta) \qquad & \min \|\alpha\|, \\ & A\alpha \;=\; \beta, \\ & \;\;\alpha \;\in\; C. \end{aligned}$$

Notice that if $\operatorname{dist}(A, \mathcal{I}) > 0$, then for every given $\beta \in Y$ the system

$$\begin{aligned} Ax &= \beta, \\ x &\in C, \end{aligned}$$

has a solution; this is immediate if $\|\beta\| < \operatorname{dist}(A, \mathcal{I})$ and hence by scaling for all $\beta \in Y$.

Therefore, $\operatorname{dist}(A, \mathcal{I}) > 0$ implies that the program (P_β) has a solution α_β. Moreover, it is easy to see that the solution is unique. We can use this α_β to construct an essentially infeasible rank-one perturbation of the desired form $\beta\gamma^T$.

Proposition 3.1. *Assume $\beta \neq 0$ is given, and let α_β denote the solution to (P_β). For all $\epsilon > 0$, the system*

$$\begin{aligned} \left(A - \frac{1}{\|\alpha_\beta\|^2}\beta\alpha_\beta^T\right) x &= \epsilon\beta, \\ x &\in C, \end{aligned} \tag{3.2}$$

is infeasible.

Proof. Suppose $x_0 \in C$ solves (3.2). Then

$$Ax_0 = \left(\epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2}\right)\beta.$$

Since

$$A\alpha_\beta = \beta,$$

we thus have

$$A(\lambda\alpha_\beta + x_0) = \left(\lambda + \epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2}\right)\beta$$

for any $\lambda > 0$. In particular, if λ is large enough so that $\lambda + \epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2} > 0$, we have

$$A\tilde{\alpha} = \beta,$$

with

$$\tilde{\alpha} = \frac{1}{\lambda + \epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2}}(\lambda\alpha_\beta + x_0) \in C.$$

However, for λ large enough,

$$\begin{aligned}\|\lambda\alpha_\beta + x_0\|^2 - \|\alpha_\beta\|^2 &\left(\lambda + \epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2}\right)^2 \\ &= \|x_0\|^2 - \epsilon^2\|\alpha_\beta\|^2 - \frac{(\alpha_\beta^T x_0)^2}{\|\alpha_\beta\|^2} - 2\epsilon\alpha_\beta^T x_0 - 2\lambda\epsilon\|\alpha_\beta\|^2 < 0,\end{aligned}$$

so

$$\|\tilde{\alpha}\| = \frac{\|\lambda\alpha_\beta + x_0\|}{\lambda + \epsilon + \frac{\alpha_\beta^T x_0}{\|\alpha_\beta\|^2}} < \|\alpha_\beta\|,$$

which contradicts the optimality of α_β. □

Notice that if we replace β by $-\beta$, then we conclude that for all $\epsilon > 0$, the system

$$\begin{aligned}\left(A + \frac{1}{\|\alpha_{-\beta}\|^2}\beta\alpha_{-\beta}^T\right)x &= -\epsilon\beta, \\ x &\in C,\end{aligned}$$

is also infeasible, where $\alpha_{-\beta}$ solves

$$\begin{aligned}(P_{-\beta}) \qquad &\min \|\alpha\|, \\ A\alpha &= -\beta, \\ \alpha &\in C.\end{aligned}$$

The next proposition tells us that one of these two rank-one perturbations is the smallest one of the form $\beta\gamma^T$ which yields an essentially infeasible system, and that it is unique in this regard.

PROPOSITION 3.2. *Assume $\beta \neq 0$ is given. Let $\bar{\alpha}$ be the larger of α_β and $\alpha_{-\beta}$, where α_β and $\alpha_{-\beta}$ are the solutions to (P_β) and $(P_{-\beta})$, respectively. If $\|\gamma\| < \frac{1}{\|\bar{\alpha}\|}$, then for all $\delta \in Y$ the system*

$$\begin{aligned} (A - \beta\gamma^T)x &= \delta, \\ x &\in C, \end{aligned} \tag{3.3}$$

is feasible. If $\|\gamma\| = \frac{1}{\|\bar{\alpha}\|}$ and the system (3.3) *is infeasible for some $\delta \in Y$, then either*

$$\gamma = \frac{1}{\|\alpha_\beta\|^2}\alpha_\beta \quad \textit{or} \quad \gamma = -\frac{1}{\|\alpha_{-\beta}\|^2}\alpha_{-\beta}.$$

This proposition will follow as an immediate consequence of the following lemma.

LEMMA 3.3. *Assume $\beta \neq 0$ and let α_β, $\alpha_{-\beta}$ solve (P_β) and $(P_{-\beta})$, respectively. If $\gamma^T\alpha_\beta < 1$ and $-\gamma^T\alpha_{-\beta} < 1$, then for all $\delta \in Y$ the system* (3.3) *is feasible.*

Proof. Since $\text{dist}(A, \mathcal{I}) > 0$, given any $\delta \in Y$ there exists $x_0 \in C$ such that $Ax_0 = \delta$. We consider two separate cases.

Case 1: $\gamma^T x_0 \geq 0$. Then take

$$\tilde{x} = x_0 + \frac{\gamma^T x_0}{1 - \gamma^T\alpha_\beta}\alpha_\beta \in C.$$

Case 2: $\gamma^T x_0 < 0$. Then take

$$\tilde{x} = x_0 - \frac{\gamma^T x_0}{1 + \gamma^T\alpha_{-\beta}}\alpha_{-\beta} \in C.$$

In either case it is clear that $\tilde{x} \in C$ satisfies $(A - \beta\gamma^T)\tilde{x} = \delta$. □

Proof of Proposition 3.2. If $\|\gamma\| < \frac{1}{\|\bar{\alpha}\|}$, then $\gamma^T\alpha_\beta < 1$ and $-\gamma^T\alpha_{-\beta} < 1$, so the first part follows from Lemma 3.3. For the second part, just observe that if $\|\gamma\| = \frac{1}{\|\bar{\alpha}\|} = \min\{\frac{1}{\|\alpha_\beta\|}, \frac{1}{\|\alpha_{-\beta}\|}\}$ and the system (3.3) is infeasible, then we must have either

$$\gamma^T\alpha_\beta = 1 \quad \text{whereby } \gamma = \frac{1}{\|\alpha_\beta\|^2}\alpha_\beta$$

or

$$-\gamma^T\alpha_{-\beta} = 1 \quad \text{whereby } \gamma = -\frac{1}{\|\alpha_{-\beta}\|^2}\alpha_{-\beta}.$$

Otherwise, by Lemma 3.3 the system would be feasible. □

A natural question now is, Can we construct an infeasible rank-one perturbation of the form $\beta\gamma^T$ without knowing α_β exactly? By Proposition 3.2 we know that such a perturbation must have $\|\gamma\| \geq \frac{1}{\|\alpha_\beta\|}$. Basically the same argument used in the proof of Proposition 3.1 yields Proposition 3.4.

PROPOSITION 3.4. *Assume $\beta \neq 0$ is given, and let α_β denote the solution to (P_β). If $\gamma - \frac{1}{\|\alpha_\beta\|^2}\alpha_\beta \in C^*$, then for any $\epsilon > 0$ the system*

$$\begin{aligned} (A - \beta\gamma^T)x &= \epsilon\beta, \\ x &\in C, \end{aligned} \tag{3.4}$$

is infeasible.

Proof. Suppose $x_0 \in C$ solves (3.4). Then

$$Ax_0 = (\epsilon + \gamma^T x_0)\beta.$$

Thus

$$A(\lambda\alpha_\beta + x_0) = (\lambda + \epsilon + \gamma^T x_0)\beta$$

for any $\lambda > 0$. In particular, if λ is large enough so that $\lambda + \epsilon + \gamma^T x_0 > 0$, we have

$$A\tilde{\alpha} = \beta,$$

with

$$\tilde{\alpha} = \frac{1}{\lambda + \epsilon + \gamma^T x_0}(\lambda\alpha_\beta + x_0) \in C.$$

However, for λ large enough,

$$\begin{aligned} &\|\lambda\alpha_\beta + x_0\|^2 - \|\alpha_\beta\|^2(\lambda + \epsilon + \gamma^T x_0)^2 \\ &\qquad = \|x_0\|^2 \ - \epsilon^2\|\alpha_\beta\|^2 - 2\epsilon\gamma^T x_0\|\alpha_\beta\|^2 - \|\alpha_\beta\|^2(\gamma^T x_0)^2 \\ &\qquad\qquad - 2\lambda\|\alpha_\beta\|^2\left(\epsilon + \left(\gamma - \frac{1}{\|\alpha_\beta\|^2}\alpha_\beta\right)^T x_0\right) < 0, \end{aligned}$$

so

$$\|\tilde{\alpha}\| = \frac{\|\lambda\alpha_\beta + x_0\|}{\lambda + \epsilon + \gamma^T x_0} < \|\alpha_\beta\|,$$

which contradicts the optimality of α_β. □

One might wonder whether rank-one perturbations say much about $\operatorname{dist}(A, \mathcal{I})$. The following result tells us that if we are interested in $\operatorname{dist}(A, \mathcal{I})$, it is enough to consider just rank-one perturbations.

PROPOSITION 3.5. *If the system*

$$\begin{aligned} (A + \Delta A)x &= \Delta b, \\ x &\in C, \end{aligned} \tag{3.5}$$

is infeasible, then there exists $\beta \in Y, \beta \neq 0$, such that the system

$$\begin{aligned} \left(A - \frac{1}{\|\alpha_\beta\|^2}\beta\alpha_\beta^T\right) x &= 0, \\ x &\in C, \end{aligned} \tag{3.6}$$

is essentially infeasible and has the property that

$$\left\|\frac{1}{\|\alpha_\beta\|^2}\beta\alpha_\beta^T\right\| = \frac{\|\beta\|}{\|\alpha_\beta\|} \leq \|\Delta A\|,$$

where α_β solves (P_β).

Proof. If we scale Δb by any arbitrarily small positive number, then the system (3.5) remains infeasible. Hence,

$$0 \in \partial\{(A + \Delta A)x : x \in C\}.$$

Since the set $S = \{(A+\Delta A)x : x \in C\}$ is convex, there exists a supporting hyperplane to S containing 0; i.e., there exists $\beta \in Y, \|\beta\| = 1$ such that

$$\beta^T s \leq 0 \quad \text{for all } s \in S.$$

Let α_β be the solution of (P_β) for this β. Since $\alpha_\beta \in C$,

$$0 \geq \beta^T(A + \Delta A)\alpha_\beta = \beta^T\beta + \beta^T \Delta A \alpha_\beta = 1 + \beta^T \Delta A \alpha_\beta.$$

Therefore,

$$1 \leq |\beta^T \Delta A \alpha_\beta| \leq \|\beta\| \, \|\Delta A\| \, \|\alpha_\beta\| = \|\Delta A\| \, \|\alpha_\beta\|.$$

Thus

$$\left\| \frac{1}{\|\alpha_\beta\|^2} \beta \alpha_\beta^T \right\| = \frac{\|\beta\|}{\|\alpha_\beta\|} = \frac{1}{\|\alpha_\beta\|} \leq \|\Delta A\|.$$

Finally, the system (3.6) is essentially infeasible by Proposition 3.1. □

As a straightforward consequence we obtain a new, simpler proof of the following characterization of $\mathrm{dist}(A, \mathcal{I})$ originally due to Renegar (see [6]).

Corollary 3.6 (Renegar [6]).

$$\mathrm{dist}(A, \mathcal{I}) = \rho(\mathcal{A}) := \inf\{\|\beta\| : \nexists § \text{ s.t. } \mathcal{A}§ = \beta, \|§\| \leq \infty, § \in \mathcal{C}\}.$$

Proof. Given β such that $Ax = \beta, x \in C, \|x\| \leq 1$ is inconsistent, the solution α_β to (P_β) must satisfy $\|\alpha_\beta\| > 1$. By Proposition 3.1, we can construct an infeasible perturbation of size arbitrarily close to

$$\left\| \frac{1}{\|\alpha_\beta\|^2} \beta \alpha_\beta^T \right\| = \frac{\|\beta\|}{\|\alpha_\beta\|} < \|\beta\|,$$

and so it follows that $\mathrm{dist}(A, \mathcal{I}) \leq \rho(A)$.

On the other hand, given any arbitrary infeasible perturbation $(\Delta A, \Delta b)$, by Proposition 3.5 there exists $\beta \neq 0$ such that if α_β solves (P_β), then

$$\frac{\|\beta\|}{\|\alpha_\beta\|} \leq \|\Delta A\| \leq \|(\Delta A, \Delta b)\|.$$

If we take $\beta_\delta = \frac{1+\delta}{\|\alpha_\beta\|}\beta$ with $\delta > 0$, then the solution α_{β_δ} to (P_{β_δ}) has norm $1 + \delta$, and therefore

$$Ax = \beta_\delta, \qquad x \in C, \qquad \|x\| \leq 1$$

is inconsistent, and thus

$$\rho(A) \leq \|\beta_\delta\| = (1+\delta)\frac{\|\beta\|}{\|\alpha_\beta\|} \leq (1+\delta)\|(\Delta A, \Delta b)\|.$$

This holds for any $\delta > 0$ and any infeasible perturbation $(\Delta A, \Delta b)$, hence $\rho(A) \leq \mathrm{dist}(A, \mathcal{I})$. □

The first part of the next proposition states how to construct minimum-size rank-one infeasible perturbations of (3.1). Pick a $u \in Y$, let $\bar{\alpha}$ be the projection of $A^T u$ onto C, and set $\beta = A\bar{\alpha}$. Then perturb A to $A - \frac{1}{\|\bar{\alpha}\|^2}\beta\bar{\alpha}^T$. The second part states that indeed all minimal rank-one infeasible perturbations of (3.1) are obtained in this fashion.

PROPOSITION 3.7.

(a) *For any given $u \in Y$ let $\bar{\alpha}$ be the closest point to $A^T u$ in C. Then $\bar{\alpha}$ is the solution of (P_β) for $\beta = A\bar{\alpha}$.*

(b) *For any given $\beta \in Y, \beta \neq 0$, let α_β be the solution to (P_β). Then α_β is the closest point to $A^T u$ in C for some $u \in Y$.*

Proof. Recall the following projection fact: For a given vector $v \in X$, $\bar{\alpha}$ solves $\min\{\|\alpha - v\| : \alpha \in C\}$ if and only if

(i) $\bar{\alpha} \in C$,

(ii) $\bar{\alpha} - v \in C^*$, and

(iii) $\bar{\alpha}^T(\bar{\alpha} - v) = 0$.

On the other hand, the Lagrangian dual of program (P_β) is

$$\begin{array}{rcl} (D_\beta) & & \max \beta^T u, \\ \|A^T u + w\| & \leq & 1, \\ w & \in & C^*. \end{array}$$

Both (P_β) and (D_β) involve only closed, convex cones, linear operators, and norms; and it is easy to prove, say, by Fenchel's duality theorem (see [7, section 31]), that strong duality holds and both programs attain their optima.

From duality it follows that $\bar{\alpha}$ solves (P_β) and $(\bar{u}, \bar{w})$ solves (D_β) if and only if

(i′) $\bar{\alpha} \in C$,

(ii′) $A\bar{\alpha} = \beta$,

(iii′) $\bar{w} \in C^*$,

(iv′) $A^T\bar{u} + \bar{w} = \frac{1}{\|\bar{\alpha}\|}\bar{\alpha}$, and

(v′) $\bar{\alpha}^T\bar{w} = 0$.

To prove (a), let us assume $\bar{\alpha} \neq 0$ (if $\bar{\alpha} = 0$, then (a) holds trivially). Set

$$\bar{w} = \frac{1}{\|\bar{\alpha}\|}(\bar{\alpha} - A^T u), \qquad \bar{u} = \frac{1}{\|\bar{\alpha}\|}u.$$

From (i)–(iii) it is easy to see that (i′)–(v′) hold, thereby proving (a).

To prove (b), suppose that α_β solves (P_β) and let $(\bar{u}, \bar{w})$ be a solution to (D_β). Now from (i′)–(v′) it is easy to see that (i)–(iii) hold for

$$\bar{\alpha} := \alpha_\beta, \qquad u := \|\bar{\alpha}\|\bar{u}, \qquad \text{and} \qquad v := A^T u. \qquad \square$$

4. Perturbations with restricted structure. In many situations the perturbations of interest are not arbitrary perturbations on the data but are restricted to have a particular structure. For example, certain predefined entries of A (e.g., determined by some sparsity pattern) may be the only ones allowed to be perturbed (see [1]).

As an early step towards developing a theory of structured perturbations, in this section we study the special cases in which only certain rows and/or certain columns of A are allowed to be perturbed.

Assume first that perturbations can be made only on certain columns of A. That is, suppose we can split $X = X_1 \times X_2$, and accordingly we decompose A into two corresponding blocks $A = \begin{bmatrix} A_1 & A_2 \end{bmatrix}$ so that our conic system is

$$\begin{array}{c} \begin{bmatrix} A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0, \\ x \in C, \end{array} \tag{4.1}$$

where C is a closed convex cone in X.

Suppose that we only allow perturbations of the form

$$\begin{aligned} \begin{bmatrix} A_1+\Delta A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \Delta b, \\ x &\in C. \end{aligned}$$

In this section we will see how most of the development presented in section 3 can be "split" to fit this context. (This is an interesting phenomenon as the cone C is *not* assumed to be the product of two cones in X_1 and X_2.)

We start with a version of the program (P_β) in the current context. Given $\beta \in Y$, let $\alpha_\beta = (\alpha_{\beta,1}, \alpha_{\beta,2})$ denote the solution of the program

$$\begin{aligned} &\min \|\alpha_1\|, \\ A\alpha &= \beta, \\ \alpha &\in C, \end{aligned} \tag{4.2}$$

where $\alpha = (\alpha_1, \alpha_2)$.

Assuming $\alpha_{\beta,1} \neq 0$ we can use α_β to construct a rank-one infeasible perturbation of (4.1), where we only perturb A_1.

PROPOSITION 4.1. *Assume $\beta \neq 0$ is given, and let α_β denote the solution to* (4.2)*. If $\alpha_{\beta,1} \neq 0$, then for all $\epsilon > 0$ the system*

$$\begin{aligned} \begin{bmatrix} A_1 - \frac{1}{\|\alpha_{\beta,1}\|^2}\beta\alpha_{\beta,1}^T & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \epsilon\beta, \\ x &\in C, \end{aligned} \tag{4.3}$$

is infeasible.

Proof. Suppose $x_0 \in C$ solves (4.3). Then

$$Ax_0 = \left(\epsilon + \frac{\alpha_{\beta,1}^T x_{0,1}}{\|\alpha_{\beta,1}\|^2}\right)\beta.$$

Since

$$A\alpha_\beta = \beta,$$

we thus have

$$A(\lambda\alpha_\beta + x_0) = \left(\lambda + \epsilon + \frac{\alpha_{\beta,1}^T x_{0,1}}{\|\alpha_{\beta,1}\|^2}\right)\beta$$

for any $\lambda > 0$. In particular, if λ is large enough so that $\lambda+\epsilon+\frac{\alpha_{\beta,1}^T x_{0,1}}{\|\alpha_{\beta,1}\|^2} > 0$, we obtain

$$A\tilde{\alpha} = \beta,$$

with

$$\tilde{\alpha} = \frac{1}{\lambda + \epsilon + \frac{\alpha_{\beta,1}^T x_{0,1}}{\|\alpha_{\beta,1}\|^2}}(\lambda\alpha_\beta + x_0) \in C.$$

However, for λ large enough,

$$\|\tilde{\alpha}_1\| = \frac{\|\lambda\alpha_{\beta,1} + x_{0,1}\|}{\lambda + \epsilon + \frac{\alpha_{\beta,1}^T x_{0,1}}{\|\alpha_{\beta,1}\|^2}} < \|\alpha_{\beta,1}\|,$$

which contradicts the optimality of α_β. □

Also letting $\alpha_{-\beta}$ denote the solution to

$$\begin{array}{rcl} & \min \|\alpha_1\|, & \\ A\alpha & = & -\beta, \\ \alpha & \in & C, \end{array} \tag{4.4}$$

if $\alpha_{-\beta,1} \neq 0$, then for all $\epsilon > 0$ we obtain another infeasible rank-one perturbation:

$$\begin{bmatrix} A_1 + \frac{1}{\|\alpha_{-\beta,1}\|^2}\beta\alpha_{-\beta,1}^T & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = -\epsilon\beta, \qquad x \in C.$$

One of these two rank-one perturbations is the smallest essentially infeasible perturbation of the form $\beta\gamma^T$ on A_1.

Proposition 4.2. *Assume $\beta \neq 0$ is given. Let $\bar{\alpha}_1$ be the larger of $\alpha_{\beta,1}$ and $\alpha_{-\beta,1}$, where α_β and $\alpha_{-\beta}$ are the solutions to* (4.2) *and* (4.4), *respectively, and suppose $\bar{\alpha}_1 \neq 0$.*

If $\|\gamma\| < \frac{1}{\|\bar{\alpha}_1\|}$, then for all $\delta \in Y$ the system

$$\begin{bmatrix} A_1 - \beta\gamma^T & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \delta, \qquad x \in C, \tag{4.5}$$

is feasible. If $\|\gamma\| = \frac{1}{\|\bar{\alpha}_1\|}$ and the system (4.5) *is infeasible for some $\delta \in Y$, then either $\gamma = \frac{1}{\|\alpha_{\beta,1}\|^2}\alpha_{\beta,1}$ or $\gamma = -\frac{1}{\|\alpha_{-\beta,1}\|^2}\alpha_{-\beta,1}$.*

The proof of this proposition mimics the proof of Proposition 3.2, and it relies on the following analogue of Lemma 3.3.

Lemma 4.3. *Assume $\beta \neq 0$ is given, and let α_β, $\alpha_{-\beta}$ solve* (4.2) *and* (4.4), *respectively. If $\gamma^T\alpha_{\beta,1} < 1$ and $-\gamma^T\alpha_{-\beta,1} < 1$, then for all $\delta \in Y$ the system* (4.5) *is feasible.*

Proof. Since $\operatorname{dist}(A, \mathcal{I}) > 0$, there exists $x_0 \in C$ such that $Ax_0 = \delta$. We consider two separate cases.

Case 1: $\gamma^T x_{0,1} \geq 0$. Then take

$$\tilde{x} = x_0 + \frac{\gamma^T x_{0,1}}{1 - \gamma^T\alpha_{\beta,1}}\alpha_\beta \in C.$$

Case 2: $\gamma^T x_{0,1} < 0$. Then take

$$\tilde{x} = x_0 - \frac{\gamma^T x_{0,1}}{1 + \gamma^T\alpha_{-\beta,1}}\alpha_{-\beta} \in C.$$

In either case it is clear that $\tilde{x} \in C$ satisfies $\begin{bmatrix} A_1 - \beta\gamma^T & A_2 \end{bmatrix}\tilde{x} = \delta$. □

Proof of Proposition 4.2. If $\|\gamma\| < \frac{1}{\|\bar{\alpha}_1\|}$, then $\gamma^T\alpha_{\beta,1} < 1$ and $-\gamma^T\alpha_{-\beta,1} < 1$, so the first part follows from Lemma 4.3. For the second part, just observe that if

$\|\gamma\| = \frac{1}{\|\bar{\alpha}_1\|} = \min\{\frac{1}{\|\alpha_{\beta,1}\|}, \frac{1}{\|\alpha_{-\beta,1}\|}\}$ and the system (3.3) is infeasible, then we must have either

$$\gamma^T \alpha_{\beta,1} = 1 \quad \text{whereby } \gamma = \frac{1}{\|\alpha_{\beta,1}\|^2}\alpha_{\beta,1}$$

or

$$-\gamma^T \alpha_{-\beta,1} = 1 \quad \text{whereby } \gamma = -\frac{1}{\|\alpha_{-\beta,1}\|^2}\alpha_{-\beta,1}.$$

Otherwise by Lemma 4.3 the system (4.5) would be feasible. □

Remark 4.4. In the statement of Proposition 4.2, if $\bar{\alpha}_1 = 0$ (i.e., $\alpha_{\beta,1} = \alpha_{-\beta,1} = 0$), then Lemma 4.3 implies that it is impossible to construct an essentially infeasible perturbation of the form $\beta\gamma^T$ on A_1.

Once again, for the purpose of determining the smallest restricted perturbation that makes the system (4.1) infeasible, it suffices to look at rank-one perturbations.

PROPOSITION 4.5. *If the system*

$$\begin{array}{c} \begin{bmatrix} A_1 + \Delta A_1 & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \Delta b, \\ x \in C, \end{array} \tag{4.6}$$

is infeasible, then there exists $\beta \in Y, \beta \neq 0$, *such that the system*

$$\begin{array}{c} \begin{bmatrix} A_1 - \frac{1}{\|\alpha_{\beta,1}\|^2}\beta\alpha_{\beta,1}^T & A_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0, \\ x \in C, \end{array} \tag{4.7}$$

is essentially infeasible and with the property that

$$\left\| \frac{1}{\|\alpha_{\beta,1}\|^2}\beta\alpha_{\beta,1}^T \right\| = \frac{\|\beta\|}{\|\alpha_{\beta,1}\|} \leq \|\Delta A_1\|,$$

where α_β *solves* (4.2).

Proof. If we scale Δb by any arbitrarily small positive number, then the system (4.6) remains infeasible. Hence,

$$0 \in \partial\{\begin{bmatrix} A_1 + \Delta A_1 & A_2 \end{bmatrix} x : x \in C\}.$$

Since the set $S = \{\begin{bmatrix} A_1 + \Delta A_1 & A_2 \end{bmatrix} x : x \in C\}$ is convex, there exists a supporting hyperplane to S containing 0; i.e., there exists $\beta \in Y, \|\beta\| = 1$ such that

$$\beta^T s \leq 0 \quad \text{ for all } s \in S.$$

Let α_β be the solution of (4.2) for this β. Since $\alpha_\beta \in C$,

$$0 \geq \beta^T \begin{bmatrix} A_1 + \Delta A_1 & A_2 \end{bmatrix} \alpha_\beta = \beta^T\beta + \beta^T \Delta A \alpha_\beta = 1 + \beta^T \Delta A_1 \alpha_{\beta,1}.$$

Therefore

$$1 \leq |\beta^T \Delta A_1 \alpha_{\beta,1}| \leq \|\beta\| \, \|\Delta A_1\| \, \|\alpha_{\beta,1}\| = \|\Delta A_1\| \, \|\alpha_{\beta,1}\|.$$

Thus $\alpha_{\beta,1} \neq 0$ and

$$\left\| \frac{1}{\|\alpha_{\beta,1}\|^2}\beta\alpha_{\beta,1}^T \right\| = \frac{\|\beta\|}{\|\alpha_{\beta,1}\|} = \frac{1}{\|\alpha_{\beta,1}\|} \leq \|\Delta A_1\|.$$

Finally, the system (4.7) is essentially infeasible by Proposition 4.1. □

As a straightforward consequence we obtain the following generalization of Corollary 3.6.

Corollary 4.6.

$$\operatorname{sdist}(A,\mathcal{I}) = s\rho(A),$$

where

$$\operatorname{sdist}(A,\mathcal{I}) := \inf\left\{\|(\Delta A_1, \Delta b)\| : \left(\begin{bmatrix} A_1+\Delta A_1 & A_2\end{bmatrix}, \Delta b\right) \in \mathcal{I}\right\}$$

and

$$s\rho(A) := \inf\{\|\beta\| : Ax=\beta, x\in C, \|x_1\|\le 1 \textit{ is inconsistent}\}.$$

Proof. For any β such that $Ax=\beta, x\in C, \|x_1\|\le 1$ is inconsistent, the solution α_β to (4.2) satisfies $\|\alpha_{\beta,1}\| > 1$. By Proposition 4.1, we can construct an infeasible perturbation of size arbitrarily close to

$$\left\|\frac{1}{\|\alpha_{\beta,1}\|^2}\beta\alpha_{\beta,1}^T\right\| = \frac{\|\beta\|}{\|\alpha_{\beta,1}\|} < \|\beta\|;$$

it follows that $\operatorname{sdist}(A,\mathcal{I}) \le s\rho(A)$.

On the other hand, given any arbitrary infeasible perturbation $(\Delta A_1, \Delta b)$, by Proposition 4.5 there exists $\beta \neq 0$ such that if $\alpha_{\beta,1}$ solves (4.2), then

$$\frac{\|\beta\|}{\|\alpha_{\beta,1}\|} \le \|\Delta A_1\| \le \|(\Delta A_1, \Delta b)\|.$$

If we take $\beta_\delta = \frac{1+\delta}{\|\alpha_\beta\|}\beta$ with $\delta > 0$, then the solution α_{β_δ} to (4.2) (with β_δ instead of β) satisfies $\|\alpha_{\beta_\delta,1}\| = 1+\delta$, and therefore

$$Ax=\beta_\delta, \qquad x\in C, \qquad \|x_1\|\le 1$$

is inconsistent, and thus

$$s\rho(A) \le \|\beta_\delta\| = (1+\delta)\frac{\|\beta\|}{\|\alpha_{\beta,1}\|} \le (1+\delta)\|(\Delta A_1, \Delta b)\|.$$

This holds for any $\delta > 0$ and any infeasible perturbation $(\Delta A_1, \Delta b)$, hence $s\rho(A) \le \operatorname{sdist}(A,\mathcal{I})$. □

Now assume that we restrict the perturbations to be only on certain columns and rows; i.e., suppose we can also split $Y = Y_1 \times Y_2$ and our linear system is

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix} x_1 \\ x_2\end{bmatrix} = \begin{bmatrix} 0 \\ 0\end{bmatrix}, \quad x \in C, \tag{4.8}$$

where C is a closed convex cone in X.

Suppose we only allow perturbations of the form

$$\begin{bmatrix} A_{11}+\Delta A_{11} & A_{12} \\ A_{21} & A_{22}\end{bmatrix}\begin{bmatrix} x_1 \\ x_2\end{bmatrix} = \begin{bmatrix} \Delta b_1 \\ 0\end{bmatrix}, \quad x \in C. \tag{4.9}$$

Given $\beta \in Y_1$, consider the program

$$\begin{array}{rcl} & \min \|\alpha_1\|, & \\ \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} & = & \begin{bmatrix} \beta \\ 0 \end{bmatrix}, \\ \alpha & \in & C. \end{array} \tag{4.10}$$

We can write (4.8), (4.9), and (4.10), respectively, as

$$\begin{bmatrix} A_{11} & A_{12} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0, \quad x \in \mathcal{C}',$$

$$\begin{bmatrix} A_{11} + \Delta A_{11} & A_{12} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \Delta b_1, \quad x \in \mathcal{C}',$$

and

$$\begin{array}{rcl} & \min \|\alpha_1\|, & \\ \begin{bmatrix} A_{11} & A_{12} \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} & = & \beta, \\ \alpha & \in & \mathcal{C}', \end{array}$$

where

$$\mathcal{C}' = \{x \in C : A_{21}x_1 + A_{22}x_2 = 0\}.$$

Observe that $\mathcal{C}'$ is a closed convex cone, so we can reduce this problem setting to the one just studied, and therefore analogous results hold.

5. Other conic systems.

An interesting application of the approach taken in section 4 is that perturbations of other kinds of conic problems can be studied as particular types of structured perturbations.

For example, one could consider the following more general conic linear system:

$$\begin{aligned} Ax - b &\in C_Y, \\ x &\in C_X, \end{aligned} \tag{5.1}$$

where C_X, C_Y are closed convex cones in X and Y, respectively. (Renegar addresses the problem written in this form in [6].)

We extend the meaning of $\operatorname{dist}((A,b),\mathcal{I})$ to fit this general context; that is, we extend the definition given in the introduction as follows:

$$\operatorname{dist}((A,b),\mathcal{I}) := \inf\{\|(\Delta A, \Delta b)\| : (A + \Delta A, b + \Delta b) \in \mathcal{I}\},$$

where

$$\mathcal{I} := \{(A,b) : \text{ the system (5.1) is infeasible}\}.$$

The definition of essentially infeasible instances is also extended in the obvious way.

Again, by homogenizing if necessary, there is no loss of generality in assuming $b = 0$ here. That is, we study the system

$$\begin{aligned} Ax &\in C_Y, \\ x &\in C_X, \end{aligned} \tag{5.2}$$

denote the corresponding distance to infeasibility by $\operatorname{dist}(A, \mathcal{I})$, and assume that $\operatorname{dist}(A, \mathcal{I}) > 0$.

By introducing slack variables, we can write (5.2) as follows:

$$\begin{aligned} \begin{bmatrix} A & -I \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} &= 0, \\ \begin{bmatrix} x \\ z \end{bmatrix} &\in C_X \times C_Y. \end{aligned} \tag{5.3}$$

Therefore, infeasible perturbations of (5.2) correspond to infeasible perturbations to (5.3), where we only allow the first block (namely, A) to be perturbed; i.e., we have a particular case of what we discussed in section 4.

We could rewrite the results we discussed in section 4 in this context. For example, Corollary 4.6 becomes

$$\operatorname{dist}(A, \mathcal{I}) = \inf\{\|\beta\| : \nexists x \text{ s.t. } Ax - \beta \in C_Y, x \in C_X, \|x\| \leq 1\}$$

(which is the way Renegar phrased his characterization of $\operatorname{dist}(A, \mathcal{I})$ in [6]).

We can also rewrite program (4.2) as

$$\begin{aligned} &\min \|\alpha\|, \\ A\alpha - \beta &\in C_Y, \\ \alpha &\in C_X. \end{aligned} \tag{5.4}$$

The next result is analogous to Proposition 3.7. The first part states how to construct minimum-size rank-one perturbations of (5.3). Pick a $u \in C_Y^*$, and let $\bar{\alpha}$ be the projection of $A^T u$ onto C_X, let $\bar{y} \in C_Y$ be such that $u^T \bar{y} = 0$, and set $\beta = A\bar{\alpha} - \bar{y}$. Then perturb A to $A - \frac{1}{\|\bar{\alpha}\|^2} \beta \bar{\alpha}^T$. The second part states that all minimal rank-one perturbations of (5.3) are indeed obtained in this fashion.

PROPOSITION 5.1.

(a) *For any given $u \in C_Y^*$ let $\bar{\alpha}$ be the closest point to $A^T u$ in C_X, and let $\bar{y} \in C_Y$ be such that $u^T \bar{y} = 0$. Then $\bar{\alpha}$ is the solution of* (5.4) *for $\beta = A\bar{\alpha} - \bar{y}$.*

(b) *For any given $\beta \in Y, \beta \neq 0$, let (α_β, y_β) be the solution to* (5.4). *Then there is a vector $u \in C_Y^*$ with the property that α_β is the closest point to $A^T u$ in C_X and $u^T y_\beta = 0$.*

Proof. This proof is similar in spirit to that of Proposition 3.7. Recall the projection fact: For a given vector $v \in X$, $\bar{\alpha}$ solves $\min\{\|\alpha - v\| : \alpha \in C_X\}$ if and only if

(i) $\bar{\alpha} \in C_X$,

(ii) $\bar{\alpha} - v \in C_X^*$, and

(iii) $\bar{\alpha}^T(\bar{\alpha} - v) = 0$.

The Lagrangian dual of (5.4) is

$$\begin{aligned} &\max \beta^T u, \\ \|A^T u + w\| &\leq 1, \\ w &\in C_X^*, \\ u &\in C_Y^*. \end{aligned} \tag{5.5}$$

By Fenchel's duality theorem (see [7, section 31]), strong duality holds and both (5.4) and (5.5) attain their optima.

From duality it follows that $\bar{\alpha}$ solves (5.4) (with $\bar{y} = A\bar{\alpha} - \beta$) and $(\bar{u}, \bar{w})$ solves (5.5) if and only if

(i′) $\bar{\alpha} \in C_X,\ \bar{y} \in C_Y$,

(ii′) $A\bar{\alpha} - \bar{y} = \beta$,

(iii′) $\bar{w} \in C_X^*,\ \bar{u} \in C_Y^*$,

(iv′) $A^T\bar{u} + \bar{w} = \frac{1}{\|\bar{\alpha}\|}\bar{\alpha}$, and

(v′) $\bar{u}^T\bar{y} = 0,\ \bar{\alpha}^T\bar{w} = 0$.

To prove (a), assume $\bar{\alpha} \neq 0$ (if $\bar{\alpha} = 0$, then (a) holds trivially). Set

$$\bar{w} = \frac{1}{\|\bar{\alpha}\|}(\bar{\alpha} - A^T u), \qquad \bar{u} = \frac{1}{\|\bar{\alpha}\|}u.$$

From conditions (i)–(iii) and $u^T\bar{y} = 0$ it is easy to see that (i′)–(v′) hold, thereby proving (a).

To prove (b), suppose that α_β solves (5.4) and let $y_\beta = A\alpha_\beta - \beta$. Let $(\bar{u}, \bar{w})$ be a solution to (5.5); now, from (i′)–(v′) it is easy to see that (i)–(iii) hold for

$$\bar{\alpha} := \alpha_\beta, \qquad u := \|\bar{\alpha}\|\bar{u}, \qquad v := A^T u, \qquad \text{and} \quad \bar{y} := y_\beta. \qquad \square$$

Acknowledgments. The author thanks Professor James Renegar at Cornell University for many helpful discussions and suggestions, as well as two anonymous referees whose comments helped improve the exposition of the original manuscript.

REFERENCES

[1] S. Filipowski, *On the complexity of solving sparse symmetric linear programs specified with approximate data*, Math. Oper. Res., 22 (1997), pp. 769–792.

[2] R. Freund and M. Nunez, *Condition measures and properties of the central trajectory of a linear program*, Math. Programming, 83 (1998), pp. 1–28.

[3] R. Freund and J. Vera, *Some characterizations and properties of the "Distance to Ill-posedness" and the condition measure of a conic linear system*, Math. Programming, to appear.

[4] G. Golub and C. Van Loan, *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, 1997.

[5] J. Renegar, *Some perturbation theory for linear programming*, Math. Programming, 65 (1994), pp. 73–92.

[6] J. Renegar, *Linear programming, complexity theory and elementary functional analysis*, Math. Programming, 70 (1995), pp. 279–351.

[7] T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[8] G. Stewart and J. Sun, *Matrix Perturbation Theory*, Academic Press, New York, 1990.

SIAM J. Optim.
Vol. 10, No. 2, pp. 551–579

AN EFFICIENT ALGORITHM FOR MINIMIZING A SUM OF p-NORMS[*]

GUOLIANG XUE[†] AND YINYU YE[‡]

Abstract. We study the problem of minimizing a sum of p-norms where p is a fixed real number in the interval $[1, \infty]$. Several practical algorithms have been proposed to solve this problem. However, none of them has a known polynomial time complexity. In this paper, we transform the problem into standard conic form. Unlike those in most convex optimization problems, the cone for the p-norm problem is *not* self-dual unless $p = 2$. Nevertheless, we are able to construct two logarithmically homogeneous self-concordant barrier functions for this problem. The barrier parameter of the first barrier function does not depend on p. The barrier parameter of the second barrier function increases with p. Using both barrier functions, we present a primal-dual potential reduction algorithm to compute an ϵ-optimal solution in polynomial time that is independent of p. Computational experiences with a Matlab implementation are also reported.

Key words. shortest network under a given topology, facilities location, Steiner minimum trees, minimizing a sum of norms, primal-dual potential reduction algorithms, polynomial time algorithms

AMS subject classifications. 68Q20, 68Q25, 90C25, 90C35

PII. S1052623497327088

1. Introduction.

Let $c_1, c_2, \dots, c_m \in R^d$ be column vectors in the Euclidean d-space and $A_1, A_2, \dots, A_m \in R^{n\times d}$ be n-by-d matrices each having full column rank. We want to find a point $u \in R^n$ such that the following sum of p-norms, $p \geq 1$, is minimized:

$$\begin{array}{ll} \min & \displaystyle\sum_{i=1}^{m} ||c_i - A_i^T u||_p \\ \text{s.t.} & u \in R^n. \end{array} \tag{1.1}$$

We recall that $|| \bullet ||_p$ is the *Hölder* or *p-norm* [14] defined by

$$||x||_p = (|x_1|^p + \cdots + |x_n|^p)^{1/p}. \tag{1.2}$$

It is clear that $u = 0$ is an optimal solution to (1.1) when all of the c_i are zero. Therefore, we will assume in the rest of this paper that not all of the c_i are zero. Problem (1.1) is a convex programming problem. Thus, it can be generally solved in "polynomial time." In this paper, we present a conic formulation of the problem, develop a specialized interior point algorithm, and analyze its computational complexity, exploring the special tree structure of the problem. One interesting feature is that the p-order cone involved in our formulation, unlike those in almost all current problems solved by interior point algorithms, is *not* self-dual. This presents certain difficulties in developing and analyzing the algorithm.

[*] Received by the editors September 5, 1997; accepted for publication (in revised form) July 27, 1998; published electronically February 24, 2000.
http://www.siam.org/journals/siopt/10-2/32708.html

[†] Department of Computer Science, The University of Vermont, Burlington, VT 05405-0156 (xue@cs.uvm.edu). The research of this author was supported in part by U.S. Army Research Office grant DOD DAAH04-96-1-0233 and by National Science Foundation grants NSF ASC-9409285 and NSF OSR-9350540.

[‡] Department of Management Sciences, The University of Iowa, Iowa City, IA 52242-1000 (yyye@dollar.biz.uiowa.edu). The research of this author was supported in part by NSF grants DMI-9522507, DMS-9703490, and OSR-9350540.

Problem (1.1) has a long history, dating back to the 17th century, when Fermat [19] studied the problem of finding the shortest network interconnecting three points on the Euclidean plane, where a fourth point may be added to minimize the sum of lengths of the network. The Fermat problem was generalized to the Euclidean facilities location problem and the Euclidean Steiner minimal tree problem. Those problems were in turn further generalized to cases where distances are measured using weighted p-norms.

The facilities location problem is one of locating N new facilities with respect to M existing facilities, the locations of which are known. The problem consists of finding locations of new facilities that will minimize the sum of (nonnegatively) weighted distances between the new and existing facilities, and between the new facilities. If there is only one new facility ($N = 1$), the problem is called a single facility location (SFL) problem. If there is more than one new facility ($N \geq 2$), the problem is called a multifacility location (MFL) problem.

For the general SFL problem with Euclidean norm, Weiszfeld [33] gave a simple closed-form iterative algorithm in 1937. This work started a chain of research on this topic [19, 27, 17, 32, 5, 6, 20]. Miehle [23] was the first to propose an extension of the Weiszfeld algorithm for the SFL problem to solve MFL problems. Again, a number of important results were obtained along this line [27, 29, 34, 10, 30]. For more details on location problems, see the books by Francis, McGinnis, and White [11] and Love, Morris, and Wesolowsky [21].

Practical algorithms for solving these problems began with the work of Calamai and Conn [3, 4] and Overton [28], where they proposed projected Newton algorithms with quadratic rate of convergence. They also generalized the location problems to one of minimizing a sum of norms. In recent years, several complexity results and numerically stable algorithms have been obtained for problem (1.1) with $p = 2$ using techniques of interior point algorithms. In [35], Xue, Rosen, and Pardalos showed that the dual of the Euclidean MFL problem is the minimization of a linear function subject to linear and convex quadratic constraints and that therefore it can be solved using interior point techniques in polynomial time. More recently, Andersen [1] used the HAP idea [10] to smooth the objective function by introducing a perturbation $\epsilon > 0$ and applied a Newton barrier method to solve the problem. Andersen and Christiansen [2] and Conn and Overton [7] also proposed a primal-dual method based on the ϵ-perturbation and presented impressive computational results.

For $p \geq 1$, den Hertog et al. [8, 9] (also see references therein) presented a polynomial time interior point Newton barrier method for solving a closely related problem, where the objective function is

$$\sum_{i=1}^{m} ||c_i - A_i^T u||_p^p,$$

that is, the sum of the p-powers of p-norms. This objective function is equivalent to ours only if $m = 1$. Nesterov and Nemirovskii [25, 24] also addressed the problem with $m = 1$. Their constructed barrier function, although different from the one used in our conic formulation, influenced our study of this problem.

In a recent paper [36], we studied the problem of minimizing a sum of Euclidean norms. The problem was formulated in standard conic form and a polynomial time primal-dual potential reduction algorithm was presented. A nice property of the Euclidean case is that the second-order cone is self-dual. We have taken advantage of this fact in our work in [36], based on the self-dual theory and primal-dual algorithm developed by Nesterov and Todd [26] (see also Kojima, Shindoh, and Hara [18]).

In some applications, the problems are better modeled with p-norms where $p \neq 2$. For example, in VLSI design, the 1-norm or Manhattan distance is used. In transportation, several general p-norms are used [21]. Since the p-order cone of the conic form of (1.1) is *not* self-dual unless $p = 2$, a natural question to ask is the following: *Can the techniques used for minimizing a sum of Euclidean norms be generalized to minimize a sum of p-norms in polynomial time?* The goal of this paper is to answer the above question affirmatively. Specifically, we present a primal-dual potential reduction algorithm that computes an ϵ-optimal solution in at most

$$O(\sqrt{md}(\log(\bar{c}/\epsilon) + \log(md)))$$

iterations, where $\bar{c} = \max_{1 \le i \le m} ||c_i||$. Note that this bound is independent of p and is increased only by a factor $\sqrt{d}$ compared to the bound for $p = 2$ [36].

The rest of this paper is organized as follows. In section 2, the basic problem (1.1) is transformed into a standard convex programming problem in conic form. In section 3, we develop two logarithmically homogeneous self-concordant barrier functions for this problem. In section 4, we present a primal-dual potential reduction algorithm for solving the problem. In section 5, we discuss the computational complexity and simplifications of the potential reduction algorithm. In section 6, we present applications to the SFL problem, the MFL problem, and the Steiner minimal tree (SMT) problem. In section 7, we present some computational examples on SMT problems. We conclude this paper in section 8.

2. Conic formulation.

We will call problem (1.1) the *basic problem* in the rest of our paper. This problem can be formulated as the maximization of a linear function subject to affine and convex cone constraints as follows.

$$\begin{aligned} \max \quad & -\sum_{i=1}^{m} t_i \\ \text{s.t.} \quad & t_1 \ge ||c_1 - A_1^T u||_p, \\ & t_2 \ge ||c_2 - A_2^T u||_p, \\ & \vdots \\ & t_m \ge ||c_m - A_m^T u||_p, \end{aligned} \tag{2.1}$$

where $t_i \in R, i = 1, 2, \ldots, m$.

In the rest of this paper, when we represent a large matrix with several small matrices, we will use a semicolon to represent column concatenation and a comma to represent row concatenation. This notation also applies to vectors. We will use 0 to represent a column vector whose elements are all zero, and e the vector of all ones. We will use R_+ to represent the set of nonnegative real numbers.

In this section, we will transform our basic problem (1.1) into a standard convex programming problem in conic form, where the cone and its associated barrier are *not self-dual* unless $p = 2$. (It is worthwhile to mention that, for $p = 1$ or $p = \infty$, the problem can be reformulated as a linear program or second-order conic program of md variables; thus, the reformulation becomes self-dual and the symmetric-scaling algorithm applies.)

Consider the p-order cone, where $p \ge 1$,

$$K = \left\{ (t \in R_+, s \in R^d) : \ t^p \ge \sum_{j=1}^{d} |s_j|^p \right\}.$$

Its interior is

$$\text{int}K := \{(t \in R_+, s \in R^d) : \ t > \|s\|_p\}.$$

The dual of K is

$$K^* = \left\{ (\tau \in R_+, x \in R^d) : \ \tau^q \geq \sum_{j=1}^{d} |x_j|^q \right\},$$

where $\frac{1}{p} + \frac{1}{q} = 1$.

Now let

$$\mathcal{B} = \begin{pmatrix} -1 \\ -1 \\ \vdots \\ -1 \\ 0_n \end{pmatrix} \in R^{m+n}, \quad \mathcal{C} = \begin{pmatrix} (0; c_1) \\ (0; c_2) \\ \vdots \\ (0; c_m) \end{pmatrix} \in R^{m+md},$$

and

$$\mathcal{A}^T = \begin{pmatrix} -1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & A_1^T \\ 0 & -1 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & A_2^T \\ & & \ddots & & \\ 0 & 0 & \cdots & -1 & 0 \\ 0 & 0 & \cdots & 0 & A_m^T \end{pmatrix} \in R^{(m+md)\times(m+n)}.$$

Then, problem (1.1) or (2.1) can be written in the standard (dual) form

$$\begin{array}{ll} \max & \mathcal{B}^T(t_1; t_2; \ldots; t_m; u) \\ \text{s.t.} & \begin{pmatrix} (t_1; s_1) \\ (t_2; s_2) \\ \vdots \\ (t_m; s_m) \end{pmatrix} = \mathcal{C} - \mathcal{A}^T(t_1; t_2; \ldots; t_m; u), \\ & (t_i; s_i) \in K, \ i = 1, 2, \ldots, m, \end{array} \tag{2.2}$$

where $u \in R^n$ and $t_i \in R_+, s_i \in R^d$, $i = 1, 2, \ldots, m$.

Let $(\tau_1; x_1), (\tau_2; x_2), \ldots, (\tau_m; x_m) \in R^{d+1}$. Then its corresponding primal problem is

$$\begin{array}{ll} \min & \mathcal{C}^T((\tau_1; x_1); (\tau_2; x_2); \ldots; (\tau_m; x_m)) \\ \text{s.t.} & \mathcal{A}((\tau_1; x_1); (\tau_2; x_2); \ldots; (\tau_m; x_m)) = \mathcal{B}, \\ & (\tau_i; x_i) \in K^*, \ i = 1, 2, \ldots, m. \end{array} \tag{2.3}$$

Thus, using $\mathcal{S} := ((t_1; s_1); (t_2; s_2); \ldots; (t_m; s_m))$, $\mathcal{Y} := (t_1; t_2; \ldots; t_m; u)$, $\mathcal{X} := ((\tau_1; x_1); (\tau_2; x_2); \ldots; (\tau_m; x_m))$, $\mathcal{K} := K^m := K \times K \times \cdots \times K$, and $\mathcal{K}^* := (K^*)^m := K^* \times K^* \times \cdots \times K^*$, we can write the two problems (2.3) and (2.2) as

$$\begin{array}{ll} \min & \mathcal{C}^T\mathcal{X} \\ \text{s.t.} & \mathcal{A}\mathcal{X} = \mathcal{B}, \\ & \mathcal{X} \in \mathcal{K}^*, \end{array} \tag{P}$$

and

$$(D) \qquad \begin{array}{ll} \max \mathcal{B}^T\mathcal{Y} & \\ \text{s.t.} & \mathcal{S} = \mathcal{C} - \mathcal{A}^T\mathcal{Y}, \\ & \mathcal{S} \in \mathcal{K}. \end{array}$$

This is the pair of problems (P) and (D) in Nesterov and Nemirovskii [25]. Unlike those in most optimization problems, $\mathcal{K} \neq \mathcal{K}^*$ (unless $p = 2$) here. Therefore, the symmetric primal-dual techniques, developed by, e.g., Nesterov and Todd [26], are no longer applicable. However, we can still use interior point algorithms to compute an ϵ-optimal solution of the problem in polynomial time.

3. Barrier functions for p-order cones.

The key to solving problems (P) and (D) is to construct a simple and efficient barrier function for K, when $p \neq 2$. This has been an open question. In this section we present two barrier functions and analyze their barrier parameters. To simplify notation, we will use s and z to denote vectors in R^d and use s_j and z_j to denote the jth single component of the vectors s and z. This notation is different than that used in the previous sections (where s_i stood for a vector in R^d) and will be used *only in this section.*

3.1. Barrier function I.

One barrier function is constructed from the following convex set:

$$G_p = \left\{ (s \in R^d, z \in R^d) : \; z_i \geq |s_i|^p, \; i = 1, \dots, d, \; \sum_{i=1}^d z_i = 1 \right\}.$$

A barrier function for this set is

$$f_p(s, z) = \sum_{i=1}^d (-2 \log z_i - \log(z_i^{2/p} - s_i^2)).$$

Its barrier parameter is $4 \cdot d$ (where each term in the summation has parameter 4); see Nesterov and Nemirovskii [25]. We will also use a special case of this function ($d = 1$):

$$f_{1,p}(s_i, z_i) = (-2 \log z_i - \log(z_i^{2/p} - s_i^2)),$$

whose barrier parameter is 4.

Consider the conic hull

$$\begin{aligned} K(G_p) &= \left\{ (t \in R_+, s \in R^d, z \in R^d) : \; t > 0, \; \left(\frac{s}{t}, \frac{z}{t}\right) \in G_p \right\} \\ &= \left\{ (t \in R_+, s \in R^d, z \in R^d) : \; t > 0, \; t^{p-1} z_i \geq |s_i|^p, \; \sum_{i=1}^d z_i = t \right\}, \end{aligned}$$

which is equivalent to K for (t, s). In what follows, we prove the following theorem.

THEOREM 3.1. *The function*

$$\hat{f}_p(t, s, z) = 25 \cdot \left(f_p \left(\frac{s}{t}, \frac{z}{t} \right) - 8d \log t \right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $200d$.

Proof. Notice that

$$\hat{f}_p(t,s,z) = 25 \cdot \left(f_p \left(\frac{s}{t}, \frac{z}{t} \right) - 8d \log t \right) = \sum_{i=1}^{d} 25 \cdot \left(f_{1,p} \left(\frac{s_i}{t}, \frac{z_i}{t} \right) - 8 \log t \right).$$

It is sufficient to prove that

$$\hat{f}_{1,p}(t \in R_+, s \in R, z \in R_+) = 25 \cdot (-2\log(z/t) - \log((z/t)^{2/p} - (s/t)^2) - 8 \log t)$$

$$= 25 \cdot (-2 \log z - \log(z^{2/p} t^{2(p-1)/p} - s^2) - 4 \log t)$$

is a logarithmically homogeneous self-concordant barrier function with parameter 200. Our proof follows from Proposition 5.1.4 of Nesterov and Nemirovskii [25]. Note that we have changed notation again: we have dropped the subscript i in s_i and z_i to simplify notation in the rest of this proof.

Let us fix $o = (t, z, s)$, $t > 0$, and $t^{p-1} z > |s|^p$, and let $w = (-d_t, d_z, d_s) \in R^3$. Let us compute the derivatives up to the order 3 of $\hat{f}_{1,p}$ at the point o in the direction w. For $\sigma = d_t/t$ and some $\alpha \in R$, we have

$$\frac{s + \alpha d_s}{t - \alpha d_t} = \frac{s}{t} + \frac{\alpha}{1 - \alpha\sigma} \left(\sigma \frac{s}{t} + \frac{d_s}{t} \right)$$

and

$$\frac{z + \alpha d_z}{t - \alpha d_t} = \frac{z}{t} + \frac{\alpha}{1 - \alpha\sigma} \left(\sigma \frac{z}{t} + \frac{d_z}{t} \right).$$

Let

$$\phi(\alpha) := f_{1,p} \left(\frac{s}{t} + \alpha \left(\sigma \frac{s}{t} + \frac{d_s}{t} \right), \frac{z}{t} + \alpha \left(\sigma \frac{z}{t} + \frac{d_z}{t} \right) \right).$$

Then

$$\hat{\phi}(\alpha) := \hat{f}_{1,p}(o + \alpha w) = 25 \cdot \left(\phi \left(\frac{\alpha}{1 - \alpha\sigma} \right) - 8 \log(1 - \alpha\sigma) - 8 \log t \right).$$

Therefore, we have

$$\hat{\pi}_1 := \nabla \hat{f}_{1,p}(o)[w] = \hat{\phi}'(0) = 25(8\sigma + \pi_1),$$

$$\hat{\pi}_2 := \nabla^2 \hat{f}_{1,p}(o)[w,w] = \hat{\phi}''(0) = 25(8\sigma^2 + 2\sigma\pi_1 + \pi_2),$$

$$\hat{\pi}_3 := \nabla^3 \hat{f}_{1,p}(o)[w,w,w] = \hat{\phi}'''(0) = 25(16\sigma^3 + 6\sigma^2\pi_1 + 6\sigma\pi_2 + \pi_3),$$

where

$$\pi_1 = \phi'(0), \quad \pi_2 = \phi''(0), \quad \pi_3 = \phi'''(0).$$

Since f_p is a self-concordant barrier with the barrier parameter 4, we have

$$\pi_2 \geq 0, \quad \pi_1^2 \leq 4\pi_2, \quad \text{and} \quad |\pi_3| \leq 2\pi_2^{3/2}.$$

Thus, we see

$$\hat{\pi}_2 \geq 25(8\sigma^2 - 4|\sigma|\sqrt{\pi_2} + \pi_2) \geq 25 \max(4\sigma^2, \pi_2/2) \geq 0,$$

which implies that $\hat{f}_{1,p}$ is a convex function. Now we need to prove that

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Note that, if $\sigma \leq 0$, we have

$$\begin{aligned}
\hat{\pi}_3/25 &= 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 8\sigma^3 + 3\sigma\pi_2 + \pi_3 \\
&\leq -8\sigma^3 + \pi_3 \\
&\leq 8|\sigma|^3 + 2\pi_2^{3/2} \\
&\leq \frac{8}{1000}\hat{\pi}_2^{3/2} + \frac{4\sqrt{2}}{125}\hat{\pi}_2^{3/2} \\
&\leq \frac{2}{25}\hat{\pi}_2^{3/2},
\end{aligned}$$

and

$$\begin{aligned}
-\hat{\pi}_3/25 &\leq -3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 3\sigma\pi_2 - \pi_3 \\
&\leq \frac{3}{250}\hat{\pi}_2^{3/2} - 3\sigma\pi_2 + 2\pi_2^{3/2}.
\end{aligned}$$

For the latter, consider the following maximum problem for any fixed $a \geq 0$:

$$\max \ -3xy^2 + 2y^3 : \ 8x^2 + 4xy + y^2 = a, \quad x \leq 0, \quad y \geq 0.$$

One can verify that the maximum value is below $8a^{3/2}$. Thus,

$$-3\sigma\pi_2 + 2\pi_2^{3/2} \leq \frac{8}{125}\hat{\pi}_2^{3/2},$$

which, together with the above inequality, implies that

$$-\hat{\pi}_3 \leq \left(\frac{3}{10} + \frac{8}{5}\right)\hat{\pi}_2^{3/2} \leq 2\hat{\pi}_2^{3/2}.$$

Thus, for the case of $\sigma \leq 0$, we have

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Now we consider the case when $\sigma > 0$:

$$\begin{aligned}
\hat{\pi}_3/25 &= 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) - 8\sigma^3 + 3\sigma\pi_2 + \pi_3 \\
&\leq 3\sigma(8\sigma^2 + 2\sigma\pi_1 + \pi_2) + 3\sigma\pi_2 + \pi_3 \\
&\leq \frac{3}{25}\sigma\hat{\pi}_2 + 3\sigma\pi_2 + 2\pi_2^{3/2} \\
&\leq \frac{3}{250}\hat{\pi}_2^{3/2} + 3\sigma\pi_2 + 2\pi_2^{3/2}.
\end{aligned}$$

Consider again the following maximum problem for any fixed $a > 0$:

$$\max \ 3xy^2 + 2y^3 : \ 8x^2 - 4xy + y^2 = a, \quad x \geq 0, \quad y \geq 0,$$

where the maximum value is below $8a^{3/2}$. Thus,

$$3\sigma\pi_2 + 2\pi_2^{3/2} \leq \frac{8}{125}\hat{\pi}_2^{3/2},$$

which, together with the above inequality, implies that

$$\hat{\pi}_3 \leq \left(\frac{3}{10} + \frac{8}{5}\right) \hat{\pi}_2^{3/2} \leq 2\hat{\pi}_2^{3/2}.$$

Also,

$$\begin{aligned} -\hat{\pi}_3/25 &\leq 8\sigma^3 - \pi_3 \\ &\leq 8|\sigma|^3 + 2\pi_2^{3/2} \\ &\leq \frac{8}{1000}\hat{\pi}_2^{3/2} + \frac{4\sqrt{2}}{125}\hat{\pi}_2^{3/2} \\ &\leq \frac{2}{25}\hat{\pi}_2^{3/2}. \end{aligned}$$

To summarize, we have proved that

$$|\hat{\pi}_3| \leq 2\hat{\pi}_2^{3/2}.$$

Finally, it is easy to verify that the barrier parameter is 200 for $\hat{f}_{1,p}$. Therefore, the barrier parameter for $\hat{f}_p$ is $200d$. This completes the proof of the theorem. ∎

3.2. Barrier function II. In this section, we first consider the set

$$\left\{ z \in R^d : \ z_i \geq 0, \ i = 1, \ldots, d, \ \sum_{i=1}^{d} z_i^p \leq 1 \right\}.$$

A barrier function for this set is

$$f_p(z) = -4 \log \left(1 - \sum_{i=1}^{d} z_i^p\right) - 4p^2 \sum_{i=1}^{d} \log z_i.$$

It is a convex barrier function. We present its barrier parameter in the following theorem.

THEOREM 3.2. *The function $f_p(z)$ is a logarithmically self-concordant function with parameter $4(1+p^2d)$ in the interior of the set $\{z \in R^d : \ \sum_{i=1}^{d} z_i^p \leq 1, \ z \geq 0\}$.*

Proof. Let $z > 0$ and $\delta = \sum_{i=1}^{d} z_i^p < 1$ be given, and let $w \in R^d$. Let

$$\phi(\alpha) := f_p(z + \alpha w).$$

We compute the derivatives up to the order 3 of f_p at the point z in the direction w:

$$\begin{aligned} \pi_1/4 &:= \nabla f_p(z)[w]/4 = \phi'(0)/4 \\ &= \frac{p}{1-\delta} \sum_{i=1}^{d} w_i z_i^{p-1} - p^2 \sum_{i=1}^{d} w_i z_i^{-1}, \end{aligned}$$

$$\begin{aligned} \pi_2/4 &:= \nabla^2 f_p(z)[w,w]/4 = \phi''(0)/4 \\ &= \frac{p^2}{(1-\delta)^2} \left(\sum_{i=1}^{d} w_i z_i^{p-1}\right)^2 + \frac{p(p-1)}{1-\delta} \sum_{i=1}^{d} w_i^2 z_i^{p-2} + p^2 \sum_{i=1}^{d} w_i^2 z_i^{-2}, \end{aligned}$$

and

$$\pi_3/4 := \nabla^3 f_p(z)[w,w,w]/4 = \phi'''(0)/4$$

$$= \frac{2p^3}{(1-\delta)^3}\left(\sum_{i=1}^{d} w_i z_i^{p-1}\right)^3 + \frac{2p^2(p-1)}{(1-\delta)^2}\left(\sum_{i=1}^{d} w_i z_i^{p-1}\right)\left(\sum_{i=1}^{d} w_i^2 z_i^{p-2}\right)$$

$$+\frac{p^2(p-1)}{(1-\delta)^2}\left(\sum_{i=1}^{d} w_i z_i^{p-1}\right)\left(\sum_{i=1}^{d} w_i^2 z_i^{p-2}\right) + \frac{p(p-1)(p-2)}{1-\delta}\sum_{i=1}^{d} w_i^3 z_i^{p-3} + p^2\sum_{i=1}^{d} w_i^3 z_i^{-3}.$$

Now let

$$\phi_1 = \frac{p}{1-\delta}\sum_{i=1}^{d} w_i z_i^{p-1}, \qquad \phi_2 = p^2 \sum_{i=1}^{d} w_i z_i^{-1},$$

$$\phi_3 = \sqrt{\frac{p(p-1)}{1-\delta}\sum_{i=1}^{d} w_i^2 z_i^{p-2}}, \quad \text{and} \quad \phi_4 = \sqrt{p^2 \sum_{i=1}^{d} w_i^2 z_i^{-2}}.$$

Then,

$$\pi_1/4 = \phi_1 - \phi_2,$$

$$\pi_2/4 = \phi_1^2 + \phi_3^2 + \phi_4^2 > 0,$$

and

$$\begin{aligned} |\pi_3/4| &\le |2\phi_1^3 + 3\phi_1\phi_3^2| + \frac{|p-2|}{p}\phi_3^2\phi_4 + \frac{1}{p}\phi_4^3 \\ &\le 2|\phi_1|(\phi_1^2+\phi_3^2) + |\phi_1|\phi_3^2 + \phi_3(\phi_3^2+\phi_4^2) \\ &\le \pi_2^{3/2}/4 + \pi_2^{3/2}/8 + \pi_2^{3/2}/8 \\ &= \pi_2^{3/2}/2; \end{aligned}$$

i.e.,

$$|\pi_3| \le 2\pi_2^{3/2}.$$

Finally, we need to prove

$$\pi_1^2 \le 4(1+p^2 d)\pi_2.$$

Note that

$$\pi_1^2 = 16(\phi_1 - \phi_2)^2.$$

Thus, if $\phi_1 = 0$, then

$$\pi_1^2 = 16\phi_2^2 \le 16p^2 d\phi_4^2 \le 4p^2 d\pi_2.$$

Consider $\phi_1 \ne 0$ and let $\bar{\delta} = \frac{|\phi_2|}{|\phi_1|}$. Then

$$\pi_2/4 \ge \phi_1^2 + \frac{\phi_2^2}{p^2 d} = \left(1 + \frac{\bar{\delta}^2}{p^2 d}\right)\phi_1^2.$$

On the other hand,

$$\pi_1^2 \leq 16(1+\bar{\delta})^2\phi_1^2.$$

Therefore,

$$\begin{aligned}\pi_1^2 &\leq \frac{(1+\bar{\delta})^2}{1+\frac{\bar{\delta}^2}{p^2 d}} \cdot 4\pi_2\\ &\leq 4(1+p^2 d)\pi_2,\end{aligned}$$

where the last inequality holds since

$$(1+\bar{\delta})^2 \leq (1+p^2 d)\left(1+\frac{\bar{\delta}^2}{p^2 d}\right). \qquad \square$$

Similarly, we can prove the following corollary.

Corollary 3.3. *If $p \leq 3$, then*

$$f_p(z) = -4\log\left(1-\sum_{i=1}^d z_i^p\right) - 4\sum_{i=1}^d \log z_i$$

is a convex, logarithmically self-concordant barrier function with parameter $4(1+d)$ in the interior of the set $\{z \in R^d : \sum_{i=1}^d z_i^p \leq 1,\ z \geq 0\}$. □

We now consider the set

$$G_p = \left\{(s \in R^d, z \in R^d) :\ z_i \geq |s_i|,\ i = 1, \ldots, d,\ \sum_{i=1}^d z_i^p \leq 1\right\}.$$

A barrier function for this set is

$$f_p(s,z) = -4\log\left(1-\sum_{i=1}^d z_i^p\right) - 4p^2\sum_{i=1}^d \log z_i - \sum_{i=1}^d \log(z_i^2 - s_i^2).$$

The barrier parameter of this function is $4(1+p^2d)+2d$, since the first two summations have parameter $4(1+p^2d)$ and the last summation has parameter $2d$.

Again consider the conic hull

$$\begin{aligned}K(G_p) &= \left\{(t, s \in R^d, z \in R^d) :\ t > 0,\ \left(\frac{s}{t}, \frac{z}{t}\right) \in G_p\right\}\\ &= \left\{(t, s \in R^d, z \in R^d) :\ t > 0,\ z_i \geq |s_i|,\ \sum_{i=1}^d z_i^p \leq t^p\right\},\end{aligned}$$

which is also equivalent to K for (t,s). Again, similar to Theorem 3.1, we can prove the following theorem.

Theorem 3.4. *The function*

$$\hat{f}_p(t,s,z) = \theta^2 \cdot \left(f_p\left(\frac{s}{t}, \frac{z}{t}\right) - (8(1+p^2 d)+4d)\log t\right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $\theta^2(8(1+p^2d)+4d)$, where θ is a positive constant, say, $\theta = 5$.

We can simplify the above barrier function further. Note that $z_i \geq |s_i|$ implies $z_i \geq 0$, and the Hessian of $-\log(z_i^2 - s_i^2)$ is

$$H(z_i, s_i) = \frac{2}{z_i^2 - s_i^2} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{4}{(z_i^2 - s_i^2)^2} \begin{pmatrix} z_i^2 & -z_i s_i \\ -z_i s_i & s_i^2 \end{pmatrix}.$$

For $z_i^2 > s_i^2$, we have

$$H(z_i, s_i) - \begin{pmatrix} z_i^{-2} & 0 \\ 0 & 0 \end{pmatrix} = \frac{2}{(z_i^2 - s_i^2)^2} \begin{pmatrix} z_i^2 + s_i^2 - \frac{(z_i^2 - s_i^2)^2}{2z_i^2} & -2z_i s_i \\ -2z_i s_i & z_i^2 + s_i^2 \end{pmatrix}.$$

We can show, from $(z_i^2 - s_i^2)^3 > 0$, that the two-by-two matrix in the right-hand-side expression is positive semidefinite. Thus, we have the following corollary, whose proof is similar to that for Theorem 3.2.

Corollary 3.5. *If $p \leq 3$, then*

$$f_p(s, z) = -4 \log \left(1 - \sum_{i=1}^{d} z_i^p\right) - 4 \sum_{i=1}^{d} \log(z_i^2 - s_i^2)$$

is a convex, logarithmically self-concordant barrier function with parameter $4(1 + 2d)$ in the interior of the set $\{(s \in R^d, z \in R^d) : \sum_{i=1}^{d} z_i^p \leq 1,\ z \geq |s_i|\}$. The function

$$\hat{f}_p(t, s, z) = 25 \cdot \left(f_p\left(\frac{s}{t}, \frac{z}{t}\right) - 8(1 + 2d) \log t\right)$$

$$= 100 \left(-\log \left(t^p - \sum_{i=1}^{d} z_i^p\right) - \sum_{i=1}^{d} \log(z_i^2 - s_i^2) - (2 - p + 2d) \log t\right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_p)$, where the barrier parameter is $200(1 + 2d)$. □

Symmetrically, we may consider the convex set

$$G_q = \left\{(x \in R^d, v \in R^d) :\ v_i \geq |x_i|^q,\ i = 1, \ldots, d,\ \sum_{i=1}^{d} v_i = 1\right\}$$

and the conic hull

$$K(G_q) = \left\{(\tau, x \in R^d, v \in R^d) :\ \tau > 0,\ \left(\frac{x}{\tau}, \frac{v}{\tau}\right) \in G_q\right\}$$

$$= \left\{(\tau, x \in R^d, v \in R^d) :\ \tau > 0,\ v_i \geq |x_i|,\ \sum_{i=1}^{d} v_i^q \leq \tau^q\right\},$$

which is also equivalent to K^* for (τ, x). A direct application of Corollary 3.5 leads to the following corollary.

Corollary 3.6. *If $q \leq 3$, then*

$$f_q(x, v) = -4 \log \left(1 - \sum_{i=1}^{d} v_i^q\right) - 4 \sum_{i=1}^{d} \log(v_i^2 - x_i^2)$$

is a convex, logarithmically self-concordant barrier function with parameter $4(1 + 2d)$ in the interior of the set $\{(x \in R^d, v \in R^d) :\ \sum_{i=1}^{d} v_i^q \leq 1,\ v \geq |x_i|\}$. The function

$$\hat{f}_q(\tau, x, v) = 25 \cdot \left(f_q\left(\frac{x}{\tau}, \frac{v}{\tau}\right) - 8(1 + 2d) \log \tau\right)$$

$$= 100 \left(-\log \left(\tau^q - \sum_{i=1}^{d} v_i^q \right) - \sum_{i=1}^{d} \log(v_i^2 - x_i^2) - (2 - q + 2d) \log \tau \right)$$

is a logarithmically homogeneous self-concordant barrier for $K(G_q)$, where the barrier parameter is $200(1 + 2d)$. □

3.3. Legendre transformations. Although it has a higher barrier parameter, barrier function II possesses a structure similar to the barrier function for the second-order cone, which we will use in the following analyses. Since we can solve both (P) and (D) using either the barrier function for the p-order cone in (D) or the barrier function for the q-order cone in (P), we will make the following conventions in this paper:

- When $p > 3$, use the barrier function of the q-order cone in (P).
- When $1 \le p \le 3$, use the barrier function of the p-order cone in (D).

It follows from Nesterov and Nemirovskii [25] that the Legendre transformation of the logarithmically homogeneous self-concordant barrier function in Corollary 3.5,

$$\hat{f}_p^*(\tau, x \in R^d) = \sup\{-\tau \cdot t - x^T s - \hat{f}_p(t, s, z) : \ (t, s, z) \in K(G_p)\},$$

is a logarithmically homogeneous self-concordant barrier for K^* with the same parameter $200(1 + 2d)$. It seems hard to find an explicit form of $\hat{f}_p^*(\tau, x)$. Fortunately, for the interior point algorithm presented in the next section, we do not need such an explicit formula.

We need only the initial barrier function value at a primal initial point in our complexity analysis. We will set the initial point $\tau = 1$ and $x = 0$. Thus, we need to evaluate

$$\hat{f}_p^*(1, 0) = \sup\{-t - \hat{f}_p(t, s, z) : \ (t, s, z) \in K(G_p)\}.$$

This is a convex optimization problem with an analytical solution

$$t^* = 200(1 + 2d), \quad s^* = 0, \quad z_j^* = \left(\frac{2}{p + 2d}\right)^{1/p} t^*, \quad j = 1, \ldots, d.$$

Thus,

(3.1)

$$\hat{f}_p^*(1, 0) = 200(1 + 2d)(\log(200(1 + 2d)) - 1) + 100 \log \frac{p}{p + 2d} + \frac{200d}{p} \log \frac{2}{p + 2d}.$$

It also follows from Nesterov and Nemirovskii [25] that the Legendre transformation of the logarithmically homogeneous self-concordant barrier function in Corollary 3.6,

$$\hat{f}_q^*(t, s \in R^d) = \sup\{-\tau \cdot t - x^T s - \hat{f}_q(\tau, x, v) : \ (\tau, x, v) \in K(G_q)\},$$

is a logarithmically homogeneous self-concordant barrier for K with the same parameter $200(1 + 2d)$. It seems hard to find the exact value of $\hat{f}_q^*(t, s)$ for $s \neq 0$. In the following, we will find an explicit formula for $\hat{f}_q^*(c, 0)$ for any $c > 0$ and prove that $\hat{f}_q^*(t, s) \le \hat{f}_q^*(t - ||s||_p, 0)$ for $(t, s) \in K$.

Note that

$$\hat{f}_q^*(c, 0) = \sup\{-\tau c - \hat{f}_q(\tau, x, v) : \ (\tau, x, v) \in K(G_q)\}.$$

This is a convex optimization problem with an analytical solution

$$\tau^* = 200(1+2d)/c, \quad x^* = 0, \quad v_j^* = \left(\frac{2}{q+2d}\right)^{1/q} \tau^*, \quad j = 1, \dots, d.$$

Thus,

$$\hat{f}_q^*(c,0) = 200(1+2d)(\log(200(1+2d)/c) - 1) + 100 \log \frac{q}{q+2d} + \frac{200d}{q} \log \frac{2}{q+2d}. \tag{3.2}$$

For any $(t,s) \in K$ and $(\tau, x) \in K^*$, we have

$$t \geq ||s||_p, \qquad \tau \geq ||x||_q.$$

Since

$$|s^T x| \leq ||s||_p ||x||_q,$$

we have

$$\begin{aligned} -t\tau - s^T x &\leq -t\tau + ||s||_p ||x||_q \\ &\leq -t\tau + ||s||_p \tau \\ &= -\tau(t - ||s||_p). \end{aligned}$$

Therefore,

$$\hat{f}_q^*(t,s) \leq \hat{f}_q^*(t - ||s||_p, 0). \tag{3.3}$$

4. A primal-dual potential reduction algorithm.

In this section, we will present a primal-dual potential reduction algorithm for computing ϵ-optimal solutions for (P) and (D) in polynomial time. We will use either dual scaling or primal scaling depending on the value of p.

4.1. Use dual scaling when $p \in [1,3]$.

When $p \in [1,3]$, we may solve (P) and (D) using the barrier function for the p-order cone. Let

$$F_p^*(\mathcal{X}) = \sum_{i=1}^m \hat{f}_p^*(\tau_i, x_i) \quad \text{and} \quad F_p(\mathcal{S}, \mathcal{Z}) = \sum_{i=1}^m \hat{f}_p(t_i, s_i, z_i \in R^d), \tag{4.1}$$

where

$$\mathcal{Z} := (z_1; z_2; \dots; z_m).$$

They are logarithmically homogeneous self-concordant barriers for (P) and (D), where the barrier parameter is $\theta := 200(1+2d)m$. A primal-dual potential function for the pair (P) and (D) is

$$\phi_\rho(\mathcal{X}, \mathcal{S}, \mathcal{Z}) := \rho \log\langle \mathcal{X}, \mathcal{S} \rangle + F_p^*(\mathcal{X}) + F_p(\mathcal{S}, \mathcal{Z}) + \theta, \tag{4.2}$$

where $\rho = \theta + \gamma\sqrt{\theta}$, $\gamma \geq 1$. Note that

$$\langle \mathcal{X}, \mathcal{S} \rangle = \mathcal{X}^T \mathcal{S} = \mathcal{C}^T \mathcal{X} - \mathcal{B}^T \mathcal{Y},$$

and from Nesterov and Nemirovskii [25],

$$\phi_\theta(\mathcal{X}, \mathcal{S}, \mathcal{Z}) \geq \theta \log \theta. \tag{4.3}$$

The main iteration of a potential reduction algorithm starts with a strictly feasible primal-dual pair $\mathcal{X}$ and $(\mathcal{Y}, \mathcal{S})$, i.e.,

$$\begin{array}{cc} \mathcal{A}\mathcal{X} = \mathcal{B}, & \mathcal{S} = \mathcal{C} - \mathcal{A}^T\mathcal{Y}, \\ \mathcal{X} \in \text{int}\mathcal{K}^*, \quad \text{and} & \mathcal{S} \in \text{int}\mathcal{K}. \end{array}$$

It computes a search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$ by solving a system of linear equations. After obtaining $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$, a new, strictly feasible, primal-dual pair $\mathcal{X}^+$ and $(\mathcal{Y}^+, \mathcal{S}^+)$ is generated from

$$\mathcal{X}^+ = \mathcal{X} + \alpha d_\mathcal{X}, \quad \mathcal{Y}^+ = \mathcal{Y} + \beta d_\mathcal{Y}, \quad \mathcal{S}^+ = \mathcal{S} + \beta d_\mathcal{S}, \quad \mathcal{Z}^+ = \mathcal{Z} + \beta d_\mathcal{Z},$$

for some step-sizes α and β, and

$$\phi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{Z}^+) \leq \phi_\rho(\mathcal{X}, \mathcal{S}, \mathcal{Z}) - \Omega(1).$$

The search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$ is determined by the following equations:

$$\mathcal{A}d_\mathcal{X} = 0, \quad d_\mathcal{S} = -\mathcal{A}^T d_\mathcal{Y} \quad \text{(feasibility)} \tag{4.4}$$

and

$$\begin{pmatrix} d_\mathcal{X} \\ 0 \end{pmatrix} + F_p''(\mathcal{S}, \mathcal{Z}) \begin{pmatrix} d_\mathcal{S} \\ d_\mathcal{Z} \end{pmatrix} = -\frac{\rho}{\mathcal{X}^T\mathcal{S}} \begin{pmatrix} \mathcal{X} \\ 0 \end{pmatrix} - F_p'(\mathcal{S}, \mathcal{Z}). \tag{4.5}$$

The *theoretical* dual-scaling potential reduction algorithm, with a specific choice of step-sizes α and β, can be described as follows (see section 4.5.3 of [25]).

```
ALGORITHM PDD
Let $\gamma$ and $\Delta$ be fixed constants such that $\gamma \geq 1$, $0 < \Delta < 1$, and $\frac{\gamma(\gamma(1-\Delta)-\Delta)}{1+\gamma} > \frac{\Delta^2}{2(1-\Delta)^2}$.
Step 1. Compute the search direction $(d_\mathcal{X}, d_\mathcal{Y}, d_\mathcal{S}, d_\mathcal{Z})$ using (4.4) and (4.5).
Step 2. Compute $\lambda = \sqrt{\begin{pmatrix} d_\mathcal{S} \\ d_\mathcal{Z} \end{pmatrix}^T F_p''(\mathcal{S}, \mathcal{Z}) \begin{pmatrix} d_\mathcal{S} \\ d_\mathcal{Z} \end{pmatrix}}$.
        If $\lambda > \Delta$, then
          $\mathcal{X}^+ = \mathcal{X}$                                  (primal step-size $\alpha = 0$),
          $\mathcal{S}^+ = \mathcal{S} + \frac{1}{1+\lambda} d_\mathcal{S}$            (dual step-size $\beta = \frac{1}{1+\lambda}$),
          $\mathcal{Y}^+ = \mathcal{Y} + \frac{1}{1+\lambda} d_\mathcal{Y}$,
          $\mathcal{Z}^+ = \mathcal{Z} + \frac{1}{1+\lambda} d_\mathcal{Z}$,
        else
          $\mathcal{X}^+ = \mathcal{X} + \frac{\langle \mathcal{S}, \mathcal{X} \rangle}{\rho} d_\mathcal{X}$   (primal step-size $\alpha = \frac{\langle \mathcal{S}, \mathcal{X} \rangle}{\rho}$),
          $\mathcal{S}^+ = \mathcal{S}$                                  (dual step-size $\beta = 0$),
          $\mathcal{Y}^+ = \mathcal{Y}$,
          $\mathcal{Z}^+ = \mathcal{Z}$.
        endif
```

According to Nesterov and Nemirovskii [25], we have the following theorem.

THEOREM 4.1. *Starting from any strictly feasible primal solution $\mathcal{X}^0$ and strictly dual feasible solution $(\mathcal{Y}^0; \mathcal{S}^0)$, an ϵ-optimal solution $(\mathcal{X}, \mathcal{Y}, \mathcal{S})$ to problem* (1.1)*, can be obtained by repeated explication of Algorithm PDD for at most* $O(\gamma\sqrt{\theta}\log(\langle \mathcal{X}^0, \mathcal{S}^0 \rangle/\epsilon) + \phi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{Z}^0) - \theta\log\theta)$ *iterations.* □

In practice, one usually finds the largest step-sizes $\bar{\alpha}$ and $\bar{\beta}$ such that

$$\mathcal{X}+\bar{\alpha}d_{\mathcal{X}}\in\mathcal{K}^* \quad \text{and} \quad \mathcal{S}+\bar{\beta}d_{\mathcal{S}}\in\mathcal{K}, \tag{4.6}$$

then takes $\alpha\in[0,\bar{\alpha}]$ and $\beta\in[0,\bar{\beta}]$ via a line-search to minimize $\phi_\rho(\mathcal{X}^+,\mathcal{S}^+,\mathcal{Z}^+)$, or simply chooses

$$\alpha=(0.5\sim 0.99)\bar{\alpha} \quad \text{and} \quad \beta=(0.5\sim 0.99)\bar{\beta} \tag{4.7}$$

as long as ϕ_ρ is reduced.

4.2. Use primal scaling when $p\in[3,\infty]$. The dual scaling algorithm is good when p is small ($1\le p\le 3$). For larger values of p, the barrier parameter for the p-order cone becomes larger. In this case, it is better to use the barrier function of the q-order cone in (P), where $\frac{1}{p}+\frac{1}{q}=1$. It is clear that $q\in[1,2]$ whenever $p\in[2,\infty]$.

Let

$$F_q(\mathcal{X},\mathcal{V})=\sum_{i=1}^m \hat{f}_q(\tau_i,x_i,v_i\in R^d) \quad \text{and} \quad F_q^*(\mathcal{S})=\sum_{i=1}^m \hat{f}_q^*(t_i,s_i), \tag{4.8}$$

where

$$\mathcal{V}:=(v_1;v_2;\dots;v_m).$$

They are logarithmically homogeneous self-concordant barriers for (P) and (D), where the barrier parameter is $\theta:=200(1+2d)m$. A primal-dual potential function for the pair (P) and (D) is

$$\psi_\rho(\mathcal{X},\mathcal{S},\mathcal{V}):=\rho\log\langle\mathcal{X},\mathcal{S}\rangle+F_q(\mathcal{X},\mathcal{V})+F_q^*(\mathcal{S})+\theta, \tag{4.9}$$

where $\rho=\theta+\gamma\sqrt{\theta}$, $\gamma\ge 1$. Note that

$$\langle\mathcal{X},\mathcal{S}\rangle=\mathcal{X}^T\mathcal{S}=\mathcal{C}^T\mathcal{X}-\mathcal{B}^T\mathcal{Y},$$

and from Nesterov and Nemirovskii [25]

$$\psi_\theta(\mathcal{X},\mathcal{S},\mathcal{V})\ge\theta\log\theta. \tag{4.10}$$

The main iteration of a potential reduction algorithm starts with a strictly feasible primal-dual pair $(\mathcal{X},\mathcal{V})$ and $(\mathcal{Y},\mathcal{S})$, i.e.,

$$\begin{aligned} \mathcal{A}\mathcal{X}&=\mathcal{B}, & \mathcal{S}&=\mathcal{C}-\mathcal{A}^T\mathcal{Y},\\ \mathcal{X}&\in\text{int}\mathcal{K}, \quad \text{and} & \mathcal{S}&\in\text{int}\mathcal{K}^*. \end{aligned}$$

It computes a search direction $(d_{\mathcal{X}},d_{\mathcal{Y}},d_{\mathcal{S}},d_{\mathcal{V}})$ by solving a system of linear equations. After obtaining $(d_{\mathcal{X}},d_{\mathcal{Y}},d_{\mathcal{S}},d_{\mathcal{V}})$, a new, strictly feasible, primal-dual pair $\mathcal{X}^+$ and $(\mathcal{Y}^+,\mathcal{S}^+)$ is generated from

$$\mathcal{X}^+=\mathcal{X}+\alpha d_{\mathcal{X}}, \quad \mathcal{Y}^+=\mathcal{Y}+\beta d_{\mathcal{Y}}, \quad \mathcal{S}^+=\mathcal{S}+\beta d_{\mathcal{S}}, \quad \mathcal{V}^+=\mathcal{V}+\alpha d_{\mathcal{V}}$$

for some step-sizes α and β, and

$$\psi_\rho(\mathcal{X}^+,\mathcal{S}^+,\mathcal{V}^+)\le\psi_\rho(\mathcal{X},\mathcal{S},\mathcal{V})-\Omega(1).$$

The search direction $(d_{\mathcal{X}},d_{\mathcal{Y}},d_{\mathcal{S}},d_{\mathcal{V}})$ is determined by (4.4) and the following equations:

$$\begin{pmatrix} d_{\mathcal{S}}\\ 0\end{pmatrix}+F_q''(\mathcal{X},\mathcal{V})\begin{pmatrix} d_{\mathcal{X}}\\ d_{\mathcal{V}}\end{pmatrix}=-\frac{\rho}{\mathcal{X}^T\mathcal{S}}\begin{pmatrix}\mathcal{S}\\ 0\end{pmatrix}-F_q'(\mathcal{X},\mathcal{V}). \tag{4.11}$$

The algorithm generates an ϵ-optimal solution $(\mathcal{X}, \mathcal{S}, \mathcal{V})$, i.e.,

$$\langle \mathcal{X}, \mathcal{S} \rangle \leq \epsilon,$$

in a guaranteed $O(\gamma\sqrt{\theta}\log(\langle \mathcal{X}^0, \mathcal{S}^0\rangle/\epsilon) + \psi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{V}^0) - \theta\log\theta)$ iterations.

In practice, one usually finds the largest step-sizes $\bar{\alpha}$ and $\bar{\beta}$ such that

$$\mathcal{X} + \bar{\alpha} d_{\mathcal{X}} \in \mathcal{K} \quad \text{and} \quad \mathcal{S} + \bar{\beta} d_{\mathcal{S}} \in \mathcal{K}^*, \tag{4.12}$$

then takes $\alpha \in [0, \bar{\alpha}]$ and $\beta \in [0, \bar{\beta}]$ via a line-search to minimize $\psi_\rho(\mathcal{X}^+, \mathcal{S}^+, \mathcal{V}^+)$, or simply chooses

$$\alpha = (0.5 \sim 0.99)\bar{\alpha} \quad \text{and} \quad \beta = (0.5 \sim 0.99)\bar{\beta} \tag{4.13}$$

as long as ψ_ρ is reduced.

The *theoretical* potential reduction algorithm using primal scaling can be described as follows.

Algorithm PDP.

Let γ and Δ be fixed constants such that $\gamma \geq 1$, $0 < \Delta < 1$, and $\frac{\gamma(\gamma(1-\Delta)-\Delta)}{1+\gamma} > \frac{\Delta^2}{2(1-\Delta)^2}$.

```
Step 1. Compute the search direction $(d_{\mathcal{X}}, d_{\mathcal{Y}}, d_{\mathcal{S}}, d_{\mathcal{V}})$ using (4.4) and (4.11).
Step 2. Compute $\lambda = \sqrt{\begin{pmatrix} d_{\mathcal{X}} \\ d_{\mathcal{V}} \end{pmatrix}^T F_q''(\mathcal{X}, \mathcal{V}) \begin{pmatrix} d_{\mathcal{X}} \\ d_{\mathcal{V}} \end{pmatrix}}$.
        If $\lambda > \Delta$, then
          $\mathcal{X}^+ = \mathcal{X} + \frac{1}{1+\lambda} d_{\mathcal{X}}$     (primal step-size $\alpha = \frac{1}{1+\lambda}$),
          $\mathcal{V}^+ = \mathcal{V} + \frac{1}{1+\lambda} d_{\mathcal{V}}$,
          $\mathcal{S}^+ = \mathcal{S}$     (dual step-size $\beta = 0$),
          $\mathcal{Y}^+ = \mathcal{Y}$,
        else
          $\mathcal{X}^+ = \mathcal{X}$     (primal step-size $\alpha = 0$),
          $\mathcal{V}^+ = \mathcal{V}$,
          $\mathcal{S}^+ = \mathcal{S} + \frac{\langle \mathcal{S}, \mathcal{X} \rangle}{\rho} d_{\mathcal{S}}$     (dual step-size $\beta = \frac{\langle \mathcal{S}, \mathcal{X} \rangle}{\rho}$),
          $\mathcal{Y}^+ = \mathcal{Y} + \frac{\langle \mathcal{S}, \mathcal{X} \rangle}{\rho} d_{\mathcal{Y}}$.
        endif
```

According to Nesterov and Nemirovskii [25], we have the following theorem.

Theorem 4.2. *Starting from any strictly feasible primal solution* $(\mathcal{X}^0; \mathcal{V}^0)$ *and strictly dual feasible solution* $(\mathcal{Y}^0; \mathcal{S}^0)$*, an* ϵ*-optimal solution to problem* (1.1) *can be obtained by repeated application of Algorithm PDP for at most* $O(\gamma\sqrt{\theta}\log(\langle \mathcal{X}^0, \mathcal{S}^0\rangle/\epsilon) + \psi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{V}^2) - \theta\log\theta)$ *iterations.* □

5. Complexity and implementation.

As we have seen, the number of iterations required to compute an ϵ-optimal solution to problem (2.1) depends on the initial point $(\mathcal{X}^0, \mathcal{S}^0, \mathcal{Z}^0)$. In this section, we discuss the initial point selection and other computational issues for solving problem (2.1) using the algorithms presented in section 4.

5.1. Initial point for dual scaling.

The algorithms discussed in the previous section all require a pair of strictly primal-dual interior feasible solutions. In the following, we give one such pair.

Let

$$\bar{c} = \max_{1 \le i \le m} \|c_i\|,$$

and for $i = 1, 2, \ldots, m$, let

$$u^0 = 0, \qquad t_i^0 = \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2} \left(\frac{p+2d}{2}\right)^{\frac{1}{p}},$$
$$s_i^0 = c_i, \qquad z_i^0 = \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2}\; e \in R^d,$$

and

$$\tau_i^0 = 1, \quad x_i^0 = 0 \in R^d.$$

Then, one can verify that $\mathcal{X}^0$ is an interior feasible solution to (P) and that $\mathcal{S}^0$ and $\mathcal{Y}^0$ form an interior feasible solution to (D). One can also verify that

$$\begin{aligned}\langle \mathcal{X}^0, \mathcal{S}^0 \rangle = (\mathcal{X}^0)^T \mathcal{S}^0 = \sum_{i=1}^m t_i^0 \tau_i^0 &= \left(\frac{p+2d}{2}\right)^{\frac{1}{p}} \sum_{i=1}^m \sqrt{\|c_i\|^2 + m(1+2d)\bar{c}^2} \\ &\le \left(\frac{p+2d}{2}\right)^{\frac{1}{p}} \bar{c} m \sqrt{1 + m(1+2d)},\end{aligned}$$

and the initial value

$$\begin{aligned}\hat{f}_p(t_i^0, s_i^0, z_i^0) &\le 100 \left(-(2+2d)\log t_i^0 - \log \frac{p}{p+2d} - d \log m(1+2d)\bar{c}^2\right) \\ &\le -100(1+2d)\log m(1+2d)\bar{c}^2 - \frac{200(1+d)}{p} \log \frac{p+2d}{2} - 100 \log \frac{p}{p+2d}.\end{aligned}$$

From this inequality and (3.1), we have

$$\begin{aligned}&F_p^*(\mathcal{X}^0) + F_p(\mathcal{S}^0, \mathcal{Z}^0) \\ &= \sum_{i=1}^m \hat{f}_p^*(1, 0) + \sum_{i=1}^m \hat{f}_p(t_i^0, s_i^0, z_i^0) \\ &\le \theta(\log(200(1+2d)) - 1) - 100(1+2d)m \log m(1+2d)\bar{c}^2 + \frac{\theta}{p} \log \frac{2}{p+2d}.\end{aligned}$$

Thus, from these inequalities,

$$\begin{aligned}&\phi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{V}^0) - \theta \log \theta \\ &= \theta \log \langle \mathcal{X}^0, \mathcal{S}^0 \rangle + F_p^*(\mathcal{X}^0) + F_p(\mathcal{S}^0, \mathcal{Z}^0) + \theta - \theta \log \theta \\ &\le \theta \log(\bar{c} m \sqrt{1 + m(1+2d)}) - 100(1+2d)m \log m(1+2d)\bar{c}^2 \\ &\quad + \theta \log(200(1+2d)) - \theta \log \theta \\ &= 100(1+2d)m \log(1 + m(1+2d)) - 100(1+2d)m \log m(1+2d) \\ &= 100(1+2d)m \log \left(1 + \frac{1}{m(1+2d)}\right) \\ &\le 100.\end{aligned}$$

With this initial point, we have the following corollary.

COROLLARY 5.1. *Let the initial feasible primal solution $\mathcal{X}^0$ and dual feasible solution $(\mathcal{Y}^0, \mathcal{S}^0, \mathcal{Z}^0)$ be given as above, and let $1 \le p \le 3$. Then, an ϵ-optimal solution to problem* (2.1) *can be obtained by the (dual) potential reduction algorithm in at most*

$$O\left(\gamma\sqrt{200(1+2d)m}\,(\log(\bar{c}/\epsilon) + \log(md))\right)$$

iterations, where

$$\bar{c} = \max_{1\le i\le m} \|c_i\|. \qquad \square$$

5.2. Initial point for primal scaling. In this section, we give a pair of strictly primal-dual interior feasible solutions for the primal scaling algorithm. It is assumed that $p \ge 2$.

Let

$$\bar{c} = \max_{1\le i\le m} \|c_i\|,$$

and for $i = 1, 2, \ldots, m$, let

$$u^0 = 0, \quad s_i^0 = c_i, \quad t_i^0 = ||c_i||_p + m(1+2d)\bar{c}$$

and

$$\tau_i^0 = 1, \quad x_i^0 = 0 \in R^d, \quad v_i^0 = \left(\frac{2}{q+2d}\right)^{\frac{1}{q}} e.$$

Then, one can verify that $\mathcal{X}^0$ is an interior feasible solution to (P) and that $\mathcal{S}^0$ and $\mathcal{Y}^0$ form an interior feasible solution to (D). One can also verify that

$$\langle \mathcal{X}^0, \mathcal{S}^0\rangle = (\mathcal{X}^0)^T\mathcal{S}^0 = \sum_{i=1}^m t_i^0\tau_i^0 \le m\bar{c} + m^2(1+2d)\bar{c},$$

and the initial value

$$\begin{aligned}
\hat{f}_q(\tau_i^0, x_i^0, v_i^0) &= 100\left(-\log\left((\tau_i^0)^q - \sum_{j=1}^d (v_j^0)^q\right) - \sum_{j=1}^d \log(v_j^0)^2 - (2-q+2d)\log\tau_i^0\right)\\
&= 100\left(-\log\left(1 - \frac{2d}{q+2d}\right) - \frac{2d}{q}\log\frac{2}{q+2d}\right)\\
&= 100\left(-\log\left(\frac{q}{q+2d}\right) - \frac{2d}{q}\log\frac{2}{q+2d}\right),\\
\hat{f}_i^*(t_i, s_i) &\le 200(1+2d)\left(\log\frac{200(1+2d)}{m(1+2d)\bar{c}} - 1\right)\\
&\quad + 100\left(\log\frac{q}{q+2d} + \frac{2d}{q}\log\frac{2}{q+2d}\right).
\end{aligned}$$

Therefore,

$$\hat{f}_q(\tau_i, x_i, v_i) + \hat{f}_i^*(t_i, s_i) \le 200(1+2d)\left(\log\frac{200(1+2d)}{m(1+2d)\bar{c}} - 1\right).$$

From these inequalities, we have

$$\begin{aligned} &F_q^*(\mathcal{S}^0) + F_q(\mathcal{X}^0, \mathcal{V}^0) \\ &= \sum_{i=1}^m \hat{f}_q^*(t_i^0, s_i^0) + \sum_{i=1}^m \hat{f}_q(\tau_i^0, x_i^0, v_i^0) \\ &\le \theta \left(\log \frac{200(1+2d)}{m(1+2d)\bar{c}} - 1 \right). \end{aligned}$$

Thus,

$$\begin{aligned} &\psi_\theta(\mathcal{X}^0, \mathcal{S}^0, \mathcal{V}^0) - \theta \log \theta \\ &= \theta \log \langle \mathcal{X}^0, \mathcal{S}^0 \rangle + F_q^*(\mathcal{S}^0) + F_q(\mathcal{X}^0, \mathcal{V}^0) + \theta - \theta \log \theta \\ &\le \theta \log(m\bar{c} + m^2(1+2d)\bar{c}) + \theta \left(\log \frac{200(1+2d)}{m(1+2d)\bar{c}} - 1 \right) + \theta - \theta \log \theta \\ &= \theta \log \left(1 + \frac{1}{m(1+2d)} \right) \\ &\le 200. \end{aligned}$$

With this initial point, we have the following corollary.

COROLLARY 5.2. *Let the initial feasible primal solution* $(\mathcal{X}^0, \mathcal{V}^0)$ *and dual feasible solution* $(\mathcal{Y}^0, \mathcal{S}^0)$ *be given as above, and let* $2 \le q \le 3$. *Then, an* ϵ*-optimal solution to problem* (2.1) *can be obtained by the (primal) potential reduction algorithm in at most*

$$O\left(\gamma \sqrt{200(1+2d)m} \, (\log(\bar{c}/\epsilon) + \log(md)) \right)$$

iterations, where

$$\bar{c} = \max_{1 \le i \le m} \|c_i\|. \qquad \square$$

5.3. Search direction. At each step of the potential reduction algorithm, we need to compute the search direction $d_\mathcal{X}$, $d_\mathcal{S}$, $d_\mathcal{Y}$, and $d_\mathcal{Z}$ by solving a system of linear equations. In what follows, we will show that this can be further simplified, taking advantage of the special structure of the problem.

Consider the search direction defined by dual scaling (4.5). For $i = 1, \ldots, m$, it can be decomposed as

$$\begin{pmatrix} d_{\tau_i} \\ 0_d \\ d_{x_i} \end{pmatrix} + \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial s_i} \end{pmatrix} \begin{pmatrix} d_{t_i} \\ d_{z_i} \\ d_{s_i} \end{pmatrix}$$

$$= -\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} \tau_i \\ 0 \\ x_i \end{pmatrix} - \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial s_i} \end{pmatrix}.$$

Note that $s_i = c_i - A_i^T u$, $d_{s_i} = -A_i^T d_u$, $\tau_i = 1$, and $d_{\tau_i} = 0$ for $i = 1, \ldots, m$. The system can be written as

$$\begin{pmatrix} 0 \\ 0 \\ d_{x_i} \end{pmatrix} + \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial s_i} \end{pmatrix} \begin{pmatrix} d_{t_i} \\ d_{z_i} \\ -A_i^T d_u \end{pmatrix}$$

$$= -\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \\ x_i \end{pmatrix} - \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial s_i} \end{pmatrix}.$$

Let

$$J_i = \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial z_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial t_i} & \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial z_i} \end{pmatrix}.$$

One can easily verify that J_i is positive definite. Therefore, we can compute J_i^{-1} in at most $O(d^3)$ time. From the first two equations, we get

(5.1)
$$\begin{pmatrix} d_{t_i} \\ d_{z_i} \end{pmatrix} = J_i^{-1} \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial t_i \partial s_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial z_i \partial s_i} \end{pmatrix} A_i^T d_u - \frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right].$$

Substituting this into the third equation, we get

$$d_{x_i} + \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix} - \frac{\partial^2 \hat{f}_p(t_i, s_i, z_i)}{\partial s_i \partial s_i} \right] A_i^T d_u$$

$$= \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \left[\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right] - \frac{\rho}{\mathcal{X}^T \mathcal{S}} x_i - \frac{\partial \hat{f}_p(t_i, s_i, z_i)}{\partial s_i}.$$

Moreover, since

$$\sum_{i=1}^m A_i x_i = 0, \qquad \sum_{i=1}^m A_i d_{x_i} = 0,$$

we have

(5.2)

$$\sum_{i=1}^{m} A_i \left[\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix} - \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial s_i} \right] A_i^T d_u$$

$$= \sum_{i=1}^{m} A_i \left(\begin{pmatrix} \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial t_i} \\ \frac{\partial^2 \hat{f}_p(t_i,s_i,z_i)}{\partial s_i \partial z_i} \end{pmatrix}^T J_i^{-1} \right.$$

$$\left. \times \left[\frac{\rho}{\mathcal{X}^T \mathcal{S}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \begin{pmatrix} \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial t_i} \\ \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial z_i} \end{pmatrix} \right] - \frac{\rho}{\mathcal{X}^T \mathcal{S}} x_i - \frac{\partial \hat{f}_p(t_i,s_i,z_i)}{\partial s_i} \right).$$

Note that the system for computing d_u may not have full rank. If that is the case, any feasible solution is acceptable.

It requires $O(m(d^3 + nd^2 + n^2 d))$ operations to set up the system (5.2) for computing d_u. Solving the system requires $O(n^3)$ operations. Once d_u is computed, $O(m(d^3 + nd^2 + n^2 d))$ operations are required to compute $d_{\mathcal{X}}$, $d_{\mathcal{S}}$, and $d_{\mathcal{Z}}$. Therefore, the number of arithmetic operations in each iteration is bounded by $O(n^3 + md^3 + md^2 n + mdn^2)$. The following theorem follows from Corollary 5.1 and the above analysis.

THEOREM 5.3. *Let the initial feasible primal solution $\mathcal{X}^0$ and dual feasible solution $(\mathcal{Y}^0; \mathcal{S}^0)$ be given as above. Then, an ϵ-optimal solution to problem* (2.1) *can be obtained by the potential reduction algorithms in at most*

$$O\left(\gamma\sqrt{200(1+2d)m}\,(\log(\bar{c}/\epsilon) + \log m + \log d)\right)$$

iterations, where

$$\bar{c} = \max_{1 \le i \le m} \|c_i\|,$$

and each iteration requires $O(n^3 + md^3 + md^2 n + mdn^2)$ arithmetic operations. □

Note that if both d and γ are constants and the problem is normalized such that $\bar{c} = 1$, i.e., all of c_i is within the unit ball in R^d, then the iteration bound is $O(\sqrt{m}(\log(1/\epsilon) + \log m))$. We will further discuss this issue in the following applications.

6. Applications.

In this section, we will apply the algorithms presented in the previous sections to solve the p-norm multifacility location (PMFL) problem and the p-norm SMT problem under a given topology. We will also take advantage of the special structures of these special problems and obtain improved computational complexity results wherever possible.

6.1. The p-norm multifacility location problem.

Let $a_1, a_2, \ldots, a_M$ be M points in R^d, the d-dimensional $l_p$space. Let w_{ji}, $j = 1, 2, \ldots, N$, $i = 1, 2, \ldots, M$, and $v_{jk}, 1 \le j < k \le N$, be given nonnegative numbers. Find a point $x = (x_1; x_2; \ldots; x_N) \in R^{dN}$ that will minimize

$$f_p(x) = \sum_{j=1}^{N} \sum_{i=1}^{M} w_{ji} ||x_j - a_i||_p + \sum_{1 \le j < k \le N} v_{jk} ||x_j - x_k||_p, \qquad p \ge 1. \tag{6.1}$$

This is the so-called PMFL. For ease of notation, we assume that $v_{jj} = 0$ for $j = 1, 2, \dots, N$ and that $v_{jk} = v_{kj}$ for $1 \le k < j \le N$.

In the PMFL problem, $a_1, a_2, \dots, a_M$ represent the locations of M existing facilities; $x_1, x_2, \dots, x_N$ represent the locations of N new facilities; and the objective function $f_p(x)$ is the sum of weighted p-norm distances from each new facility to each existing facility and those between each pair of new facilities. Our goal is to find optimal locations for the new facilities, i.e., to minimize $f_p(x)$.

In problem (6.1), some of the weights w_{ji} and v_{jk} may be zero. Let m be the number of nonzero weights in (6.1). Then the PMFL problem (6.1) is the minimization of a sum of m p-norms. Without loss of generality, we assume that for each $j \in \{1, 2, \dots, N\}$, there exists a nonzero w_{ji} for some $i \in \{1, 2, \dots, M\}$ or a nonzero v_{jk} for some $k \in \{1, 2, \dots, N\}$.

To transform the PMFL problem (6.1) into an instance of problem (1.1), we simply do the following. Let $u = (x_1; x_2; \dots; x_N)$. It is clear that $u \in R^n$ where $n = dN$. For each nonzero w_{ji}, there is a corresponding term of p-norm $||c(w_{ji}) - A(w_{ji})^T u||_p$ where $c(w_{ji}) = w_{ji} a_i$ and $A(w_{ji})^T$ is a row of N blocks of d by d matrices whose jth block is $w_{ji} I_d$ and whose other blocks are all zero. For each nonzero v_{jk}, there is a corresponding term of p-norm $||c(v_{jk}) - A(v_{jk})^T u||_p$, where $c(v_{jk}) = 0$, and $A(v_{jk})^T$ is a row of N blocks of d by d matrices whose jth and kth blocks are $-v_{jk} I_d$ and $v_{jk} I_d$, respectively, and whose other blocks are all zero.

Now it is clear that we have transformed the PMFL problem (6.1) into an instance of (2.1), where $n = dN$, and m is the number of nonzero weights w_{ji} and v_{jk}. Note that the system (5.2) can be set up with $O(md^2)$ operations. Therefore, it follows from Theorem 5.1 that we have Theorem 6.1.

THEOREM 6.1. *An ϵ-optimal solution to the PMFL problem* (6.1) *can be computed using any of our algorithms in at most*

$$O\left(\gamma\sqrt{200(1+2d)MN}\,(\log(\bar{c}/\epsilon) + \log(MN) + \log d)\right)$$

iterations, where $\bar{c} = \max_{1 \le j \le n \quad 1 \le i \le m} \|w_{ji} a_i\|$, and each iteration requires $O(d^3N^3 + MNd^2)$ arithmetic operations. □

6.2. The p-norm SMT problem. The *p-norm SMT problem* is given by a set of points $P = \{p_1, p_2, \dots, p_N\}$ in the l_p-plane and asks for the shortest planar straight-line graph spanning P. The solution takes the form of a tree, called the *Steiner minimal tree*, that includes all the given points, called *regular points*, along with some extra vertices, called *Steiner points*. It is known that there are at most $N - 2$ Steiner points and that the degree of each Steiner point is at most 3. See [13, 22] for details.

DEFINITION 6.2 (see [13, 15, 16]). *A* full Steiner topology *of point set P is a tree graph whose vertex set contains P and $N - 2$ Steiner points; the degree of each vertex in P is exactly* 1, *and the degree of each Steiner vertex is exactly* 3.

Computing a SMT for a given set of N points in the l_p-plane is NP-hard [12]. However, the problem of computing the shortest network under a given full Steiner topology can be solved efficiently. Recently, there have been increased interests in this latter problem and several algorithms have been proposed [15, 16, 31]. We will formulate this problem as a special case of problem (1.1).

Let $m = 2N - 3$, $d = 2$, $n = 2N - 4$. Let $u \in R^{2N-4}$ represent the locations of the $N - 2$ Steiner points. Without loss of generality, we may order the edges in the given full Steiner topology in such a way that each of the first N edges connects a regular point to a Steiner point. For $i = 1, 2, \dots, N$, c_i is p_{i_1}, where i_1 is the index of

Table 1
The coordinates of the 10 *regular points.*

Index	x-coordinate	y-coordinate	Index	x-coordinate	y-coordinate
9	2.30946900	9.20821100	14	7.59815200	0.61583600
10	0.57736700	6.48093800	15	8.56812900	3.07917900
11	0.80831400	3.51906200	16	4.75750600	3.75366600
12	1.68591200	1.23167200	17	3.92609700	7.00879800
13	4.11085500	0.82111400	18	7.43649000	7.68328400

Table 2
The tree topology.

Edge-index	ea-index	eb-index	Edge-index	ea-index	eb-index
1	9	7	10	18	8
2	10	1	11	5	6
3	11	2	12	6	4
4	12	3	13	4	3
5	13	4	14	3	2
6	14	5	15	2	1
7	15	5	16	1	7
8	16	6	17	7	8
9	17	8			

the regular point on the ith edge; $A_i^T \in R^{2\times n}$ is a row of $N-2$ 2 by 2 block matrices, where only the i_2th block is I_2 and the rest are all zero, where i_2 is the index of the Steiner point on the ith edge. For $i = N+1, N+2, \dots, m$, $c_i = 0$, and $A_i^T \in R^{2\times n}$ is a row of $N-2$ 2 by 2 block matrices, where the i_1st block is $-I_2$, the i_2nd block is I_2, and the rest of the blocks are all zero, and where i_1 and i_2 are the indices of the two Steiner points on the ith edge. It is clear that we have transformed the problem of computing a shortest network under a full Steiner topology into an instance of (2.1), where $d = 2$, $n = 2N-4$, and $m = 2N-3$. Therefore, it can be solved efficiently using our interior point algorithm.

Note that we can move the point set P so that its gravitational center is the origin. Therefore, the l_p-norms (as well as the Euclidean norms) of the regular points are bounded by the largest pairwise (p-norm or Euclidean) distance among the points in P, which corresponds to the constant $\bar{c}$ in previous theorems. Furthermore, as illustrated in [36], the search direction can be computed in $O(N)$ arithmetic operations using a technique known as *Gaussian elimination on leaves of a tree* [31]. Therefore, we have the following theorem.

Theorem 6.3. *An ϵ-optimal solution to the shortest network under a given full Steiner topology of N regular points in the l_p-plane can be computed using our potential reduction algorithms in at most $O(\sqrt{N}(\log(\bar{c}/\epsilon) + \log N))$ iterations, where $\bar{c}$ is the largest pairwise distance among the regular points and each iteration requires $O(N)$ arithmetic operations. Therefore, the computation of an ϵ-optimal solution requires $O(N\sqrt{N}(\log(\bar{c}/\epsilon) + \log N))$ arithmetic operations.* □

The problem of computing the shortest network under a full Steiner topology was first studied by Hwang [15], Hwang and Weng [16], and Smith [31]. Hwang [15] presented a linear-time exact algorithm that can output the shortest network under a given full Steiner topology if there exists a nondegenerate SMT corresponding to that given topology and that quits otherwise. Hwang and Weng [16] presented an $O(N^2)$-time graphical algorithm that can output the shortest network under a

TABLE 3
Output for $p = 1.01$ using dual scaling.

Iteration	Network-cost	Duality-gap	dstep-max	pstep-max
1	1.9810e+03	1.9863e+03	3.1250e-01	9.7656e-03
2	9.3134e+02	9.3948e+02	3.1250e-01	9.7656e-03
3	4.6604e+02	4.6352e+02	3.1250e-01	7.8125e-02
4	2.6185e+02	2.5141e+02	3.1250e-01	1.9531e-02
5	1.6673e+02	1.5389e+02	3.1250e-01	4.8828e-03
6	1.1164e+02	1.0050e+02	3.1250e-01	1.9531e-02
7	8.2320e+01	7.0048e+01	3.1250e-01	4.8828e-03
8	6.3980e+01	4.8173e+01	3.1250e-01	4.8828e-03
9	5.1500e+01	3.2779e+01	3.1250e-01	4.8828e-03
10	4.3166e+01	2.2726e+01	3.1250e-01	2.4414e-03
11	3.7916e+01	1.5499e+01	3.1250e-01	1.2207e-03
12	3.4452e+01	1.0726e+01	3.1250e-01	6.1035e-04
13	3.2373e+01	6.5718e+00	3.1250e-01	6.1035e-04
14	3.0782e+01	3.5208e+00	3.1250e-01	6.1035e-04
15	2.9691e+01	2.0491e+00	3.1250e-01	3.0518e-04
16	2.9255e+01	1.3093e+00	3.1250e-01	7.6294e-05
17	2.9023e+01	6.9142e-01	3.1250e-01	7.6294e-05
18	2.8824e+01	3.9670e-01	3.1250e-01	3.8147e-05
19	2.8737e+01	2.3084e-01	3.1250e-01	1.9073e-05
20	2.8682e+01	1.4074e-01	3.1250e-01	9.5367e-06
21	2.8649e+01	9.0313e-02	3.1250e-01	4.7684e-06
22	2.8630e+01	5.2046e-02	3.1250e-01	4.7684e-06
23	2.8615e+01	2.7875e-02	3.1250e-01	4.7684e-06
24	2.8605e+01	1.7067e-02	3.1250e-01	2.3842e-06
25	2.8601e+01	1.1235e-02	3.1250e-01	5.9605e-07
26	2.8599e+01	6.6474e-03	3.1250e-01	5.9605e-07
27	2.8597e+01	3.5436e-03	3.1250e-01	5.9605e-07
28	2.8596e+01	2.0801e-03	3.1250e-01	5.9605e-07
29	2.8595e+01	1.3643e-03	3.1250e-01	7.4506e-08
30	2.8594e+01	5.2467e-04	6.2500e-01	7.4506e-08
31	2.8594e+01	2.8420e-04	6.2500e-01	1.8626e-08
32	2.8594e+01	1.5954e-04	6.2500e-01	9.3132e-09
33	2.8594e+01	9.4942e-05	6.2500e-01	4.6566e-09
34	2.8594e+01	4.2928e-05	6.2500e-01	4.6566e-09
35	2.8594e+01	1.6635e-05	6.2500e-01	2.3283e-09
36	2.8594e+01	9.0572e-06	6.2500e-01	5.8208e-10

given full Steiner topology if the shortest network under the given topology is a tree with maximum vertex degree 3 and that quits otherwise. Xue and Ye [36] proposed a primal-dual potential reduction algorithm that can always output an ϵ-optimal network under a given topology in $O(N\sqrt{N}(\log(\bar{c}/\epsilon) + \log N))$ operations, where $\bar{c}$ is the largest pairwise distance among the given points. It seems hard to generalize the graphical methods to the p-norm case. Therefore, our generalization of the result of [36] is important.

7. Computational examples. We have implemented all three versions of our algorithm using Matlab. Our intention here is to justify the theory developed in this paper. Therefore, our primary interest is in the number of iterations required by the algorithms. Our implementation here is very preliminary. Extensive computational study of the algorithms will be reported in a separate paper. For test problems, we have taken the 10-regular-points SMT problem from [36]. The coordinates of the regular points are given in Table 1. The tree topology is given in Table 2, where for each edge, indices of its two vertices are shown next to the index of the edge.

In our implementation, we used $\gamma = 2m$ to take *long steps* instead of using the

TABLE 4
Output for $p = 1.50$ using dual scaling.

Iteration	Network-cost	Duality-gap	dstep-max	pstep-max
1	6.3022e+02	6.3501e+02	3.1250e-01	9.7656e-03
2	3.0949e+02	3.0829e+02	3.1250e-01	7.8125e-02
3	2.4246e+02	2.3339e+02	1.5625e-01	9.7656e-03
4	1.4290e+02	1.2897e+02	3.1250e-01	9.7656e-03
5	9.0251e+01	7.7152e+01	3.1250e-01	1.9531e-02
...				
23	2.6561e+01	6.2128e-03	3.1250e-01	5.9605e-07
24	2.6559e+01	3.7579e-03	3.1250e-01	2.9802e-07
25	2.6558e+01	2.3707e-03	3.1250e-01	1.4901e-07
26	2.6557e+01	7.8395e-04	6.2500e-01	1.4901e-07
27	2.6557e+01	3.5326e-04	6.2500e-01	3.7253e-08
28	2.6557e+01	1.4104e-04	6.2500e-01	1.8626e-08
29	2.6557e+01	7.4709e-05	6.2500e-01	4.6566e-09
30	2.6557e+01	4.4346e-05	6.2500e-01	2.3283e-09
31	2.6557e+01	1.8446e-05	6.2500e-01	2.3283e-09
32	2.6557e+01	9.9644e-06	3.1250e-01	1.1642e-09

TABLE 5
Output for $p = 2.00$ using dual scaling.

Iteration	Network-cost	Duality-gap	dstep-max	pstep-max
1	3.5099e+02	3.5513e+02	3.1250e-01	9.7656e-03
2	2.6731e+02	2.6493e+02	1.5625e-01	9.7656e-03
3	2.0753e+02	1.9761e+02	1.5625e-01	9.7656e-03
4	1.1948e+02	1.0726e+02	3.1250e-01	4.8828e-03
5	7.5355e+01	6.1541e+01	3.1250e-01	4.8828e-03
...				
28	2.5357e+01	1.2411e-03	3.1250e-01	7.4506e-08
29	2.5357e+01	7.0021e-04	3.1250e-01	7.4506e-08
30	2.5356e+01	3.7630e-04	3.1250e-01	7.4506e-08
31	2.5356e+01	2.2183e-04	3.1250e-01	3.7253e-08
32	2.5356e+01	1.4140e-04	3.1250e-01	9.3132e-09
33	2.5356e+01	7.8081e-05	3.1250e-01	9.3132e-09
34	2.5356e+01	4.8133e-05	3.1250e-01	4.6566e-09
35	2.5356e+01	2.9645e-05	3.1250e-01	2.3283e-09
36	2.5356e+01	1.5629e-05	3.1250e-01	2.3283e-09
37	2.5356e+01	8.3688e-06	3.1250e-01	4.6566e-09

conservative theoretical parameter $\gamma = 1$. Also, we used 0.75 times the largest feasible step-size as the actual step-size rather than using the theoretical step-size or a line-search. The algorithm stops whenever the absolute duality gap is smaller than 10^{-5}.

To test the flexibility of our algorithm, we have used the values of $p = 1.01$, 1.5, 2.0, 3.0, and 101 on the 10-point case. Although with a larger parameter, the algorithm based on barrier function 2 works better in our examples. All five cases were solved within 30 to 40 iterations. The output of the algorithm is presented in Tables 3–7. Note that for the last case, we have used the primal scaling algorithm, where $q = 1.01$. The second column in the tables shows the cost of the current network (i.e., the sum of p-norms in the current network). The third column shows the duality gap, which is an upper bound of the error in the cost of the current network to the cost of the optimal (shortest) network. The last two columns show the largest dual and primal feasible step-sizes, $\bar{\beta}$ and $\bar{\alpha}$; see the discussion at the end of section 4.

The shortest networks for the cases $p = 1.01, 1.50, 3.0, 101$ are plotted in Figure 1, where regular points are labeled by + and Steiner points are labeled by o. The case

Table 6
Output for $p = 3.00$ using dual scaling.

Iteration	Network-cost	Duality-gap	dstep-max	pstep-max
1	3.0811e+02	3.1301e+02	1.5625e-01	1.9531e-02
2	2.3295e+02	2.3135e+02	1.5625e-01	9.7656e-03
3	1.7991e+02	1.7375e+02	1.5625e-01	4.8828e-03
4	1.4208e+02	1.3316e+02	1.5625e-01	2.4414e-03
5	8.4541e+01	7.0951e+01	3.1250e-01	4.8828e-03
...				
24	2.3930e+01	5.6315e-03	3.1250e-01	5.9605e-07
25	2.3928e+01	2.9602e-03	3.1250e-01	5.9605e-07
26	2.3927e+01	1.6010e-03	3.1250e-01	5.9605e-07
27	2.3927e+01	1.0595e-03	3.1250e-01	3.7253e-08
28	2.3926e+01	4.9997e-04	6.2500e-01	3.7253e-08
29	2.3926e+01	2.0940e-04	1.2500e+00	1.8626e-08
30	2.3926e+01	6.3721e-05	2.5000e+00	9.3132e-09
31	2.3926e+01	2.7372e-05	2.5000e+00	2.3283e-09
32	2.3926e+01	1.2168e-05	1.2500e+00	1.1642e-09
33	2.3926e+01	5.4989e-06	6.2500e-01	5.8208e-10

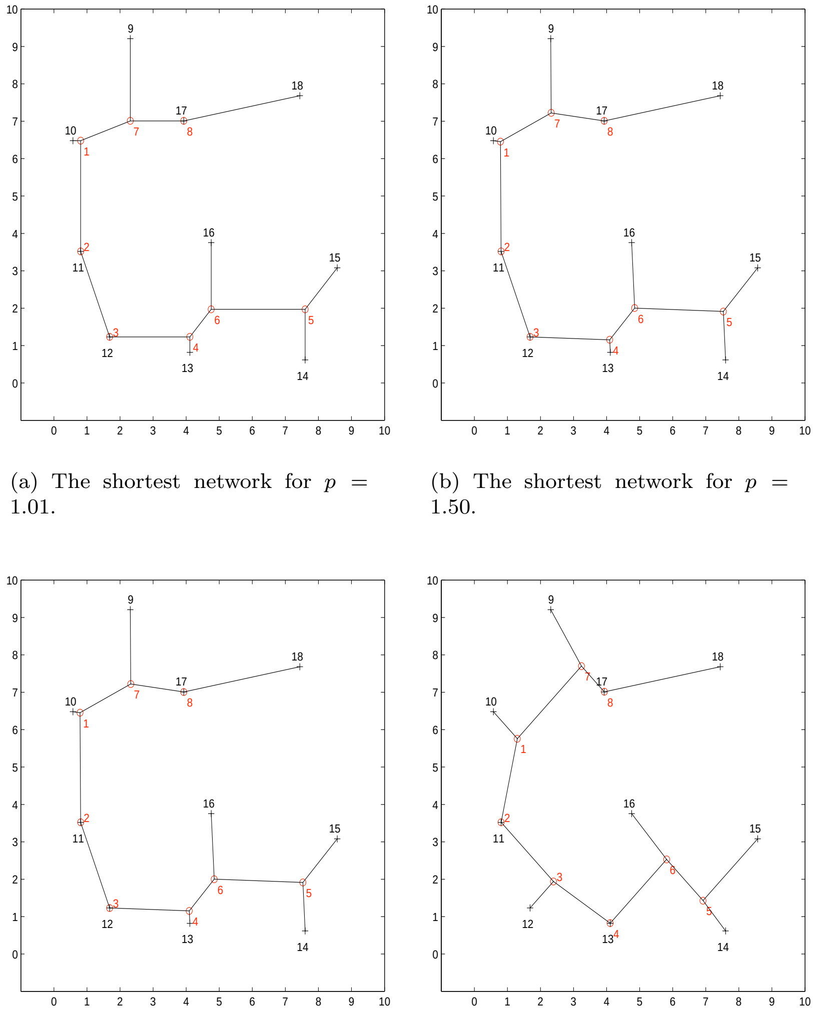
Table 7
Output for $p = 101$ ($q = 1.01$) using primal scaling.

Iteration	Network-cost	Duality-gap	dstep-max	pstep-max
1	9.8742e+01	9.4987e+01	4.8828e-03	6.2500e-01
2	6.9168e+01	5.7033e+01	2.4414e-03	1.2500e+00
3	3.9986e+01	2.4935e+01	2.4414e-03	1.2500e+00
4	2.7451e+01	1.0022e+01	1.2207e-03	6.2500e-01
5	2.5071e+01	6.3698e+00	3.0518e-04	3.1250e-01
...				
23	2.1182e+01	1.3274e-04	9.3132e-09	3.1250e-01
24	2.1182e+01	5.1656e-05	9.3132e-09	3.1250e-01
25	2.1182e+01	2.6681e-05	9.3132e-09	1.5625e-01
26	2.1183e+01	2.7794e-04	1.4901e-07	1.5625e-01
27	2.1183e+01	1.6209e-04	9.3132e-09	5.0000e+00
28	2.1182e+01	9.0529e-05	4.6566e-09	1.2500e+00
29	2.1182e+01	5.8100e-05	2.3283e-09	2.5000e+00
30	2.1182e+01	2.7823e-05	2.3283e-09	1.2500e+00
31	2.1182e+01	1.3559e-05	1.1642e-09	6.2500e-01
32	2.1182e+01	6.2516e-06	5.8208e-10	6.2500e-01

$p = 2.0$ is similar but we choose not to illustrate it here to save space. It is interesting to see the slight change in the network when p is changed from 1.01 to 1.5 and eventually to 101.

8. Conclusions.

In this paper, we have transformed the problem of minimizing a sum of p-norms into a standard convex programming problem in conic forms. Unlike those in most convex optimization problems, the cone for this problem is not self-dual. We have constructed two barrier functions and studied its associated parameters. Using these barrier functions, we have presented a polynomial time primal-dual potential reduction algorithm for solving this problem. In particular, the number of iterations required to produce an ϵ-optimal solution is at most $O(\sqrt{md}(\log(\bar{c}/\epsilon) + \log(md)))$. As applications, we have shown that computing an ϵ-optimal solution of the shortest p-norm network under a tree topology interconnecting N regular points on the l_p-plane requires only $O(N^{1.5}(\log(\bar{c}/\epsilon) + \log N))$ arithmetic operations, where $\bar{c}$ is the largest pairwise l_p-distance among the given point set. Our implementation is only preliminary. Computational issues of our algorithm are under investigation and will be reported in another paper.



(a) The shortest network for $p =$ 1.01.

(b) The shortest network for $p =$ 1.50.

(c) The shortest network for $p =$ 3.00.

(d) The shortest network for $p =$ 101.

FIG. 1. *Shortest networks for different values of p.*

Acknowledgments. The authors would like to thank two referees and the associate editor for their helpful comments and remarks on the first version of the paper.

REFERENCES

[1] K.D. ANDERSEN, *An efficient Newton barrier method for minimizing a sum of Euclidean norms*, SIAM J. Optim., 6 (1996), pp. 74–95.

[2] K.D. ANDERSEN AND E. CHRISTIANSEN, *A Symmetric Primal-Dual Newton Method for Minimizing a Sum of Norms*, manuscript, Odense University, Denmark, 1995.

[3] P.H. CALAMAI AND A.R. CONN, *A second-order method for solving the continuous multifacility location problem*, in Numerical Analysis: Proceedings of the Ninth Biennial Conference, Dundee, Scotland, Lecture Notes in Mathematics 912, G.A. Watson, ed., Springer-

Verlag, New York, 1982, pp. 1–25.

[4] P.H. Calamai and A.R. Conn, *A projected Newton method for l_p norm location problems*, Math. Programming, 38 (1987), pp. 75–109.

[5] R. Chandrasekaran and A. Tamir, *Open questions concerning Weiszfeld's algorithm for the Fermat-Weber location problem*, Math. Programming, 44 (1989), pp. 293–295.

[6] R. Chandrasekaran and A. Tamir, *Algebraic optimization: The Fermat-Weber location problem*, Math. Programming, 46 (1990), pp. 219–224.

[7] K.D. Anderson, E. Christiansen, A.R. Conn, and M.L. Overton, *An efficient primal-dual interior-point method for minimizing a sum of Euclidean norms*, SIAM J. Sci. Comput., to appear.

[8] D. den Hertog, *Interior Point Approach to Linear, Quadratic and Convex Programming*, Kluwer Academic Publishers, Norwell, MA, 1994.

[9] D. den Hertog, F. Jarre, C. Roos, and T. Terlaky, *A sufficient condition for self-concordance, with application to some classes of structured convex programming problems*, Math. Programming, 69 (1995), pp. 75–88.

[10] J.W. Eyster, J.A. White, and W.W. Wierwille, *On solving multifacility location problems using a hyperboloid approximation procedure*, AIIE Transactions, 5 (1973), pp. 1–6.

[11] R.L. Francis, L.F. McGinnis, Jr., and J.A. White, *Facility Layout and Location: An Analytical Approach*, Prentice-Hall, Englewood Cliffs, NJ, 1991.

[12] M.R. Garey, R.L. Graham, and D.S. Johnson, *The complexity of computing Steiner minimal trees*, SIAM J. Appl. Math., 32 (1977), pp. 835–859.

[13] E.N. Gilbert and H.O. Pollak, *Steiner minimal trees*, SIAM J. Appl. Math., 16 (1968), pp. 1–29.

[14] G.H. Golub and C.F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1983.

[15] F.K. Hwang, *A linear time algorithm for full Steiner trees*, Oper. Res. Lett., 4 (1986), pp. 235–237.

[16] F.K. Hwang and J.F. Weng, *The shortest network under a given topology*, J. Algorithms, 13 (1992), pp. 468–488.

[17] I.N. Katz, *Local convergence in Fermat's problem*, Math. Programming, 6 (1974), pp. 89–104.

[18] M. Kojima, S. Shindoh, and S. Hara, *Interior-Point Methods for the Monotone Semidefinite Linear Complementarity Problem in Symmetric Matrices*, Research Reports on Information Sciences, No. B-282, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan, 1994; revised April, 1995.

[19] H.W. Kuhn, *A note on Fermat's problem*, Math. Programming, 4 (1973), pp. 98–107.

[20] Y. Li, *A Newton Acceleration of the Weiszfeld Algorithm for Minimizing the Sum of Euclidean Distances*, Technical report TR95-1552, Computer Science Department, Cornell University, Ithaca, NY, November 1995.

[21] R.F. Love, J.G. Morris, and G.O. Wesolowsky, *Facilities Location: Models & Methods*, North-Holland, Amsterdam, 1988.

[22] Z.A. Melzak, *On the problem of Steiner*, Canad. Math. Bull., 16 (1961), pp. 143–148.

[23] W. Miehle, *Link length minimization in networks*, Oper. Res., 6 (1958), pp. 232–243.

[24] A. Nemirovskii, *Polynomial time methods in convex programming*, in The Mathematics of Numerical Analysis, Lectures in Applied Mathematics 32, J. Renegar, M. Shub, and S. Smale, eds., AMS, Providence, RI, 1996.

[25] Yu. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, PA, 1994.

[26] Yu. E. Nesterov and M. J. Todd, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res. 22 (1997), pp. 1–42.

[27] L.M. Ostresh, *The multifacility location problem: Applications and descent theorems*, J. Regional Science, 17 (1977), pp. 409–419.

[28] M.L. Overton, *A quadratically convergent method for minimizing a sum of Euclidean norms*, Math. Programming, 27 (1983), pp. 34–63.

[29] J.B. Rosen and G.L. Xue, *On the convergence of Miehle's algorithm for the Euclidean multifacility location problem*, Oper. Res., 40 (1992), pp. 188–191.

[30] J.B. Rosen and G.L. Xue, *On the convergence of a hyperboloid approximation procedure for solving the perturbed Euclidean multifacility location problem*, Oper. Res., 41 (1993), pp. 1164–1171.

[31] W.D. Smith, *How to find Steiner minimal trees in Euclidean d-space*, Algorithmica, 7 (1992), pp. 137–177.

[32] C.Y. Wang et al., *On the convergence and rate of convergence of an iterative algorithm for the plant location problem*, Qufu Shiyun Xuebao, 2 (1975), pp. 14–25 (in Chinese).

[33] E. Weiszfeld, *Sur le point par lequel le somme des distances de n points donnes est minimum*, Tohoku Math. J., 43 (1937), pp. 355–386.

[34] G.L. Xue, *Algorithms for Computing Extreme Points of Convex Hulls and the Euclidean Facilities Location Problem*, Ph.D. thesis, Computer Science Department, University of Minnesota, Minneapolis, 1991.

[35] G.L. Xue, J.B. Rosen, and P.M. Pardalos, *A polynomial time dual algorithm for the Euclidean multifacility location problem*, in Proceedings of Second Conference on Integer Programming and Combinatorial Optimization, Pittsburgh, PA, 1992, pp. 227–236.

[36] G. Xue and Y. Ye, *An efficient algorithm for minimizing a sum of Euclidean norms with applications*, SIAM J. Optim., 7 (1997), pp. 1017–1036.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 580–604

STABILITY OF LOCALLY OPTIMAL SOLUTIONS*

A. B. LEVY[†], R. A. POLIQUIN[‡], AND R. T. ROCKAFELLAR[§]

Abstract. Necessary and sufficient conditions are obtained for the Lipschitzian stability of local solutions to finite-dimensional parameterized optimization problems in a very general setting. Properties of prox-regularity of the essential objective function and positive definiteness of its coderivative Hessian are the keys to these results. A previous characterization of tilt stability arises as a special case.

Key words. parameterized optimization, Lipschitzian stability, tilt stability, coderivative Hessians, prox-regular functions, amenable functions

AMS subject classifications. Primary, 49A52, 58C06, 58C20; Secondary, 90C30

PII. S1052623498348274

1. Introduction.

In theory, any problem of optimization in n real variables can be represented as a problem of minimizing, over the entire space $\mathbb{R}^n$, a function f with values in $\overline{\mathbb{R}} = [-\infty, \infty]$. Points x that should not be candidates in the minimization can effectively be excluded by setting $f = \infty$. Such a representation is especially useful in getting to the heart of theoretical issues in parametric optimization, because it allows problem parameters to be viewed as just additional variables on which f depends.

Our aim is to try to understand, in this abstract setting and on the most fundamental level of variational analysis, the circumstances in which locally optimal solutions behave in a "stable" manner with respect to shifts in parameter values. The model we adopt is that of a family of minimization problems in $x \in \mathbb{R}^n$ parameterized by $u \in \mathbb{R}^d$, as specified by a function $f : \mathbb{R}^n \times \mathbb{R}^d \to \overline{\mathbb{R}}$. Within the family we single out a problem

$$\overline{\mathcal{P}} \qquad\qquad \text{minimize } f(x, \bar{u}) \text{ over } x \in \mathbb{R}^n,$$

and compare it to perturbed versions that come from shifting the associated parameter vector $\bar{u}$ to some nearby vector u. For technical reasons, we further consider, along with such *basic* perturbations, *tilt* perturbations that correspond to adding a small linear term to the objective. Thus, we regard $\overline{\mathcal{P}}$ as imbedded in the larger family of problems

$$\mathcal{P}(u, v) \qquad\qquad \text{minimize } f(x, u) - \langle v, x \rangle \text{ over } x \in \mathbb{R}^n,$$

with both $u \in \mathbb{R}^d$ and $v \in \mathbb{R}^n$ parameters, so that $\overline{\mathcal{P}} = \mathcal{P}(\bar{u}, \bar{v})$ for $\bar{v} = 0$. In the developments that follow, however, $\bar{v}$ might just as well be any vector, so we refer to the unperturbed problem around which we work as $\mathcal{P}(\bar{u}, \bar{v})$ rather than $\overline{\mathcal{P}}$.

[*] Received by the editors December 2, 1998; accepted for publication (in revised form) June 18, 1999; published electronically February 24, 2000.

http://www.siam.org/journals/siopt/10-2/34827.html

[†] Department of Mathematics, Bowdoin College, Brunswick, ME 04011 (alevy@.bowdoin.edu).

[‡] Department of Mathematical Sciences, University of Alberta, Edmonton, Alberta, Canada T6G 2G1 (rene.poliquin@math.ualberta.ca). This author was supported by the Natural Sciences and Engineering Research Council of Canada under grant OGP41983.

[§] Department of Mathematics, University of Washington, Seattle, WA 98195 (rtr@math.washington.edu). This author was supported by the National Science Foundation under grant DMS–9803089 and by the Binational Science Foundation under grant 94-00237.

Throughout, we assume that f is lower semicontinuous (lsc) and proper, i.e., not identically ∞ and nowhere taking on $-\infty$. The set of *feasible solutions* to $\mathcal{P}(u,v)$ consists then, by definition, of the points x such that $f(u,x)$ is finite. We denote by $\bar{x}$ a feasible solution to $\mathcal{P}(\bar{u},\bar{v})$ and investigate it in terms of the functions $m_\delta : \mathbb{R}^d \times \mathbb{R}^n \to \mathbb{R}$ and mappings $M_\delta : \mathbb{R}^d \times \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ (set-valued) that are defined for $\delta > 0$ by

$$\begin{aligned} m_\delta(u,v) &= \inf_{|x-\bar{x}|\le\delta} \big\{ f(x,u) - \langle v,x\rangle \big\}, \\ M_\delta(u,v) &= \operatorname{argmin}_{|x-\bar{x}|\le\delta} \big\{ f(x,u) - \langle v,x\rangle \big\}. \end{aligned} \tag{1.1}$$

Here $M_\delta(u,v)$ could consist of a single-point x, in which case M_δ is *single-valued* at (u,v), but it might contain many points or be empty. By convention, argmin $= \emptyset$ when the expression being minimized can be only ∞; that ensures having $M_\delta(u,v)$ be empty when $\mathcal{P}(u,v)$ has no feasible solutions x satisfying $|x-\bar{x}| \le \delta$, i.e., when $m_\delta(u,v) = \infty$. Aside from that case, $M_\delta(u,v)$ is nonempty and $m_\delta(u,v)$ is finite.

In such notation, to say that $\bar{x}$ is a *locally optimal solution* to $\mathcal{P}(\bar{u},\bar{v})$ is to say that $\bar{x} \in M_\delta(\bar{u},\bar{v})$ for some $\delta > 0$ (sufficiently small). The stability properties of locally optimal solutions that we target for study revolve around $\bar{x}$ being the *only* point of $M_\delta(\bar{u},\bar{v})$ and having this single-valuedness of the mapping M_δ at $(\bar{u},\bar{v})$ persist in a Lipschitzian manner with respect to certain parameter shifts away from $(\bar{u},\bar{v})$.

DEFINITION 1.1 (solution stability). *A point $\bar{x}$ is a stable locally optimal solution to $\mathcal{P}(\bar{u},\bar{v})$ (in the basic sense, i.e., relative to the specified parameterization in u only) if there is a $\delta > 0$ such that, on some neighborhood U of $\bar{u}$, the mapping $u \mapsto M_\delta(u,\bar{v})$ is single-valued and Lipschitz continuous with $M_\delta(\bar{u},\bar{v}) = \bar{x}$, and the function $u \mapsto m_\delta(u,\bar{v})$ is likewise Lipschitz continuous on U.*

It is a tilt stable locally optimal solution if these properties hold with respect to v instead of u, i.e., for the mapping $v \mapsto M_\delta(\bar{u},v)$ and the function $v \mapsto m_\delta(\bar{u},v)$ on some neighborhood V of $\bar{v}$. It is a fully stable locally optimal solution if these properties hold with respect to (u,v) for the full mapping $(u,v) \mapsto M_\delta(u,v)$ and function $(u,v) \mapsto m_\delta(u,v)$ on some neighborhood $U \times V$ of $(\bar{u},\bar{v})$.

Full stability implies both (basic) stability and tilt stability but in general may differ from those properties. With x and u in $\mathbb{R}$ and $(\bar{x},\bar{u}) = (0,0)$, for instance, the case of $f(x,u) = (x-u)^4$ exhibits stability without full stability, whereas $f(x,u) = (x-u^{1/3})^2$ has tilt stability without full stability.

Note that in the definition of tilt stability it would not really be necessary to say anything about m_δ, since the formula for this function in (1.1) implies that $m_\delta(\bar{u},v)$ is finite and concave in v (as long as $f(\bar{x},\bar{u})$ is finite). In other situations the Lipschitz continuity of m_δ is not automatic, however, even in the face of Lipschitz continuity of M_δ. For example, the lsc, proper function $f : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ defined by $f(x,u) = x^2$ when $u = 0$ but $f(x,u) = 1 + x^2$ when $u \neq 0$ has, for $(\bar{x},\bar{u}) = (0,0)$ and $\bar{v} = 0$, that $M_\delta(u,\bar{v}) = 0$ for all u, yet $m_\delta(u,\bar{v})$ is discontinuous at $u = \bar{u}$.

Stability properties of one kind or another have been extensively investigated for optimal solutions to conventional nonlinear programming problems as well as for Karush–Kuhn–Tucker pairs in such problems or, more broadly, solutions to "generalized equations" and variational inequalities. The pioneering contribution of Robinson [1] put the focus on single-valued Lipschitzian behavior of optimal solutions. The literature on the subject is vast; the articles of Klatte and Kummer [2] and Dontchev and Rockafellar [3] provide an overview with many references to Lipschitzian behavior and also to calmness ("upper Lipschitzian" behavior) under perturbations.

The approach we take to stability differs from most of that literature, not merely because we adopt the format of extended-real-valued functions, but also in the tools we use. Crucial among them is the form of localized Lipschitz continuity for set-valued mappings that was defined by Aubin [4] and the criterion for it that was derived by Mordukhovich [5] in terms of his coderivative mappings. These tools of variational analysis have already been applied to stability issues by those authors in some general ways and also by Rockafellar and Wets in their recent book [6], which offers a thorough exposition of the concepts and their history (in finite dimensions). In other work, Dontchev and Rockafellar [7] have applied such methodology in finer detail to nonlinear programming and variational inequalities over polyhedral sets. Closest to our present effort, however, is the paper of Poliquin and Rockafellar [8], where tilt stability was first explored—in the simpler framework of a minimization problem perturbed by tilt vectors only.

The chief contribution of [8] was a characterization of tilt stability of locally optimal solutions in terms of positive definiteness of the generalized Hessian for f in the sense of Mordukhovich [5]. Here we build on the results in [8] by adding a parameterization in u alongside the tilt perturbations in v. As in [8], a function property called prox-regularity turns out to be essential. That property, which was introduced by Poliquin and Rockafellar [9] for the sake of fundamental developments in second-order nonsmooth analysis, must be adapted, however, to the additional parameterization. Likewise, the generalized Hessian in x is no longer enough and must be extended as part of the effort to make sure that the functions $f(\cdot, u)$ depend reasonably on u.

We concentrate on characterizing full stability, being content with the fact that necessary and sufficient conditions for full stability immediately yield sufficient conditions for basic stability. The task of characterizing basic stability on its own appears much more difficult and perhaps inappropriate. After all, tilt perturbations are a special case of other perturbations (one could have $f(x, u) = f_0(x) - \langle u, x\rangle$, say), so a universal result about basic stability could not escape having to account for them somehow. Indeed, it might well be that such a result would require a sort of extra "constraint qualification" that is tantamount to insisting on good tilt behavior. Nonetheless, from a practical point of view, as in connection with numerical methodology, for instance, there is likely to be little interest in situations where tilt stability is absent.

The assumptions behind our characterization of full stability, stated in Theorem 2.3, cover a very broad range of parameterized optimization problems expressible in the pattern of $\mathcal{P}(u, v)$. That includes not only nonlinear programming models in standard formats but also extended nonlinear programming models in which the objective function can be represented as the composition of a $\mathcal{C}^2$ mapping with a proper, lsc, convex function. We establish this in Proposition 2.2.

In order to apply our results to such special cases, one has to invoke a calculus of generalized Hessian mappings to see what one gets for the particular forms of $f(x, u)$ that come up. We have not undertaken to do that because it is a major project in itself and is better reserved for other papers in which the calculus rules suited for the job can systematically be laid out. Here, as a critical first step, we identify the underpinnings to stability at a depth not previously plumbed.

2. Main results.

In dealing with subgradients, we follow the notation and terminology of [6]. For a function $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ and a point $x \in \mathbb{R}^n$, a vector $v \in \mathbb{R}^n$ is a *regular subgradient* of g at x if $g(x)$ is finite and $g(x + w) \geq g(x) + \langle v, w\rangle + o(|w|)$.

It is a (*general*) *subgradient* at x if $g(x)$ is finite and there exist sequences $\{x^\nu\}_{\nu=1}^\infty$ and $\{v^\nu\}_{\nu=1}^\infty$ with v^ν a regular subgradient of g at x^ν, such that $v^\nu \to v$, $x^\nu \to x$, and $g(x^\nu) \to g(x)$. The set of all such (general) subgradients of g at x includes the regular subgradients at x and is denoted by $\partial g(x)$. A set-valued *subgradient mapping* $\partial g : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is thereby defined, which is empty-valued outside of $\operatorname{dom} g = \{x \mid g(x) < \infty\}$. The *graph* of ∂g is the set $\operatorname{gph} \partial g \subset \mathbb{R}^n \times \mathbb{R}^n$ consisting of the pairs (x, v) such that $v \in \partial g(x)$.

Also of use to us will be the concept of v being a *horizon subgradient* of g at x. This refers to the existence of sequences $\{x^\nu\}_{\nu=1}^\infty$ and $\{v^\nu\}_{\nu=1}^\infty$ with v^ν a regular subgradient of g at x^ν, such that $x^\nu \to x$, $g(x^\nu) \to g(x)$, and $\lambda^\nu v^\nu \to v$ for some scalar sequence $\{\lambda^\nu\}_{\nu=1}^\infty$ with $\lambda^\nu \downarrow 0$. The set of horizon subgradients v of g at x is denoted by $\partial^\infty g(x)$.

Prox-regularity arises from consideration of regular subgradients with a second-order aspect. A *proximal subgradient* of g at x is a regular subgradient v for which the error term $o(|w|)$ can be specialized to $(r/2)|w|^2$. Prox-regularity refers to a situation in which proximal subgradients prevail locally and with the same r. Specifically, g is *prox-regular* at $\bar{x}$ for $\bar{v}$ if it is locally lsc at $\bar{x}$ (cf. [6, Def. 1.33, Exercise 1.34]), has $\bar{v} \in \partial g(\bar{x})$, and there are neighborhoods X of $\bar{x}$ and V of $\bar{v}$ along with $\epsilon > 0$ and $r \geq 0$ such that

$$\begin{aligned} &g(x') \geq g(x) + \langle v,\ x' - x\rangle - \frac{r}{2}|x' - x|^2 \quad \text{for all} \quad x' \in X \\ &\qquad\qquad \text{when} \quad v \in \partial g(x),\ v \in V,\ x \in X,\ g(x) \leq g(\bar{x}) + \epsilon. \end{aligned} \tag{2.1}$$

It is *continuously prox-regular* at $\bar{x}$ for $\bar{v}$ if, in addition, $g(x)$ is continuous as a function of $(x, v) \in \operatorname{gph} \partial g$ at $(\bar{x}, \bar{v})$. (The latter property, by itself, is known as the *subdifferential continuity* of g at $\bar{x}$ for $\bar{v}$.) In that case one can arrange, by a shrinking of the neighborhoods X and V if necessary, that

$$\begin{aligned} &g(x') \geq g(x) + \langle v,\ x' - x\rangle - \frac{r}{2}|x' - x|^2 \quad \text{for all} \quad x' \in X \\ &\qquad\qquad\qquad\qquad \text{when} \quad v \in \partial g(x),\ v \in V,\ x \in X. \end{aligned} \tag{2.2}$$

The class of continuously prox-regular functions is very wide and includes not only convex functions, $\mathcal{C}^2$ functions, and lower-$\mathcal{C}^2$ functions, but also any such function plus the indicator of a set defined by finitely many $\mathcal{C}^2$ constraints under a constraint qualification. Many, if not most, of the essential objective functions in finite-dimensional optimization are covered. An overview is provided in [6, Chap. 13]. An elaboration on the parametric situation at hand will be given below in Proposition 2.2.

For the indicator δ_D of a set $D \subset \mathbb{R}^n$, the subgradient set $\partial \delta_D(x)$ is denoted by $N_D(x)$ and its elements are called the *normal vectors* to D at x.

Generalized Hessians are derived from normal vectors to the graphs of subgradient mappings. For any mapping $S : \mathbb{R}^m \rightrightarrows \mathbb{R}^p$, we denote by $\operatorname{gph} S$ the set of all pairs $(z, w) \in \mathbb{R}^m \times \mathbb{R}^n$ such that $w \in S(z)$. For any such pair (z, w), the *coderivative* of S at z for w is the mapping $D^*S(z \,|\, w) : \mathbb{R}^p \rightrightarrows \mathbb{R}^m$ defined by

$$D^*S(z \,|\, w)(w') = \{z' \mid (z', -w') \in N_{\operatorname{gph} S}(z, w)\}. \tag{2.3}$$

When S is single-valued and $\mathcal{C}^1$ around z with Jacobian matrix $\nabla S(z)$, the coderivative for $w = S(z)$ reduces to the adjoint linear mapping $w' \mapsto \nabla S(z)^* w'$.

For a subgradient mapping $\partial g : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ and a pair $(x, v) \in \operatorname{gph} \partial g$, the mapping $D^*(\partial g)(x \,|\, v)$ is the *coderivative Hessian* associated with g at x for v in the sense of

Mordukhovich [5] and is denoted by $\partial^2 g(x \,|\, v)$. If g is $\mathcal{C}^2$ around x with Hessian matrix $\nabla^2 g(x)$, then $\partial^2 g(x \,|\, v)$ for $v = \nabla g(x)$ reduces to the linear mapping $v' \mapsto \nabla^2 g(x)v'$.

In the context of our parametric model, as specified by the function $f : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}$, these concepts need some adaptation. The spotlight there is on the *partial* subgradient mapping $\partial_x f : \mathbb{R}^n \times \mathbb{R}^d \rightrightarrows \mathbb{R}^n$ defined by

$$(2.4) \qquad \partial_x f(x,u) = \{\text{set of subgradients } v \text{ of } f_u := f(\cdot,u) \text{ at } x\} = \partial f_u(x).$$

The importance of $\partial_x f$ comes from the elementary rule that wherever a function on $\mathbb{R}^n$ has a local minimum, its subgradient set must contain 0. Application of that rule to $f(\cdot,u) - \langle v, \cdot \rangle$ yields the first-order necessary condition with which we must work:

$$(2.5) \qquad x \text{ locally optimal in } \mathcal{P}(u,v) \implies v \in \partial_x f(x,u).$$

In particular, *any local optimal solution $\bar{x}$ to $\mathcal{P}(\bar{u},\bar{v})$ must have $\bar{v} \in \partial_x f(\bar{x},\bar{u})$.*

Although the constraints in $\mathcal{P}(u,v)$ are only implicit in our general framework, as signaled by ∞ values of f, a notion of "constraint qualification" comes in nonetheless. The *basic constraint qualification* at a feasible solution x to $\mathcal{P}(u,v)$ is the condition

$$\mathcal{Q}(x,u) \qquad (0,y) \in \partial^\infty f(x,u) \implies y = 0.$$

In our reference problem $\mathcal{P}(\bar{u},\bar{v})$, we will be concerned primarily with $\bar{x}$ and $\mathcal{Q}(\bar{x},\bar{u})$.

Note that $\partial^\infty f(x,u)$ refers to horizon subgradients of f as a function of both arguments, not just in x. As demonstrated in [6, Example 10.12], the constraint qualification $\mathcal{Q}(x,u)$ guarantees, in connection with the optimality condition in (2.5), the existence of y such that $(v,y) \in \partial f(x,u)$. In other words, it implies that

$$\partial_x f(x,u) \subset \{v \mid \exists\, y \text{ with } (v,y) \in \partial f(x,u)\}.$$

In the circumstances with which we will ultimately be working (in Theorem 2.3), this inclusion will turn out actually to be an equation (cf. Proposition 3.4). Nonetheless, the mapping $\partial_x f : \mathbb{R}^n \times \mathbb{R}^d \rightrightarrows \mathbb{R}^n$ rather than the mapping $\partial f : \mathbb{R}^n \times \mathbb{R}^d \rightrightarrows \mathbb{R}^n \times \mathbb{R}^d$ will be the vehicle for stating our results.

In analyzing the parametric behavior of locally optimal solutions on the platform of the optimality condition in (2.5), we will inevitably be concerned not only with $\partial_x f$ but also with its partial inverse

$$(2.6) \qquad M : (u,v) \mapsto \{x \mid v \in \partial_x f(x,u)\}.$$

Because the first-order condition $v \in \partial_x f(x,u)$ is also necessary for optimality in the minimization problem that defines $M_\delta(u,v)$ in (1.1) when $|x - \bar{x}| < \delta$, we know that

$$(2.7) \qquad x \in M_\delta(u,v),\ |x - \bar{x}| < \delta \implies x \in M(u,v).$$

Much will hinge on ascertaining when the graphs of M_δ and M actually coincide around $(\bar{u},\bar{v},\bar{x})$ for small δ, with M single-valued and Lipschitz continuous in such localization. The analysis will center on the coderivative mappings $D^*(\partial_x f)(x,u \,|\, v) : \mathbb{R}^n \rightrightarrows \mathbb{R}^n \times \mathbb{R}^d$ at points $(x,u,v) \in \operatorname{gph} \partial_x f$ near $(\bar{x},\bar{u},\bar{v})$.

It should be observed that the mapping $D^*(\partial_x f)(x,u \,|\, v)$ is not the same as the coderivative Hessian mapping

$$(2.8) \qquad \partial_x^2 f(x,u \,|\, v) := D^*(\partial f_u)(x \,|\, v) = \partial^2 f_u(x \,|\, v) \quad \text{for } f_u = f(\cdot,u).$$

With $f \in \mathcal{C}^2$ and $v = \nabla_x f(x,u)$ for instance, $\partial_x^2 f(x,u \,|\, v)$ comes out as $v' \mapsto \nabla_{xx}^2 f(x,u)v'$, while $D^*(\partial_x f)(x,u \,|\, v)$ comes out as $v' \mapsto (\nabla_{xx}^2 f(x,u)v', \nabla_{ux}^2 f(x,u)v')$. However, the mapping $\partial_x^2 f(x,u \,|\, v)(v')$ cannot even be identified, in general, with the mapping

$$v' \mapsto \{x' \mid \exists\, u',\ (x',u') \in D^*(\partial_x f)(x,u \,|\, v)(v')\}. \tag{2.9}$$

The former has u fixed in its definition, whereas the latter, which for comparison might be denoted by $\tilde{\partial}_x^2 f(x,u \,|\, v)$, depends on limits being taken in the u argument as well, and its graph may therefore be larger. Limits in u are a source of strength, however. The positive definiteness that we eventually require will be imposed on $\tilde{\partial}_x^2 f(\bar{x},\bar{u} \,|\, \bar{v})$ instead of $\partial_x^2 f(\bar{x},\bar{u} \,|\, \bar{v})$, although the notation $\tilde{\partial}_x^2 f(\bar{x},\bar{u} \,|\, \bar{v})$ will not be employed in expressing it.

The notion of prox-regularity must now be expanded in order for it to account for parametric effects in u.

Definition 2.1 (parametric prox-regularity). *The lsc expression $f(x,u)$ is prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$ if $\bar{v} \in \partial_x f(\bar{x},\bar{u})$ and there exist neighborhoods U of $\bar{u}$, X of $\bar{x}$, and V of $\bar{v}$, along with $\epsilon > 0$ and $r \geq 0$ such that*

$$\begin{aligned} f(x',u) \ &\geq\ f(x,u) + \langle v,\, x'-x\rangle - \frac{r}{2}|x'-x|^2 \quad \textit{for all} \quad x' \in X \\ &\qquad \textit{when} \quad v \in \partial_x f(x,u),\ v \in V,\ x \in X,\ u \in U,\ f(x,u) \leq f(\bar{x},\bar{u}) + \epsilon. \end{aligned} \tag{2.10}$$

It is continuously prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$ if, in addition, $f(x,u)$ is continuous as a function of $(x,u,v) \in \operatorname{gph} \partial_x f$ at $(\bar{x},\bar{u},\bar{v})$.

Our attention will be focused on the parametric version of continuous prox-regularity, which obviously entails continuous prox-regularity of $f(\cdot,\bar{u})$ at $\bar{x}$ for $\bar{v}$, in particular, but spreads some of it uniformly to subgradients of neighboring functions $f(\cdot,u)$. According to its definition, it provides the existence of a neighborhood $X \times U \times V$ of $(\bar{x},\bar{u},\bar{v}) \in \operatorname{gph} \partial_x f$ such that, for a certain $r \geq 0$, one has

$$\begin{aligned} f(x',u) \ &\geq\ f(x,u) + \langle v, x'-x\rangle - \frac{r}{2}|x'-x|^2 \quad \text{for all} \quad x' \in X \\ &\qquad \text{when} \quad (x,u,v) \in [X \times U \times V] \cap \operatorname{gph} \partial_x f. \end{aligned} \tag{2.11}$$

Strongly amenable functions furnish a prime source of examples for parametric continuous prox-regularity, as we show next. Amenable functions were first studied as a class in [10]. Parametric amenability, as defined in the next proposition, was introduced in [11].

Proposition 2.2 (prox-regularity from amenability). *Suppose that $f(x,u)$ is strongly amenable in x at $\bar{x}$ with compatible parameterization by u at $\bar{u}$, in the sense that on some neighborhood of $(\bar{x},\bar{u})$ there is a composite representation $f(x,u) = g(F(x,u))$ in which $F : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}^m$ is a $\mathcal{C}^2$ mapping and $g : \mathbb{R}^m \to \overline{\mathbb{R}}$ is a convex, proper, lsc function for which $F(\bar{x},\bar{u}) \in D := \operatorname{dom} g$ and*

$$z \in N_D(F(\bar{x},\bar{u})), \quad \nabla_x F(\bar{x},\bar{u})^* z = 0 \quad \Longrightarrow \quad z = 0. \tag{2.12}$$

Then, as long as $\bar{v} \in \partial_x f(\bar{x},\bar{u})$, one has $f(x,u)$ continuously prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$. Moreover, $\mathcal{Q}(\bar{x},\bar{u})$ holds.

Proof. From (2.12) we have, in particular, that f is strongly amenable at $(\bar{x}, \bar{u})$ as a function of (x, u), since that property by definition (cf. [6, Def. 10.23]) concerns a representation $f = g \circ F$ of the same kind but which need only satisfy

$$z \in N_D(F(\bar{x}, \bar{u})), \quad \nabla_x F(\bar{x}, \bar{u})^* z = 0, \quad \nabla_u F(\bar{x}, \bar{u})^* z = 0 \quad \Longrightarrow \quad z = 0,$$

where $N_D = \partial^\infty g$ (because g is convex; cf. [6, Prop. 8.12]). This condition implies by the subgradient chain rule in [6, Thm. 10.6] that $\partial^\infty f(\bar{x}, \bar{u})$ consists of all (v, y) such that there exists $z \in N_D(F(\bar{x}, \bar{u}))$ with $\nabla_x F(\bar{x}, \bar{u})^* z = v$ and $\nabla_u F(\bar{x}, \bar{u})^* z = y$. Clearly, then, it is impossible to have $(0, y) \in \partial^\infty f(\bar{x}, \bar{u})$ unless $y = 0$. Thus, $\mathcal{Q}(\bar{x}, \bar{u})$ is satisfied.

The condition in (2.12) carries over from $(\bar{x}, \bar{u})$ to all nearby (x, u) with $F(x, u) \in D$, for if not there would be a contradiction based on a simple argument of taking limits. This condition ensures by the same subgradient chain rule that for such (x, u) one has

$$\partial_x f(x, u) = \nabla_x F(x, u)^* \partial g(F(x, u)) = \{ v = \nabla_x F(x, u)^* z \mid z \in \partial g(F(x, u)) \}. \tag{2.13}$$

Assuming $\bar{v} \in \partial f(\bar{x}, \bar{u})$, let S be the mapping that associates with (x, u, v) the set of vectors z on the right-hand side of (2.13). We argue that S is locally bounded at $(\bar{x}, \bar{u}, \bar{v})$, i.e., that there exist $\epsilon > 0$ and $\zeta > 0$ such that

$$|(x, u, v) - (\bar{x}, \bar{u}, \bar{v})| \leq \epsilon, \quad z \in S(x, u, v) \quad \Longrightarrow \quad |z| \leq \zeta, \tag{2.14}$$

moreover, with (2.13) holding under these circumstances. Our reasoning is that if we had sequences $(x^\nu, u^\nu, v^\nu) \to (\bar{x}, \bar{u}, \bar{v})$ and $z^\nu \in S(x^\nu, u^\nu, v^\nu)$ with $0 < |z^\nu| \to \infty$, the vectors $\lambda^\nu z^\nu$ for $\lambda^\nu = 1/|z^\nu| \downarrow 0$ would cluster at some $\bar{z} \neq 0$. Then from having $\nabla_x F(x^\nu, u^\nu)^* [\lambda^\nu z^\nu] = \lambda^\nu v^\nu$ and $z^\nu \in \partial g(F(x^\nu, u^\nu))$ we would get $\nabla_x F(\bar{x}, \bar{u})^* \bar{z} = 0$ and $\bar{z} \in \partial^\infty g(F(\bar{x}, \bar{u}))$. Here we have $\partial^\infty g(F(\bar{x}, \bar{u})) = N_D(F(\bar{x}, \bar{u})$, so this would contradict (2.12).

Now let $X \times U \times V$ be a neighborhood of $(\bar{x}, \bar{u}, \bar{v})$ small enough that $f = g \circ F$ on $X \times U$ and $|(x, u, v) - (\bar{x}, \bar{u}, \bar{v})| \leq \epsilon$ when $(x, u, v) \in X \times U \times V$. Suppose $(x^\nu, u^\nu, v^\nu) \to (\bar{x}, \bar{u}, \bar{v})$ in $X \times U \times V$ with $v^\nu \in \partial_x f(x^\nu, u^\nu)$. Is it true that $f(x^\nu, u^\nu) \to f(\bar{x}, \bar{u})$? Taking advantage of the formula in (2.13) at (x^ν, u^ν, v^ν), select $z^\nu \in \partial g(F(x^\nu, u^\nu))$ such that $\nabla_x F(x^\nu, u^\nu)^* z^\nu = v^\nu$. We have $|z^\nu| \leq \zeta$ through (2.14), so by passing to subsequences we can reduce to the case where z^ν converges to some $\bar{z}$. The pairs $(F(x^\nu, u^\nu), z^\nu) \in \operatorname{gph} \partial g$ then converge to $(F(\bar{x}, \bar{u}), \bar{z})$, and since g is convex (hence subdifferentially continuous) this implies that $g(F(x^\nu, u^\nu)) \to g(F(\bar{x}, \bar{u}))$. Thus, $f(x^\nu, u^\nu) \to f(\bar{x}, \bar{u})$ as required.

Observe next that because F is of class $\mathcal{C}^2$ and the neighborhood U is bounded, there exists $r > 0$ such that, for all z with $|z| \leq \zeta$ and $u \in U$, the function $h_{zu} : x \mapsto \langle z, F(x, u) \rangle$ has $h_{zu}(x') \geq h_{zu}(x) + \langle \nabla h_{zu}(x), x' - x \rangle - \frac{r}{2} |x' - x|^2$ for all $x, x' \in X$. This tells us that

$$\begin{aligned} \langle z, F(x', u) - F(x, u) \rangle \geq \langle \nabla_x F(x, u)^* z, x' - x \rangle - \frac{r}{2} |x' - x|^2 \\ \text{when} \quad x, x' \in X, \ u \in U, \ |z| \leq \zeta. \end{aligned} \tag{2.15}$$

For any $x, x' \in X$, $u \in U$, and $v \in V$ with $v \in \partial_x f(x, u)$, we have $v = \nabla_x F(x, u)^* z$ for some $z \in \partial g(F(x, u))$, necessarily satisfying $|z| \leq \zeta$ by the local boundedness of S in (2.14). The convexity of g yields $g(F(x', u)) \geq g(F(x, u)) + \langle z, \, F(x', u) - F(x, u) \rangle$, and in combination with (2.15) we therefore have $f(x',u) - f(x,u) = g(F(x',u)) - g(F(x,u)) \geq \langle v,\, x'-x\rangle - \frac{r}{2}|x'-x|^2$. In other words we have (2.11), as required. □

As obvious very special cases of Proposition 2.2, f could be any $\mathcal{C}^2$ function (take $F = f$ and let $g(t) = t$ on $\mathbb{R}$) or any lsc, proper, convex function (take $g = f$ and $F = I$). For a broader discussion of the rich possibilities, see [11] and [6, Example 10.24].

THEOREM 2.3 (full stability). *Let $\bar{x}$ be a feasible solution to $\mathcal{P}(\bar{u},\bar{v})$ at which the first-order condition $\bar{v} \in \partial_x f(\bar{x},\bar{u})$ is satisfied along with the constraint qualification $\mathcal{Q}(\bar{x},\bar{u})$. Suppose $f(x,u)$ is continuously prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$. Then for $\bar{x}$ to be a locally optimal solution to $\mathcal{P}(\bar{u},\bar{v})$ that is fully stable, it is necessary and sufficient that the following second-order conditions be fulfilled:*

(a) $(x',u') \in D^*(\partial_x f)(\bar{x},\bar{u}\,|\,\bar{v})(v'),\ v' \neq 0 \ \Rightarrow\ \langle v', x'\rangle > 0$,

(b) $(0,u') \in D^*(\partial_x f)(\bar{x},\bar{u}\,|\,\bar{v})(0) \ \Rightarrow\ u' = 0$.

Moreover, in that case it follows, when $\delta > 0$ is sufficiently small, that for all (u,v) in some neighborhood of $(\bar{u},\bar{v})$ one has $M_\delta(u,v) = M(u,v) \cap \{x \mid |x-\bar{x}| < \delta\}$. In addition, the Lipschitz modulus of M_δ at $(\bar{u},\bar{v})$ is then given by

$$(2.16)\qquad (\operatorname{lip} M_\delta)(\bar{u},\bar{v}) = \max\left\{ \frac{|(u',v')|}{|x'|}(x',u') \in D^*(\partial_x f)(\bar{x},\bar{u}\,|\,\bar{v})(v'),\ x' \neq 0 \right\},$$

where $(\operatorname{lip}\, M_\delta)(\bar{u},\bar{v})$ *is the upper limit of* $|M_\delta(u_1,v_1) - M_\delta(u_2,v_2)|/|(u_1,v_1)-(u_2,v_2)|$ *as* $(u_1,v_1) \to (\bar{u},\bar{v})$ *and* $(u_2,v_2) \to (\bar{u},\bar{v})$ *with* $(u_1,v_1) \neq (u_2,v_2)$.

This is our main result. It will be proved in section 5. The proof of equivalence really centers just on the single-valuedness and Lipschitz continuity of M_δ. The local Lipschitz continuity of m_δ that has been incorporated into the definition of full stability is already a consequence of merely assuming $\mathcal{Q}(\bar{x},\bar{u})$ (cf. Proposition 3.5).

Theorem 2.3 covers the chief characterization of tilt stability in [8] as the case where the parameterization in u drops out and only the tilt vectors v remain. It adds to that characterization the corresponding specialization of the modulus formula in (2.16), i.e., $(\operatorname{lip} M_\delta)(\bar{v}) = \max\{|v'|/|x'| \mid x' \in \partial^2 f(\bar{x}\,|\,\bar{v})(v'),\ x' \neq 0\}$. Of course, it also provides a criterion for the basic form of stability in Definition 1.1.

COROLLARY 2.4 (basic stability). *The properties in Theorem* 2.3 *suffice for $\bar{x}$ to be a locally optimal solution to $\mathcal{P}(\bar{u},\bar{v})$ that is stable (in the basic sense).*

COROLLARY 2.5 (amenable case). *Suppose $f(x,u)$ is strongly amenable in x at $\bar{x}$ with compatible parameterization by u at $\bar{u}$. Then for $\bar{x}$ to be a locally optimal solution to $\mathcal{P}(\bar{u},\bar{v})$ that is fully stable, it is necessary and sufficient that the second-order conditions* (a) *and* (b) *of Theorem* 2.3 *be fulfilled along with the first-order condition $\bar{v} \in \partial_x f(\bar{x},\bar{u})$.*

Proof. This is immediate from Theorem 2.3 and Proposition 2.2. □

COROLLARY 2.6 (smooth case). *Let f be of class $\mathcal{C}^2$ around $(\bar{x},\bar{u})$. In order for $\bar{x}$ to be a locally optimal solution to $\mathcal{P}(\bar{u},\bar{v})$ that is fully stable, it is necessary and sufficient that $\nabla_x f(\bar{x},\bar{u}) = \bar{v}$ with $\nabla^2_{xx} f(\bar{x},\bar{u})$ positive definite.*

Proof. For f of this type we have the amenability in Corollary 2.5. The coderivative mapping $D^*(\partial_x f)(\bar{x},\bar{u})$ reduces to the mapping $v' \mapsto (\nabla^2_{xx} f(\bar{x},\bar{u})v', \nabla^2_{ux} f(\bar{x},\bar{u})v')$ as noted earlier. Condition (a) of Theorem 2.3 turns into the positive definiteness of $\nabla^2_{xx} f(\bar{x},\bar{u})$, while condition (b) is trivialized. □

It would be possible to derive the fact in Corollary 2.6 by classical methods, but we present it this way to show how it fits into the broader scene. The direct argument is not as easy as might be imagined, however; cf. the corresponding case of tilt stability in [8].

Corollary 2.6 brings attention to the "positive definiteness" in (a) of Theorem 2.3 as expressing a *second-order sufficient condition for optimality*, at least in combination with (b). This role was observed previously by Poliquin and Rockafellar in their tilt stability setting in [8]. Although second-order conditions in terms of coderivative Hessians can, in general, be far from the sharpest conditions for confirming local optimality, if that were the only issue, our results show that they are sharp for confirming local optimality together with stability. In the unconstrained optimization in Corollary 2.6, especially the tilt case with u suppressed, such a gap between stable and unstable second-order sufficient conditions is absent, but it appears to prevail almost everywhere else.

Theorem 2.3 requires f to belong to a class of prox-regular functions. Proposition 2.2 underscores the breadth of this class. Still, one can ask whether the stability conclusions might hold for an even larger class. The answer is essentially negative, however.

THEOREM 2.7 (effective need for prox-regularity). *Let $\bar{x}$ be a locally optimal solution to $\mathcal{P}(\bar{u}, \bar{v})$ that is fully stable and satisfies $\mathcal{Q}(\bar{x}, \bar{u})$. Then there is a proper, lsc function $\widehat{f}$ that has the prox-regularity ascribed to f in Theorem* 2.3 *and is locally equivalent to f for purposes of optimization, in the following sense: For the problems $\widehat{\mathcal{P}}(u, v)$ obtained with $\widehat{f}$ in place of f, the associated $\widehat{m}_\delta$ and $\widehat{M}_\delta$ for δ sufficiently small agree with m_δ and M_δ on a neighborhood of $(\bar{u}, \bar{v})$. Indeed, one can take $\widehat{f}(x, u)$ convex in x and such that, for (u, v) near $(\bar{u}, \bar{v})$, if $v \in \partial_x \widehat{f}(x, u)$ then $v \in \partial_x f(x, u)$ and $\widehat{f}(x, u) = f(x, u)$.*

This theorem will be proved in section 5 as well. The need for replacing f with a "locally equivalent" function $\widehat{f}$ to get a converse result can be seen already from examples focused on tilt stability. On $\mathbb{R}^2$, let $f(x, u) = |x| \sin(1/x) + 2|x|$ with $f(0, u) = 0$. The increasingly wild oscillations prevent f from having the prox-regularity demanded in Theorem 2.3 relative to $(\bar{x}, \bar{u}) = (0, 0)$ and $\bar{v} = 0$. The function $\widehat{f}(x, u) = |x|$ does have all the properties, however. (It is convex and therefore covered by Proposition 2.2.) For any $\delta > 0$ and $(u, v) \in W = \mathbb{R} \times (-1, 1)$ we have $\widehat{m}_\delta(u, v) = m_\delta(u, v) = 0$ and $\widehat{M}_\delta(u, v) = M_\delta(u, v) = 0$. Thus, f and $\widehat{f}$ are equivalent in the sense described in Theorem 2.7.

3. Prox-regularity under the constraint qualification. Laying the groundwork for the proof of Theorem 2.3, we show that the combination of parametric prox-regularity with the constraint qualification $\mathcal{Q}(\bar{x}, \bar{u})$ produces even more uniformity than has been explicitly built into Definition 2.1. The analysis revolves around a form of "graphically localized Lipschitz continuity" of set-valued mappings which will also be important later in the study of the mappings $\partial_x f$ and M but for now is utilized in an epigraphical context.

A mapping $S : \mathbb{R}^m \rightrightarrows \mathbb{R}^p$ has the *Aubin property* at $\bar{z}$ for $\bar{w}$, an element of $S(\bar{z})$, if there are neighborhoods Z of $\bar{z}$ and W of $\bar{w}$ along with $\kappa \geq 0$ such that

$$S(z') \cap W \subset S(z) + \kappa |z' - z| \mathbb{B} \quad \text{for all } z, z' \in Z. \tag{3.1}$$

Here $\mathbb{B}$ is the closed unit ball in $\mathbb{R}^p$. This property, which Aubin called "pseudo-Lipschitz continuity" in [4], reduces for single-valued S to Lipschitz continuity around $\bar{z}$. A powerful criterion has been found by Mordukhovich [5], [12], [13]: As long as gph S is closed relative to a neighborhood of $(\bar{z}, \bar{w})$, the Aubin property holds if and only if

$$z' \in D^*S(\bar{z} \,|\, \bar{w})(0) \implies z' = 0, \tag{3.2}$$

where, moreover, the lowest limiting value at $(\bar{z}, \bar{w})$ of the moduli κ that work in (3.1) has been characterized as the "norm" of the coderivative mapping $D^*S(\bar{z}\,|\,\bar{w})$. (That characterization will ultimately be the source of formula (2.16) in Theorem 2.3.) The great advantage of the Mordukhovich criterion is that, because coderivatives of S arise from normal vectors to $\operatorname{gph} S$, it can be invoked in tandem with the calculus of coderivatives that comes out of the calculus of normal vectors. See [6, Chap. 9] as well as [14].

The constraint qualification $\mathcal{Q}(\bar{x}, \bar{u})$ has an interpretation in this context in terms of the epigraphs

$$\operatorname{epi} f_u = \operatorname{epi} f(\cdot, u) := \{(x, \alpha) \in \mathbb{R}^n \times \mathbb{R} \mid f(x, u) \leq \alpha\}.$$

As shown in [6, Prop. 10.16], it amounts to the Mordukhovich criterion for the epigraphical mapping $E : u \mapsto \operatorname{epi} f_u$ at $\bar{u}$ for $(\bar{x}, f(\bar{x}, \bar{u}))$ and therefore to the Aubin property holding there. (The graph of this mapping is closed because f is lsc.)

Proposition 3.1 (consequences of the basic constraint qualification). *Under the constraint qualification $\mathcal{Q}(\bar{x}, \bar{u})$, there exist neighborhoods X_1 of $\bar{x}$ and U_1 of $\bar{u}$ along with $\epsilon > 0$ and $\kappa \geq 0$ such that*

$$\left.\begin{array}{r} x \in X_1,\ \ u, u' \in U_1 \\ f(x, u) \leq f(\bar{x}, \bar{u}) + \epsilon \end{array}\right\} \quad \Longrightarrow \quad \exists\, x' \text{ with } \left\{\begin{array}{l} |x' - x| \leq \kappa |u' - u|, \\ f(x', u') \leq f(x, u) + \kappa |u' - u|. \end{array}\right. \tag{3.3}$$

Proof. We have just observed that $\mathcal{Q}(\bar{x}, \bar{u})$ corresponds to having the Aubin property of the set-valued mapping $E : u \mapsto \operatorname{epi} f_u$ hold at $\bar{u}$ for $(\bar{x}, \bar{\alpha})$, where $\bar{\alpha} := f(\bar{x}, \bar{u})$, so the task is to show that this yields (3.3).

With convenient adjustments of notation to fit the epigraphical setting, the Aubin property in question can be identified with the existence of neighborhoods X_1 of $\bar{x}$ and U_1 of $\bar{u}$ along with $\epsilon > 0$ and $\kappa \geq 0$ such that, for all $u, u' \in U_1$, one has

$$[\operatorname{epi} f_u] \cap \big([X_1 \times [\bar{\alpha} - \epsilon, \bar{\alpha} + \epsilon]\big) \subset [\operatorname{epi} f_{u'}] + \kappa |u' - u| \big(\mathbb{B} \times [-1, 1]\big),$$

or in other words, the implication

$$\left.\begin{array}{r} x \in X_1 \\ \alpha \geq f(x, u) \\ |\alpha - \bar{\alpha}| \leq \epsilon \end{array}\right\} \quad \Longrightarrow \quad \exists (x', \alpha') \text{ with } \left\{\begin{array}{l} f(x', u') \leq \alpha', \\ |x' - x| \leq \kappa |u' - u|, \\ |\alpha' - \alpha| \leq \kappa |u' - u|. \end{array}\right. \tag{3.4}$$

Because f is lsc in this implication, we can arrange (by shrinking X_1 and U_1 if necessary) that $f(x, u) \geq \bar{\alpha} - \epsilon$ when $(x, u) \in X_1 \times U_1$. Then only the inequality $\alpha \leq \bar{\alpha} + \epsilon$ has force on the left. Conversely, only the upper bound provided by the inequality $|\alpha' - \alpha| \leq \kappa |u' - u|$ has force on the right. Thus, we can enhance (3.4) to

$$\left.\begin{array}{r} x \in X_1 \\ \alpha \geq f(x, u) \\ \alpha \leq \bar{\alpha} + \epsilon \end{array}\right\} \quad \Longrightarrow \quad \exists (x', \alpha') \text{ with } \left\{\begin{array}{l} f(x', u') \leq \alpha', \\ |x' - x| \leq \kappa |u' - u|, \\ \alpha' \leq \alpha + \kappa |u' - u|. \end{array}\right. \tag{3.5}$$

When (3.5) is invoked in the case of $\alpha = f(x, u)$, the x' it produces has $f(x', u') \leq f(x, u) + \kappa |u' - u|$. Since (3.5) holds for arbitrary $u, u' \in U_1$, we have (3.3). □

We use this now to bring out some important consequences of parametric prox-regularity.

PROPOSITION 3.2 (persistence of prox-regularity). *Let the constraint qualification $\mathcal{Q}(\bar{x},\bar{u})$ hold with $\bar{v}\in\partial_x f(\bar{x},\bar{u})$, and suppose that $f(x,u)$ is continuously prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$. Then an open neighborhood $X\times U\times V$ of $(\bar{x},\bar{u},\bar{v})$ can be found for which the uniform proximal subgradient property in* (2.11) *holds and, in addition,*

(a) *$f(x,u)$ is continuous as a function of $(x,u,v)\in[X\times U\times V]\cap\operatorname{gph}\partial_x f$,*

(b) *$\operatorname{gph}\partial_x f$ is closed relative to $X\times U\times V$.*

In particular, then, one has for all $(\tilde{x},\tilde{u},\tilde{v})\in[X\times U\times V]\cap\operatorname{gph}\partial_x f$ that $f(x,u)$ is continuously prox-regular in x at $\tilde{x}$ for $\tilde{v}$ with compatible parameterization by u at $\tilde{u}$.

Proof. Let X_0, U_0, and V_0 be neighborhoods as in the definition of continuous prox-regularity so that (2.11) holds for them and a certain r. Let X_1, U_1, λ, and κ have the property of Proposition 3.1. Choose an open neighborhood $X\times U\times V$ of $(\bar{x},\bar{u},\bar{v})$ such that $X\times U\times V\subset X_0\times U_0\times V_0$, $X\times U\subset X_1\times U_1$, and

$$(x,u,v)\in[X\times U\times V]\cap\operatorname{gph}\partial_x f \implies f(x,u)<f(\bar{x},\bar{u})+\lambda,$$

the latter being possible because $f(x,u)$ is continuous at $(\bar{x},\bar{u})$ as a function of $(x,u,v)\in[X\times U\times V]\cap\operatorname{gph}\partial_x f$. Then (2.11) holds for the neighborhoods X, U, and V, and (3.3) can be invoked in the simplified form

$$\text{(3.6)}\qquad \left.\begin{array}{r}(x,u)\in X\times U,\ u'\in U\\ (x,u,v)\in\operatorname{gph}\partial_x f,\ v\in V\end{array}\right\}\implies \exists\, x' \text{ with } \left\{\begin{array}{l}|x'-x|\le\kappa|u'-u|,\\ f(x',u')\le f(x,u)+\kappa|u'-u|.\end{array}\right.$$

Consider any sequence of points $(x^\nu,u^\nu,v^\nu)\in[X\times U\times V]\cap\operatorname{gph}\partial_x f$ that converges to a point $(\tilde{x},\tilde{u},\tilde{v})\in[X\times U\times V]$. We have to demonstrate that $f(x^\nu,u^\nu)\to f(\tilde{x},\tilde{u})$ and $(\tilde{x},\tilde{u},\tilde{v})\in\operatorname{gph}\partial_x f$.

We first apply (3.6) to $x=\tilde{x}$, $u=\tilde{u}$, and $u'=u^\nu$ to obtain for each ν the existence of $\tilde{x}^\nu$ such that $|\tilde{x}^\nu-\tilde{x}|\le\kappa|u^\nu-\tilde{u}|$ and $f(\tilde{x}^\nu,u^\nu)\le f(\tilde{x},\tilde{u})+\kappa|u^\nu-\tilde{u}|$. Then $\tilde{x}^\nu\to\tilde{x}$ and $f(\tilde{x}^\nu,u^\nu)\to f(\tilde{x},\tilde{u})$ (because f is lsc). Eventually, $\tilde{x}^\nu\in X$ so that we have

$$f(\tilde{x}^\nu,u^\nu)\ \ge\ f(x^\nu,u^\nu)+\langle v^\nu,\,\tilde{x}^\nu-x^\nu\rangle-\frac{r}{2}|\tilde{x}^\nu-x^\nu|^2.$$

The second and third terms on the right tend to 0 as $(x^\nu,u^\nu)\to(\tilde{x},\tilde{u})$, so from knowing that $f(\tilde{x}^\nu,u^\nu)\to f(\tilde{x},\tilde{u})$ we may conclude that $f(x^\nu,u^\nu)\to f(\tilde{x},\tilde{u})$ (because f is lsc). This establishes (a).

Next we consider any point $\hat{x}\in X$ and apply (3.6) to $x=\hat{x}$, $u=\tilde{u}$, and $u'=u^\nu$ to get for each ν the existence of $\hat{x}^\nu$ such that $|\hat{x}^\nu-\tilde{x}|\le\kappa|u^\nu-\tilde{u}|$ and $f(\hat{x}^\nu,u^\nu)\le f(\tilde{x},\tilde{u})+\kappa|u^\nu-\tilde{u}|$. We have $\hat{x}^\nu\to\tilde{x}$ and $f(\hat{x}^\nu,u^\nu)\to f(\hat{x},\tilde{u})$ (again because f is lsc). Furthermore, we have from (2.11) that

$$f(\hat{x}^\nu,u^\nu)\ \ge\ f(x^\nu,u^\nu)+\langle v^\nu,\,\hat{x}^\nu-x^\nu\rangle-\frac{r}{2}|\hat{x}^\nu-x^\nu|^2.$$

Limits are known for all the terms in this inequality, and in passing to them we obtain

$$f(\hat{x},\tilde{u})\ \ge\ f(\tilde{x},\tilde{u})+\langle\tilde{v},\,\hat{x}-\tilde{x}\rangle-\frac{r}{2}|\hat{x}-\tilde{x}|^2.$$

This has been shown to hold for arbitrary $\hat{x}$ in X, which is a neighborhood of $\tilde{x}$, so it follows that $\tilde{v}$ is a regular subgradient of $f(\cdot,\tilde{u})$ at $\tilde{x}$ and hence, in particular, that $\tilde{v}\in\partial_x f(\tilde{x},\tilde{u})$. This establishes (b). □

COROLLARY 3.3 (nonparametric case). *Suppose that a function $g : \mathbb{R}^n \to \overline{\mathbb{R}}$ is continuously prox-regular at $\bar{x}$ for $\bar{v}$. Then an open neighborhood $X \times V$ of $(\bar{x}, \bar{v})$ can be found for which the uniform proximal subgradient property in* (2.2) *holds and, in addition,*

(a) $g(x)$ *is continuous as a function of* $(x, v) \in [X \times V] \cap \operatorname{gph} \partial g$,

(b) $\operatorname{gph} \partial g$ *is closed relative to* $X \times V$.

In particular, then, one has for all $(\tilde{x}, \tilde{v}) \in [X \times V] \cap \operatorname{gph} \partial g$ *that* g *is continuously prox-regular at* $\tilde{x}$ *for* $\tilde{v}$.

Proof. Here we take $f(x, u) \equiv g(x)$. □

PROPOSITION 3.4 (subgradients under parametric prox-regularity). *Under the hypothesis of Proposition* 3.2, *there is a neighborhood of* $(\bar{x}, \bar{u}, \bar{v})$ *such that, as long as* (x, u, v) *lies in this neighborhood, one has*

$$v \in \partial_x f(x, u) \iff \exists\, y \text{ with } (v, y) \in \partial f(x, u). \tag{3.7}$$

Proof. Because $\mathcal{Q}(\bar{x}, \bar{u})$ holds, the constraint qualification $\mathcal{Q}(x, u)$ also holds when (x, u) is close enough to $(\bar{x}, \bar{u})$ with $f(x, u)$ close enough to $f(\bar{x}, \bar{u})$. (Otherwise, a contradiction can be reached by a simple argument based on the definition of $\partial^\infty f(\bar{x}, \bar{u})$.) As part of the continuous prox-regularity that is assumed, we know that when (x, u, v) approaches $(\bar{x}, \bar{u}, \bar{v})$ within $\operatorname{gph} \partial_x f$, $f(x, u)$ automatically approaches $f(\bar{x}, \bar{u})$, so the proviso about $f(x, u)$ being close enough to $f(\bar{x}, \bar{u})$ is superfluous.

The constraint qualification $\mathcal{Q}(x, u)$ guarantees that "⇒" holds in (3.7); see [6, Cor. 10.11]. For the converse, suppose that $(v, y) \in \partial f(x, u)$ with (x, u, v) in an open neighborhood $X \times U \times V$ of $(\bar{x}, \bar{u}, \bar{v})$ of the kind in Proposition 3.2. Then by definition there is a sequence of points $(x^\nu, u^\nu, v^\nu, y^\nu) \to (x, u, v, y)$ with $f(x^\nu, u^\nu) \to f(x, u)$ and (v^ν, y^ν) a regular subgradient of f at (x^ν, u^ν). Then v^ν is a regular subgradient of $f(\cdot, u^\nu)$ at x^ν and, in particular, $v^\nu \in \partial_x f(x^\nu, u^\nu)$. Eventually (x^ν, u^ν, v^ν) belongs to the neighborhood $X \times U \times V$, and by appealing to (b) of Proposition 3.2 we see that the limit (x, u, v) still lies in $\operatorname{gph} \partial_x f$. Thus, "⇐" holds in (3.7) when $(x, u, v) \in X \times U \times V$. □

We finish with a result about the behavior of the functions m_δ and mappings M_δ in (1.1), which will be needed later in the proof of Theorem 2.3.

PROPOSITION 3.5 (convergence in local optimality). *Suppose* $M_\delta(\bar{u}, \bar{v}) = \{\bar{x}\}$ *for some* $\delta > 0$ *and the constraint qualification* $\mathcal{Q}(\bar{x}, \bar{u})$ *is satisfied. Then* m_δ *is Lipschitz continuous around* $(\bar{u}, \bar{v})$, *and for every* $\epsilon > 0$ *there is a neighborhood* W *of* $(\bar{u}, \bar{v})$ *such that*

$$(u, v) \in W \implies \emptyset \neq M_\delta(u, v) \subset \{x \mid |x - \bar{x}| < \epsilon\}.$$

Proof. In terms of the function

$$g_\delta(u, v, x) := \begin{cases} f(x, u) - \langle v, x \rangle & \text{if } |x - \bar{x}| \leq \delta, \\ \infty & \text{if } |x - \bar{x}| > \delta, \end{cases}$$

we have $m_\delta(u, v) = \inf_x g_\delta(u, v, x)$ and $M_\delta(u, v) = \operatorname{argmin}_x g_\delta(u, v, x)$. Here g_δ is lsc and proper on $\mathbb{R}^d \times \mathbb{R}^n \times \mathbb{R}^n$, and for each (u, v) the level sets of the form $\{x \mid g_\delta(u, v, x) \leq \alpha\}$, $\alpha \in \mathbb{R}$, are of course all contained in the ball $\{x \mid |x - \bar{x}| \leq \delta\}$. Further, we have

$$\partial^\infty g_\delta(\bar{u}, \bar{v}, \bar{x}) = \{(y, 0, w) \mid (w, y) \in \partial^\infty f(\bar{x}, \bar{u})\}$$

by the calculus rule in [6, Exercise 8.8(c)] so that, from $\mathcal{Q}(\bar{x},\bar{u})$, g_δ has $(y,z,0) \in \partial^\infty g_\delta(\bar{u},\bar{v},\bar{x})$ only for $(y,z) = (0,0)$. On the basis of this constraint qualification we know that m_δ is Lipschitz continuous on some neighborhood of $(\bar{u},\bar{v})$; cf. [6, Thm. 10.13]. The rest then follows from the fundamental theorem on parametric optimization in [6, Thm. 1.17]. □

4. Coderivative analysis of subgradient mappings.

Our investigation shifts now to coderivatives of the mapping $\partial_x f$ and its partial inverse M introduced in (2.6).

PROPOSITION 4.1 (partial inverse mapping). *The mapping M has its coderivatives related to those of $\partial_x f$ by*

$$(u',-v') \in D^*M(u,v\,|\,x)(-x') \iff (x',u') \in D^*(\partial_x f)(x,u\,|\,v)(v'). \tag{4.1}$$

When $\operatorname{gph}\partial_x f$ *is closed locally around* (x,u,v)*, the condition*

$$(0,u') \in D^*(\partial_x f)(x,u\,|\,v)(v') \quad\Longrightarrow\quad (u',v') = (0,0) \tag{4.2}$$

is necessary and sufficient for M to have the Aubin property at (u,v) for x.

Proof. By definition, $(u',-v') \in D^*M(u,v\,|\,x)(-x')$ means that $(u',-v',x')$ belongs to $N_{\operatorname{gph} M}(u,v,x)$. Since the elements (u,v,x) of $\operatorname{gph} M$ correspond simply to the elements (x,u,v) of $\operatorname{gph}\partial_x f$, this is the same as having $(x',u',-v') \in N_{\operatorname{gph}\partial_x f}(x,u,v)$. However, that means $(x',u') \in D^*(\partial_x f)(x,u\,|\,v)(v')$.

Local closedness of $\operatorname{gph}\partial_x f$ around (x,u,v) corresponds to local closedness of $\operatorname{gph} M$ around (u,v,x) and allows the Aubin property of M at (u,v) for x to be captured by the Mordukhovich criterion: $(u',-v') \in D^*M(u,v\,|\,x)(0)$ only for $(u',-v') = (0,0)$. When the latter is translated through (4.1), it comes out as (4.2). □

COROLLARY 4.2 (Aubin property of the partial inverse). *Under the hypothesis of Theorem* 2.3*, conditions* (a) *and* (b) *in the statement of that theorem guarantee that M has the Aubin property at $(\bar{u},\bar{v})$ for $\bar{x}$.*

Proof. The hypothesis guarantees through Proposition 3.2 that $\operatorname{gph}\partial_x f$ is closed locally around $(\bar{x},\bar{u},\bar{v})$. The issue then is whether (4.2) holds there. Let $(0,u') \in D^*(\partial_x f)(\bar{x},\bar{u}\,|\,\bar{v})(v')$. From condition (a) of Theorem 2.3, we must have $v' = 0$. However, then by condition (b) of Theorem 2.3 we must have $u' = 0$. Thus, (4.2) is correct. □

PROPOSITION 4.3 (partial coderivatives). *Consider in terms of $f_u = f(\cdot,u)$ the set-valued mapping $G : \mathbb{R}^d \rightrightarrows \mathbb{R}^n \times \mathbb{R}^n$ defined by*

$$G(u) = \operatorname{gph}\partial f_u = \{(x,v) \mid (x,u,v) \in \operatorname{gph}\partial_x f\}. \tag{4.3}$$

When $\operatorname{gph}\partial_x f$ *is closed locally around* $(\bar{x},\bar{u},\bar{v})$*, condition* (b) *of Theorem* 2.3 *is equivalent to G having the Aubin property at $\bar{u}$ for $(\bar{x},\bar{v})$. Furthermore,* (b) *ensures that for all $(x,u,v) \in \operatorname{gph}\partial_x f$ in some neighborhood of $(\bar{x},\bar{u},\bar{v})$, one has*

$$\partial^2 f_u(x\,|\,v)(v') \subset \{x' \mid \exists\, u' \text{ with } (x',u') \in D^*(\partial_x f)(x,u\,|\,v)(v')\}. \tag{4.4}$$

Proof. The elements (u,x,v) of $\operatorname{gph} G$ correspond under permutation to the elements (x,u,v) of $\operatorname{gph}\partial_x f$. From this we get

$$\begin{aligned} (x',u') \in D^*(\partial_x f)(x,u\,|\,v)(v') &\iff (x',u',-v') \in N_{\operatorname{gph}\partial_x f}(x,u,v) \\ &\iff (u',x',-v') \in N_{\operatorname{gph} G}(u,x,v) \\ &\iff u' \in D^*G(u\,|\,x,v)(-x',v'). \end{aligned} \tag{4.5}$$

The local closedness of $\operatorname{gph}\partial_x f$ around $(\bar{x},\bar{u},\bar{v})$ corresponds to the local closedness of $\operatorname{gph}G$ around $(\bar{u},\bar{x},\bar{v})$. With such closedness, G has the Aubin property at $\bar{u}$ for $(\bar{x},\bar{v})$ if and only if the Mordukhovich criterion is satisfied, namely, that $u' \in D^*G(\bar{u}\,|\,\bar{x},\bar{v})(0,0)$ only for $u'=0$. This is identical under (4.5) to condition (b) of Theorem 2.3.

The Aubin property of G at $\bar{u}$ for $(\bar{x},\bar{v})$ entails the Aubin property at u for (x,v) whenever (u,x,v) is near enough to $(\bar{u},\bar{x},\bar{v})$ in $\operatorname{gph}G$. Thus, for all such (u,x,v) in $\operatorname{gph}G$, and also within the neighborhood of $(\bar{u},\bar{x},\bar{v})$ where $\operatorname{gph}G$ is locally closed, the Mordukhovich criterion is satisfied; we can write this as

$$(4.6)\qquad (u',0,0)\in N_{\operatorname{gph}G}(u,x,v) \implies u'=0.$$

Fix any such element of $\operatorname{gph}G$, say $(\tilde{u},\tilde{x},\tilde{v})$. By determining the normal vectors to the set $G(\tilde{u})=\operatorname{gph}\partial f_{\tilde{u}}$ at $(\tilde{x},\tilde{v})$, we can determine the coderivative mapping $D^*(\partial f_{\tilde{u}})(\tilde{x}\,|\,\tilde{v})=\partial^2 f_{\tilde{u}}(\tilde{x}\,|\,\tilde{v})$. Observing that

$$(4.7)\qquad G(\tilde{u})=\{(x,v)\mid F(x,v)\in\operatorname{gph}G\}\quad \text{for } F:(x,v)\mapsto(\tilde{u},x,v),$$

we apply the chain rule for normal vectors in [6, Thm. 6.14]. Because $\operatorname{gph}G$ is locally closed around $(\tilde{u},\tilde{x},\tilde{v})$, this chain rule is valid as long as the constraint qualification holds that

$$(u',x',v')\in N_{\operatorname{gph}G}(\tilde{u},\tilde{x},\tilde{v}),\ \nabla F(\tilde{x},\tilde{v})^*(u',x',v')=(0,0) \implies (u',x',v')=(0,0,0).$$

Trivially, however, $\nabla F(\tilde{x},\tilde{v})^*(u',x',v')=(0,0)$ only when $(x',v')=(0,0)$, so this constraint qualification comes out as (4.6) in the case of $(u,x,v)=(\tilde{u},\tilde{x},\tilde{v})$ and thus is indeed satisfied. The chain rule allows us to deduce from (4.7) that

$$(4.8)\qquad \begin{aligned} N_{G(\tilde{u})}(\tilde{x},\tilde{v}) &\subset \{(x'',v'')\mid \exists\,(u',x',v')\in N_{\operatorname{gph}G}(\tilde{u},\tilde{x},\tilde{v}) \\ &\qquad\qquad \text{with } \nabla F(\tilde{x},\tilde{v})^*(u',x',v')=(x'',v'')\} \\ &= \{(x',v')\mid \exists\,u' \text{ with } (u',x',v')\in N_{\operatorname{gph}G}(\tilde{u},\tilde{x},\tilde{v})\}. \end{aligned}$$

Noting that $\operatorname{gph}D^*(\partial f_{\tilde{u}})(\tilde{x}\,|\,\tilde{v})$ consists of the pairs (v',x') with $(x',-v')\in N_{G(\tilde{u})}(\tilde{x},\tilde{v})$, whereas $\operatorname{gph}D^*(\partial_x f)(\tilde{x},\tilde{u}\,|\,\tilde{v})$ consists by (4.5) of all (v',x',u') such that $(u',x',-v')\in N_{\operatorname{gph}G}(\tilde{u},\tilde{x},\tilde{v})$, we obtain from (4.8) that (4.4) holds. □

In support of the final proposition in this section, the following lemma will be crucial.

LEMMA 4.4 (positive definiteness estimate). *Let $g:\mathbb{R}^n\to\mathbb{R}$ be continuously prox-regular at $\tilde{x}$ for $\tilde{v}$ and let $\epsilon>0$. If the inequality $\langle x',v'\rangle\ge\epsilon|v'|^2$ holds for all $(v',x')\in\operatorname{gph}\partial^2 g(\tilde{x},\tilde{v})$ such that $x'=\lambda v'$ for some $\lambda\in\mathbb{R}$, then it also holds without that restriction.*

Proof. Consider any $\mu\in(0,\epsilon)$. Let $G=\operatorname{gph}\partial g$. Under our inequality assumption there must be an open neighborhood $X_0\times V_0$ of $(\tilde{x},\tilde{v})$ such that

$$(4.9)\quad (x,v)\in[X_0\times V_0]\cap\operatorname{gph}\partial g,\quad (\lambda v',v')\in\operatorname{gph}\partial^2 g(x\,|\,v),\quad |v'|=1 \implies \lambda\ge\mu,$$

inasmuch as $\operatorname{gph}\partial^2 g(x\,|\,v)$ consists of the vectors (v',x') with $(x',-v')\in N_{\operatorname{gph}\partial g}(x,v)$, and the graph of the mapping $N_{\operatorname{gph}\partial g}$ is closed (by the general definition of normal cones).

We can suppose (by shrinking X_0 and V_0 if necessary) that $X_0\times V_0$ lies within a neighborhood $X\times V$ for which the continuous prox-regularity property in (2.2) is operational and, moreover, through Corollary 3.3, makes g continuously prox-regular at x

for v when $(x,v) \in [X \times V] \cap \operatorname{gph} \partial g$. Consider now within $[X_0 \times V_0] \cap \operatorname{gph} \partial g$ any point (x,v) with the special property that the ∂g is proto-differentiable at x for v and the corresponding derivative mapping $D(\partial g)(x \,|\, v) : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is generalized linear. (This property is known actually to hold in an almost everywhere sense because continuous prox-regularity makes $\operatorname{gph} \partial g$ a graphically Lipschitzian manifold of dimension n in its localization relative to $X \times V$; cf. [9, Prop. 4.8]. The points (x,v) in question are the "Rademacher points" of $\operatorname{gph} \partial g$ near $(\tilde{x}, \tilde{v})$. Proto-differentiability is the graphical counterpart to function differentiability; see [6]. A mapping is generalized linear when its graph is a subspace.)

In this situation, three facts are at our disposal. First, according to a theorem of Rockafellar and Zagrodny [15], the graph of $D(\partial g)(x \,|\, v)$ is included in the graph of $D^*(\partial g)(x \,|\, v) = \partial^2 g(x \,|\, v)$, so that by (4.9) we have

$$(\lambda v', v') \in \operatorname{gph} D(\partial g)(x \,|\, v), \quad |v'| = 1 \quad \Longrightarrow \quad \lambda \geq \mu. \tag{4.10}$$

Second, because of the proto-differentiability, $D(\partial g)(x \,|\, v)$ is the subgradient mapping ∂h for $h = d^2 g(x \,|\, v)$, the second subderivative function associated with g at x for v; this holds through prox-regularity as shown in [9, Cor. 6.2]. Third, the generalized linearity of ∂h corresponds to h being a generalized (purely) quadratic function: the sum of a purely quadratic function on $\mathbb{R}^n$ and the indicator of a subspace. Thus, there is a subspace L of $\mathbb{R}^n$ along with a symmetric, positive semidefinite matrix $Q \in \mathbb{R}^{n \times n}$ such that

$$D(\partial g)(x \,|\, v)(v') = \begin{cases} Qv' + L^{\perp} & \text{when } v' \in L, \\ \emptyset & \text{when } v' \notin L. \end{cases} \tag{4.11}$$

In combining (4.11) with (4.10), we see that the eigenvalues λ of Q relative to L must all satisfy $\lambda \geq \mu$. This tells us that the generalized linear mapping $D(\partial g)(x \,|\, v)$ is μ-strongly monotone. We invoke next the criterion of [9, Prop. 5.7]: Because the mappings $D(\partial g)(x \,|\, v)$ of the special type just investigated are all μ-strongly monotone, the localization of ∂g that we are working with is itself μ-strongly monotone.

A monotone mapping T has $\langle x', v' \rangle \geq 0$ whenever $x' \in D^*T(x \,|\, v)(v')$, as shown by Poliquin and Rockafellar [8, Thm. 2.1]; therefore, a μ-monotone mapping T (for which $T - \mu I$ is monotone) has $\langle x', v' \rangle \geq \mu |v'|^2$ whenever $x' \in D^*T(x \,|\, v)(v')$. In particular, then, in taking T to be our localization of ∂g, we see that

$$(x,v) \in [X_0 \times V_0] \cap \operatorname{gph} \partial g, \quad x' \in \partial^2 g(x \,|\, v)(v') \quad \Longrightarrow \quad \langle x', v' \rangle \geq \mu |v'|^2.$$

Applying this at $(x,v) = (\tilde{x}, \tilde{v})$ and recalling that μ was an arbitrary value in $(0, \epsilon)$, we reach the desired conclusion that $\langle x', v' \rangle \geq \epsilon |v'|^2$ whenever $x' \in \partial^2 g(\tilde{x} \,|\, \tilde{v})(v')$. □

PROPOSITION 4.5 (uniform positive definiteness). *Let the constraint qualification $\mathcal{Q}(\bar{x}, \bar{u})$ hold with $\bar{v} \in \partial_x f(\bar{x}, \bar{u})$, and suppose that $f(x,u)$ is continuously prox-regular in x at $\bar{x}$ for $\bar{v}$ with compatible parameterization by u at $\bar{u}$. If conditions* (a) *and* (b) *of Theorem* 2.3 *hold as well, there must actually exist a constant $\epsilon > 0$ and a neighborhood $X \times U \times V$ of $(\bar{x}, \bar{u}, \bar{v})$ for which, in terms of $f_u = f(\cdot, u)$, one has*

$$\left. \begin{array}{r} x' \in \partial^2 f_u(x \,|\, v)(v') \\ (x,u,v) \in [X \times U \times V] \cap \operatorname{gph} \partial_x f \end{array} \right\} \quad \Longrightarrow \quad \langle x', v' \rangle \geq \epsilon |v'|^2. \tag{4.12}$$

Conversely, if this property holds, then condition (a) *of Theorem* 2.3 *must hold with*

$$(x', u') \in D^*(\partial_x f)(\bar{x}, \bar{u} \,|\, \bar{v})(v') \quad \Longrightarrow \quad \langle x', v' \rangle \geq \epsilon |v'|^2. \tag{4.13}$$

Proof. Our hypothesis ensures through Proposition 3.2 that for all (x, u, v) near enough to $(\bar{x}, \bar{u}, \bar{v})$ with $v \in \partial f_u(x)$ the function f_u is continuously prox-regular at x for v. In combining it with condition (b) of Theorem 2.3 and invoking Proposition 4.3, we get the coderivative inclusion in (4.4) to hold locally. Suppose now that condition (a) of Theorem 2.3 is satisfied along with condition (b). To justify the locally uniform positive definiteness property claimed in that case, we will rely on Lemma 4.4, according to which we can obtain (4.12) by demonstrating that

$$\left.\begin{array}{r} \lambda z \in \partial^2 f_u(x \,|\, v)(z) \\ (x, u, v) \in [X \times U \times V] \cap \operatorname{gph} \partial_x f \end{array}\right\} \implies \lambda \geq \epsilon.$$

Through the inclusion in (4.4), it suffices to verify the existence of $\epsilon > 0$ such that

$$\left.\begin{array}{r} (\lambda z, w) \in D^*(\partial_x f)(x, u \,|\, v)(z) \\ (x, u, v) \in [X \times U \times V] \cap \operatorname{gph} \partial_x f \end{array}\right\} \implies \lambda \geq \epsilon. \tag{4.14}$$

Suppose there is no such ϵ. Then there must exist sequences $(x^\nu, u^\nu, v^\nu) \to (\bar{x}, \bar{u}, \bar{v})$ in $\operatorname{gph} \partial_x f$ along with scalars $\lambda^\nu \downarrow 0$ and vectors z^ν and w^ν with $z^\nu \neq 0$, such that $(\lambda^\nu z^\nu, w^\nu) \in D^*(\partial_x f)(x^\nu, u^\nu \,|\, v^\nu)(z^\nu)$. The latter means by definition that $(\lambda^\nu z^\nu, w^\nu, -z^\nu)$ is a normal vector to $\operatorname{gph} \partial_x f$ at (x^ν, u^ν, v^ν). Rescaling, we can make $|z^\nu| = 1$.

By passing to subsequences, we can suppose z^ν converges to some z with $|z| = 1$ and, as for w^ν, reduce to two cases: either w^ν converges to some w or $0 < |w^\nu| \to \infty$. In the first case we have in the limit that $(0, w, -z)$ is normal to $\operatorname{gph} \partial_x f$ at $(\bar{x}, \bar{u}, \bar{v})$, so $(0, w) \in D^*(\partial_x f)(\bar{x}, \bar{u} \,|\, \bar{v})(z)$. However, that is excluded by condition (a) of Theorem 2.3. In the second case, let $\hat{w}^\nu = w^\nu / |w^\nu|$ and $\hat{z}^\nu = z^\nu / |w^\nu|$. Then $\hat{z}^\nu \to 0$, whereas, by passing once more to subsequences if necessary, we can suppose $\hat{w}^\nu$ converges to some $\hat{w}$ with $|\hat{w}| = 1$. We have $(\lambda^\nu \hat{z}^\nu, \hat{w}^\nu, -\hat{z}^\nu)$ normal to $\operatorname{gph} \partial_x f$ at (x^ν, u^ν, v^ν), and hence in the limit that $(0, \hat{w}, 0)$ is normal to $\operatorname{gph} \partial_x f$ at $(\bar{x}, \bar{u}, \bar{v})$. Then $(0, \hat{w}) \in D^*(\partial_x f)(\bar{x}, \bar{u} \,|\, \bar{v})(0)$, but that is impossible under condition (b) of Theorem 2.3.

Turning now to the converse claim at the end of the proposition, we drop the assumption that (a) and (b) of Theorem 2.3 hold and suppose instead that (4.12) is satisfied by ϵ and a neighborhood $X \times U \times V$. Let $(x', u') \in D^*(\partial_x f)(\bar{x}, \bar{u} \,|\, \bar{v})(v')$, so that $(x', u', -v')$ is a normal vector to $\operatorname{gph} \partial_x f$ at $(\bar{x}, \bar{u}, \bar{v})$. By definition, then, there exist sequences $(\bar{x}^\nu, \bar{u}^\nu, \bar{v}^\nu) \to (\bar{x}, \bar{u}, \bar{v})$ in $X \times U \times V$ and $(\tilde{x}^\nu, \tilde{u}^\nu, \tilde{v}^\nu) \to (x', u', v')$ in which $(\tilde{x}^\nu, \tilde{u}^\nu, -\tilde{v}^\nu)$ is a *regular* normal vector to $\operatorname{gph} \partial_x f$ at $(\bar{x}^\nu, \bar{u}^\nu, \bar{v}^\nu)$. Since $\operatorname{gph} \partial f_u$ is merely the cross section of $\operatorname{gph} \partial_x f$ obtained by fixing the u argument, $(\tilde{x}^\nu, -\tilde{v}^\nu)$ is then a regular normal vector to $\operatorname{gph} \partial f_{\bar{u}^\nu}$ at $(\bar{x}^\nu, \bar{v}^\nu)$. This implies $\tilde{x}^\nu \in D^*(\partial f_{\bar{u}^\nu})(\bar{x}^\nu \,|\, \bar{v}^\nu)(\tilde{v}^\nu) = \partial^2 f_{\bar{u}^\nu}(\bar{x}^\nu \,|\, \bar{v}^\nu)(\tilde{v}^\nu)$, so $\langle \tilde{x}^\nu, \tilde{v}^\nu \rangle \geq \epsilon |\tilde{v}^\nu|^2$ by (4.12). Taking the limit we get the inequality in (4.13), as desired. □

5. Proof of the main result. Two auxiliary facts still have to be established in order to set the stage completely for the proof of necessity and sufficiency in Theorem 2.3. We first deal with one needed in the sufficiency argument. We denote by $\mathbb{B}(v, \lambda)$ the closed ball of radius λ around v.

Lemma 5.1 (subgradient inversion estimate). *Let $g : \mathbb{R}^n \to \mathbb{R}$ be convex and let O be an open convex set on which g is finite and strongly convex with modulus μ. Suppose $v_0 \in O$ and $w_0 \in \partial g(v_0)$, and let $\lambda > 0$ be small enough that the $\mathbb{B}(v_0, \lambda)$ lies in O. Then for every $w \in \mathbb{B}(w_0, \lambda\mu)$ there is a unique $v \in \mathbb{B}(v_0, \lambda)$ with $w \in \partial g(v)$. Furthermore, the single-valued mapping $w \mapsto v$ defined in this way is Lipschitz continuous on $\mathbb{B}(w_0, \lambda\mu)$ with constant $1/\mu$.*

Proof. Fix any $\lambda_0 > \lambda$ small enough that $\mathbb{B}(v_0, \lambda_0)$ still lies in O. Define $g_0(v)$ to be $g(v_0+v) - \langle w_0, v\rangle$ when $v \in \lambda_0\mathbb{B}$ but ∞ otherwise. Then $\partial g_0(v) = \partial g(v_0+v) - w_0$ for $v \in \lambda\mathbb{B}$ and in particular $0 \in \partial g_0(0)$. It will suffice to prove that for every $w \in \mu\lambda\mathbb{B}$ there is a unique $v \in \lambda\mathbb{B}$ with $w \in \partial g_0(v)$ and that the associated mapping $w \mapsto v$ has the Lipschitz property claimed.

Because g is continuous on O by virtue of its convexity, g_0 is lsc on $\mathbb{R}^n$ as well as μ-strongly convex on its effective domain $\lambda_0\mathbb{B}$. Then the subgradient mapping ∂g_0 is μ-strongly monotone, and the conjugate convex function g_0^* is differentiable on $\mathbb{R}^n$, its gradient mapping being globally Lipschitz continuous with constant $1/\mu$; see [6, Thm. 11.13, Prop. 12.60]. This makes ∂g_0^* reduce to ∇g_0^*, and since $\partial g_0^* = (\partial g_0)^{-1}$ in general, it follows that we have $v = \nabla g_0^*(w)$ if and only if $w \in \partial g_0(v)$.

Our task reduces to demonstrating that in these circumstances we have $v \in \lambda\mathbb{B}$ when $w \in \lambda\mu\mathbb{B}$. We know in general from the theory of conjugate functions that

$$\partial g_0^*(w) = \operatorname{argmin}_v\{g_0(v) - \langle v, w\rangle\} = \operatorname{argmin}_v\{g(v_0+v) - \langle v,\ w_0+w\rangle + \delta_{\lambda_0\mathbb{B}}(v)\}.$$

The rules of subgradient calculus tell us that the minimum is attained at v if and only if $0 \in \partial g(v_0+v) - [w_0+w] + N_{\lambda_0\mathbb{B}}(v)$. Therefore, $v = \nabla g_0^*(w)$ if and only if $v \in \lambda_0\mathbb{B}$ and there exists $\theta \geq 0$ such that $w_0 + w - \theta v \in \partial g(v_0+v)$, in which case $w - \theta v \in \partial g_0(v)$. Here necessarily $\theta = 0$ unless $|v| = \lambda_0$. Thus we can finish off by showing that if $w - \theta v \in \partial g_0(v)$ and $|v| = \lambda_0$, then $|w| > \mu\lambda$.

We accomplish this by appealing to the fact that ∂g_0 is μ-strongly monotone with $0 \in \partial g_0(0)$. In combination with the relation $w - \theta v \in \partial g_0(v)$ this yields $\langle w - \theta v, v\rangle \geq \mu|v|^2$, and hence $\langle w, v\rangle \geq (\mu+\theta)|v|^2$. That implies $|w| \geq (\mu+\theta)|v| = (\mu+\theta)|\lambda_0| > \mu\lambda$. □

Proof of sufficiency in Theorem 2.3. Assume the hypothesis of Theorem 2.3 along with conditions (a) and (b). Full stability will be demonstrated, and the assertion about $M_\delta(u,v)$ equaling $M(u,v)$ will be obtained as a by-product.

Our assumptions yield the uniform positive definiteness property in Proposition 4.5. In particular, in order to get started we observe that this implies for the function $f_{\bar{u}}$ that

$$x' \in \partial^2 f_{\bar{u}}(\bar{x}\,|\,\bar{v})(v'),\ v' \neq 0 \implies \langle v', x'\rangle > 0.$$

Since $f_{\bar{u}}$ is continuously prox-regular at $\bar{x}$ for $\bar{v}$ in consequence of the parametric continuous prox-regularity of f, we have everything in place to apply the main result of Poliquin and Rockafellar in [8] and conclude that we at least have tilt stability. All that we really need from this, however, is the fact that, for $\delta > 0$ sufficiently small, we have $M_\delta(\bar{u}, \bar{v}) = \{\bar{x}\}$. Then we can invoke Proposition 3.5 in tandem with (2.7) to see that, for some neighborhood W of $(\bar{u}, \bar{v})$, we have m_δ Lipschitz continuous on W and

$$\begin{aligned}\emptyset \neq M_\delta(u,v) &= M_\delta(u,v) \cap \{x \mid |x - \bar{x}| < \delta\}\\ &\subset\ M(u,v) \cap \{x \mid |x-\bar{x}| < \delta\} \quad \text{ for all } (u,v) \in W.\end{aligned} \tag{5.1}$$

Conversely, we know from Corollary 4.2 that M has the Aubin property at $(\bar{u}, \bar{v})$ for $\bar{x}$. The Lipschitz modulus of M at $(\bar{u}, \bar{v})$ for $\bar{x}$ (in the set-valued sense of the Aubin property—see [6, Def. 9.36]) is given then by the "norm," $|D^*M(\bar{u}, \bar{v})|^+$ in [6, Thm. 9.40]. By virtue of the equivalence in (4.1) and this "norm" value can be expressed as the max on the right-hand side of (2.16).

Thus, if we can prove that the mapping $(u,v) \mapsto M(u,v) \cap \{x \mid |x - \bar{x}| < \delta\}$ is single-valued around $(\bar{u}, \bar{v})$, it will follow that, on some neighborhood of $(\bar{u}, \bar{v})$, this single-valued mapping is Lipschitz continuous and agrees with M_δ, as claimed. Furthermore, we will have the formula in (2.16) for the Lipschitz modulus of M_δ at $(\bar{u}, \bar{v})$, and be done.

Everything therefore hinges on establishing this single-valuedness. From [8], as already noted, we already have it for $M(\bar{u}, v)$ as a function of v around $\bar{v}$. It might seem an easy step to go from that to the local single-valuedness of $M(u, v)$ in v for parameter vectors u near $\bar{u}$, using the fact that functions f_u, like $f_{\bar{u}}$, exhibit prox-regularity locally by Proposition 3.2, together with the fact that the coderivative Hessians associated with these functions are positive definite by Proposition 4.5. At best, however, we could get from such an argument only a separate domain of single-valuedness of $M(u, v)$ in v for each u, whereas we require that these domains come together as a neighborhood of $(\bar{u}, \bar{v})$ in (u, v) jointly. That makes everything much more complicated.

Let $X \times U \times V$ be a bounded open neighborhood of $(\bar{x}, \bar{u}, \bar{v})$ small enough to ensure the properties in Proposition 3.2 (for a certain prox-regularity parameter $r \geq 0$) and also the uniform positive definiteness in Proposition 4.5. Suppose further that $U \times V$ is small enough that it lies in the neighborhood W where (5.1) holds. Fix any $s > r$ and let

$$\begin{aligned} \bar{f}(x, u) &:= \begin{cases} f(x, u) & \text{if } |x - \bar{x}| \leq \delta, \\ \infty & \text{if } |x - \bar{x}| > \delta, \end{cases} \\ k(x, u, v) &:= \bar{f}(x, u) - \langle v, x \rangle + (s/2)|x - \bar{x}|^2. \end{aligned} \tag{5.2}$$

Further, in terms of this define

$$\phi(u, v) := \inf\nolimits_x k(x, u, v), \qquad \Phi(u, v) := \operatorname{argmin}_x k(x, u, v). \tag{5.3}$$

Our first objective is to show by techniques of variational analysis that ϕ is Lipschitz continuous on a neighborhood of $(\bar{u}, \bar{v})$.

To this end we note first that when $(u, v) \in U \times V$ there exists x with $k(x, u, v) < \infty$; indeed, any $x \in M_\delta(u, v)$ has this property, since (5.1) holds and $U \times V \subset W$. Therefore $\phi < \infty$ on $U \times V$. Conversely, k is lsc and we have for each $\alpha \in \mathbb{R}$ that the set $\{(v, u, x) \mid (v, u) \in V \times U,\ k(v, u, x) \leq \alpha\}$ is bounded. This guarantees by the basic theorem on parametric optimization in [6, Thm. 1.17] that ϕ is lsc on $U \times V$ with $\phi > -\infty$ and

$$\Phi(u, v) \neq \emptyset \text{ when } (u, v) \in U \times V, \text{ where } \Phi(\bar{u}, \bar{v}) = \{\bar{x}\}. \tag{5.4}$$

Moreover, we have then from [6, Thm. 10.13] that

$$\begin{aligned} \partial\phi(u, v) &\subset \{(y, w) \mid \exists\, x \in \Phi(u, v) \text{ with } (0, y, w) \in \partial k(x, u, v)\}, \\ \partial^\infty\phi(u, v) &\subset \{(y, w) \mid \exists\, x \in \Phi(v, u) \text{ with } (0, y, w) \in \partial^\infty k(x, u, v)\}, \end{aligned} \tag{5.5}$$

where we calculate via [6, Exercise 8.8(c)] that

$$\begin{aligned} (0, y, w) \in \partial k(x, u, v) &\iff (0, y, w) \in (\partial\bar{f}(x, u), 0) + (s[x - \bar{x}] - v, 0, -x) \\ &\iff (v - s[x - \bar{x}], y) \in \partial\bar{f}(x, u) \text{ and } w = -x, \\ (0, y, w) \in \partial^\infty k(x, u, v) &\iff (0, y) \in \partial^\infty\bar{f}(x, u) \text{ and } w = 0. \end{aligned} \tag{5.6}$$

Applying the last formula to $(\bar{u}, \bar{v})$ and observing that $\partial^\infty\bar{f}(\bar{x}, \bar{u}) = \partial^\infty f(\bar{x}, \bar{u})$ because $\Phi(\bar{u}, \bar{v}) = \{\bar{x}\}$, we see through the constraint qualification $\mathcal{Q}(\bar{x}, \bar{u})$ that the only choice of (y,w) satisfying $(0,y,w) \in \partial^\infty k(\bar{x},\bar{u},\bar{v})$ is $(y,w)=(0,0)$. The second formula in (5.5) then yields $\partial^\infty \phi(\bar{u},\bar{v}) = (0,0)$. A function is Lipschitz continuous on a neighborhood of any point where it is finite, lsc, and has no nonzero horizon subgradient [6, Thm. 9.13], so we conclude that ϕ is Lipschitz continuous around $(\bar{u},\bar{v})$.

Continuity of ϕ at $(\bar{u},\bar{v})$ implies continuity of the set-valued mapping Φ at $(\bar{u},\bar{v})$, where it is single-valued; see [6, Thm. 1.17(b)]. Thus, for some open neighborhood $U_0 \times V_0$ of $(\bar{v},\bar{u})$ within $U \times V$, which can be taken to be convex, we have

$$\Phi(u,v) \subset \{x \mid |x-\bar{x}| < \delta\} \text{ when } (u,v) \in U_0 \times V_0. \tag{5.7}$$

By choosing $U_0 \times V_0$ even smaller, we can arrange to have the additional property, needed below, that

$$x \in \Phi(u,v) \implies x \in X,\ \ v - s[x-\bar{x}] \in V. \tag{5.8}$$

Under (5.7), $\partial \bar{f}(x,u)$ reduces to $\partial f(x,u)$ in (5.6), and we then obtain from (5.5) that

$$\begin{aligned}\partial\phi(u,v) \subset \{(y,-x) \mid x \in \Phi(v,u),\ (v,y) \in \partial f(x,u) + (s[x-\bar{x}],0)\}\\ \text{when } (u,v) \in U_0 \times V_0.\end{aligned} \tag{5.9}$$

The Lipschitz continuity of ϕ on $U_0 \times V_0$ implies that $\partial^\infty\phi(u,v) = \{(0,0)\}$ for $(u,v) \in U_0 \times V_0$ [6, Thm. 9.13] and allows us to apply the partial subgradient rule in [6, Cor. 10.11] to see that $\emptyset \neq \partial_v \phi(u,v) \subset \{w \mid \exists\, y \text{ with } (y,w) \in \partial\phi(u,v)\}$ and then get from (5.9) that

$$\partial_v\phi(u,v) \subset \{-x \mid x \in \Phi(u,v)\} \text{ when } (u,v) \in U_0 \times V_0. \tag{5.10}$$

Next we determine what it means for x to belong to $\Phi(u,v)$ when $(u,v) \in U_0 \times V_0$. Because of (5.7), the subgradient optimality condition for x to furnish the minimum in (5.3) takes the form of requiring $0 \in \partial_x f(x,u) - v + s[x-\bar{x}]$. Hence

$$\Phi(u,v) \subset \{x \mid v - s[x-\bar{x}] \in \partial_x f(x,u)\} \text{ when } (u,v) \in U_0 \times V_0. \tag{5.11}$$

It will be demonstrated that this makes Φ single-valued. Fix any $(u,v) \in U_0 \times V_0$ and suppose that $x, x' \in \Phi(v,u)$. In particular, we have $(x,u,v-s[x-\bar{x}])$ and $(x',u,v-s[x'-\bar{x}])$ in $X \times U \times V$ by (5.8) and therefore by prox-regularity

$$\begin{aligned} f(x',u) &\geq f(x,u) + \langle v - s[x-\bar{x}],\, x'-x\rangle - \frac{r}{2}|x'-x|^2,\\ f(x,u) &\geq f(x',u) + \langle v - s[x'-\bar{x}],\ x-x'\rangle - \frac{r}{2}|x'-x|^2,\end{aligned}$$

from which it follows (by adding the two inequalities) that $0 \geq (s-r)|x'-x|^2$. Thus $x' = x$ (inasmuch as $s > r$), and the single-valuedness of Φ is confirmed.

The single-valuedness of Φ on $U_0 \times V_0$ produces the single-valuedness of the mapping $\partial_v\phi$ on that set by (5.10) and reveals that for each $u \in U_0$ the function $\phi_u = \phi(\cdot,u)$ is strictly differentiable with respect to $v \in V_0$ [6, Thm. 9.18]—in fact with gradient $\nabla\phi_u(u,v) = -x$ for the unique $x \in \Phi(u,v)$. Strict differentiability at every point of an open set is equivalent to continuous differentiability on that set [6, Cor. 9.19].

The achievement so far can be summarized as follows in terms of ϕ and its "slices" ϕ_u. We have an open neighborhood $U_0 \times V_0$ of $(\bar{v},\bar{u})$ on which ϕ is finite and Lipschitz continuous and such that, for each $u \in U_0$, ϕ_u is continuously differentiable on V_0 with

$$\begin{aligned} -\nabla\phi_u(v) &= \text{ unique } x \in \Phi(u,v)\\ &= \text{ unique } x \text{ with } |x-\bar{x}| < \delta,\ v - s[x-\bar{x}] \in \partial f_u(x).\end{aligned} \tag{5.12}$$

In particular, $-\nabla\phi_{\bar{u}}(\bar{v}) = \bar{x}$.

Keeping u as an arbitrary element of U_0, let $F_u(v) = -\nabla\phi_u(v)$ on V_0 for simplicity. Then $F_{\bar{u}}(\bar{v}) = \bar{x}$ and F_u is a continuous, single-valued mapping from V_0 to $\mathbb{R}^n$ with its graph related to that of ∂f_u through (5.12) by

$$\begin{gathered}(v,x) \in \operatorname{gph} F_u \iff (v,x) \in \Omega,\ L(v,x) \in \operatorname{gph} f_u,\ \text{where} \\ \Omega = V_0 \times \{x \mid |x-\bar{x}| < \delta\}, \quad L(v,x) = \big(x, v - s[x-\bar{x}]\big).\end{gathered} \tag{5.13}$$

The affine mapping L is invertible and gives a "change of coordinates" through which normal cones to $\operatorname{gph} F_u$ can be identified with normal cones to $\operatorname{gph} \partial f_u$; by way of the rule in [6, Excercise 6.7] we obtain

$$(v', -x') \in N_{\operatorname{gph} F_u}(v,x) \iff (sv'-x', v') \in N_{\operatorname{gph} \partial f_u}\big(x, v - s[x-\bar{x}]\big)$$

and can write this in coderivative form as

$$v' \in D^*F_u(v\,|\,x)(x') \iff sv' - x' \in D^*(\partial f_u)(x\,|\,w)(-v') \text{ for } w = v - s[x-\bar{x}]. \tag{5.14}$$

Appealing now to the fact that the pairs (v,x) in this situation have $x \in X$ and $w \in V$ by (5.7), we make use of the uniform positive definiteness of $D^*(\partial f_u)(x\,|\,w)$ for such (x,w) (as we arranged by making our neighborhoods be such that (4.12) holds) to see from (5.14) that

$$\begin{aligned} v' \in D^*F_u(v\,|\,x)(x') &\implies \langle sv'-x', -v'\rangle \geq \epsilon|-v'|^2 \\ &\implies \langle -x', -v'\rangle \geq s|v'|^2 + \epsilon|v'|^2 \\ &\implies |x'||v'| \geq (s+\epsilon)|v'|^2 \implies |v'| \leq (s+\epsilon)^{-1}|x'|. \end{aligned}$$

This inequality on the coderivatives of F_u guarantees, in the face of the stipulated convexity of V_0, that F_u itself is Lipschitz continuous on V_0 with constant $(s+\epsilon)^{-1}$. That is an immediate outcome of the calculus of the Lipschitz modulus in [6, Thms. 9.31, 9.38, 9.40] as specialized to the case of a single-valued mapping like F_u.

We now introduce on V_0 the mapping $G_u : v \mapsto v - s[F_u(v) - \bar{x}]$, noting that $G_u(v) = \nabla_v\psi(u,v)$ for the function $\psi : (u,v) \mapsto \frac{1}{2}|v|^2 + s\phi(u,v) + s\langle v, \bar{x}\rangle$. The choice of this mapping is motivated by the fact that $w = G_u(v)$ if and only if $w = v - s[x-\bar{x}]$ for the unique x such that $|x-\bar{x}| < \delta$ and $v - s[x-\bar{x}] \in \partial f_u(x)$. Then obviously $w \in \partial f_u(x)$, so that $x \in M(u,w)$. In particular, we have $G_{\bar{u}}(\bar{v}) = \bar{v}$. If we can determine a neighborhood V_1 of $\bar{v}$ along with a neighborhood U_1 of $\bar{u}$ such that for each $(u,w) \in U_1 \times V_1$ there is a unique $v \in V_0$ with $G_u(v) = w$, we will conclude that for such (u,w) there is a unique $x \in M(u,w)$ with $|x-\bar{x}| < \delta$. That will confirm that the mapping $(u,w) \mapsto M(u,w) \cap \{x \mid |s-\bar{x}| < \delta\}$ is single-valued on $U_1 \times V_1$, and we will be finished.

Our key to this final stage will be Lemma 5.1. As preparation for using it, we demonstrate that the gradient mapping G_u is strongly monotone: for $v, v' \in V_0$ we have

$$\begin{aligned} \big\langle G_u(v') - G_u(v),\, v'-v\big\rangle &= \big\langle v' - sF_u(v') + s\bar{x} - v + sF_u(v) - s\bar{x},\, v'-v\big\rangle \\ &= |v'-v|^2 - s\big\langle F_u(v') - F_u(v),\, v'-v\big\rangle \\ &\geq |v'-v|^2 - s|F_u(v') - F_u(v)||v'-v| \\ &\geq |v'-v|^2 - s(s+\epsilon)^{-1}|v'-v|^2 = \epsilon(s+\epsilon)^{-1}|v'-v|^2. \end{aligned}$$

This monotonicity implies that $\psi(u,v)$ is μ-strongly convex in $v \in V_0$ with modulus $\mu = \epsilon(s+\epsilon)^{-1}$. Since $\psi(u,v)$ is continuous in $(u,v) \in U_0 \times V_0$ (it inherits this from ϕ), the vector $G_u(v) = \nabla_v \psi(u,v)$ depends continuously on $(u,v) \in U_0 \times V_0$ as well [16, Thm. 25.7].

Take $\lambda > 0$ small enough that $\mathbb{B}(\bar{v}, 2\lambda) \subset V_0$. Let $g_u(v) = \psi(u,v)$ if $v \in \mathbb{B}(\bar{v}, 2\lambda)$ but $g_u(v) = \infty$ otherwise. Then g_u is convex as a function on $\mathbb{R}^n$ and agrees with $\psi(u,\cdot)$ on the open set $O_u = \{v \mid |v - \bar{v}| < 2\lambda\}$. There, g_u is strongly convex with modulus μ, and its gradient mapping is G_u; the unique subgradient $w \in \partial g_u(v)$ is $w = G_u(v)$ when $v \in O_u$. By virtue of Lemma 5.1, there exists then for each $w \in \mathbb{B}(G_u(\bar{v}), \lambda\mu)$ a unique $v \in \mathbb{B}(\bar{v}, \lambda)$ with $w = G_u(v)$.

All that remains is to observe that by choosing U_1 to be a small enough neighborhood of $\bar{u}$ within U_0 we can obtain (through the continuous dependence of $G_u(\bar{v})$ on u) the existence of a neighborhood V_1 of $\bar{v}$ within V_0 such that, for all $u \in U_1$, we have $\mathbb{B}(G_u(\bar{v}), \lambda\mu) \supset V_1$. □

In moving on to the necessity in Theorem 2.3, we will need help from a different auxiliary result.

LEMMA 5.2 (dual criterion for localized strong convexity). *Let $h : \mathbb{R}^n \to \mathbb{R}$ be a proper, lsc, convex function whose conjugate h^* is differentiable on a certain open convex set $O \subset \mathbb{R}^n$—moreover, with its gradient mapping $\nabla h^* : O \to \mathbb{R}^n$ Lipschitz continuous on O with constant $1/\sigma$ (for some $\sigma > 0$). Let $\lambda > 0$ and $O_\lambda = \{v \mid \mathbb{B}(v,\lambda) \subset O\}$. Then*

$$(5.15) \qquad h(x') \geq h(x) + \langle v,\, x' - x\rangle + \frac{\sigma}{2}|x' - x|^2 \text{ if } v \in \partial h(x) \cap O_\lambda,\ |x' - x| \leq \frac{\lambda}{\sigma},$$

and therefore also

$$(5.16) \qquad \langle x' - x,\, v' - v\rangle \geq \sigma |x' - x|^2 \text{ whenever } \begin{cases} v \in \partial h(x),\ v' \in \partial h(x'), \\ v, v' \in O_\lambda,\ |x' - x| \leq \lambda/\sigma. \end{cases}$$

Proof. For any $v, v' \in O$ we have $h^*(v') - h^*(v) = \int_0^1 \langle \nabla h^*(v + t[v' - v]),\, v' - v\rangle dt$. The estimate $\langle \nabla h^*(v + t[v' - v]),\, v' - v\rangle \leq \langle \nabla h^*(v),\, v' - v\rangle + (t/\sigma)|v' - v|^2$ holds under our assumptions, so the integral gives us

$$h^*(v') - h^*(v) \;\leq\; \langle \nabla h^*(v),\, v' - v\rangle + \frac{1}{2\sigma}|v' - v|^2.$$

Therefore, in terms of the indicator function $\delta_{\lambda\mathbb{B}}$ of the closed λ-ball around 0 and the function $j(w) = \frac{1}{2}|w|^2$, we have for any choice of $v \in O_\lambda$ that

$$(5.17) \qquad \begin{aligned} &h^*(v') \;\leq\; k(v' - v) \text{ for all } v' \in \mathbb{R}^n, \text{ where} \\ &\qquad\qquad k(w) := h^*(v) + \langle \nabla h^*(v), w\rangle + \sigma^{-1} j(w) + \delta_{\lambda\mathbb{B}}(w). \end{aligned}$$

Fix $v \in O_\lambda$ and take conjugates of both sides of (5.17) as convex functions of v', using x' as the variable to describe the conjugate functions. That produces the inequality

$$(5.18) \qquad h^{**}(x') \;\geq\; k^*(x') + \langle v, x'\rangle \text{ for all } x' \in \mathbb{R}^n.$$

Here $h^{**} = h$ because h is lsc, proper, and convex, and k^* calculates to

$$k^*(x') = -h^*(v) + \big(\sigma^{-1} j + \delta_{\lambda\mathbb{B}}\big)^*\big(x' - \nabla h^*(v)\big).$$

The function conjugate to $\sigma^{-1}j$ is σj and the function conjugate to $\delta_{\lambda\mathbb{B}}$ is $\lambda|\cdot|$, and consequently $\big(\sigma^{-1}j + \delta_{\lambda\mathbb{B}}\big)^* = \sigma j \# \lambda|\cdot|$, with "$\#$" denoting the operation of epi-addition (inf-convolution):

(5.19)
$$\big(\sigma j \# \lambda|\cdot|\big)(u) = \inf_{u'}\{\sigma j(u') + \lambda|u-u'|\} = \begin{cases}\sigma j(u) & \text{when } |u| \le \lambda/\sigma,\\ \lambda(\sigma^{-1} + |u|) & \text{when } |u| \ge \lambda/\sigma.\end{cases}$$

Let $x = \nabla h^*(v)$; this relation is the same as $x \in \partial h^*(v)$ when h^* is differentiable at v, and hence is equivalent also to $v \in \partial h(x)$ as well as to $h(x) + h^*(v) = \langle x, v\rangle$ (by convex analysis; cf. [6, Prop. 11.3]). We obtain from (5.18) and our calculations that

$$h(x') \;\ge\; h(x) + \langle v,\, x'-x\rangle + \big(\sigma j \# \lambda|\cdot|\big)(x'-x) \text{ for all } x' \in \mathbb{R}^n.$$

This yields (5.15) through (5.19). By symmetry, of course, we also have

$$h(x) \ge h(x') + \langle v',\, x-x'\rangle + \frac{\sigma}{2}|x-x'|^2 \text{ if } v' \in \partial h(x') \cap O_\lambda,\ |x-x'| \le \frac{\lambda}{\sigma}.$$

In combining this inequality with the one in (5.15) we obtain (5.16). □

Proof of necessity in Theorem 2.3. The hypothesis furnishes for us a neighborhood $X \times U \times V$ of $(\bar{x}, \bar{u}, \bar{v})$ for which the properties in Proposition 3.2 hold. An additional assumption now is that, for some $\delta > 0$ sufficiently small, the mapping M_δ is single-valued and Lipschitz continuous around $(\bar{u}, \bar{v})$, its value at $(\bar{u}, \bar{v})$ being $\bar{x}$. Without loss of generality, we can suppose these properties hold for M_δ on $U \times V$ and that

(5.20)
$$M_\delta(u,v) \in \{x \mid |x-\bar{x}| < \delta\} \subset X \text{ for } (u,v) \in U \times V.$$

We can also arrange that (5.1) holds for $W = U \times V$, through Proposition 3.5 and (2.7).

Define $\bar{f}$, k, ϕ, and Φ as in (5.2) and (5.3) but with $s = 0$, so $\phi = m_\delta$ and $\Phi = M_\delta$. The subgradient calculus used in the proof of sufficiency after those definitions remains valid and reveals that ϕ, which is Lipschitz continuous on an open convex neighborhood $U_0 \times V_0$, say, of $(\bar{u}, \bar{v})$ in $U \times V$, exhibits as instances of (5.10) and (5.11) the relations

(5.21)
$$\begin{aligned}&\partial_v \phi(u,v) = -M_\delta(u,v),\\ &M_\delta(u,v) \in \{x \mid v \in \partial_x f(x,u)\} = M(u,v).\end{aligned}$$

The first of these implies, moreover, that for each $u \in U_0$ the function $\phi_u = \phi(u,\cdot)$ is continuously differentiable on V_0 with gradient $\nabla\phi_u(v) = -M_\delta(u,v)$. In fact, our Lipschitz assumption on M_δ gives us a constant $\kappa > 0$ such that for each $u \in U_0$ the mapping $\nabla\phi_u$ is Lipschitz continuous on V_0 with constant κ.

Let $g_u = -\phi_u$ so that $g_u(v) = \sup_x\{\langle v, x\rangle - \bar{f}(x,u)\}$, or in other words, g_u is conjugate to $\bar{f}_u$ under the Legendre–Fenchel transform. In particular, g_u is a proper, lsc, convex function on $\mathbb{R}^n$ that is differentiable on V_0 with $\nabla g_u(v) = M_\delta(u,v)$. Let h_u be conjugate in turn to g_u. Then $h_u = g_u^* = \bar{f}_u^{**}$ and $g_u = h_u^* = \bar{f}_u^*$, and we have by the usual relation between subgradients of conjugate convex functions that $v \in \partial h_u(x)$ if and only if $x \in \partial g_u(v)$, so that

(5.22)
$$v \in \partial h_u(x) \iff x = \nabla g_u(v) = M_\delta(u,v), \text{ as long as } u \in U_0,\ v \in V_0.$$

We now apply Lemma 5.2 to h_u and its conjugate function g_u on the set $O = V_0$ with $1/\sigma = \kappa$. Let $\lambda > 0$ be small enough that $\mathbb{B}(\bar{v}, \lambda) \subset V_0$, so the set $O_\lambda = \{v \mid \mathbb{B}(\bar{v}, \lambda) \subset V_0\}$ is an open neighborhood of $\bar{v}$. Then (5.16) holds for h_u, where by (5.24) the relations $v \in \partial h(x)$ and $v' \in \partial h(x')$ can be written as $x = M_\delta(u, v)$ and $x' = M_\delta(u, v')$.

Choose X_1 to be a neighborhood of $\bar{x}$ within X so small that $|x' - x| \le \lambda/\sigma$ when $x, x' \in X_1$. Let $U_1 \times V_1$ be a neighborhood of $(\bar{u}, \bar{v})$ within $U_0 \times O_\lambda$ small enough that $(u, v) \in U_1 \times V_1$ implies $M_\delta(u, v) \in X_1$. Then (5.16) yields the inequality

$$\langle x' - x,\, v' - v\rangle \ge \sigma|x' - x|^2 \text{ when } \begin{cases} x = M_\delta(u, v),\ x' = M_\delta(u, v'), \\ u \in U_1,\ v, v' \in V_1 \end{cases}. \tag{5.23}$$

In terms of the mapping T_u obtained by restricting $M_\delta(u, \cdot)$ to V_1, (5.23) says that T_u^{-1} is strongly monotone with constant σ. Let S_u be the mapping whose graph is the intersection of $\operatorname{gph} M(u, \cdot)$ with $V_1 \times \{x \mid |x - \bar{x}| < \delta\}$, so that S_u^{-1} is the mapping whose graph is the intersection of $\operatorname{gph} \partial f_u$ with $\{x \mid |x - \bar{x}| < \delta\} \times V_1$. We have $\operatorname{gph} T_u \subset \operatorname{gph} S_u$ by (5.20) and the second relation in (5.21); hence also $T_u^{-1}(x) \subset S_u^{-1}(x) \subset \partial f_u(x)$ for all x.

For the constant r in the prox-regularity of f, we know that the mappings ∂f_u are monotone when $u \in U_1$. Let $s > r$ and consider the mappings $T_u^{-1} + sI$ and $S_u^{-1} + sI$. As long as $u \in U_1$, both of these are strongly monotone, the first with constant $\sigma + s$ and the second surely with constant $s - r$. Hence, the inverses $(T_u^{-1} + sI)^{-1}$ and $(S_u^{-1} + sI)^{-1}$ are single-valued on their domains. Because $\operatorname{gph} T_u^{-1} \subset \operatorname{gph} S_u^{-1}$, we have $\operatorname{gph}(T_u^{-1} + sI)^{-1} \subset \operatorname{gph}(S_u^{-1} + sI)^{-1}$, so it follows that

$$x \in (T_u^{-1} + sI)^{-1}(z) \implies (T_u^{-1} + sI)^{-1}(z) = (S_u^{-1} + sI)^{-1}(z) = \{x\}. \tag{5.24}$$

Expressing z in the form $v + sx$, we find that this means

$$x \in (T_u^{-1} + sI)^{-1}(v + sx) \iff v + sx \in (T_u^{-1} + sI)(x) \iff x \in T_u(v),$$

and similarly with S_u substituted for T_u. Thus, (5.24) asserts that whenever $x \in T_u(v)$ we have $T_u(v) = S_u(v) = \{x\}$. This has been established for arbitrary $u \in U_1$, so in recalling the definitions of T_u and S_u we are able to conclude that

$$M(u, v) \cap \{x \mid |x - \bar{x}| < \delta\} = M_\delta(u, v) \text{ for all } (u, v) \in U_1 \times V_1. \tag{5.25}$$

This localization of M therefore inherits the Lipschitz continuity of M_δ. Hence, in particular, the Aubin property holds for M at $(\bar{u}, \bar{v})$ for $\bar{x}$. That implies by Proposition 4.1 that condition (b) of Theorem 2.3 must hold. Furthermore, in terms of the inverse mappings, (5.25) states that $\operatorname{gph} T_u = \big[\{x \mid |x - \bar{x}| < \delta\} \times V_1\big] \cap \operatorname{gph} \partial f_u$ when $u \in U_1$. This reveals that the coderivatives of these truncated mappings must coincide: $DT_u(x \,|\, v) = D^*(\partial f_u)(x \,|\, v) = \partial^2 f_u(x \,|\, v)$ at the common graph points (x, v). Because T_u is strongly monotone with constant σ we have $\langle x', v'\rangle \ge \sigma|v'|^2$ for $x' \in DT_u(x \,|\, v)(v')$; hence likewise

$$\langle x', v'\rangle \ge \sigma|v'|^2 \text{ for } x' \in \partial^2 f_u(x \,|\, v)(v') \text{ when } v \in \partial f_u(x) = \partial_x f(x, u),$$

provided that $(u, v) \in U_1 \times V_1$. That guarantees, through the converse part of Proposition 4.5 that the positive definiteness condition (a) holds in Theorem 2.3. □

Proof of Theorem 2.7. This is really just an extension of the proof of necessity in Theorem 2.3. That proof utilized the function $\bar{f}$ in (5.2) and, in terms of $\bar{f}_u = \bar{f}(\cdot, u)$, introduced the conjugate functions $g_u = \bar{f}^* = -m_\delta(u,\cdot)$ and $h_u = g_u^* = \bar{f}_u^{**}$. The conjugacy relations imply in turn that $h_u^* = g_u$ and also that $h_u(x) = \infty$ when $|x - \bar{x}| > \delta$, since $\bar{f}_u(x)$ has this property by definition. Hence

$$g_u(v) = \sup_{x \in \mathbb{R}^n} \{\langle v, x\rangle - h_u(x)\} = \sup_{|x-\bar{x}| \le \delta} \{\langle v, x\rangle - h_u(x)\}, \tag{5.26}$$

where the maximum is attained at x if and only if $v \in \partial h_u(x)$.

Take $\widehat{f}(x,u) = h_u(x)$. Then $\partial_x \widehat{f}(x,u) = \partial h_u(x)$, and since $g_u(v) = -m_\delta(u,v)$ the conjugacy formula $h_u(x) = \sup_v\{\langle v, x\rangle - g_u(x)\}$ converts to

$$\widehat{f}(x,u) = \sup_v\{\langle v, x\rangle + m_\delta(u,v)\}, \tag{5.27}$$

while from (5.26) we get

$$\begin{aligned} \inf_{|x-\bar{x}| \le \delta} \{\widehat{f}(x,u) - \langle v, x\rangle\} &= m_\delta(u,v), \\ \operatorname*{argmin}_{|x-\bar{x}| \le \delta} \{\widehat{f}(x,u) - \langle v, x\rangle\} &= \{x \mid v \in \partial_x \widehat{f}(x,u)\}. \end{aligned} \tag{5.28}$$

For the problems $\widehat{\mathcal{P}}(u,v)$, the expressions on the left-hand side of (5.28) are $\widehat{m}_\delta(u,v)$ and $\widehat{M_\delta}(u,v)$. Conversely, according to (5.22), the right-hand side of the second equation in (5.28) gives $M_\delta(u,v)$ when (u,v) lies in a certain neighborhood $U_0 \times V_0$ of $(\bar{u}, \bar{v})$. Therefore, $\widehat{m}_\delta(u,v)$ and $\widehat{M_\delta}(u,v)$ agree with $m_\delta(u,v)$ and $M_\delta(u,v)$ around $(\bar{u}, \bar{v})$; thus, $\widehat{f}$ is equivalent to f in the sense described in Theorem 2.7.

Furthermore, $\widehat{f}$ is lsc on $\mathbb{R}^n \times \mathbb{R}^d$; by (5.27) because m_δ is lsc on $\mathbb{R}^d \times \mathbb{R}^n$; which is true because m_δ is a special case of the function ϕ defined in (5.3) through (5.2) (namely for $s = 0$), and ϕ was shown to be lsc in the argument leading up to (5.4). In addition we have, for $(x,u,v) \in \operatorname{gph} \partial_x \widehat{f}$ with $(u,v) \in U_0 \times V_0$, that

$$\widehat{f}(x,u) - \langle v, x\rangle = m_\delta(u,v) = f(x,u) - \langle v, x\rangle$$

and consequently $\widehat{f}(x,u) = f(x,u) = m_\delta(u,v) + \langle v, x\rangle$, an expression that is continuous with respect to the elements (x,u,v) in question. The convexity of $\widehat{f}(x,u)$ in x combined with that continuity makes $\widehat{f}$ continuously prox-regular at $(\bar{x}, \bar{u})$ for $\bar{v}$. (Convexity allows the constant r in the definition of prox-regularity to be taken to be 0.) □

REFERENCES

[1] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

[2] D. Klatte and B. Kummer, *Strong stability in nonlinear programming revisited*, J. Austral. Math. Soc. Ser. B, 40 (1999), pp. 336–352.

[3] A. Dontchev and R. T. Rockafellar, *Characterizations of Lipschitzian stability in nonlinear programming*, in Mathematical Programming with Data Perturbations, Lecture Notes in Pure and Appl. Math. 195, A. V. Fiacco, ed., Marcel Dekker, New York, 1997, pp. 65–82.

[4] J.-P. Aubin, *Lipschitz behavior of solutions to convex minimization problems*, Math. Oper. Res., 9 (1984), pp. 87–111.

[5] B. S. Mordukhovich, *Sensitivity analysis in nonsmooth optimization*, in Theoretical Aspects of Industrial Design, Proceedings of the SIAM Regional Conference on Industrial Theory, April 25–26, 1990, Wright–Patterson Air Force Base, OH, D. A. Field and V. Komkov, eds., SIAM, Philadelphia, 1992, pp. 32–46.

[6] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Grundlehren Math. Wiss. 317, Springer-Verlag, New York, 1998.

[7] A. Dontchev and R. T. Rockafellar, *Characterizations of strong regularity for variational inequalities over polyhedral convex sets*, SIAM J. Optim., 6 (1996), pp. 1087–1105.

[8] R. A. Poliquin and R. T. Rockafellar, *Tilt stability of a local minimum*, SIAM J. Optim., 8 (1998), pp. 287–299.

[9] R. A. Poliquin and R. T. Rockafellar, *Prox-regular functions in variational analysis*, Trans. Amer. Math. Soc., 348 (1996), pp. 1805–1838.

[10] R. A. Poliquin and R. T. Rockafellar, *Amenable functions in optimization*, in Nonsmooth Optimization Methods and Applications, F. Giannessi, ed., Gordon and Breach, Philadelphia, 1992, pp. 338–353.

[11] A. B. Levy and R. T. Rockafellar, *Variational conditions and the proto-differentiation of partial subgradient mappings*, Nonlin. Anal., 26 (1996), pp. 1951–1964.

[12] B. S. Mordukhovich, *Complete characterization of openness, metric regularity, and Lipschitzian properties of multifunctions*, Trans. Amer. Math. Soc., 340 (1993), pp. 1–36.

[13] B. S. Mordukhovich, *Lipschitzian stability of constraint systems and generalized equations*, Nonlinear Anal., 22 (1994), pp. 173–206.

[14] B. S. Mordukhovich, *Generalized differential calculus for nonsmooth and set-valued mappings*, J. Math. Anal. Appl., 183 (1994), pp. 250–288.

[15] R. T. Rockafellar and D. Zagrodny, *A derivative-coderivative inclusion in second-order nonsmooth analysis*, Set-Valued Anal., 5 (1997), pp. 89–105.

[16] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

SIAM J. OPTIM.
Vol. 10, No. 2, pp. 605–625

A TRULY GLOBALLY CONVERGENT NEWTON-TYPE METHOD FOR THE MONOTONE NONLINEAR COMPLEMENTARITY PROBLEM[*]

M. V. SOLODOV[†] AND B. F. SVAITER[†]

Abstract. The Josephy–Newton method for solving a nonlinear complementarity problem consists of solving, possibly inexactly, a sequence of linear complementarity problems. Under appropriate regularity assumptions, this method is known to be locally (superlinearly) convergent. To enlarge the domain of convergence of the Newton method, some globalization strategy based on a chosen merit function is typically used. However, to ensure global convergence to a solution, some additional restrictive assumptions are needed. These assumptions imply boundedness of level sets of the merit function and often even (global) uniqueness of the solution. We present a new globalization strategy for monotone problems which is not based on any merit function. Our linesearch procedure utilizes the regularized Newton direction and the monotonicity structure of the problem to force global convergence by means of a (computationally explicit) projection step which reduces the distance to the solution set of the problem. The resulting algorithm is truly globally convergent in the sense that the subproblems are always solvable, and the whole sequence of iterates converges to a solution of the problem without any regularity assumptions. In fact, the solution set can even be unbounded. Each iteration of the new method has the same order of computational cost as an iteration of the damped Newton method. Under natural assumptions, the local superlinear rate of convergence is also achieved.

Key words. nonlinear complementarity problem, Newton method, proximal point method, projection method, global convergence, superlinear convergence

AMS subject classifications. 90C30, 90C33

PII. S1052623498337546

1. Introduction.

The classical nonlinear complementarity problem [28, 7], NCP(F), is to find a point $x \in \Re^n$ such that

$$x \geq 0, \quad F(x) \geq 0, \quad \langle x, F(x) \rangle = 0, \tag{1.1}$$

where $F : \Re^n \to \Re^n$ and $\langle \cdot, \cdot \rangle$ denotes the usual inner product in $\Re^n$. Throughout this paper, we shall assume that $F(\cdot)$ is continuous and monotone, i.e.,

$$\langle F(x) - F(y), x - y \rangle \geq 0 \quad \text{for all} \quad x, y \in \Re^n.$$

Note that under this assumption, the solution set of (1.1) is convex.

While there exists a wide range of approaches to solving NCP(F) (see [14, 28, 7]), some of the most successful and widely used are Newton-type algorithms based on solving successive linearizations of the problem (see, for example, the more detailed discussion in [8]). Given a point x^k, the (Josephy–) Newton method [17, 30, 26, 14, 1] generates the next iterate x^{k+1} by solving the linear complementarity problem

$$x \geq 0, \quad F_k(x) \geq 0, \quad \langle x, F_k(x) \rangle = 0, \tag{1.2}$$

[*] Received by the editors April 16, 1998; accepted for publication (in revised form) June 1, 1999; published electronically February 24, 2000.

http://www.siam.org/journals/siopt/10-2/33754.html

[†] Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, RJ 22460-320, Brazil (solodov@impa.br, benar@impa.br). The research of the first author was supported by CNPq grant 300734/95-6 and by PRONEX–Optimization. The research of the second author was supported by CNPq grant 301200/93-9(RN) and by PRONEX–Optimization.

where $F_k(\cdot)$ is the first-order approximation of $F(\cdot)$ at x^k,

$$F_k(x) := F(x^k) + \nabla F(x^k)(x - x^k), \tag{1.3}$$

assuming $F(\cdot)$ is differentiable. If the starting point is sufficiently close to some *regular* [35] solution $\bar{x}$ of (1.1), the sequence generated by the Newton method is well defined and converges to $\bar{x}$ superlinearly or quadratically, depending on further assumptions. We note, in passing, that monotonicity of F is not needed for such local analysis.

There are two key difficulties with using the Newton method given by (1.2), (1.3) for solving NCP(F). First, in the absence of regularity even local convergence cannot be ensured. Second, even if regularity holds at a solution, there are serious problems with ensuring global convergence. In particular, far from a regular solution of the problem, there is no guarantee that the linearization subproblems are solvable. And even if the subproblem solution exists, there is no guarantee that it actually constitutes some progress toward solving the original problem, NCP(F). In this paper, we shall successfully address each of these difficulties in the context of monotone problems.

To enlarge the domain of convergence of the Newton method, some globalization strategy has to be used. However, as pointed out in [14, p. 182], "the global convergence remains a rare property for most of the modified methods." As mentioned in [14, p. 185], "the trouble with general variational inequalities and nonlinear complementarity problems is that valid merit functions which are relatively easy to compute are very difficult if not impossible to find." Although considerable progress has been made in the theory and numerical use of merit functions in recent years [11, 23, 10, 19, 6, 44, 22, 24, 45, 46] (see [12] for a survey of merit functions for variational inequality and complementarity problems), most of the known merit functions do not appear to be useful for the specific task of globalizing the Newton method given by (1.2), (1.3). At this point, it is worth emphasizing that the Newton method under consideration should not be confused with other Newton-like methods which are not of the Josephy type—for example, Newton methods for minimizing a particular merit function or Newton methods for equation-based NCP reformulations [18, 34, 21, 4, 6, 15, 49, 53, 16, 20, 13, 48].

Perhaps the most natural globalization strategy for (1.2), (1.3) is a linesearch procedure in the obtained Newton direction (if it exists!) aimed at decreasing the value of some valid merit function. In this regard, we note the following. First, many of the known NCP merit functions are not differentiable, e.g., those based on the natural residual [26] and the normal map [36]. This makes linesearch difficult, although an alternative pathsearch approach is possible [34, 3]. Second, for some differentiable merit functions, e.g., (the square of) the Fischer–Burmeister function [9, 19, 6], it appears that the Newton direction need not be a direction of descent, particularly far from a solution or in the absence of regularity. Thus the use of this function results in a rather indirect globalization of the Newton method [8]. (More on this later.)

To our knowledge, the only merit functions which have been used to globalize the Newton method of the form (1.2), (1.3) are the gap function [25], the regularized gap function [51, 50], the D-gap function [31, 32], and (the square of) the Fischer–Burmeister function [8]. (We note that some of these methods were developed for the more general mixed complementarity or variational inequality setting.) However, each of these globalizations has certain drawbacks. Using the gap function [25] requires exact minimization along the line to compute the stepsize, which is not implementable. Moreover, the gap function itself is not easy to compute in general.

In addition, compactness of the feasible set of the variational inequality problem is required, which precludes an application to complementarity problems. Using the regularized gap function [51, 50] admits inexact Armijo-type linesearch but requires strong monotonicity of F for global convergence (see also [52]). In addition, methods of [25, 51, 50] also need the (restrictive) strict complementarity assumption to establish superlinear/quadratic rate of convergence. Note that the subproblems in [25, 51, 50] are solvable due to the compactness of the feasible set and the strong monotonicity of F, respectively.

The methods based on the D-gap function [31, 32] and the Fischer–Burmeister function [8] globalize the Newton method in a rather indirect way. In [31, 32, 8] there is no guarantee that the subproblems are solvable. If the Newton direction does not exist (which cannot be checked a priori, so some wasteful computing will inevitably be done), the method resorts to an "escape" mechanism of taking a gradient descent step for the merit function. Even if the Newton direction exists, it still may happen that it does not satisfy conditions needed to obtain sufficient descent of the merit function by means of a linesearch procedure. In this case, the Newton point will be discarded altogether, and again a gradient descent step will be taken. In a sense, this is a rather indirect globalization of the Newton method, because such a strategy does not correspond to the damped Newton methodology. As for the convergence results in [31, 32, 8], every accumulation point of the generated sequence of iterates is a stationary point of the employed merit function (even without the monotonicity assumption on F). However, the existence of such accumulation points, and the equivalence of stationary points of the merit functions to solutions of NCP(F), cannot be guaranteed without further assumptions. For example, in [31, 32], F is assumed to be a uniform P-function, which implies that NCP(F) has a (globally) unique solution.

Finally, we briefly comment on the interesting regularization approach proposed in [5], which converges globally when the solution set of the NCP is compact and F is a P_0-function. First, it is important to note that the subproblems in the method of [5] are *nonlinear* complementarity problems, which are structurally just as difficult to solve as the original problem itself (although they are better behaved due to regularization). Therefore, this method is not of the Newton type. Second, the global convergence result of [5] states that the sequence of iterates remains bounded, and its every accumulation point solves the NCP. This is weaker than the full convergence of the whole sequence which we shall establish for our method. On the other hand, the P_0 assumption on F is, of course, weaker than our assumption of monotonicity. It should also be mentioned that it is often unknown whether the solution set is compact. When F is monotone, this holds if the NCP is strictly feasible, i.e., there exists an x such that $x > 0$ and $F(x) > 0$. However, this condition is not easy to verify in general. In any case, the key conceptual difference between our method and that of [5] is in the linear versus nonlinear structure of subproblems solved at each iteration. It is worth noting that the latter is also the important difference between our method and the proximal point algorithm [37], which does converge globally under the monotonicity assumption only.

We emphasize that for each of the cited Josephy–Newton algorithms, to ensure global convergence of the whole sequence of iterates to a solution of the problem, one needs assumptions which, among other things, imply that the solution is unique. In fact, relatively restrictive assumptions are required even to prove boundedness of the iterates and convergence to zero of the distance to the solution set.

In this paper, we present a new globalization strategy for monotone problems which overcomes the above mentioned drawbacks. In particular, our algorithm is *truly globally convergent* in the sense that from any starting point the whole sequence of iterates converges to a solution of NCP(F) under no assumptions other than monotonicity and continuity of F (and of course, the existence of a solution); see Theorem 3.2. In particular, no regularity-type assumptions are needed. In fact, the solution set may even be unbounded. In addition, the linear complementarity subproblems are always solvable, and our algorithm allows for their inexact solution similar to the setting of [26]. This feature of approximate subproblem solutions is of particular importance for large-scale problems. (We note that the effect of an inexact subproblem solution has not been analyzed in globalizations proposed in [51, 50, 31, 32, 8].) Under the assumptions of positive definiteness of the Jacobian of F at the solution, and its Hölder continuity in some neighborhood of it, the local superlinear rate of convergence is also established (see Theorem 4.3). Each iteration of our algorithm (see Algorithm 2.2) consists of an approximate solution of a (strongly monotone) linear complementarity problem, followed by a linesearch procedure and a (computationally trivial) projection step. Thus computational cost of each iteration is of the same order as that of the damped Newton method.

2. The algorithm. We start with some equivalent formulations of NCP(F), each of which will be useful in the subsequent analysis. In particular, the following five statements are equivalent:

1. $\bar{x}$ solves NCP(F).
2. $\bar{x}$ is a solution of the variational inequality problem over the nonnegative orthant $\Re^n_+$:

$$\bar{x} \in \Re^n_+, \quad \langle F(\bar{x}), x - \bar{x} \rangle \geq 0 \quad \text{for all} \quad x \in \Re^n_+.$$

3. $\bar{x}$ yields a zero of the (maximal monotone) operator $F + N$:

$$0 \in F(\bar{x}) + N(\bar{x}),$$

   where $N(x)$ is the normal cone to $\Re^n_+$ at the point x.
4. $\bar{x}$ is a zero of the natural (projection) residual:

$$0 = r(\bar{x}) := \min\{\bar{x}; F(\bar{x})\} = \bar{x} - [\bar{x} - F(\bar{x})]^+,$$

   where the minimum is taken componentwise and $[\cdot]^+$ stands for the orthogonal projection map onto $\Re^n_+$.

The approach presented in this paper is in some ways motivated by the hybrid projection–proximal point method of [43] and the projection method of [47], which already proved to be useful for developing globally (and locally superlinearly) convergent Newton methods for systems of monotone equations [40]. In fact, in a sufficiently small neighborhood of a regular solution of NCP(F) our Newton-type Algorithm 2.2 takes steps which can be viewed, in a certain sense, as iterations of the hybrid projection–proximal point method. This will prove to be the key to the local superlinear rate of convergence. We therefore first state the algorithm of [43] in the more general context of finding zeros of set-valued maximal monotone operators in a Hilbert space. Let T be a maximal monotone operator on a real Hilbert space $\mathcal{H}$. And consider, for a moment, the problem of finding an $x \in \mathcal{H}$ such that $0 \in T(x)$. Note that NCP(F) considered here is a particular instance of this problem with $T(x) = (F + N)(x)$ and $\mathcal{H} = \Re^n$.

Algorithm 2.1 (hybrid projection–proximal point method [43]). *Choose any* $x^0 \in \mathcal{H}$ *and* $\sigma \in [0, 1)$*; set* $k := 0$.

Inexact Proximal Step. *Choose* $\mu_k > 0$ *and find* $y^k \in \mathcal{H}$ *and* $v^k \in T(y^k)$ *such that*

$$0 = v^k + \mu_k(y^k - x^k) + \varepsilon^k, \tag{2.1}$$

where

$$\|\varepsilon^k\| \le \sigma \max\{\|v^k\|, \mu_k\|y^k - x^k\|\}. \tag{2.2}$$

Stop if $v^k = 0$ *or* $y^k = x^k$*. Otherwise,*

Projection Step. *Compute*

$$x^{k+1} = x^k - \frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|^2} v^k.$$

Set $k := k + 1$*; and repeat.*

If problem $0 \in T(x)$ has a solution and the sequence $\{\mu_k\}$ is bounded above, then the generated sequence $\{x^k\}$ either is finite and terminates at a solution or is infinite and converges (weakly) to a solution. The linear/superlinear rate of convergence is also achieved under standard assumptions. For complete properties of the method, see [43]. The idea of Algorithm 2.1 is to use an approximate proximal iteration to construct a hyperplane

$$H_k := \{x \mid \langle v^k, x - y^k\rangle = 0\}, \tag{2.3}$$

which separates the current iterate x^k from the solutions of $0 \in T(x)$. Let us make this more precise. If $0 \in T(\bar{x})$, then, by monotonicity of T,

$$\langle v^k, \bar{x} - y^k\rangle \le 0$$

for any y^k and any $v^k \in T(y^k)$. Here we will consider a condition somewhat stronger than (2.2), namely,

$$\|\varepsilon^k\| \le \sigma\mu_k\|y^k - x^k\|, \tag{2.4}$$

because this will be the only condition used in our Algorithm 2.2. Using (2.1) and the Cauchy–Schwarz inequality, we obtain

$$\begin{aligned} \langle v^k, x^k - y^k\rangle &= \mu_k\|x^k - y^k\|^2 - \langle \varepsilon^k, x^k - y^k\rangle \\ &\ge \mu_k\|x^k - y^k\|^2 - \|\varepsilon^k\|\|x^k - y^k\| \\ &\ge \mu_k(1 - \sigma)\|x^k - y^k\|^2 \\ &> 0, \qquad (2.5) \end{aligned}$$

where the last inequality follows from (2.4). Thus whenever (2.4) holds, we have (2.5), and so H_k given by (2.3) indeed separates x^k from zeros of T. The last step of Algorithm 2.1 is equivalent to projecting x^k onto this hyperplane. Separation arguments show that the distance to the solution set for a sequence thus constructed monotonically decreases, which essentially ensures global convergence of the algorithm. Algorithm 2.1 has certain advantages over the classical proximal point method [37] in the sense of less restrictive and more constructive tolerance requirements imposed on approximate solutions of proximal subproblems. See [43] for a more detailed discussion. Other related works are [39, 41, 38, 42].

Of course, in the context of this paper straightforward application of Algorithm 2.1 to solving NCP(F) is not practical, since this would involve solving a sequence of *nonlinear* subproblems which are in general just as difficult as the original NCP, even if they are better behaved due to the regularization. In this sense, the situation is similar to the method of [5] discussed in the introduction. However, the regularization and projection methodology of Algorithm 2.1 would prove to be crucial for devising a globally convergent method.

Given a current iterate x^k and a regularization parameter $\mu_k > 0$, consider the regularized linear complementarity problem LCP(φ_k)

$$x \geq 0, \quad \varphi_k(x) \geq 0, \quad \langle x, \varphi_k(x)\rangle = 0, \tag{2.6}$$

where

$$\varphi_k(x) := F(x^k) + G_k(x - x^k) + \mu_k(x - x^k) \tag{2.7}$$

with G_k being a positive semidefinite matrix (presumably, the Jacobian of F or its approximation, if F is differentiable at x^k). Suppose $z^k \geq 0$ is some approximate solution of this problem with e^k being the associated natural residual [2, 26]:

$$\min\{z^k; \varphi_k(z^k)\} = e^k, \tag{2.8}$$

where the minimum is taken componentwise (see (2.9), (2.10) in Algorithm 2.2 for conditions imposed on the error tolerance e^k). Note that since the matrix $G_k + \mu_k I$ is positive definite, LCP(φ_k) always has (unique) solution $\hat{z}^k$ [2] for which the residual is zero:

$$0 = \min\{\hat{z}^k; \varphi_k(\hat{z}^k)\}.$$

Hence, LCP(φ_k) always has inexact solutions such that $\|e^k\| \leq \delta_k$, whichever $\delta_k \geq 0$ we choose. Therefore this inexact Newton step is well defined.

The next step of our algorithm is checking whether the inexact Newton point obtained by solving LCP(φ_k) provides an acceptable (in the sense of Algorithm 2.1) approximate solution for the proximal point subproblem

$$0 \in (F + N)(x) + \mu_k(x - x^k).$$

If this is the case, the previously described separation property holds, and our algorithm proceeds to make the projection step prescribed by Algorithm 2.1.

To make this more precise, we provide the following considerations. Since

$$e^k = \min\{z^k; \varphi_k(z^k)\} = z^k - [z^k - \varphi_k(z^k)]^+,$$

we have that

$$z^k - e^k = [z^k - \varphi_k(z^k)]^+ \in \Re^n_+.$$

By properties of the projection operator [33, p. 121],

$$\langle z^k - \varphi_k(z^k) - (z^k - e^k), x - (z^k - e^k)\rangle \leq 0 \quad \text{for all } x \in \Re^n_+.$$

Therefore

$$\langle -\varphi_k(z^k) + e^k, x - (z^k - e^k)\rangle \leq 0 \quad \text{for all } x \in \Re^n_+,$$

which implies that

$$h^k := -\varphi_k(z^k) + e^k \in N(z^k - e^k),$$

so that we have available an element in the normal cone N at the point $z^k - e^k \in \Re^n_+$. Consider now the pair

$$y^k = z^k - e^k$$

and

$$v^k = F(z^k - e^k) - \varphi_k(z^k) + e^k \in (F + N)(z^k - e^k) = (F + N)(y^k).$$

Let us analyze $y^k,\ v^k$ as an approximate solution of the proximal subproblem (2.1) in Algorithm 2.1 with $T = F + N$. We have to check whether (2.4), the stronger version of condition (2.2), is satisfied with

$$\varepsilon^k = -v^k - \mu_k(y^k - x^k)$$

and $y^k,\ v^k$ given above. If (2.4) is satisfied, then (2.5) holds and the hyperplane H_k given by (2.3) separates the current iterate x^k from zeros of $T = F + N$ or, equivalently, from solutions of NCP(F). Therefore we can make progress toward a solution of NCP(F) by making the projection step of Algorithm 2.1 with y^k and v^k defined above (followed by projection onto $\Re^n_+$ to preserve feasibility). As we shall see, the test (2.4) and the resulting step will be crucial for obtaining fast local convergence when the regularity assumption holds.

However, far from the solution or if regularity does not hold, an approximate (and even exact) Newton point obtained from solving the *linear* model may not satisfy the tolerance requirements (2.4) for the *nonlinear* proximal subproblem. In this case, the preceding separation arguments are not valid, and the Newton point has to be refined. For this task, we employ a linesearch procedure in the approximate Newton direction $z^k - x^k$ (see (2.12) in Algorithm 2.2) which computes a point $y^k = x^k + \alpha_k(z^k - x^k)$ such that

$$0 < \langle F(y^k), x^k - y^k \rangle.$$

A similar linesearch technique was used in [47, 40]. Note that for any $\bar{x}$ which solves NCP(F) we have

$$0 \geq \langle F(\bar{x}), \bar{x} - y^k \rangle,$$

where $y^k \in \Re^n_+$. Hence, by monotonicity of F,

$$0 \geq \langle F(y^k), \bar{x} - y^k \rangle.$$

Therefore in this case another hyperplane, namely,

$$H_k := \{x \in \Re^n \mid \langle F(y^k), x - y^k \rangle = 0\},$$

strictly separates the current iterate x^k from solutions of the problem. Once the separating hyperplane H_k is obtained, the next iterate x^{k+1} is computed by projecting x^k onto H_k and then onto the nonnegative orthant $\Re^n_+$.

We now formally state the algorithm.

Algorithm 2.2. *Choose any* $x^0 \in \Re^n$, $\sigma \in (0,1)$, $\beta \in (0,1)$, *and* $\lambda \in (0,1)$. *Set* $k := 0$.

Inexact Newton Step. *Choose a positive semidefinite matrix* G_k *and* $\mu_k > 0$. *Choose* $\rho_k \in [0,1)$ *and compute* $z^k \in \Re^n_+$, *an inexact solution of LCP*(φ_k) *given by* (2.6)–(2.8), *such that*

$$\|e^k\| \leq \rho_k \mu_k \|z^k - x^k\| \tag{2.9}$$

and

$$\langle e^k, \varphi_k(z^k) + z^k - x^k \rangle \leq \rho_k \mu_k \|z^k - x^k\|^2. \tag{2.10}$$

Stop if $z^k = x^k$. *Otherwise,*

Linesearch Step. *Set*

$$v^k := F(z^k - e^k) - \varphi_k(z^k) + e^k \quad \textit{and} \quad y^k := z^k - e^k.$$

Let

$$\varepsilon^k = -v^k - \mu_k(y^k - x^k).$$

If

$$\|\varepsilon^k\| \leq \sigma \mu_k \|y^k - x^k\|, \tag{2.11}$$

then go to the Projection Step.

Otherwise, find $y^k = x^k + \alpha_k(z^k - x^k)$, *where* $\alpha_k = \beta^{m_k}$ *with* m_k *being the smallest nonnegative integer* m *such that*

$$\langle F(x^k + \beta^m(z^k - x^k)), x^k - z^k \rangle \geq \lambda(1 - \rho_k)\mu_k \|z^k - x^k\|^2. \tag{2.12}$$

Set $v^k := F(y^k)$ *and go to the Projection Step.*

Projection Step. *Compute*

$$x^{k+1} = \max\left\{0; x^k - \frac{\langle v^k, x^k - y^k \rangle}{\|v^k\|^2} v^k \right\}. \tag{2.13}$$

Set $k := k + 1$; *and repeat.*

To compute z^k, an approximate solution of LCP(φ_k) satisfying (2.6)–(2.10), one can employ any appropriate algorithm known to converge for the strongly monotone linear complementarity problem. There are many algorithms which would generate a sequence converging to the unique solution of LCP(φ_k) with a (global) quadratic rate. This guarantees that after finitely many iterations (just a few, one hopes), the LCP(φ_k) residual e^k would be small enough, so that (2.9)–(2.10) holds.

Note that Algorithm 2.2 has computational cost per iteration of the same order as any other damped Newton method: solving, possibly inexactly, a linear complementarity problem followed by a simple linesearch procedure (if (2.11) is not satisfied). The projection step is explicit, and therefore its computational cost is negligible. The advantage of Algorithm 2.2 over other Newton-type methods is that it is truly globally convergent under minimal assumptions.

3. Global convergence analysis. We start with the global convergence analysis. Throughout we assume that the solution set of the problem is nonempty. First note that if Algorithm 2.2 terminates with $z^k = x^k$, then $e^k = 0$ by (2.9), and we have that z^k is the exact solution of LCP(φ_k). Therefore $\langle \varphi_k(z^k), x - z^k \rangle \geq 0$ for all $x \in \Re^n_+$. Because $z^k = x^k$ implies that $\varphi_k(z^k) = F(z^k)$, it follows from the latter inequality that z^k solves NCP(F). From now on, we assume that $z^k \neq x^k$ for all k, and an infinite sequence $\{x^k\}$ is generated.

We first state a preliminary result [43] whose simple proof we include for completeness.

LEMMA 3.1. *Let $x, y, v, \bar{x}$ be any elements of $\Re^n$ such that*

$$\langle v, x - y \rangle > 0 \quad \text{and} \quad \langle v, \bar{x} - y \rangle \leq 0.$$

Let

$$\hat{x} = x - \frac{\langle v, x - y \rangle}{\|v\|^2} v.$$

Then

$$\|\hat{x} - \bar{x}\|^2 \leq \|x - \bar{x}\|^2 - \|\hat{x} - x\|^2.$$

Proof. It follows from the hypothesis that the hyperplane $H = \{s \mid \langle v, s - y \rangle = 0\}$ separates x from $\bar{x}$. Moreover, $\hat{x}$ is the projection of x onto the half-space $\{s \mid \langle v, s - y \rangle \leq 0\}$. Since $\bar{x}$ belongs to this half-space, it follows from the basic properties of the projection operator (see [33, p. 121]) that $\langle x - \hat{x}, \hat{x} - \bar{x} \rangle \geq 0$. Therefore

$$\begin{aligned} \|x - \bar{x}\|^2 &= \|x - \hat{x}\|^2 + \|\hat{x} - \bar{x}\|^2 + 2\langle x - \hat{x}, \hat{x} - \bar{x} \rangle \\ &\geq \|x - \hat{x}\|^2 + \|\hat{x} - \bar{x}\|^2, \end{aligned}$$

which completes the proof. □

We are now ready to prove our main global convergence result. The remarkable property established in Theorem 3.2 is that the *whole* sequence of iterates always converges to some solution of NCP(F) under the assumptions of merely monotonicity and continuity of F. NCP(F) need not have a unique solution; in fact, the solution set may even be unbounded. In the latter case the level sets of any merit function for NCP(F) are also unbounded, so that for typical Newton methods with globalization strategies based on merit functions, even boundedness of iterates cannot be established.

THEOREM 3.2. *Suppose that F is continuous and monotone. Then any sequence $\{x^k\}$ generated by Algorithm* 2.2 *is bounded.*

Suppose further that there exist constants $C_1, C_2, C_3 > 0$ and $t > 0$ such that $\|G_k\| \leq C_1$ for all k and, starting with some index k_0,

$$C_2 \geq \mu_k \geq C_3 \|r(x^k)\|^t.$$

Suppose that

$$\min\{1; 1/C_2\} > \limsup_{k \to \infty} \rho_k.$$

Then $\{x^k\}$ converges to some $\bar{x}$, which is a solution of NCP(F).

Proof. First note that because

$$z^k - e^k = [z^k - \varphi_k(z^k)]^+$$

and $x^k \in \Re^n_+$, by properties of the projection operator [33, p. 121] it follows that

$$\langle z^k - \varphi_k(z^k) - (z^k - e^k), x^k - (z^k - e^k)\rangle \leq 0.$$

Therefore

$$\langle -\varphi_k(z^k) + e^k, x^k - z^k + e^k\rangle \leq 0.$$

Making use of the latter inequality, we further obtain

$$\begin{aligned}
\langle F(x^k), x^k - z^k\rangle &\geq \langle F(x^k) - \varphi_k(z^k) + e^k, x^k - z^k + e^k\rangle - \langle F(x^k), e^k\rangle \\
&= \langle (G_k + \mu_k I)(x^k - z^k), x^k - z^k + e^k\rangle + \langle e^k, x^k - z^k - F(x^k)\rangle + \|e^k\|^2 \\
&\geq \mu_k\|z^k - x^k\|^2 - \langle e^k, (G_k + \mu_k I)(z^k - x^k) + F(x^k) + z^k - x^k\rangle \\
&= \mu_k\|z^k - x^k\|^2 - \langle e^k, \varphi_k(z^k) + z^k - x^k\rangle \\
&\geq \mu_k(1-\rho_k)\|z^k - x^k\|^2, \qquad (3.1)
\end{aligned}$$

where the last inequality follows from (2.10) and the next to last follows from positive semidefiniteness of G_k.

We next show that the linesearch procedure (2.12), if activated, always terminates with a positive stepsize α_k. Suppose that for some iteration index k this is not the case. That is, for all integers m we have

$$\langle F(x^k + \beta^m(z^k - x^k)), x^k - z^k\rangle < \lambda(1-\rho_k)\mu_k\|z^k - x^k\|^2.$$

Since F is continuous, passing onto the limit as $m \to \infty$, we obtain

$$\langle F(x^k), x^k - z^k\rangle \leq \lambda(1-\rho_k)\mu_k\|z^k - x^k\|^2. \qquad (3.2)$$

Now since $\lambda \in (0,1)$, $\rho_k \in [0,1)$, and $z^k \neq x^k$, (3.2) contradicts (3.1). Therefore the linesearch step is well defined.

Denote

$$\hat{x}^k := x^k - \frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|^2} v^k. \qquad (3.3)$$

Observe that this is well defined because $v^k \neq 0$. Indeed, if $v^k = 0$ in the case when the linesearch is not activated, then (2.11) must hold. Furthermore, in that case $v^k = 0$ implies $\varepsilon^k = -\mu_k(y^k - x^k)$, which together with (2.11) implies that $\varepsilon^k = y^k - x^k = 0$. Therefore, using the definition of y^k, we conclude that $x^k = z^k - e^k$. If $\mu_k \leq 1$, then $\rho_k\mu_k < 1$, so (2.9) implies that $z^k = x^k$, in contradiction with the stopping test. Suppose now that $\mu_k > 1$. Note that $0 = v^k = F(x^k) - \varphi_k(z^k) + e^k = (-G_k + (1-\mu_k)I)e^k$. It then follows that $0 = \langle(-G_k + (1-\mu_k)I)e^k, e^k\rangle \leq (1-\mu_k)\|e^k\|^2$. Since $1 < \mu_k$, we again have that $e^k = 0$, i.e., $z^k = x^k$, which contradicts the stopping rule. If the linesearch is used to compute y^k, then $v^k = F(y^k)$, which again cannot be zero by (2.12). The observation that $v^k \neq 0$ concludes the proof that the whole algorithm is well defined.

With the notation (3.3), we have that $x^{k+1} = [\hat{x}^k]^+$. Let $\bar{x}$ be any solution of NCP(F). Since $\bar{x} \in \Re^n_+$, it is easy to see that

$$\|x^{k+1} - \bar{x}\| \leq \|\hat{x}^k - \bar{x}\|. \qquad (3.4)$$

Note that if (2.11) is satisfied, then $\langle v^k, x^k - y^k\rangle > 0$ (see (2.5)). If (2.11) is not satisfied, then $\langle v^k, x^k - y^k\rangle = \alpha_k\langle F(y^k), x^k - z^k\rangle > 0$ still holds by (2.12). In this respect, the only difference is the choice of y^k and v^k. As discussed before, in either case $\langle v^k, \bar{x} - y^k\rangle \leq 0$, so that by Lemma 3.1, it follows that

$$\|\hat{x}^k - \bar{x}\|^2 \leq \|x^k - \bar{x}\|^2 - \|\hat{x}^k - x^k\|^2.$$

Combining the latter relation with (3.4), we obtain

$$\|x^{k+1} - \bar{x}\|^2 \leq \|x^k - \bar{x}\|^2 - \|\hat{x}^k - x^k\|^2. \tag{3.5}$$

It immediately follows that the sequence $\{\|x^k - \bar{x}\|\}$ is monotone, so it converges. Therefore, $\{x^k\}$ is bounded.

We next consider the two possible cases:

$$0 = \liminf_{k\to\infty} \|r(x^k)\| \tag{3.6}$$

and

$$0 < \liminf_{k\to\infty} \|r(x^k)\|. \tag{3.7}$$

In the first case, by continuity of $r(\cdot)$ and boundedness of $\{x^k\}$, there exists $\tilde{x}$, an accumulation point of $\{x^k\}$, such that $r(\tilde{x}) = 0$. Therefore $\tilde{x}$ is a solution of NCP(F). Since $\bar{x}$ was an arbitrary solution, we can now choose $\bar{x} = \tilde{x}$ in (3.5). Because the sequence $\{\|x^k - \tilde{x}\|\}$ converges and $\tilde{x}$ is an accumulation point of $\{x^k\}$, it must be the case that $\{x^k\}$ converges to $\tilde{x}$, which is a solution of NCP(F).

Consider now the second case. By (3.5), it follows that

$$0 = \lim_{k\to\infty} \|\hat{x}^k - x^k\|$$

or, equivalently,

$$0 = \lim_{k\to\infty} \frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|}. \tag{3.8}$$

By the choice of μ_k and (3.7), it then follows that $\mu_k \geq C_3\|r(x^k)\|^t \geq C_4 > 0$ for all k. By (3.1) and the Cauchy–Schwarz inequality, we obtain

$$\begin{aligned}\|F(x^k)\|\|x^k - z^k\| &\geq \langle F(x^k), x^k - z^k\rangle \\ &\geq \mu_k(1-\rho_k)\|x^k - z^k\|^2.\end{aligned}$$

Hence,

$$\|F(x^k)\| \geq C_4(1-\rho_k)\|x^k - z^k\|.$$

Taking into account boundedness of $\{x^k\}$ and continuity of F, and the fact that $1 > \limsup_{k\to\infty}\rho_k$, we conclude that the sequence $\{x^k - z^k\}$ is bounded. It now easily follows that the sequences $\{z^k\}$, $\{e^k\}$, and $\{y^k\}$ are all bounded.

By the triangle and Cauchy–Schwarz inequalities, and the nonexpansiveness of the projection operator, we have

$$\begin{aligned}\|x^k - z^k\| &\geq \|x^k - (z^k - e^k)\| - \|e^k\| \\ &= \|x^k - [z^k - \varphi_k(z^k)]^+\| - \|e^k\| \\ &\geq \|x^k - [x^k - F(x^k)]^+\| - \|[x^k - F(x^k)]^+ - [z^k - \varphi_k(z^k)]^+\| - \|e^k\| \\ &\geq \|r(x^k)\| - \|x^k - z^k - F(x^k) + \varphi_k(z^k)\| - \|e^k\| \\ &\geq \|r(x^k)\| - (1+\rho_k\mu_k)\|x^k - z^k\| - \|(G_k + \mu_k I)(x^k - z^k)\| \\ &\geq \|r(x^k)\| - (1 + C_1 + 2C_2)\|x^k - z^k\|.\end{aligned}$$

Hence,

$$(2 + C_1 + 2C_2)\|x^k - z^k\| \geq \|r(x^k)\|.$$

It follows from (3.7) that

$$0 < \liminf_{k\to\infty} \|x^k - z^k\|. \tag{3.9}$$

Suppose that condition (2.11) in Algorithm 2.2 holds an infinite number of times. For such iterations k, by (2.5), we have (recall also that $v^k \neq 0$)

$$\begin{aligned}\frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|} &\geq \frac{\mu_k(1-\sigma)\|x^k - y^k\|^2}{\|v^k\|}\\ &\geq \frac{C_4(1-\sigma)\|x^k - z^k + e^k\|^2}{\|F(z^k - e^k) - \varphi_k(z^k) + e^k\|}.\end{aligned} \tag{3.10}$$

Since $\|G_k\| \leq C_1$, $\mu_k \leq C_2$ and $\{z^k\}$, $\{e^k\}$ are bounded, it follows that $\{F(z^k - e^k)\}$ and $\{\varphi_k(z^k)\}$ are bounded. Therefore, for some $C_5 > 0$,

$$\frac{C_4(1-\sigma)}{\|F(z^k - e^k) - \varphi_k(z^k) + e^k\|} \geq C_5.$$

Passing onto the limit in (3.10) (along the indices k for which (2.11) holds) and taking into account (3.8), we obtain that

$$0 = \liminf_{k\to\infty} \|x^k - z^k + e^k\|.$$

By the triangle inequality and (2.9), we have

$$\begin{aligned}\|x^k - z^k + e^k\| &\geq \|x^k - z^k\| - \|e^k\|\\ &\geq (1 - \rho_k\mu_k)\|x^k - z^k\|\\ &\geq (1 - \rho_k C_2)\|x^k - z^k\|.\end{aligned}$$

Because $1/C_2 > \limsup_{k\to\infty} \rho_k$, we further conclude that

$$0 = \liminf_{k\to\infty} \|x^k - z^k\|$$

which contradicts (3.9). We conclude that if $0 < \liminf_{k\to\infty} \|r(x^k)\|$, then condition (2.11) in Algorithm 2.2 may hold no more than a finite number of times.

Hence, we can assume that for all k sufficiently large, y^k and v^k are obtained through the linesearch step (2.12), in which case

$$\begin{aligned}\frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|} &= \frac{\alpha_k\langle F(y^k), x^k - z^k\rangle}{\|F(y^k)\|}\\ &\geq \frac{\alpha_k\lambda(1-\rho_k)\mu_k\|x^k - z^k\|^2}{\|F(y^k)\|}.\end{aligned}$$

Using (3.8) and taking into account the boundedness of $\{F(y^k)\}$ and the fact that $\mu_k \geq C_4$ and $1 > \limsup_{k\to\infty} \rho_k$, we obtain that

$$0 = \lim_{k\to\infty} \alpha_k\|x^k - z^k\|.$$

Now, because of (3.9), we conclude that it must be the case that

$$0 = \lim_{k\to\infty} \alpha_k.$$

The latter is equivalent to saying that $m_k \to \infty$. It follows that for every sufficiently large k, the stepsize is decreased at least twice, i.e., $m_k \geq 2$. Hence, the stepsize rule (2.12) is not satisfied for the value of $\beta^{m_k - 1}$, i.e.,

$$\langle F(x^k + \beta^{m_k-1}(z^k - x^k)), x^k - z^k\rangle < \lambda(1-\rho_k)\mu_k\|z^k - x^k\|^2.$$

Taking into account boundedness of the sequences $\{x^k\}$, $\{\mu_k\}$, $\{\rho_k\}$, and $\{z^k\}$, and passing onto a subsequence if necessary, as $k \to \infty$ we obtain

$$\langle F(\hat{x}), \hat{x} - \hat{z}\rangle \leq \lambda(1-\hat{\rho})\hat{\mu}\|\hat{z} - \hat{x}\|^2,$$

where $\hat{x}$, $\hat{\mu}$, $\hat{\rho}$, and $\hat{z}$ are limits of corresponding subsequences. On the other hand, passing onto the limit in (3.1), we have that

$$\langle F(\hat{x}), \hat{x} - \hat{z}\rangle \geq (1-\hat{\rho})\hat{\mu}\|\hat{z} - \hat{x}\|^2.$$

Taking into account that $\hat{\mu} > 0$ and $\|\hat{z} - \hat{x}\| > 0$ (by (3.9)), and $\hat{\rho} \leq \limsup_{k\to\infty} \rho_k < 1$, the last two relations are a contradiction because $\lambda \in (0, 1)$. Hence the case $0 < \liminf_{k\to\infty} \|r(x^k)\|$ is not possible.

This completes the proof. □

Remark. Since boundedness of $\{x^k\}$ was established without any boundedness assumptions on $\{G_k\}$, for the special choice of $G_k = \nabla F(x^k)$ the condition $\|G_k\| \leq C_1$ can be removed, due to the continuity of $\nabla F(\cdot)$.

4. Local convergence analysis.

The following *error bound* [29] result will be crucial for establishing the superlinear rate of convergence of our algorithm. Note that this error bound actually holds under the more general assumption of $\bar{x}$ being a *regular* solution [35, 26, 14]. Positive definiteness of $\nabla F(\bar{x})$ is a simple sufficient condition for $\bar{x}$ to be regular [31]. Here we state only the simplified result.

LEMMA 4.1 ([14, Proposition 4.4]). *If $\bar{x}$ is a solution of NCP(F) where $\nabla F(\bar{x})$ is positive definite, then there exist a constant $\theta > 0$ and a neighborhood B of $\bar{x}$ such that*

$$\|x - \bar{x}\| \leq \theta\|r(x)\|$$

for all $x \in B$.

We will also need the following error bound result for strongly monotone linear complementarity problems [27]. This result is actually related to Lemma 4.1, but note that the constant θ can be estimated explicitly and the error bound holds globally.

LEMMA 4.2. *Let $\hat{z}$ be the solution of the linear complementarity problem*

$$z \geq 0, \quad Mz + q \geq 0, \quad \langle Mz + q, z\rangle = 0,$$

where M is a positive definite matrix and q is an arbitrary vector. For all $z \in \Re^n$ it holds that

$$\|z - \hat{z}\| \leq \frac{1 + \|M\|}{C(M)}\|\min\{z; Mz + q\}\|,$$

where $C(M) > 0$ is the smallest eigenvalue of $(M + M^\top)/2$.

We are now ready to establish the superlinear convergence of our algorithm for solving the monotone nonlinear complementarity problems under the assumptions of positive definiteness of ∇F at the solution and its local Hölder continuity. The proof relies on the fact that in a sufficiently small neighborhood of such a solution, the approximate Newton point computed by Algorithm 2.2 satisfies the error tolerance requirements of Algorithm 2.1 for solving the corresponding nonlinear proximal point subproblem. Therefore the corresponding projection step in Algorithm 2.2 is taken immediately, and the linesearch procedure is not used.

THEOREM 4.3. *Let F be monotone and continuous on $\Re^n$. Let $\bar{x}$ be the (unique) solution of NCP(F) at which F is differentiable with $\nabla F(\bar{x})$ positive definite. Let ∇F be locally Hölder continuous around $\bar{x}$ with degree $p \in (0,1]$. Suppose that the assumptions of Theorem* 3.2 *are satisfied. In addition, suppose that*

$$\mu_k = \|r(x^k)\|^t, \quad t \in (0,p),$$

$$0 = \lim_{k\to\infty} \rho_k,$$

and starting with some index k_0, $G_k = \nabla F(x^k)$.

Then the sequence $\{x^k\}$ converges to $\bar{x}$ Q-superlinearly.

Proof. By Theorem 3.2, we already know that the sequence $\{x^k\}$ converges to $\bar{x}$. (Note that monotonicity of F and nonsingularity of $\nabla F(\bar{x})$ imply that $\bar{x}$ is the unique solution.) Note that by the choice of μ_k, we have that $0 = \lim_{k\to\infty} \mu_k$.

By Hölder continuity of ∇F in the neighborhood of $\bar{x}$, it follows that there exists some $L > 0$ such that for all $u \in \Re^n$ sufficiently small and all indices k sufficiently large

$$\|\nabla F(x^k + u) - \nabla F(x^k)\| \le L\|u\|^p, \quad p \in (0,1].$$

Therefore

$$\begin{aligned} F(x^k + u) - F(x^k) &= \int_0^1 \nabla F(x^k + su)u\, ds \\ &= \nabla F(x^k)u + \int_0^1 (\nabla F(x^k + su) - \nabla F(x^k))u\, ds \\ &= \nabla F(x^k)u + R^k(u), \end{aligned}$$

where

$$\begin{aligned} \|R^k(u)\| &\le \int_0^1 \|\nabla F(x^k + su) - \nabla F(x^k)\|\|u\|\, ds \\ &\le L\int_0^1 s^p\|u\|^{1+p}\, ds \\ &= \frac{L}{1+p}\|u\|^{1+p}. \end{aligned}$$

Hence, defining $C_6 := L/(1+p)$, we have that

$$\begin{aligned} F(x^k + u) - F(x^k) - \nabla F(x^k)u &= R^k(u), \\ \|R^k(u)\| &\le C_6\|u\|^{1+p}. \end{aligned} \tag{4.1}$$

We next show that (2.11) is always satisfied for all k sufficiently large, so that the linesearch step is never used. Thus let $y^k = z^k - e^k$ and $v^k = F(z^k - e^k) - \varphi_k(z^k) + e^k$. Assume that k is sufficiently large so that (4.1) holds and $y^k - x^k$ is sufficiently small (later we shall verify that $\{y^k - x^k\} \to 0$). Then we have that

$$\begin{aligned}
-\varepsilon^k &= v^k + \mu_k(y^k - x^k) \\
&= F(z^k - e^k) - \varphi_k(z^k) + e^k + \mu_k(z^k - e^k - x^k) \\
&= F(z^k - e^k) - F(x^k) - \nabla F(x^k)(z^k - e^k - x^k) - \nabla F(x^k)e^k + (1 - \mu_k)e^k \\
&= R^k(z^k - e^k - x^k) - \nabla F(x^k)e^k + (1 - \mu_k)e^k. \qquad (4.2)
\end{aligned}$$

By (4.1) and the Cauchy–Schwarz and triangle inequalities, it follows that

$$\begin{aligned}
\|\varepsilon^k\| &\leq C_6\|y^k - x^k\|^{1+p} + (C_1 + 1)\|e^k\| \\
&\leq C_6\|y^k - x^k\|^{1+p} + (C_1 + 1)\rho_k\mu_k\|z^k - x^k\|. \qquad (4.3)
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\|z^k - x^k\| &\leq \|y^k - x^k\| + \|e^k\| \\
&\leq \|y^k - x^k\| + \rho_k\mu_k\|z^k - x^k\|.
\end{aligned}$$

Hence,

$$\|z^k - x^k\| \leq C_7\|y^k - x^k\|, \qquad (4.4)$$

where $C_7 \geq 1/(1 - \rho_k\mu_k)$ (recall that $\rho_k\mu_k \to 0$). Defining $C_8 := C_7(C_1 + 1)$, and combining (4.4) with (4.3), we obtain

$$\begin{aligned}
\|\varepsilon^k\| &\leq (C_6\|y^k - x^k\|^p + C_8\rho_k\mu_k)\|y^k - x^k\| \\
&= \left(C_6\frac{\|y^k - x^k\|^p}{\|r(x^k)\|^t} + C_8\rho_k\right)\mu_k\|y^k - x^k\|.
\end{aligned}$$

Clearly, condition (2.11) is satisfied whenever

$$C_6\frac{\|y^k - x^k\|^p}{\|r(x^k)\|^t} + C_8\rho_k \leq \sigma.$$

Since

$$0 = \lim_{k\to\infty} \rho_k,$$

the latter relation is satisfied for all indices k sufficiently large if it holds that

$$0 = \lim_{k\to\infty} \frac{\|y^k - x^k\|^p}{\|r(x^k)\|^t}. \qquad (4.5)$$

Thus for establishing that the linesearch procedure is never used for indices k sufficiently large, it is left to prove (4.5). (This would also imply that $\{y^k - x^k\} \to 0$.)

Let $\hat{z}^k$ be the exact solution of LCP(φ_k). It holds that

$$\langle F(\bar{x}) - \varphi_k(\hat{z}^k), \bar{x} - \hat{z}^k\rangle = -\langle \hat{z}^k, F(\bar{x})\rangle - \langle \bar{x}, \varphi_k(\hat{z}^k)\rangle \leq 0,$$

because $\bar{x}, F(\bar{x}), \hat{z}^k, \varphi_k(\hat{z}^k)$ are all nonnegative and $\langle \bar{x}, F(\bar{x})\rangle = \langle \hat{z}^k, \varphi_k(\hat{z}^k)\rangle = 0$. Furthermore, by (4.1) we obtain

$$\begin{aligned} F(\bar{x}) - \varphi_k(\hat{z}^k) &= F(\bar{x}) - F(x^k) - (\nabla F(x^k) + \mu_k I)(\hat{z}^k - x^k) \\ &= F(\bar{x}) - F(x^k) - \nabla F(x^k)(\bar{x} - x^k) - \nabla F(x^k)(\hat{z}^k - \bar{x}) - \mu_k(\hat{z}^k - x^k) \\ &= R^k(\bar{x} - x^k) - \nabla F(x^k)(\hat{z}^k - \bar{x}) - \mu_k(\hat{z}^k - x^k). \end{aligned}$$

Hence,

$$\begin{aligned} 0 &\geq \langle F(\bar{x}) - \varphi_k(\hat{z}^k), \bar{x} - \hat{z}^k\rangle \\ &= \langle R^k(\bar{x} - x^k), \bar{x} - \hat{z}^k\rangle - \langle \nabla F(x^k)(\hat{z}^k - \bar{x}), \bar{x} - \hat{z}^k\rangle - \mu_k\langle \hat{z}^k - x^k, \bar{x} - \hat{z}^k\rangle. \end{aligned} \tag{4.6}$$

By positive definiteness of $\nabla F(\bar{x})$, it follows that for some $C_9 > 0$ and all indices k sufficiently large

$$\langle \nabla F(x^k)d, d\rangle \geq C_9\|d\|^2 \quad \text{for all } d \in \Re^n. \tag{4.7}$$

By (4.6) and the latter relation we obtain

$$\begin{aligned} C_9\|\hat{z}^k - \bar{x}\|^2 &\leq \langle \nabla F(x^k)(\hat{z}^k - \bar{x}), \hat{z}^k - \bar{x}\rangle \\ &\leq \langle R^k(\bar{x} - x^k), \hat{z}^k - \bar{x}\rangle + \mu_k\langle \hat{z}^k - x^k, \bar{x} - \hat{z}^k\rangle \\ &\leq C_6\|x^k - \bar{x}\|^{1+p}\|\hat{z}^k - \bar{x}\| + \mu_k\|\hat{z}^k - x^k\|\|\hat{z}^k - \bar{x}\|, \end{aligned}$$

where the last inequality follows from (4.1) and the Cauchy–Schwarz inequality. Therefore,

$$C_9\|\hat{z}^k - \bar{x}\| \leq C_6\|x^k - \bar{x}\|^{1+p} + \mu_k\|\hat{z}^k - x^k\|. \tag{4.8}$$

By the triangle inequality,

$$\|\hat{z}^k - x^k\| \leq \|\hat{z}^k - \bar{x}\| + \|x^k - \bar{x}\|.$$

Combining the latter inequality with (4.8), we obtain

$$\|\hat{z}^k - x^k\| \leq C_9^{-1}\left(C_6\|x^k - \bar{x}\|^{1+p} + \mu_k\|\hat{z}^k - x^k\|\right) + \|x^k - \bar{x}\|.$$

Hence,

$$(1 - \mu_k/C_9)\|\hat{z}^k - x^k\| \leq C_6/C_9\|x^k - \bar{x}\|^{1+p} + \|x^k - \bar{x}\|.$$

And for $C_{11} \geq 1/(1 - \mu_k/C_9)$ (recall that $\mu_k \to 0$), we obtain

$$\|\hat{z}^k - x^k\| \leq C_{11}\left(1 + C_6/C_9\|x^k - \bar{x}\|^p\right)\|x^k - \bar{x}\|. \tag{4.9}$$

By the triangle inequality,

$$\|z^k - x^k\| \leq \|\hat{z}^k - z^k\| + \|\hat{z}^k - x^k\|. \tag{4.10}$$

Furthermore by Lemma 4.2, we have that

$$\|z^k - \hat{z}^k\| \leq \frac{1 + \|\nabla F(x^k) + \mu_k I\|}{c_k}\|e^k\|,$$

where c_k is the smallest eigenvalue of $(\nabla F(x^k)+\nabla F(x^k)^\top)/2+\mu_k I$. In view of (4.7), clearly $c_k \geq C_9$. Therefore

$$\|z^k - \hat{z}^k\| \leq \frac{2+C_1}{C_9}\|e^k\|.$$

Furthermore, defining $C_{10} := (2+C_1)/C_9$, we obtain

$$\|z^k - \hat{z}^k\| \leq C_{10}\rho_k\mu_k\|z^k - x^k\|. \tag{4.11}$$

Combining the latter relation with (4.10), we have that

$$\|z^k - x^k\| \leq C_{10}\rho_k\mu_k\|z^k - x^k\| + \|\hat{z}^k - x^k\|.$$

Since $\rho_k\mu_k \to 0$, taking into account (4.9), it is now clear that there exists $C_{12} > 0$ such that

$$\|z^k - x^k\| \leq C_{12}\|x^k - \bar{x}\|. \tag{4.12}$$

Note that

$$\begin{aligned} \|y^k - x^k\| &\leq \|z^k - x^k\| + \|e^k\| \\ &\leq (1+\rho_k\mu_k)\|z^k - x^k\| \\ &\leq 2\|z^k - x^k\| \\ &\leq 2C_{12}\|x^k - \bar{x}\|, \end{aligned} \tag{4.13}$$

where the last inequality follows from (4.12). Hence, by Lemma 4.1,

$$\|y^k - x^k\| \leq 2C_{12}\theta\|r(x^k)\|. \tag{4.14}$$

Therefore

$$\frac{\|y^k - x^k\|^p}{\|r(x^k)\|^t} \leq 2C_{12}\theta\|r(x^k)\|^{p-t}, \tag{4.15}$$

which establishes (4.5) by the choice of $t \in (0,p)$.

Hence, from now on, we can assume that the linesearch is never used, so that $y^k = z^k - e^k$ and $v^k = F(z^k - e^k) - \varphi_k(z^k) + e^k$. Recalling the definition

$$\hat{x}^k := x^k - \frac{\langle v^k, x^k - y^k\rangle}{\|v^k\|^2}v^k,$$

it is easy to see that

$$\|x^{k+1} - \bar{x}\| \leq \|\hat{x}^k - \bar{x}\|$$

because $\bar{x} \in \Re^n_+$ and x^{k+1} is the orthogonal projection of $\hat{x}^k$ onto $\Re^n_+$. Applying further the triangle inequality, we obtain

$$\|x^{k+1} - \bar{x}\| \leq \|\hat{x}^k - y^k\| + \|y^k - z^k\| + \|z^k - \hat{z}^k\| + \|\hat{z}^k - \bar{x}\|. \tag{4.16}$$

We proceed to analyze the four terms in the right-hand side of (4.16).

For the second term in (4.16) we have

$$\|y^k - z^k\| = \|e^k\| \leq \rho_k\mu_k\|z^k - x^k\|,$$

and for the third term (see (4.11)) we have

$$\|z^k - \hat{z}^k\| \le C_{10}\|e^k\| \le C_{10}\rho_k\mu_k\|z^k - x^k\|.$$

Using (4.12) we obtain

$$\begin{aligned}\|y^k - z^k\| + \|z^k - \hat{z}^k\| &\le (1 + C_{10})\rho_k\mu_k\|z^k - x^k\| \\ &\le (1 + C_{10})C_{12}\rho_k\|r(x^k)\|^t\|x^k - \bar{x}\|.\end{aligned} \tag{4.17}$$

Next, we consider the last term in (4.16). By (4.8), we have

$$\begin{aligned}C_9\|\hat{z}^k - \bar{x}\| &\le C_6\|x^k - \bar{x}\|^{1+p} + \mu_k\|\hat{z}^k - x^k\| \\ &\le C_6\|x^k - \bar{x}\|^{1+p} + \mu_k(\|\hat{z}^k - \bar{x}\| + \|x^k - \bar{x}\|),\end{aligned}$$

where the last inequality follows from the triangle inequality. Therefore,

$$(C_9 - \mu_k)\|\hat{z}^k - \bar{x}\| \le C_6\|x^k - \bar{x}\|^{1+p} + \mu_k\|x^k - \bar{x}\|.$$

Since $\mu_k \to 0$, for $C_{13} \ge \max\{1; C_6\}/(C_9 - \mu_k)$, we have

$$\|\hat{z}^k - \bar{x}\| \le C_{13}\left(\|x^k - \bar{x}\|^p + \|r(x^k)\|^t\right)\|x^k - \bar{x}\|. \tag{4.18}$$

Finally, we consider the first term in the right-hand side of (4.16). Since the point $\hat{x}^k$ is the projection of x^k onto the hyperplane $H_k = \{x \mid \langle v^k, x - y^k\rangle = 0\}$, and $y^k \in H_k$, the vectors $\hat{x}^k - x^k$ and $\hat{x}^k - y^k$ are orthogonal. Hence,

$$\|\hat{x}^k - y^k\| = \|y^k - x^k\|\sin\xi_k, \tag{4.19}$$

where ξ_k is the angle between $\hat{x}^k - x^k$ and $y^k - x^k$. Because $\hat{x}^k - x^k = -s_k v^k$ for a certain $s_k > 0$, the angle between the vectors v^k and $-\mu_k(y^k - x^k)$ is also ξ_k. Observe from (4.2) that

$$v^k = -\mu_k(y^k - x^k) + R^k(y^k - x^k) - \nabla F(x^k)e^k + (1 - \mu_k)e^k.$$

Given the above relation, sin of the angle between v^k and $-\mu_k(y^k - x^k)$ can be easily bounded:

$$\begin{aligned}\sin\xi_k &\le \frac{\|R^k(y^k - x^k) + ((1 - \mu_k)I - \nabla F(x^k))e^k\|}{\mu_k\|y^k - x^k\|} \\ &\le \frac{C_6\|y^k - x^k\|^{1+p} + (1 + C_1)\|e^k\|}{\mu_k\|y^k - x^k\|} \\ &\le C_6\frac{\|y^k - x^k\|^p}{\|r(x^k)\|^t} + (1 + C_1)\frac{\rho_k\|z^k - x^k\|}{\|y^k - x^k\|} \\ &\le 2C_6C_{12}\theta\|r(x^k)\|^{p-t} + C_7(1 + C_1)\rho_k,\end{aligned}$$

where the second inequality follows from (4.1) and the last inequality follows from (4.15) and (4.4). Hence, for some $C_{14} > 0$,

$$\sin\xi_k \le C_{14}(\|r(x^k)\|^{p-t} + \rho_k). \tag{4.20}$$

By (4.19), using (4.13) and (4.20), we obtain

$$\begin{aligned}\|\hat{x}^k - y^k\| &= \|y^k - x^k\|\sin\xi_k \\ &\le 2C_{12}C_{14}\left(\|r(x^k)\|^{p-t} + \rho_k\right)\|x^k - \bar{x}\|.\end{aligned} \tag{4.21}$$

By (4.16), combining (4.21), (4.17), and (4.18), we conclude that there exists $C_{15} > 0$ such that

$$\begin{aligned} \|x^{k+1} - \bar{x}\| \leq C_{15} \left(\|r(x^k)\|^{p-t} + \rho_k + (1+\rho_k)\|r(x^k)\|^t + \|x^k - \bar{x}\|^p \right) \\ \times \|x^k - \bar{x}\|, \end{aligned} \tag{4.22}$$

which means that $\{x^k\}$ converges to $\bar{x}$ at least superlinearly. □

Remark. Finally, we note that

$$\begin{aligned} \|r(x)\| &= \|x - [x - F(x)]^+ - \bar{x} + [\bar{x} - F(\bar{x})]^+\| \\ &\leq \|x - \bar{x}\| + \|[x - F(x)]^+ - [\bar{x} - F(\bar{x})]^+\| \\ &\leq (2 + C_{16})\|x - \bar{x}\|, \end{aligned}$$

where C_{16} is the constant of local Lipschitz continuity of F. Therefore if one further chooses

$$\rho_k = \|r(x^k)\|^p,$$

relation (4.22) implies that the order of superlinear convergence is at least $1 + p - t$, where $t \in (0, p)$.

5. Concluding remarks. We presented a new globalization strategy for the Newton method applied to the monotone nonlinear complementarity problem. Our strategy is based on the projection–proximal point methodology and makes full use of the monotonicity structure of the problem. The resulting hybrid algorithm is truly globally convergent to a solution without any additional assumptions, even if the solution set is unbounded. This is an important property which is not possessed by globalization approaches based on merit functions. Under natural assumptions, locally superlinear rate of convergence was also established.

Acknowledgments. We thank the two anonymous referees and the associate editor for constructive suggestions which helped us to improve the paper.

REFERENCES

[1] J. F. Bonnans, *Local analysis of Newton-type methods for variational inequalities and nonlinear programming*, Appl. Math. Optim., 29 (1994), pp. 161–186.

[2] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, New York, 1992.

[3] S. P. Dirkse and M. C. Ferris, *The PATH solver: A nonmonotone stabilization scheme for mixed complementarity problems*, Optim. Methods Softw., 5 (1995), pp. 123–156.

[4] F. Facchinei, A. Fischer, and C. Kanzow, *Inexact Newton methods for semismooth equations with applications to variational inequality problems*, in Nonlinear Optimization and Applications, G. D. Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 125–139.

[5] F. Facchinei and C. Kanzow, *Beyond monotonicity in regularization methods for nonlinear complementarity problems*, SIAM J. Control Optim., 37 (1999), pp. 150–1161.

[6] F. Facchinei and J. Soares, *A new merit function for nonlinear complementarity problems and a related algorithm*, SIAM J. Optim., 7 (1997), pp. 225–247.

[7] M. C. Ferris and J.-S. Pang, eds., *Complementarity and Variational Problems: State of the Art*, SIAM, Philadelphia, PA, 1997.

[8] M. C. Ferris, C. Kanzow, and T. S. Munson, *Feasible descent algorithms for mixed complementarity problems*, Math. Programming, 86 (1999), pp. 475–497.

[9] A. Fischer, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.

[10] A. Fischer, *An NCP function and its use for the solution of complementarity problems*, in Recent Advances in Nonsmooth Optimization, D.-Z. Du, L. Qi, and R. Womersley, eds., World Scientific Publishers, Singapore, 1995, pp. 88–105.

[11] M. Fukushima, *Equivalent differentiable optimization problems and descent methods for asymmetric variational inequality problems*, Math. Programming, 53 (1992), pp. 99–110.

[12] M. Fukushima, *Merit functions for variational inequality and complementarity problems*, in Nonlinear Optimization and Applications, G. D. Pillo and F. Giannessi, eds., Plenum Press, New York, 1996, pp. 155–170.

[13] J. Han and D. Sun, *Newton-Type Methods for Variational Inequalities*, manuscript, 1997.

[14] P. Harker and J.-S. Pang, *Finite-dimensional variational inequality problems: A survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.

[15] H. Jiang and L. Qi, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Control Optim., 35 (1997), pp. 178–193.

[16] H. Jiang and D. Ralph, *Global and local superlinear convergence analysis of Newton-type methods for semismooth equations with smooth least squares*, in Reformulation—Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 181–210.

[17] N. Josephy, *Newton's Method for Generalized Equations*, Technical Summary Report 1965, Mathematics Research Center, University of Wisconsin, Madison, WI, 1979.

[18] C. Kanzow, *Some equation-based methods for the nonlinear complementarity problem*, Optim. Methods Softw., 3 (1994), pp. 327–340.

[19] C. Kanzow, *Nonlinear complementarity as unconstrained optimization*, J. Optim. Theory Appl., 88 (1996), pp. 139–155.

[20] C. Kanzow and M. Fukushima, *Theoretical and numerical investigation of the D-gap function for box constrained variational inequalities*, Math. Programming, 83 (1998), pp. 55–87.

[21] T. D. Luca, F. Facchinei, and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problems*, Math. Programming, 75 (1996), pp. 407–439.

[22] Z.-Q. Luo and P. Tseng, *A new class of merit functions for the nonlinear complementarity problem*, in Complementarity and Variational Problems: State of the art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 204–225.

[23] O. L. Mangasarian and M. V. Solodov, *Nonlinear complementarity as unconstrained and constrained minimization*, Math. Programming, 62 (1993), pp. 277–297.

[24] O. L. Mangasarian and M. V. Solodov, *A linearly convergent derivative-free descent method for strongly monotone complementarity problems*, Comput. Optim. Appl., 14 (1999), pp. 5–16.

[25] P. Marcotte and J.-P. Dussault, *A note on a globally convergent Newton method for solving monotone variational inequalities*, Oper. Res. Lett., 6 (1987), pp. 35–42.

[26] J.-S. Pang, *Inexact Newton methods for the nonlinear complementarity problem*, Math. Programming, 36 (1986), pp. 54–71.

[27] J.-S. Pang, *A posteriori error bounds for the linearly-constrained variational inequality problem*, Math. Oper. Res., 12 (1987), pp. 474–484.

[28] J.-S. Pang, *Complementarity problems*, in Handbook of Global Optimization, R. Horst and P. Pardalos, eds., Kluwer Academic Publishers, Boston, MA, 1995, pp. 271–338.

[29] J.-S. Pang, *Error bounds in mathematical programming*, Math. Programming, 79 (1997), pp. 299–332.

[30] J.-S. Pang and D. Chan, *Iterative methods for variational and complementarity problems*, Math. Programming, 24 (1982), pp. 284–313.

[31] J.-M. Peng and M. Fukushima, *A hybrid Newton method for solving the variational inequality problem via the D-gap function*, Math. Programming, 86 (1999), pp. 367–386.

[32] J.-M. Peng, C. Kanzow, and M. Fukushima, *A hybrid Josephy-Newton method for solving box constrained variational inequality problem via the D-gap function*, Optim. Methods Softw., 10 (1999), pp. 687–710.

[33] B. T. Polyak, *Introduction to Optimization*, Optimization Software, Inc., Publications Division, New York, 1987.

[34] D. Ralph, *Global convergence of damped Newton's method for nonsmooth equations via the path search*, Math. Oper. Res., 19 (1994), pp. 352–389.

[35] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

[36] S. M. Robinson, *Normal maps induced by linear transformations*, Math. Oper. Res., 17 (1992), pp. 691–714.

[37] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.

[38] M. Solodov and B. Svaiter, *A comparison of rates of convergence of two inexact proximal point algorithms*, in Nonlinear Optimization and Related Topics, G. D. Pillo and F. Gian-

nessi, eds., Kluwer Academic Publishers, Norwell, MA, to appear.

[39] M. Solodov and B. Svaiter, *Forcing strong convergence of proximal point iterations in a Hilbert space*, Math. Programming, to appear.

[40] M. Solodov and B. Svaiter, *A globally convergent inexact Newton method for systems of monotone equations*, in Reformulation—Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 355–369.

[41] M. Solodov and B. Svaiter, *A hybrid approximate extragradient–proximal point algorithm using the enlargement of a maximal monotone operator*, Set-Valued Anal., 7 (1999), pp. 323–345.

[42] M. Solodov and B. Svaiter, *Error bounds for proximal point subproblems and associated inexact proximal point algorithms*, Math. Programming, submitted.

[43] M. Solodov and B. Svaiter, *A hybrid projection–proximal point algorithm*, J. Convex Anal., 6 (1999), pp. 59–70.

[44] M. V. Solodov, *Stationary points of bound constrained reformulations of complementarity problems*, J. Optim. Theory Appl., 94 (1997), pp. 449–467.

[45] M. V. Solodov, *Implicit Lagrangian*, in Encyclopedia of Optimization, C. Floudas and P. Pardalos, eds., Kluwer Academic Publishers, Norwell, MA, 2000.

[46] M. V. Solodov, *Some optimization reformulations of the extended linear complementarity problem*, Comput. Optim. Appl., 12 (1999), pp. 187–200.

[47] M. V. Solodov and B. F. Svaiter, *A new projection method for variational inequality problems*, SIAM J. Control Optim., 37 (1999), pp. 765–776.

[48] D. Sun, *A regularization Newton method for solving nonlinear complementarity problems*, Appl. Math. Optim., 40 (1999), pp. 315–339.

[49] D. Sun, M. Fukushima, and L. Qi, *A computable generalized Hessian of the D-gap function and Newton-type methods for variational inequality problems*, in Complementarity and Variational Problems: State of the Art, M. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 452–473.

[50] K. Taji and M. Fukushima, *Optimization based globally convergent methods for the nonlinear complementarity problems*, J. Oper. Res. Soc. Japan, 37 (1994), pp. 310–331.

[51] K. Taji, M. Fukushima, and T. Ibaraki, *A globally convergent Newton method for solving strongly monotone variational inequalities*, Math. Programming, 58 (1993), pp. 369–383.

[52] J. Wu, M. Florian, and P. Marcotte, *A general descent framework for the monotone variational inequality problem*, Math. Programming, 61 (1993), pp. 281–300.

[53] N. Yamashita and M. Fukushima, *Modified Newton methods for solving a semismooth reformulation of monotone complementarity problems*, Math. Programming, 76 (1997), pp. 469–491.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 627–642

GRADIENT CONVERGENCE IN GRADIENT METHODS WITH ERRORS[*]

DIMITRI P. BERTSEKAS[†] AND JOHN N. TSITSIKLIS[†]

Abstract. We consider the gradient method $x_{t+1} = x_t + \gamma_t(s_t + w_t)$, where s_t is a descent direction of a function $f : \Re^n \to \Re$ and w_t is a deterministic or stochastic error. We assume that ∇f is Lipschitz continuous, that the stepsize γ_t diminishes to 0, and that s_t and w_t satisfy standard conditions. We show that either $f(x_t) \to -\infty$ or $f(x_t)$ converges to a finite value and $\nabla f(x_t) \to 0$ (with probability 1 in the stochastic case), and in doing so, we remove various boundedness conditions that are assumed in existing results, such as boundedness from below of f, boundedness of $\nabla f(x_t)$, or boundedness of x_t.

Key words. gradient methods, incremental gradient methods, stochastic approximation, gradient convergence

AMS subject classifications. 62L20, 903C0

PII. S1052623497331063

1. Introduction.

We consider the problem

$$\text{(1.1)} \qquad \begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \in \Re^n, \end{array}$$

where $\Re^n$ denotes the n-dimensional Euclidean space and $f : \Re^n \mapsto \Re$ is a continuously differentiable function, such that for some constant L we have

$$\text{(1.2)} \qquad \|\nabla f(x) - \nabla f(\overline{x})\| \le L\|x - \overline{x}\| \qquad \forall\, x, \overline{x} \in \Re^n.$$

The purpose of this paper is to sharpen the existing convergence theory for the classical descent method

$$\text{(1.3)} \qquad x_{t+1} = x_t + \gamma_t(s_t + w_t),$$

where

(a) γ_t is a positive stepsize sequence satisfying

$$\text{(1.4)} \qquad \sum_{t=0}^{\infty} \gamma_t = \infty, \qquad \sum_{t=0}^{\infty} \gamma_t^2 < \infty;$$

(b) s_t is a descent direction satisfying for some positive scalars c_1 and c_2, and all t,

$$\text{(1.5)} \qquad c_1\|\nabla f(x_t)\|^2 \le -\nabla f(x_t)'s_t, \qquad \|s_t\| \le c_2\|\nabla f(x_t)\|;$$

(c) w_t either is a deterministic error satisfying for some positive scalars p and q, and all t,

$$\text{(1.6)} \qquad \|w_t\| \le \gamma_t\big(q + p\|\nabla f(x_t)\|\big)$$

[*] Received by the editors December 4, 1997; accepted for publication (in revised form) May 24, 1999; published electronically February 29, 2000. This work was supported by NSF under grant DMI-9625489.

http://www.siam.org/journals/siopt/10-3/33106.html

[†] Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA 02139 (dimitrib@mit.edu, jnt@mit.edu).

or is a stochastic error satisfying conditions that are standard in stochastic gradient and stochastic approximation methods.

Our main result is that either $f(x_t) \to -\infty$ or $f(x_t)$ converges to a finite value and $\lim_{t\to\infty} \nabla f(x_t) = 0$ (with probability 1 on the stochastic case).

The method where the errors w_t are deterministic includes as a special case the standard incremental gradient/backpropagation method for neural network training, the convergence of which has been the object of much recent analysis [Luo91], [Gai94], [Gri94], [LuT94], [MaS94], [Man93], [Ber95a] (see [BeT96] for our discussion of incremental gradient methods and their application to neural network training). The method where the errors w_t are stochastic includes as a special case the classical Robbins–Monro/stochastic gradient method, as well as methods involving scaling of the gradient and satisfying the pseudogradient condition of Poljak and Tsypkin [PoT73]; see section 4 for a precise statement of our assumptions. Basically, the entire spectrum of unconstrained gradient methods is considered, with the only restriction being the diminishing stepsize condition (1.4) (which is essential for convergence in gradient methods with errors) and the attendant Lipschitz condition (1.2) (which is necessary for showing any kind of convergence result under the stepsize condition (1.4)).

To place our analysis in perspective, we review the related results of the literature for gradient-like methods with errors and in the absence of convexity. Our results relate to two types of analysis:

(1) Results that are based on some type of deterministic or stochastic descent argument, such as the use of a Lyapunov function or a supermartingale convergence theorem. All of the results of this type known to us assume that f is bounded below and in some cases require a boundedness assumption on the sequence $\{x_t\}$ or show only that $\liminf_{t\to\infty} \|\nabla f(x_t)\| = 0$. By contrast, we show that $\lim_{t\to\infty} \|\nabla f(x_t)\| = 0$ and we also deal with the case where f is unbounded below and $\{x_t\}$ is unbounded. In fact, a principal aim of our work has been to avoid any type of boundedness assumption. For example, the classical analysis of Poljak and Tsypkin [PoT73], under essentially the same conditions as ours, shows that if f is bounded below, then $f(x_t)$ converges and $\liminf_{t\to\infty} \|\nabla f(x_t)\| = 0$ (see Poljak [Pol87, p. 51]). The analysis of Gaivoronski [Gai94], for stochastic gradient and incremental gradient methods, under similar conditions to ours shows that $\lim_{t\to\infty} \|\nabla f(x_t)\| = 0$, but it also assumes that $f(x)$ is bounded below and that $\|\nabla f(x)\|$ is bounded over $\Re^n$. The analysis of Luo and Tseng [LuT94] for the incremental gradient method shows that $\lim_{t\to\infty} \|\nabla f(x_t)\| = 0$, but it also assumes that $f(x)$ is bounded below, and it makes some additional assumptions on the stepsize γ_t. The analyses by Grippo [Gri94] and by Mangasarian and Solodov [MaS94] for the incremental gradient method (with and without a momentum term) make assumptions that are different from ours and include boundedness of the generated sequence x_t. The analysis of Walk [Wal92, p. 2] (see also Pflug [Pfl96, p. 282]) shows that $\lim_{t\to\infty} \|\nabla f(x_t)\| = 0$, assuming that $s_t = -\nabla f(x_t)$, that w_t is deterministic and satisfies somewhat different conditions than ours, and that f is bounded below. Our method of proof for the case of deterministic errors is similar to the method of Walk. (The assumption that f is bounded below is not critical for Walk's analysis.) However, in the case of stochastic errors, standard stochastic descent proofs rely critically on the boundedness of f from below, and we have used a new line of proof for our result (see the discussion in section 4).

(2) Results based on the so-called ODE analysis [Lju77], [KuC78], [BMP90], [KuY97] that relate the evolution of the algorithm to the trajectories of a differ-

ential equation $dx/dt = h(x)$. For example, if we are dealing with the stochastic steepest descent method $x_{t+1} = x_t - \gamma_t(\nabla f(x_t) - w_t)$, the corresponding ODE is $dx/dt = -\nabla f(x)$. This framework typically involves an explicit or implicit assumption that the average direction of update $h(x)$ is a well-defined function of the current iterate x. It cannot be applied, for example, to a gradient method with diagonal scaling, where the scaling may depend in a complicated way on the past history of the algorithm, unless one works with differential inclusions—rather than differential equations—for which not many results are available. For another example, an asynchronous gradient iteration that updates a single component at a time (selected by some arbitrary or hard-to-model mechanism) does not lead to a well-defined average direction of update $h(x)$, unless one makes some very special assumptions, e.g., the stepsize assumptions of Borkar [Bor95]. In addition to the above described difficulty, the ODE approach relies on the assumption that the sequence of iterates x_t is bounded or recurrent, something that must be independently verified. Let us also mention the following more recent results by Delyon [Del96], which have some similarities with ours: they are proved using a potential function argument and can establish the convergence of $\nabla f(x_t)$ to zero. Similar to the ODE approach, these results assume a well-defined average update direction $h(x)$ and are based on boundedness or recurrence assumptions.

The paper is organized as follows. In the next section, we focus on the method where there is a nonrandom error w_t satisfying the condition (1.6). The convergence result obtained is then applied in section 3 to the case of incremental gradient methods for minimizing the sum of a large number of functions. In section 4, we focus on stochastic gradient methods. Finally, in section 5, a stochastic version of the incremental gradient method is discussed.

2. Deterministic gradient methods with errors. Throughout the paper, we focus on the unconstrained minimization of a continuously differentiable function $f : \Re^n \mapsto \Re$, satisfying for some constant L

$$\|\nabla f(x) - \nabla f(\overline{x})\| \leq L\|x - \overline{x}\| \qquad \forall\, x, \overline{x} \in \Re^n. \tag{2.1}$$

As mentioned in the preceding section, the line of proof of the following proposition is known, although some of our assumptions differ slightly from those in the literature. We will need the following known lemma, which we prove for completeness.

LEMMA 1. *Let Y_t, W_t, and Z_t be three sequences such that W_t is nonnegative for all t. Assume that*

$$Y_{t+1} \leq Y_t - W_t + Z_t, \qquad t = 0, 1, \ldots,$$

and that the series $\sum_{t=0}^{T} Z_t$ converges as $T \to \infty$. Then either $Y_t \to -\infty$ or else Y_t converges to a finite value and $\sum_{t=0}^{\infty} W_t < \infty$.

Proof. Let $\bar{t}$ be any nonnegative integer. By adding the relation $Y_{t+1} \leq Y_t + Z_t$ over all $t \geq \bar{t}$ and by taking the limit superior as $t \to \infty$, we obtain

$$\limsup_{t\to\infty} Y_t \leq Y_{\bar{t}} + \sum_{t=\bar{t}}^{\infty} Z_t < \infty.$$

By taking the limit inferior of the right-hand side as $\bar{t} \to \infty$ and using the fact $\lim_{\bar{t}\to\infty} \sum_{t=\bar{t}}^{\infty} Z_t = 0$, we obtain

$$\limsup_{t\to\infty} Y_t \leq \liminf_{\bar{t}\to\infty} Y_{\bar{t}} < \infty.$$

This implies that either $Y_t \to -\infty$ or else Y_t converges to a finite value. In the latter case, by adding the relation $Y_{i+1} \le Y_i - W_i + Z_i$ from $i = 0$ to $i = t$, we obtain

$$\sum_{i=0}^{t} W_i \le Y_0 + \sum_{i=0}^{t} Z_i - Y_{t+1}, \qquad t = 0, 1, \ldots,$$

which implies that $\sum_{i=0}^{\infty} W_i \le Y_0 + \sum_{i=0}^{\infty} Z_i - \lim_{t\to\infty} Y_t < \infty$. □

We have the following result.

PROPOSITION 1. *Let x_t be a sequence generated by the method*

$$x_{t+1} = x_t + \gamma_t(s_t + w_t),$$

where s_t is a descent direction satisfying for some positive scalars c_1 and c_2, and all t,

$$c_1\|\nabla f(x_t)\|^2 \le -\nabla f(x_t)' s_t, \qquad \|s_t\| \le c_2\bigl(1 + \|\nabla f(x_t)\|\bigr), \tag{2.2}$$

and w_t is an error vector satisfying for some positive scalars p and q, and all t,

$$\|w_t\| \le \gamma_t\bigl(q + p\|\nabla f(x_t)\|\bigr). \tag{2.3}$$

Assume that the stepsize γ_t is positive and satisfies

$$\sum_{t=0}^{\infty} \gamma_t = \infty, \qquad \sum_{t=0}^{\infty} \gamma_t^2 < \infty.$$

Then either $f(x_t) \to -\infty$ or else $f(x_t)$ converges to a finite value and $\lim_{t\to\infty} \nabla f(x_t) = 0$. *Furthermore, every limit point of x_t is a stationary point of f.*

Proof. Fix two vectors x and z, let ξ be a scalar parameter, and let $g(\xi) = f(x + \xi z)$. The chain rule yields $(dg/d\xi)(\xi) = z'\nabla f(x + \xi z)$. We have

$$\begin{aligned}
f(x+z) - f(x) &= g(1) - g(0) \\
&= \int_0^1 \frac{dg}{d\xi}(\xi)\, d\xi \\
&= \int_0^1 z'\nabla f(x + \xi z)\, d\xi \\
&\le \int_0^1 z'\nabla f(x)\, d\xi + \left| \int_0^1 z'\bigl(\nabla f(x+\xi z) - \nabla f(x)\bigr)\, d\xi \right| \\
&\le z'\nabla f(x) + \int_0^1 \|z\| \cdot \|\nabla f(x + \xi z) - \nabla f(x)\| d\xi \\
&\le z'\nabla f(x) + \|z\| \int_0^1 L\xi\|z\|\, d\xi \\
&= z'\nabla f(x) + \frac{L}{2}\|z\|^2.
\end{aligned} \tag{2.4}$$

We apply (2.4) with $x = x_t$ and $z = \gamma_t(s_t + w_t)$. We obtain

$$f(x_{t+1}) \le f(x_t) + \gamma_t \nabla f(x_t)'(s_t + w_t) + \frac{\gamma_t^2 L}{2}\|s_t + w_t\|^2.$$

Using our assumptions, we have

$$\begin{aligned}\nabla f(x_t)'(s_t + w_t) &\le -c_1\|\nabla f(x_t)\|^2 + \|\nabla f(x_t)\|\,\|w_t\| \\ &\le -c_1\|\nabla f(x_t)\|^2 + \gamma_t q\|\nabla f(x_t)\| + \gamma_t p\|\nabla f(x_t)\|^2.\end{aligned}$$

Furthermore, using the relations $\|s_t\|^2 \le 2c_2^2\big(1 + \|\nabla f(x_t)\|^2\big)$ and $\|w_t\|^2 \le 2\gamma_t^2\big(q^2 + p^2\|\nabla f(x_t)\|^2\big)$, which follow from (2.2) and (2.3), respectively, we have

$$\begin{aligned}\|s_t + w_t\|^2 &\le 2\|s_t\|^2 + 2\|w_t\|^2 \\ &\le 4c_2^2\big(1 + \|\nabla f(x_t)\|^2\big) + 4\gamma_t^2\big(q^2 + p^2\|\nabla f(x_t)\|^2\big).\end{aligned}$$

Combining the above relations, we obtain

$$\begin{aligned}f(x_{t+1}) \le f(x_t) &- \gamma_t(c_1 - \gamma_t p - 2\gamma_t c_2^2 L - 2\gamma_t^3 p^2 L)\|\nabla f(x_t)\|^2 \\ &+ \gamma_t^2 q\|\nabla f(x_t)\| + 2\gamma_t^2 c_2^2 L + 2\gamma_t^4 q^2 L.\end{aligned}$$

Since $\gamma_t \to 0$, we have for some positive constant c and all t sufficiently large

$$f(x_{t+1}) \le f(x_t) - \gamma_t c\|\nabla f(x_t)\|^2 + \gamma_t^2 q\|\nabla f(x_t)\| + 2\gamma_t^2 c_2^2 L + 2\gamma_t^4 q^2 L.$$

Using the inequality $\|\nabla f(x_t)\| \le 1 + \|\nabla f(x_t)\|^2$, the above relation yields for all t

$$f(x_{t+1}) \le f(x_t) - \gamma_t(c - \gamma_t q)\|\nabla f(x_t)\|^2 + \gamma_t^2(q + 2c_2^2 L) + 2\gamma_t^4 q^2 L,$$

which for sufficiently large t can be written as

$$f(x_{t+1}) \le f(x_t) - \gamma_t\beta_1\|\nabla f(x_t)\|^2 + \gamma_t^2\beta_2, \tag{2.5}$$

where β_1 and β_2 are some positive scalars.

By using (2.5), Lemma 1, and the assumption $\sum_{t=0}^{\infty}\gamma_t^2 < \infty$, we see that either $f(x_t) \to -\infty$ or else $f(x_t)$ converges and

$$\sum_{t=0}^{\infty}\gamma_t\|\nabla f(x_t)\|^2 < \infty. \tag{2.6}$$

If there existed an $\epsilon > 0$ and an integer $\bar{t}$ such that $\|\nabla f(x_t)\| \ge \epsilon$ for all $t \ge \bar{t}$, we would have

$$\sum_{t=\bar{t}}^{\infty}\gamma_t\|\nabla f(x_t)\|^2 \ge \epsilon^2\sum_{t=\bar{t}}^{\infty}\gamma_t = \infty,$$

which contradicts (2.6). Therefore, $\liminf_{t\to\infty}\|\nabla f(x_t)\| = 0$.

To show that $\lim_{t\to\infty}\nabla f(x_t) = 0$, assume the contrary; that is, $\limsup_{t\to\infty}\|\nabla f(x_t)\| > 0$. Then there exists an $\epsilon > 0$ such that $\|\nabla f(x_t)\| < \epsilon/2$ for infinitely many t and also $\|\nabla f(x_t)\| > \epsilon$ for infinitely many t. Therefore, there is an infinite subset of integers $\mathcal{T}$ such that for each $t \in \mathcal{T}$, there exists an integer $i(t) > t$ such that

$$\|\nabla f(x_t)\| < \epsilon/2, \qquad \|\nabla f(x_{i(t)})\| > \epsilon,$$

$$\epsilon/2 \le \|\nabla f(x_i)\| \le \epsilon \qquad \text{if } t < i < i(t).$$

Since

$$\begin{aligned}\|\nabla f(x_{t+1})\| - \|\nabla f(x_t)\| &\le \|\nabla f(x_{t+1}) - \nabla f(x_t)\| \\ &\le L\|x_{t+1} - x_t\| \\ &= \gamma_t L\|s_t\| \\ &\le \gamma_t L c_2\big(1 + \|\nabla f(x_t)\|\big),\end{aligned}$$

it follows that for all $t \in \mathcal{T}$ that are sufficiently large so that $\gamma_t L c_2 < \epsilon/4$, we have

$$\epsilon/4 \le \|\nabla f(x_t)\|;$$

otherwise, the condition $\epsilon/2 \le \|\nabla f(x_{t+1})\|$ would be violated. Without loss of generality, we assume that the above relations as well as (2.5) hold for all $t \in \mathcal{T}$.

We have for all $t \in \mathcal{T}$, using the condition $\|s_t\| \le c_2\big(1 + \|\nabla f(x_t)\|\big)$ and the Lipschitz condition (2.1),

$$\begin{aligned}\frac{\epsilon}{2} &\le \|\nabla f(x_{i(t)})\| - \|\nabla f(x_t)\| \\ &\le \|\nabla f(x_{i(t)}) - \nabla f(x_t)\| \\ &\le L\|x_{i(t)} - x_t\| \\ &\le L\sum_{i=t}^{i(t)-1} \gamma_i(\|s_i\| + \|w_i\|) \\ &\le Lc_2 \sum_{i=t}^{i(t)-1} \gamma_i\big(1 + \|\nabla f(x_i)\|\big) + L\sum_{i=t}^{i(t)-1} \gamma_i^2\big(q + p\|\nabla f(x_i)\|\big) \\ &\le Lc_2(1+\epsilon)\sum_{i=t}^{i(t)-1} \gamma_i + L(q + p\epsilon)\sum_{i=t}^{i(t)-1} \gamma_i^2.\end{aligned} \tag{2.7}$$

From this it follows that

$$\frac{1}{2Lc_2(1+\epsilon)} \le \liminf_{t\to\infty} \sum_{i=t}^{i(t)-1} \gamma_i. \tag{2.8}$$

Using (2.5), we see that

$$f\big(x_{i(t)}\big) \le f(x_t) - \beta_1\left(\frac{\epsilon}{4}\right)^2 \sum_{i=t}^{i(t)-1} \gamma_i + \beta_2 \sum_{i=t}^{i(t)-1} \gamma_i^2 \qquad \forall\ t \in \mathcal{T}.$$

Using the convergence of $f(x_t)$ already shown and the assumption $\sum_{t=0}^{\infty} \gamma_t^2 < \infty$, this relation implies that

$$\lim_{t\to\infty,\, t\in\mathcal{T}} \sum_{i=t}^{i(t)-1} \gamma_i = 0$$

and contradicts (2.8).

Finally, if $\overline{x}$ is a limit point of x_t, then $f(x_t)$ converges to the finite value $f(\overline{x})$. Thus we have $\nabla f(x_t) \to 0$, implying that $\nabla f(\overline{x}) = 0$. □

3. Incremental gradient methods. In this section, we apply the results of the preceding section to the case where f has the form

$$f(x) = \sum_{i=1}^{m} f_i(x),$$

where $f_i : \Re^n \mapsto \Re$ is for every i a continuously differentiable function satisfying the Lipschitz condition

$$\|\nabla f_i(x) - \nabla f_i(\overline{x})\| \le L\|x - \overline{x}\| \qquad \forall\ x, \overline{x} \in \Re^n \tag{3.1}$$

for some constant L.

In situations where there are many component functions f_i, it may be attractive to use an incremental method that does not wait to process the entire set of components before updating x; instead, the method cycles through the components in sequence and updates the estimate of x after each component is processed. In particular, given x_t, we may obtain x_{t+1} as

$$x_{t+1} = \psi_m,$$

where ψ_m is obtained at the last step of the algorithm

$$\psi_i = \psi_{i-1} - \gamma_t \nabla f_i(\psi_{i-1}), \qquad i = 1, \dots, m, \tag{3.2}$$

and

$$\psi_0 = x_t. \tag{3.3}$$

This method can be written as

$$x_{t+1} = x_t - \gamma_t \sum_{i=1}^{m} \nabla f_i(\psi_{i-1}). \tag{3.4}$$

It is referred to as the *incremental gradient method*, and it is used extensively in the training of neural networks. It should be compared with the ordinary gradient method, which is

$$x_{t+1} = x_t - \gamma_t \nabla f(x_t) = x_t - \gamma_t \sum_{i=1}^{m} \nabla f_i(x_t). \tag{3.5}$$

Thus, a cycle of the incremental gradient method through the components f_i differs from an ordinary gradient iteration only in that the evaluation of ∇f_i is done at the corresponding current estimates ψ_{i-1} rather than at the estimate x_t available at the start of the cycle. The advantages of incrementalism in enhancing the speed of convergence (at least in the early stages of the method) are well known; see, for example, the discussions in [Ber95a], [Ber95b], [BeT96].

The main idea of the following convergence proof is that the incremental gradient method can be viewed as the regular gradient iteration where the gradient is perturbed by an error term that is proportional to the stepsize. In particular, if we compare the incremental method (3.4) with the ordinary gradient method (3.5), we see that the error term in the gradient direction is bounded by

$$\sum_{i=1}^{m} \big\|\nabla f_i(\psi_{i-1}) - \nabla f_i(x_t)\big\|.$$

In view of our Lipschitz assumption (3.1), this term is bounded by

$$L\sum_{i=1}^{m}\|\psi_{i-1}-x_t\|,$$

which from (3.2) is seen to be proportional to γ_t. (A more precise argument is given below.)

PROPOSITION 2. *Let x_t be a sequence generated by the incremental gradient method* (3.2)–(3.4). *Assume that for some positive constants C and D, and all $i = 1,\dots,m$, we have*

$$\|\nabla f_i(x)\| \le C + D\|\nabla f(x)\| \qquad \forall\ x \in \Re^n. \tag{3.6}$$

Assume also that

$$\sum_{t=0}^{\infty}\gamma_t = \infty, \qquad \sum_{t=0}^{\infty}\gamma_t^2 < \infty.$$

Then either $f(x_t) \to -\infty$ or else $f(x_t)$ converges to a finite value and $\lim_{t\to\infty}\nabla f(x_t) = 0$. *Furthermore, every limit point of x_t is a stationary point of f.*

Proof. We formulate the incremental gradient method as a gradient method with errors that are proportional to the stepsize and then apply Proposition 1. For simplicity we will assume that there are only two functions f_i, that is, $m = 2$. The proof is similar when $m > 2$. We have

$$\psi_1 = x_t - \gamma_t\nabla f_1(x_t),$$

$$x_{t+1} = \psi_1 - \gamma_t\nabla f_2(\psi_1).$$

By adding these two relations, we obtain

$$x_{t+1} = x_t + \gamma_t\big(-\nabla f(x_t) + w_t\big),$$

where

$$w_t = \nabla f_2(x_t) - \nabla f_2(\psi_1).$$

We have

$$\|w_t\| \le L\|x_t - \psi_1\| = \gamma_t L\|\nabla f_1(x_t)\| \le \gamma_t\big(LC + LD\|\nabla f(x_t)\|\big).$$

Thus Proposition 1 applies. ☐

Condition (3.6) is guaranteed to hold if each f_k is of the form

$$f_k(x) = x'Q_kx + g_k'x + h_k,$$

where each Q_k is a positive semidefinite matrix, each g_k is a vector, and each h_k is a scalar. (This is the generic situation encountered in linear least squares problems.) If $\sum_{k=1}^{K} Q_k$ is positive definite, there exists a unique minimum to which the algorithm must converge. In the absence of positive definiteness, we obtain $\nabla f(x_t) \to 0$ if the optimal cost is finite. If, on the other hand, the optimal cost is $-\infty$, it can be shown that $\|\nabla f(x)\| \ge \alpha$ for some $\alpha > 0$ and for all x. This implies that $f(x) \to -\infty$ and that $\|x\| \to \infty$.

4. Stochastic gradient methods. In this section, we study stochastic gradient methods. Our main result is similar to Proposition 1 except that we let the noise term w_t be of a stochastic nature. Once more, we will prove that $f(x_t)$ converges and, if the limit is finite, $\nabla f(x_t)$ converges to 0. We comment on the technical issues that arise in establishing such a result. The sequence $f(x_t)$ can be shown to be approximately a supermartingale. The variance of the underlying noise is allowed to grow with $\|\nabla f(x_t)\|$ and therefore can be unbounded. While such unboundedness has been successfully handled in past works on related methods, new complications arise because no lower bound on $f(x_t)$ is assumed. For that reason, the supermartingale convergence theorem cannot be used in a simple manner. Our approach is to show that whenever $\|\nabla f(x_t)\|$ is large, it remains so for a sufficiently long time interval, guaranteeing a decrease in the value of $f(x_t)$ which is significant and dominates the noise effects.

Proposition 3. *Let x_t be a sequence generated by the method*

$$x_{t+1} = x_t + \gamma_t(s_t + w_t),$$

where γ_t is a deterministic positive stepsize, s_t is a descent direction, and w_t is a random noise term. Let $\mathcal{F}_t$ be an increasing sequence of σ-fields. We assume the following:

(a) *x_t and s_t are $\mathcal{F}_t$-measurable.*

(b) *There exist positive scalars c_1 and c_2 such that*

$$c_1\|\nabla f(x_t)\|^2 \le -\nabla f(x_t)' s_t, \qquad \|s_t\| \le c_2(1 + \|\nabla f(x_t)\|) \qquad \forall\ t. \tag{4.1}$$

(c) *We have, for all t and with probability* 1,

$$E[w_t \mid \mathcal{F}_t] = 0, \tag{4.2}$$

$$E[\|w_t\|^2 \mid \mathcal{F}_t] \le A(1 + \|\nabla f(x_t)\|^2), \tag{4.3}$$

where A is a positive deterministic constant.

(d) *We have*

$$\sum_{t=0}^{\infty} \gamma_t = \infty, \qquad \sum_{t=0}^{\infty} \gamma_t^2 < \infty.$$

Then, either $f(x_t) \to -\infty$ or else $f(x_t)$ converges to a finite value and $\lim_{t\to\infty} \nabla f(x_t) = 0$. *Furthermore, every limit point of x_t is a stationary point of f.*

Remarks. (a) The σ-field $\mathcal{F}_t$ should be interpreted as the history of the algorithm up to time t, just before w_t is generated. In particular, conditioning on $\mathcal{F}_t$ can be thought of as conditioning on $x_0, s_0, w_0, \ldots, x_{t-1}, s_{t-1}, w_{t-1}, x_t, s_t$.

(b) Strictly speaking, the conclusions of the proposition only hold "with probability 1." For simplicity, an explicit statement of this qualification often will be omitted.

(c) Our assumptions on w_t are of the same type as those considered in [PoT73].

Proof of Proposition 3. We apply (2.4) with $x = x_t$ and $z = \gamma_t(s_t + w_t)$. We obtain

$$\begin{aligned} f(x_{t+1}) &\le f(x_t) + \gamma_t \nabla f(x_t)'(s_t + w_t) + \frac{\gamma_t^2 L}{2}\|s_t + w_t\|^2 \\ &\le f(x_t) - \gamma_t c_1 \|\nabla f(x_t)\|^2 + \gamma_t \nabla f(x_t)' w_t + \gamma_t^2 L(\|s_t\|^2 + \|w_t\|^2) \\ &\le f(x_t) - \gamma_t c_1 \|\nabla f(x_t)\|^2 + \gamma_t \nabla f(x_t)' w_t + \gamma_t^2 2Lc_2^2 \\ &\qquad + \gamma_t^2 2Lc_2^2 \|\nabla f(x_t)\|^2 + \gamma_t^2 L\|w_t\|^2 \\ &\le f(x_t) - \gamma_t \frac{c_1}{2}\|\nabla f(x_t)\|^2 + \gamma_t \nabla f(x_t)' w_t + \gamma_t^2 2Lc_2^2 + \gamma_t^2 L \|w_t\|^2, \end{aligned} \tag{4.4}$$

where the last inequality is valid only when t is large enough so that $\gamma_t 2Lc_2^2 \le c_1/2$. Without loss of generality, we will assume that this is the case for all $t \ge 0$.

Let $\delta > 0$ be an arbitrary positive number that will be kept constant until the very end of this proof. Let η be a positive constant defined, in terms of δ, by

$$\eta c_2 \left(\frac{1}{\delta} + 2\right) + \eta = \frac{1}{2L}. \tag{4.5}$$

We will partition the set of all times t (the nonnegative integers) into a set S of times at which $\|\nabla f(x_t)\|$ is "small" and intervals $I_k = \{\tau_k, \tau_k + 1, \ldots, \tau_k'\}$ during which $\|\nabla f(x_t)\|$ stays "large." The definition of the times τ_k and τ_k' is recursive and is initialized by letting $\tau_0' = -1$. We then let, for $k = 1, 2, \ldots$,

$$\tau_k = \min\left\{t > \tau_{k-1}' \;\middle|\; \|\nabla f(x_t)\| \ge \delta\right\}.$$

(We leave τ_k undefined if $\|\nabla f(x_t)\| < \delta$ for all $t > \tau_{k-1}'$.) We also let

$$\tau_k' = \max\Bigg\{t \ge \tau_k \;\Bigg|\; \sum_{i=\tau_k}^{t} \gamma_i \le \eta, \text{ and} \\ \frac{\|\nabla f(x_{\tau_k})\|}{2} \le \|\nabla f(x_r)\| \le 2\|\nabla f(x_{\tau_k})\| \,\forall\, r \text{ with } \tau_k \le r \le t\Bigg\}.$$

We say that the interval I_k is *full* if $\sum_{t=\tau_k}^{\tau_k'+1} \gamma_t > \eta$. Let S be the set of all times that do not belong to any of the intervals I_k.

We define a sequence G_t, used to scale the noise terms w_t, by

$$G_t = \begin{cases} \delta & \text{if } t \in S, \\ \|\nabla f(x_{\tau_k})\| = H_k & \text{if } t \in I_k, \end{cases}$$

where the last equality should be taken as the definition of H_k. In particular, G_t is constant during an interval I_t. Note that $G_t \ge \delta$ for all t.

We now collect a few observations that are direct consequences of our definitions.

(P1) For all $t \in S$, we have $\|\nabla f(x_t)\| < \delta = G_t$.

(P2) For all $t \in I_k$, we have

$$\frac{G_t}{2} = \frac{H_k}{2} \le \|\nabla f(x_t)\| \le 2H_k = 2G_t.$$

Combining this with (P1), we also see that the ratio $\|\nabla f(x_t)\|/G_t$ is bounded above by 2.

(P3) If τ_k is defined and I_k is a full interval, then

$$\frac{\eta}{2} \leq \eta - \gamma_{\tau'_k+1} < \sum_{t=\tau_k}^{\tau'_k} \gamma_t \leq \eta, \tag{4.6}$$

where the leftmost inequality holds when k is large enough so that $\gamma_{\tau'_k+1} \leq \eta/2$. Without loss of generality, we will assume that this condition actually holds for all k.

(P4) The value of G_t is completely determined by $x_0, x_1, \ldots, x_t$ and is therefore $\mathcal{F}_t$-measurable. Similarly, the indicator function

$$\chi_t = \begin{cases} 1 & \text{if } t \in S, \\ 0 & \text{otherwise} \end{cases}$$

is also $\mathcal{F}_t$-measurable.

Lemma 2. *Let r_t be a sequence of random variables with each r_t being $\mathcal{F}_{t+1}$-measurable, and suppose that $E[r_t \mid \mathcal{F}_t] = 0$ and $E[\|r_t\|^2 \mid \mathcal{F}_t] \leq B$, where B is some deterministic constant. Then, the sequences*

$$\sum_{t=0}^{T} \gamma_t r_t \qquad \text{and} \qquad \sum_{t=0}^{T} \gamma_t^2 \|r_t\|^2, \qquad T = 0, 1, \ldots,$$

converge to finite limits (with probability 1*).*

Proof. It is seen that $\sum_{t=0}^{T} \gamma_t r_t$ is a martingale whose variance is bounded by $B \sum_{t=0}^{\infty} \gamma_t^2$. It must therefore converge by the martingale convergence theorem. Furthermore,

$$E\left[\sum_{t=0}^{\infty} \gamma_t^2 \|r_t\|^2\right] \leq B \sum_{t=0}^{\infty} \gamma_t^2 < \infty,$$

which shows that $\sum_{t=0}^{\infty} \gamma_t^2 \|r_t\|^2$ is finite with probability 1. This establishes convergence of the second sequence. □

Using Lemma 2, we obtain the following.

Lemma 3. *The following sequences converge (with probability* 1*):*

(a) $\sum_{t=0}^{T} \chi_t \gamma_t \nabla f(x_t)' w_t$;

(b) $\sum_{t=0}^{T} \gamma_t \dfrac{w_t}{G_t}$;

(c) $\sum_{t=0}^{T} \gamma_t \dfrac{\nabla f(x_t)' w_t}{G_t^2}$;

(d) $\sum_{t=0}^{T} \gamma_t^2 \dfrac{\|w_t\|^2}{G_t^2}$;

(e) $\sum_{t=0}^{T} \gamma_t^2 \chi_t \|w_t\|^2$.

Proof. (a) Let $r_t = \chi_t \nabla f(x_t)' w_t$. Since χ_t and $\nabla f(x_t)$ are $\mathcal{F}_t$-measurable and $E[w_t \mid \mathcal{F}_t] = 0$, we obtain $E[r_t \mid \mathcal{F}_t] = 0$. Whenever $\chi_t = 1$, we have $\|\nabla f(x_t)\| \leq \delta$

and $E[\|w_t\|^2 \mid \mathcal{F}_t] \le A(1+\delta^2)$. It follows easily that $E[|r_t|^2 \mid \mathcal{F}_t]$ is bounded. The result follows from Lemma 2.

(b) Let $r_t = w_t/G_t$. Since G_t is $\mathcal{F}_t$-measurable and $E[w_t \mid \mathcal{F}_t] = 0$, we obtain $E[r_t \mid \mathcal{F}_t] = 0$. Furthermore,

$$E[\|r_t\|^2 \mid \mathcal{F}_t] \le \frac{A(1+\|\nabla f(x_t)\|^2)}{G_t^2}.$$

Since the ratio $\|\nabla f(x_t)\|/G_t$ is bounded above [cf. observation (P2)], Lemma 2 applies and establishes the desired convergence result.

(c) Let $r_t = \nabla f(x_t)' w_t/G_t^2$. Note that

$$\frac{\nabla f(x_t)' w_t}{G_t^2} \le \frac{\|\nabla f(x_t)\| \cdot \|w_t\|}{G_t^2} \le 2\frac{\|w_t\|}{G_t}.$$

The ratio in the left-hand side has bounded conditional second moment, by the same argument as in the proof of part (b). The desired result follows from Lemma 2.

(d) This follows again from Lemma 2. The needed assumptions have already been verified while proving part (b).

(e) This follows from Lemma 2 because $\chi_t w_t$ has bounded conditional second moment, by an argument similar to the one used in the proof of part (a). □

We now assume that we have removed the zero probability set of sample paths for which the series in Lemma 3 does not converge. For the remainder of the proof, we will concentrate on a single sample path outside this zero probability set. Let ϵ be a positive constant that satisfies

$$\epsilon \le \eta, \qquad 2\epsilon + 2L\epsilon \le \frac{c_1\eta}{48}, \qquad 4Lc_2^2\epsilon \le \frac{c_1\delta^2\eta}{48}. \tag{4.7}$$

Let us choose some t_0 after which all of the series in Lemma 3, as well as the series $\sum_{t=0}^{T}\gamma_t^2$, stay within ϵ from their limits.

LEMMA 4. *Let t_0 be as above. If τ_k is defined and is larger than t_0, then the interval I_k is full.*

Proof. Recall that for $t \in I_k = \{\tau_k, \dots, \tau_k'\}$ we have $G_t = H_k = \|\nabla f(x_{\tau_k})\| \ge \delta$ and $\|s_t\| \le c_2(1+\|\nabla f(x_t)\|) \le c_2(1+2H_k)$. Therefore,

$$\begin{aligned}
\|x_{\tau_k'+1} - x_{\tau_k}\| &\le \sum_{t=\tau_k}^{\tau_k'} \gamma_t\|s_t\| + \left\|\sum_{t=\tau_k}^{\tau_k'}\gamma_t w_t\right\| \\
&= \sum_{t=\tau_k}^{\tau_k'} \gamma_t\|s_t\| + H_k\left\|\sum_{t=\tau_k}^{\tau_k'}\gamma_t \frac{w_t}{G_t}\right\| \\
&\le \eta c_2(1+2H_k) + H_k\epsilon \\
&\le \eta c_2 H_k\left(\frac{1}{\delta}+2\right) + \eta H_k \\
&= \frac{H_k}{2L},
\end{aligned}$$

where the last equality follows from our choice of η (cf. (4.5)). Thus,

$$\|\nabla f(x_{\tau_k'+1}) - \nabla f(x_{\tau_k})\| \le L\|x_{\tau_k'+1} - x_{\tau_k}\| \le \frac{H_k}{2} = \frac{\|\nabla f(x_{\tau_k})\|}{2},$$

which implies that

$$\frac{1}{2}\|\nabla f(x_{\tau_k})\| \leq \|\nabla f(x_{\tau_k'+1})\| \leq 2\|\nabla f(x_{\tau_k})\|.$$

If we also had $\sum_{t=\tau_k}^{\tau_k'+1} \gamma_t \leq \eta$, then $\tau_k'+1$ should be an element of I_k, which it isn't. This shows that $\sum_{t=\tau_k}^{\tau_k'+1} \gamma_t > \eta$ and that I_k is a full interval. □

Our next lemma shows that after a certain time, $f(x_t)$ is guaranteed to decrease by at least a constant amount during full intervals.

LEMMA 5. *Let t_0 be the same as earlier. If τ_k is defined and larger than t_0, then*

$$f(x_{\tau_k'+1}) \leq f(x_{\tau_k}) - h,$$

where h is a positive constant that depends only on δ.

Proof. Note that I_k is a full interval by Lemma 4. Using (4.4), we have

$$f(x_{t+1}) - f(x_t) \leq -\gamma_t \frac{c_1}{2}\|\nabla f(x_t)\|^2 + \gamma_t \nabla f(x_t)' w_t + \gamma_t^2 2Lc_2^2 + \gamma_t^2 L\|w_t\|^2.$$

We will sum (from τ_k to τ_k') the terms in the right-hand side of the above inequality and provide suitable upper bounds. Recall that for $t \in I_k$, we have $\|\nabla f(x_t)\| \geq H_k/2$. Thus, also using (4.6),

$$(4.8) \qquad -\sum_{t=\tau_k}^{\tau_k'} \gamma_t \frac{c_1}{2}\|\nabla f(x_t)\|^2 \leq -\frac{c_1 H_k^2}{8}\sum_{t=\tau_k}^{\tau_k'} \gamma_t \leq -\frac{c_1 H_k^2 \eta}{16}.$$

Furthermore,

$$(4.9) \qquad \sum_{t=\tau_k}^{\tau_k'} \gamma_t \nabla f(x_t)' w_t \leq 2H_k^2 \epsilon,$$

which follows from the convergence of the series in Lemma 3(c) and the assumption that after time t_0 the series is within ϵ of its limit. By a similar argument based on Lemma 3(d), we also have

$$(4.10) \qquad L\sum_{t=\tau_k}^{\tau_k'} \gamma_t^2 \|w_t\|^2 \leq 2LH_k^2 \epsilon.$$

Finally,

$$(4.11) \qquad 2Lc_2^2 \sum_{t=\tau_k}^{\tau_k'} \gamma_t^2 \leq 4Lc_2^2 \epsilon.$$

We add (4.8)–(4.11) and obtain

$$\begin{aligned} f(x_{\tau_k'+1}) &\leq f(x_{\tau_k}) - \frac{c_1 \eta H_k^2}{16} + (2\epsilon + 2L\epsilon)H_k^2 + 4Lc_2^2\epsilon \\ &\leq f(x_{\tau_k}) - \frac{2c_1\eta H_k^2}{48} + \frac{c_1 \eta \delta^2}{48} \\ &\leq f(x_{\tau_k}) - \frac{c_1\eta\delta^2}{48}. \end{aligned}$$

The second inequality made use of (4.7); the third made use of $H_k \geq \delta$. □

LEMMA 6. *For almost every sample path, $f(x_t)$ converges to a finite value or to $-\infty$. If $\lim_{t\to\infty} f(x_t) \neq -\infty$, then $\limsup_{t\to\infty} \|\nabla f(x_t)\| \leq \delta$.*

Proof. Suppose that there are only finitely many intervals I_k and, in particular,

$$\limsup_{t\to\infty} \|\nabla f(x_t)\| \leq \delta.$$

Let t^* be some time such that $t \in S$ for all $t \geq t^*$. We then have $\chi_t = 1$ for all $t \geq t^*$. We use (4.4) to obtain

$$\begin{aligned} f(x_{t+1}) &\leq f(x_t) + \gamma_t \chi_t \nabla f(x_t)' w_t + \gamma_t^2 2Lc_2^2 + \chi_t \gamma_t^2 L \|w_t\|^2 \\ &= f(x_t) + Z_t \qquad \text{for } t \geq t^*, \end{aligned}$$

where the last equality can be taken as the definition of Z_t. Using parts (a) and (e) of Lemma 3, the series $\sum_t Z_t$ converges. Lemma 1 then implies that $f(x_t)$ converges to a finite value or to $-\infty$. This proves Lemma 6 for the case where there are finitely many intervals.

We consider next the case where there are infinitely many intervals. We will prove that $f(x_t)$ converges to $-\infty$. We first establish such convergence along a particular subsequence. Let $\mathcal{T} = S \cup \{\tau_1, \tau_2, \ldots\}$. We will show that the sequence $\{f(x_t)\}_{t\in\mathcal{T}}$ converges to $-\infty$. To see why this must be the case, notice that whenever $t \in S$, we have $f(x_{t+1}) \leq f(x_t) + Z_t$, where Z_t is as in the preceding paragraph and is summable. Also, whenever $t \in \mathcal{T}$ but $t \notin S$, then $t = \tau_k$ for some k, and the next element of $\mathcal{T}$ is the time $\tau_k' + 1$. Using Lemma 5, $f(x_t)$ decreases by at least h during this interval (for k large enough). We are now in the situation captured by Lemma 1, with $W_t = h$ whenever $t = \tau_k$. The convergence of the subsequence $\{f(x_t)\}_{t\in\mathcal{T}}$ follows. Furthermore, since $W_t = h$ infinitely often, the limit can be only $-\infty$.

Having shown that $f(x_{\tau_k})$ converges to $-\infty$, it now remains to show that the fluctuations of $f(x_t)$ during intervals I_k cannot be too large. Because the technical steps involved here are very similar to those given earlier, we provide only an outline. In order to carry out this argument, we consider the events that immediately precede an interval I_k.

Let us first consider the case where I_k is preceded by an element of S, i.e., $\tau_k - 1 \in S$. By replicating the first half of the proof of Lemma 4, we can show that $x_t - x_{\tau_k - 1}$ for $t \in I_k$ is bounded by a constant multiple of δ (for k large enough). Since $\|\nabla f(x_{\tau_k - 1})\| \leq \delta$, this leads to a $c\delta^2$ bound on the difference $f(x_t) - f(x_{\tau_k - 1})$, where c is some absolute constant. Since $f(x_{\tau_k - 1}) \to -\infty$, the same must be true for $f(x_t)$, $t \in I_k$.

Let us now consider the case where I_k is immediately preceded by an interval I_{k-1}. By replicating the proof of Lemma 5 (with a somewhat smaller choice of ϵ), we can show that (for k large enough) we will have $f(x_t) \leq f(x_{\tau_{k-1}})$ for all $t \in I_k$. Once more, since $f(x_{\tau_{k-1}})$ converges to $-\infty$, the same must be true for $f(x_t)$, $t \in I_k$. □

According to Lemma 6, $f(x_t)$ converges and if

$$\lim_{t\to\infty} f(x_t) \neq -\infty,$$

then $\limsup_{t\to\infty} \|\nabla f(x_t)\| \leq \delta$. Since this has been proved for an arbitrary $\delta > 0$, we conclude that if $\lim_{t\to\infty} f(x_t) \neq -\infty$, then $\limsup_{t\to\infty} \|\nabla f(x_t)\| = 0$, that is, $\nabla f(x_t) \to 0$.

Finally, if x^* is a limit point of x_t, this implies that $f(x_t)$ has a subsequence that converges to $f(x^*)$. Therefore, the limit of the entire sequence $f(x_t)$, which we have shown to exist, must be finite and equal to $f(x^*)$. We have shown that in this case $\nabla f(x_t)$ converges to zero. By taking the limit of $\nabla f(x_t)$ along a sequence of times such that x_t converges to x^*, we conclude that $\nabla f(x^*) = 0$. $\square$

5. The incremental gradient method revisited. We now provide an alternative view of the incremental gradient method that was discussed in section 4.

Consider again a cost function f of the form

$$f(x) = \frac{1}{m} \sum_{i=1}^{m} f_i(x),$$

where each f_i is a function from $\Re^n$ into $\Re$ that satisfies the Lipschitz condition (4.1). In contrast to the setting of section 4, we now assume that each update is based on a single component function f_i, chosen at random. More specifically, let $k(t)$, $t = 1, 2, \ldots$, be a sequence of independent random variables, each distributed uniformly over the set $\{1, \ldots, m\}$. The algorithm under consideration is

$$x_{t+1} = x_t - \gamma_t \nabla f_{k(t)}(x_t), \tag{5.1}$$

where γ_t is a nonnegative scalar stepsize. We claim that this is a special case of the stochastic gradient algorithm. Indeed, the algorithm (5.1) can be rewritten as

$$x_{t+1} = x_t - \frac{\gamma_t}{m} \sum_{i=1}^{m} \nabla f_i(x_t) - \gamma_t \left(\nabla f_{k(t)}(x_t) - \frac{1}{m} \sum_{i=1}^{m} \nabla f_i(x_t) \right),$$

which is of the form

$$x_{t+1} = x_t - \gamma_t \nabla f(x_t) - \gamma_t w_t,$$

where

$$w_t = \nabla f_{k(t)}(x_t) - \frac{1}{m} \sum_{i=1}^{m} \nabla f_i(x_t).$$

We now verify that w_t satisfies the assumptions of Proposition 3. Due to the way that $k(t)$ is chosen, we have

$$E\big[\nabla f_{k(t)}(x_t) \mid \mathcal{F}_t\big] = \frac{1}{m} \sum_{i=1}^{m} \nabla f_i(x_t),$$

from which it follows that $E[w_t \mid \mathcal{F}_t] = 0$. We also have

$$\begin{aligned} E\big[\|w_t\|^2 \mid \mathcal{F}_t\big] &= E\big[\big\|\nabla f_{k(t)}(r_t)\big\|^2 \mid \mathcal{F}_t\big] - \big\|E\big[\nabla f_{k(t)}(r_t) \mid \mathcal{F}_t\big]\big\|^2 \\ &\leq E\big[\big\|\nabla f_{k(t)}(r_t)\big\|^2 \mid \mathcal{F}_t\big], \end{aligned}$$

which yields

$$E\big[\|w_t\|^2 \mid \mathcal{F}_t\big] \leq \max_k \big\|\nabla f_k(x_t)\big\|^2.$$

Let us assume that there exist constants C and D such that

$$\big\|\nabla f_i(x)\big\| \leq C + D\big\|\nabla f(x)\big\| \qquad \forall\ i,\, x \tag{5.2}$$

(cf. the assumption of Proposition 2). It follows that

$$E\big[\|w_t\|^2 \mid \mathcal{F}_t\big] \le 2C^2 + 2D^2\big\|\nabla f(x_t)\big\|^2$$

so that condition (4.3) is satisfied and the assertion of Proposition 3 holds.

REFERENCES

[BMP90] A. Benveniste, M. Metivier, and P. Priouret, *Adaptive Algorithms and Stochastic Approximations*, Springer-Verlag, New York, 1990.

[BeT89] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice–Hall, Englewood Cliffs, NJ, 1989.

[BeT96] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, MA, 1996.

[Ber95a] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[Ber95b] D. P. Bertsekas, *A new class of incremental gradient methods for least squares problems*, SIAM J. Optim., 7 (1997), pp. 913–926.

[Bor95] V. S. Borkar, *Asynchronous stochastic approximations*, SIAM J. Control Optim., 36 (1998), pp. 840–851.

[Del96] B. Delyon, *General results on the convergence of stochastic algorithms*, IEEE Trans. Automat. Control, 41 (1996), pp. 1245–1255.

[Gai94] A. A. Gaivoronski, *Convergence analysis of parallel backpropagation algorithm for neural networks*, Optim. Methods Software, 4 (1994), pp. 117–134.

[Gri94] L. Grippo, *A class of unconstrained minimization methods for neural network training*, Optim. Methods Software, 4 (1994), pp. 135–150.

[KuC78] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Springer-Verlag, New York, 1978.

[KuY97] H. J. Kushner and G. Yin, *Stochastic Approximation Methods*, Springer-Verlag, New York, 1996.

[Lju77] L. Ljung, *Analysis of recursive stochastic algorithms*, IEEE Trans. Automat. Control, 22 (1977), pp. 551–575.

[LuT94] Z. Q. Luo and P. Tseng, *Analysis of an approximate gradient projection method with applications to the backpropagation algorithm*, Optim. Methods Software, 4 (1994), pp. 85–101.

[Luo91] Z. Q. Luo, *On the convergence of the LMS algorithm with adaptive learning rate for linear feedforward networks*, Neural Comput., 3 (1991), pp. 226–245.

[MaS94] O. L. Mangasarian and M. V. Solodov, *Serial and parallel backpropagation convergence via nonmonotone perturbed minimization*, Optim. Methods Software, 4 (1994), pp. 103–116.

[Pfl96] G. Pflug, *Optimization of Stochastic Models. The Interface Between Simulation and Optimization*, Kluwer, Boston, 1996.

[PoT73] B. T. Poljak and Y. Z. Tsypkin, *Pseudogradient adaptation and training algorithms*, Automat. Remote Control, 12 (1973), pp. 83–94.

[Pol87] B. T. Poljak, *Introduction to Optimization*, Optimization Software Inc., New York, 1987.

[TBA86] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, *Distributed asynchronous deterministic and stochastic gradient optimization algorithms*, IEEE Trans. Automat. Control, 31 (1986), pp. 803–812.

[Wal92] H. Walk, *Foundations of stochastic approximation*, in Stochastic Approximation and Optimization of Random Systems, L. Ljung, G. Pflug, and H. Walk, eds., Birkhauser, Boston, 1992, pp. 53–93.

SIAM J. Optim.
Vol. 10, No. 3, pp. 643–657

INEXACT CUTS IN BENDERS DECOMPOSITION[*]

GOLBON ZAKERI[†], ANDREW B. PHILPOTT[‡], and DAVID M. RYAN[‡]

Abstract. Benders decomposition is a well-known technique for solving large linear programs with a special structure. In particular, it is a popular technique for solving multistage stochastic linear programming problems. Early termination in the subproblems generated during Benders decomposition (assuming dual feasibility) produces valid cuts that are inexact in the sense that they are not as constraining as cuts derived from an exact solution. We describe an inexact cut algorithm, prove its convergence under easily verifiable assumptions, and discuss a corresponding Dantzig–Wolfe decomposition algorithm. The paper is concluded with some computational results from applying the algorithm to a class of stochastic programming problems that arise in hydroelectric scheduling.

Key words. stochastic programming, Benders decomposition, inexact cuts

AMS subject classifications. 90C15, 90C05, 90C06, 90C90

PII. S1052623497318700

1. Introduction.

Many large linear programming problems exhibit a block-diagonal structure that makes them amenable to decomposition techniques such as Dantzig–Wolfe decomposition [5, 6] or its dual, Benders decomposition [3]. The latter technique has become increasingly popular in stochastic linear programming, starting with the independent publication of the *L-shaped method* by Van Slyke and Wets [17] for two-stage stochastic linear programming. (The L-shaped method is often referred to as *stochastic Benders decomposition*.)

In this paper we are concerned with Benders decomposition applied to linear programs of the form

$$\begin{aligned} \text{P:}\quad & \text{minimize} && c^T x + q^T y \\ & \text{subject to} && Ax = b, \\ & && Tx + Wy = h, \\ & && x \geq 0, \quad y \geq 0. \end{aligned}$$

If we define

$$\mathcal{Q}(x) = \min\{q^T y \mid Wy = h - Tx,\ y \geq 0\},$$

then P can be written as

$$\begin{aligned} \text{P:}\quad & \text{minimize} && c^T x + \mathcal{Q}(x) \\ & \text{subject to} && Ax = b, \\ & && x \geq 0. \end{aligned}$$

Throughout this paper we assume that $X = \{x \geq 0 \mid Ax = b\}$ is contained in $\operatorname{dom} \mathcal{Q} = \{x \mid \mathcal{Q}(x) < \infty\}$. Under this assumption the Benders decomposition algorithm can be defined as follows.

[*] Received by the editors March 25, 1997; accepted for publication (in revised form) January 10, 1999; published electronically February 29, 2000. This work was supported by the New Zealand Public Good Science Fund, FRST contract 403.

http://www.siam.org/journals/siopt/10-3/31870.html

[†] Mathematical Sciences Division, Argonne National Lab, Argonne, IL 60439 (zakeri@mcs.anl.gov).

[‡] Operations Research Group, Department of Engineering Science, University of Auckland, Private Bag 92019, Auckland, New Zealand (a.philpott@auckland.ac.nz, d.ryan@auckland.ac.nz).

Benders decomposition algorithm.

```
Set $i := 0$, $U_0 := \infty$, $L_0 := -\infty$, $F := \mathbb{R}^n \times [L_0, \infty)$.
While $U_i - L_i > 0$
    (1) Set $i := i + 1$.
    (2) Solve the master problem
            MP:  minimize    $c^T x + \theta$
                 subject to  $Ax = b$,
                             $(x, \theta) \in F$,
                             $x \geq 0$
        to obtain optimal primal variables $(x_i, \theta_i)$.
    (3) Set $L_i := c^T x_i + \theta_i$.
    (4) Solve the subproblem
            SP($x_i$):  minimize    $q^T y$
                        subject to  $Wy = h - Tx_i$,
                                    $y \geq 0$
        to obtain optimal primal variables $y_i$ and dual variables $\pi_i$.
    (5) Set $U_i := \min\{U_{i-1}, c^T x_i + q^T y_i\}$.
    (6) Set $F := F \cap \{(x, \theta) \mid \pi_i^T (h - Tx) \leq \theta\}$.
```

In the classical case the cut defined by step 6 comes from an optimal basic feasible solution to the subproblem. Since there is a finite number of basis matrices for this problem, finite termination of the algorithm at the optimal solution can be guaranteed (see, e.g., [13]).

In this paper we explore the Benders decomposition algorithm in the case where the cuts are not computed from an optimal extreme-point solution to a linear programming subproblem. For example, when the subproblems are very large, it makes sense to determine the cuts by applying a primal-dual interior-point method to the subproblem. Terminating this procedure when it yields a feasible dual solution will still define a valid cut. We call this an *inexact* cut. If the dual solution is close to optimal, then an inexact cut will also separate the optimal solution from the current iterate (except when this is optimal). As observed by a number of authors (see, e.g., [2]), inexact cuts may be less effort to compute than the exact cuts, especially for linear programming algorithms that yield an approximately optimal dual feasible solution before termination.

In theoretical terms, Benders decomposition is a special case of a more general class of convex cutting plane algorithms first introduced by Kelley [14]. Cutting plane algorithms construct a sequence of hyperplanes that separate the current iterate from the optimal solution. In the case where the cutting planes are computed inexactly, the asymptotic convergence of this process to the optimal solution has been investigated by a number of authors [1, 7, 8, 11, 14]. In the context of Benders decomposition applied to linear programs of the form P, all of the convergence results in these papers assume that the sets containing x and $\pi^T T$ are both bounded. In the convergence theorem that we prove for inexact cuts, we require that $X = \{x \geq 0 \mid Ax = b\}$ be bounded and that $X \subseteq \operatorname{dom} \mathcal{Q}$. The latter assumption, which is known as relatively complete recourse in stochastic programming, is weaker than requiring that $\pi^T T$ be bounded.

To avoid possible confusion, we remark that our use of the term *inexact* is less general here than that of Au, Higle, and Sen [1]. At each iteration i of their inexact subgradient algorithm (applied to minimize a general objective function $f(x)$), they construct an approximate subgradient at the current point x_i by computing a subgradient to an approximating function f_i and taking a projected step from x_i in (the negative of) that direction. With certain restrictions on the convergence of $\{f_i\}$ to f they prove convergence of x_i to the minimum of f under the assumption that the subgradients of f_i at x_i form a bounded sequence. Our results are confined to Benders decomposition (where $f(x) = c^T x + \mathcal{Q}(x)$ and each f_i is defined by the inexact cuts at iteration i), but we do not require that the subgradients of f_i at each iterate (namely, $c - \pi_i^T T$ in our special case) form a bounded sequence.

In the next section we describe a Benders decomposition algorithm that terminates the solution of the subproblem before optimality to produce an inexact cut. The steps of the algorithm ensure that this cut separates the optimal solution from the current iterate. In section 3 we consider the convergence of the inexact cut algorithm under the above assumptions, and in section 4 we discuss the implications of our results for Dantzig–Wolfe decomposition. In section 5 we give some computational results.

2. The algorithm.

We start the inexact cut algorithm by choosing a convergence tolerance δ, setting an iteration counter $i := 0$, and choosing some decreasing sequence $\{\epsilon_i\}$ that converges to 0. We also set $U_0 := \infty$ and $L_0 := -\infty$. The remaining steps of the algorithm are as follows.

Inexact cut algorithm.

```
While $U_i - L_i > \delta$
    (1) Set $i := i + 1$.
    (2) Solve MP to obtain $(x_i, \theta_i)$.
    (3) Set $L_i := c^T x_i + \theta_i$.
    (4) Perform an inexact optimization to generate a vector $\pi_i$ feasible for the dual
        of SP$(x_i)$ such that
        (1)        $\pi_i^T(h - Tx_i) + \epsilon_i > \mathcal{Q}(x_i)$.
    (5) Set $U_i := \min\{U_{i-1}, c^T x_i + \pi_i^T(h - Tx_i) + \epsilon_i\}$.
    (6) If $\pi_i^T(h - Tx_i) > \theta_i$, then add the cut $\pi_i^T(h - Tx) \le \theta$ to MP,
        else set $i := i + 1$, $x_{i+1} := x_i$, $\theta_{i+1} := \theta_i$, $L_{i+1} := L_i$, $U_{i+1} := U_i$ and go to
        step 4.[1]
```

We denote by v_i the value of the inexact optimization in step 4. Thus $v_i = \pi_i^T(h - Tx_i)$. In step 6 of each iteration of this method we check to see if $v_i > \theta_i$, which ensures that the hyperplane $\pi_i^T(h - Tx) = \theta$ will strictly separate the current iterate (x_i, θ_i) from any optimal solution of P. If this check fails, then we decrease the duality gap tolerance ϵ and continue with the solution of SP(x_i), until either $\epsilon_i \to 0$ with no change in (x_i, θ_i) or (x_i, θ_i) is separated from an optimal solution of P by a cut.

To show that this algorithm converges we make use of the following simple results.

Lemma 2.1. *$-\pi_i^T T$ is an ϵ_i-subgradient of $\mathcal{Q}$ at x_i.*

Proof. Since π_i is dual feasible for SP(x_i), it is dual feasible for every possible subproblem SP(x). Hence for every x of suitable dimension, we have

$$\mathcal{Q}(x) \ge \pi_i^T(h - Tx),$$

[1] Note that in this case x and θ remain fixed and only (possibly) ϵ changes.

and by (1)

$$\mathcal{Q}(x_i) \le v_i + \epsilon_i.$$

Thus

$$\mathcal{Q}(x) - \mathcal{Q}(x_i) \ge \pi_i^T(h - Tx) - \pi_i^T(h - Tx_i) - \epsilon_i,$$

giving

$$\mathcal{Q}(x) \ge \mathcal{Q}(x_i) - \pi_i^T T(x - x_i) - \epsilon_i,$$

which gives the result. □

LEMMA 2.2. *Let U_i, L_i, x_i, and θ_i be generated by applying the inexact cut algorithm with ϵ_i. Then*

$$0 \le U_i - L_i \le v_i + \epsilon_i - \theta_i.$$

Proof. Since U_i is an upper bound on the value of P and L_i is a lower bound, $U_i - L_i \ge 0$. Moreover, since

$$U_i \le c^T x_i + v_i + \epsilon_i$$

and $L_i = c^T x_i + \theta_i$, we have

$$0 \le U_i - L_i \le c^T x_i + v_i + \epsilon_i - c^T x_i - \theta_i$$

and the result follows. □

3. Convergence of the algorithm. In this section we prove that the sequence $\{(x_i, \theta_i)\}$ generated by the inexact cut algorithm converges to an optimal solution to P. As alluded to above, abstract proofs of convergence for cutting plane methods (see [14]) typically invoke a compactness argument that in our context relies on an assumption that the sets containing x and $\pi^T T$ are both bounded. A general convergence theory that might avoid these assumptions is developed in Higle and Sen [10], who prove several convergence results for algorithms similar to the inexact cut algorithm. Unfortunately, the direct application of these results to our algorithm is not straightforward.

The difficulty with applying the results in [10] lies in demonstrating the key assumption that the sequence $\{\max_{j<i}(\pi_j^T(h - Tx_i))\}$ converges to $\mathcal{Q}(\bar{x})$ whenever $\{x_i\}$ converges to $\bar{x}$. (Observe that $\epsilon_i \to 0$ implies $\{\max_{j<i}(\pi_j^T(h-Tx_{i-1}))\}$ converges to $\mathcal{Q}(\bar{x})$ when $\{x_i\}$ converges to $\bar{x}$, but this is different from the above assertion, since x_{i-1} is not the minimizer of $\max_{j<i}(\pi_j^T(h - Tx))$.) To deal with situations like this (for a slightly different class of algorithms from ours) [10, Theorem 9] relaxes the assumption to

$$\lim_{i\in\mathcal{K}}\left\{\max_{j<i}(\pi_j^T(h - Tx_i)) - \max_{j<i}(\pi_j^T(h - Tx_{i-1}))\right\} = 0,$$

where $\mathcal{K}$ is some infinite index set. This can be shown to be equivalent to our equation (6) below. (Lemma 3.9 shows that this equation holds for our algorithm.) As [10, Theorem 9] is not directly applicable, we present a self-contained proof of our convergence result.

To illuminate the role that the boundedness of $\{\pi^T T\}$ plays in the proof we begin by showing that the sequence $\{\pi_i^T T\}$ generated by the inexact cut algorithm is bounded provided that the set $X = \{x \geq 0 \mid Ax = b\}$ is bounded, and $\operatorname{dom} \mathcal{Q}$ is $\mathbb{R}^n$. (In stochastic programming the latter is known as complete recourse.) In what follows we shall relax the latter assumption to $X \subseteq \operatorname{dom} \mathcal{Q}$, for which we may still prove convergence although we are no longer guaranteed a bound on $\{\pi^T T\}$. We make use of the following technical result.

LEMMA 3.1. *If for some given pair (b, β) the epigraph of*

$$f(x) = \max_{1 \leq k \leq N} \{b_k^T x + \beta_k\}$$

lies in the half-space $H = \{(x, \mu) \mid \mu \geq b^T x + \beta\}$, then $\|b\| \leq \max_{1 \leq k \leq N} \|b_k\|$.

Proof. Suppose $\|b\| > \max_{1 \leq k \leq N} \|b_k\| = M$ and let $\tilde{\beta} = \max_{1 \leq k \leq N} |\beta_k|$. Let $n > \frac{|\tilde{\beta} - \beta|}{\|b\|^2 - M\|b\|}$ and define $z := nb$. We will show that $f(z) < b^T z + \beta$, contradicting the hypothesis. Formally,

$$\begin{aligned} b^T z + \beta = n\|b\|^2 + \beta &> n\|b\|M + \beta + |\tilde{\beta} - \beta| \\ &\geq nM\|b\| + \tilde{\beta} \\ &\geq \max_{1 \leq k \leq N} [(nb^T) b_k] + \tilde{\beta} \\ &= \max_{1 \leq k \leq N} [(nb^T) b_k] + \max_{1 \leq k \leq N} |\beta_k| \\ &\geq \max_{1 \leq k \leq N} [(nb^T) b_k + \beta_k] \\ &= f(z). \end{aligned}$$

This contradicts the assumption that the epigraph of $\max_{1 \leq k \leq N} \{b_k^T x + \beta_k\}$ lies in H. Hence we must have $\|b\| \leq M$ as required. □

LEMMA 3.2. *If $\operatorname{dom} \mathcal{Q} = \mathbb{R}^n$, then the sequence $\{-\pi_i^T T\}$ is bounded.*

Proof. Let $\{\hat{\pi}_k \mid 1 \leq k \leq N\}$ be the set of basic feasible solutions of $W^T \pi \leq q$. Recall that for every $x \in \mathbb{R}^n$, π_i is dual feasible for SP(x). So for every such x we have

$$\pi_i^T (h - Tx) \leq \max_{1 \leq k \leq N} \hat{\pi}_k^T (h - Tx) = \mathcal{Q}(x),$$

where the equation follows by virtue of $\operatorname{dom} \mathcal{Q} = \mathbb{R}^n$. Therefore the epigraph of $\mathcal{Q}$ lies in the half-space

$$H = \{(x, \mu) \mid \mu \geq b^T x + \beta\},$$

where $b = -\pi_i^T T$ and $\beta = \pi_i^T h$. The conclusion is then immediate from Lemma 3.1. □

Next we will show that the inexact cut algorithm terminates in a finite number of iterations with a δ-optimal solution. If the inexact cut algorithm does not terminate in a finite number of iterations, then it will produce an infinite sequence $\{(x_i, \theta_i)\}$ that satisfies one of the following conditions:

(1) There exists m such that $\theta_i \geq v_i$ for all $i \geq m$.

(2) There exists a subsequence $\{(x_{\sigma(i)}, \theta_{\sigma(i)})\}$ such that $\theta_{\sigma(i)} < v_{\sigma(i)}$.

The following lemmas show that a contradiction results in either case, namely, the algorithm eventually yields a δ-optimal solution.

LEMMA 3.3. *If there exists m such that $\theta_i \geq v_i$ for all $i \geq m$, then $U_i - L_i \downarrow 0$.*

Proof. Since $\theta_i \geq v_i$, Lemma 2.2 implies

$$0 \leq U_i - L_i \leq v_i + \epsilon_i - \theta_i \leq \epsilon_i.$$

The result follows since $\epsilon_i \to 0$. □

LEMMA 3.4. *If there exists a convergent subsequence $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ such that $\theta_{\tau(i)} < v_{\tau(i)}$, then*

(1) $0 < v_{\tau(i)} - \theta_{\tau(i)} \leq v_{\tau(i)} - v_{\tau(i-1)} + \pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)})$;

(2) $\lim v_{\tau(i)} - v_{\tau(i-1)} = 0$;

(3) $\liminf \pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)}) \geq 0$.

Proof. It is clear that $0 < v_{\tau(i)} - \theta_{\tau(i)}$ from the assumption. To obtain the second inequality, observe that $(x_{\tau(i)}, \theta_{\tau(i)})$ is constrained to satisfy the cut we added at iteration $\tau(i-1)$. Therefore

$$\theta_{\tau(i)} \geq \pi^T_{\tau(i-1)}(h - Tx_{\tau(i)}),$$

which implies

$$\begin{aligned} v_{\tau(i)} - \theta_{\tau(i)} &\leq v_{\tau(i)} - \pi^T_{\tau(i-1)}(h - Tx_{\tau(i)}) \\ &= v_{\tau(i)} - v_{\tau(i-1)} + \pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)}). \end{aligned} \tag{2}$$

Now $(x_{\tau(i)}, \theta_{\tau(i)}) \to (x^*, \theta^*)$ by assumption. Furthermore, from the algorithm we have

$$\mathcal{Q}(x_{\tau(i)}) - \epsilon_{\tau(i)} \leq v_{\tau(i)} \leq \mathcal{Q}(x_{\tau(i)})$$

and therefore

$$v_{\tau(i)} \to \mathcal{Q}(x^*), \tag{3}$$

which implies $\lim v_{\tau(i)} - v_{\tau(i-1)} = 0$. Furthermore, (2) and (3) imply

$$\liminf \pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)}) \geq 0. \qquad \square$$

LEMMA 3.5. *Suppose $X = \{x \geq 0 \mid Ax = b\}$ is bounded and* $\operatorname{dom} \mathcal{Q}$ *is $\mathbb{R}^n$. If there exists a subsequence $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ such that $\theta_{\tau(i)} < v_{\tau(i)}$, then $U_i - L_i \downarrow 0$.*

Proof. The subsequence $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ is bounded since X is bounded. Thus we may assume, by extracting a further subsequence if necessary, that $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ is convergent to (x^*, θ^*), say. We proceed to show that $U_{\tau(i)} - L_{\tau(i)}$ converges to zero, which implies the result. By Lemma 2.2 we have that

$$0 \leq U_{\tau(i)} - L_{\tau(i)} \leq v_{\tau(i)} + \epsilon_{\tau(i)} - \theta_{\tau(i)},$$

so if we let

$$V_{\tau(i)} = v_{\tau(i)} + \epsilon_{\tau(i)} - \theta_{\tau(i)},$$

then by Lemma 3.4

$$0 < V_{\tau(i)} \leq v_{\tau(i)} - v_{\tau(i-1)} + \pi_{\tau(i-1)}^T T(x_{\tau(i)} - x_{\tau(i-1)}) + \epsilon_{\tau(i)} \tag{4}$$

and

$$\lim v_{\tau(i)} - v_{\tau(i-1)} = 0. \tag{5}$$

Furthermore, since $\operatorname{dom} \mathcal{Q} = \mathbb{R}^n$, by Lemma 3.2, $\pi_{\tau(i-1)}^T T$ is bounded, and so

$$\pi_{\tau(i-1)}^T T(x_{\tau(i)} - x_{\tau(i-1)}) \to 0.$$

Substituting into (4) and taking the limit as $\tau(i) \to \infty$ yields $V_{\tau(i)} \to 0$. Since $U_{\tau(i)} - L_{\tau(i)}$ is bounded above by $V_{\tau(i)}$ and below by 0, it must converge to 0. Now by their definitions, $\{U_i\}$ is decreasing and $\{L_i\}$ is increasing. Hence $\{U_i - L_i\}$ is decreasing, and since a subsequence of this sequence converges, it follows that the whole sequence converges. □

THEOREM 3.6. *If* $\{x \geq 0 \mid Ax = b\}$ *is bounded and* $\operatorname{dom} \mathcal{Q} = \mathbb{R}^n$, *the inexact cut algorithm terminates in a finite number of iterations with a* δ*-optimal solution of* P.

Proof. From Lemma 3.3 and Lemma 3.5 we have that $U_i - L_i \downarrow 0$. Therefore there exists some I such that $U_I - L_I < \delta$, so the algorithm terminates in at most I iterations. Let x_k be such that $U_I = c^T x_k + v_k + \epsilon_k$. Then

$$c^T x_k + \mathcal{Q}(x_k) \leq c^T x_k + v_k + \epsilon_k < L_I + \delta,$$

and so $c^T x_k + \mathcal{Q}(x_k)$ is within δ of the optimum. □

We shall now consider relaxing the assumption that $\operatorname{dom} \mathcal{Q} = \mathbb{R}^n$ to $X \subseteq \operatorname{dom} \mathcal{Q}$. (We retain the assumption that X is bounded.) In this case we are no longer guaranteed that $\{-\pi_i^T T\}$ is a bounded sequence, since $\{x_i\}$ could lie on the boundary of the domain of $\mathcal{Q}$. At such points it is possible to have unbounded ϵ_i-subgradients. Since Lemma 3.4 remains valid without our assumption, in what follows we confine our attention to the term

$$\pi_{\tau(i-1)}^T T(x_{\tau(i)} - x_{\tau(i-1)})$$

and demonstrate that for some subsequence $\{x_{\sigma(i)}\}$ of $\{x_{\tau(i)}\}$,

$$\lim_{i\to\infty} -\pi_{\sigma(i-1)}^T T(x_{\sigma(i)} - x_{\sigma(i-1)}) = 0. \tag{6}$$

We do this by showing in Lemma 3.9 that for some subsequence $\{x_{\sigma(i)}\}$ of $\{x_{\tau(i)}\}$,

$$\liminf -\pi_{\sigma(i-1)}^T T(x_{\sigma(i)} - x_{\sigma(i-1)}) \geq 0. \tag{7}$$

Since by virtue of Lemma 3.4

$$\liminf \pi_{\sigma(i-1)}^T T(x_{\sigma(i)} - x_{\sigma(i-1)}) \geq 0,$$

we get

$$\limsup -\pi_{\sigma(i-1)}^T T(x_{\sigma(i)} - x_{\sigma(i-1)}) \leq 0,$$

which with (7) yields (6).

The proof of Lemma 3.9 uses a subsequence of $\{x_i\}$ lying in the relative interior of a face of X. Each face of X is a bounded polyhedral set. To derive the inequality (7) we make use of the following two lemmas for polyhedral sets.

LEMMA 3.7. *Let*

$$K = \{x \mid b_j^T x \leq \beta_j,\ 1 \leq j \leq k\}$$

and suppose $b_j^T x^* = \beta_j$, $1 \leq j \leq k$. *Then* $x^* + y \in K$ *implies that* y *is in the recession cone of* K.

Proof. Since $x^* + y \in K$ it follows that for every $j = 1, 2, \ldots, k$, $b_j^T y \leq 0$, and so for any $x \in K$, $\lambda \geq 0$ and any j,

$$\begin{aligned} b_j^T(x + \lambda y) &= b_j^T x + \lambda b_j^T y \\ &\leq \beta_j + \lambda b_j^T y \\ &\leq \beta_j, \end{aligned} \tag{8}$$

which shows that y is in the recession cone of K. □

LEMMA 3.8. *Suppose* $\{x_i\}$ *is a sequence of points in* $G = \{x \mid b_j^T x \leq \beta_j,\ 1 \leq j \leq m\}$, *converging to* x^*, *such that for some* $k \leq m$

$$\begin{aligned} b_j^T x^* &= \beta_j \quad \text{if } 1 \leq j \leq k, \\ b_j^T x^* &< \beta_j \quad \text{otherwise.} \end{aligned}$$

Then there is some $\lambda > 0$ *and* N *such that for every* y *in the recession cone of* $\{x \mid b_j^T x \leq \beta_j,\ 1 \leq j \leq k\}$,

$$i > N \Rightarrow x_i + \lambda \frac{y}{\|y\|} \in G.$$

Proof. If $k = m$, then the result is trivial. Otherwise $k < m$, so let

$$K = \{x \mid b_j^T x \leq \beta_j,\ 1 \leq j \leq k\},$$

and define $\mathcal{C}$ to be the recession cone of K. Since

$$G = K \cap \{x \mid b_j^T x \leq \beta_j,\ k < j \leq m\},$$

every member of $\{x_i\}$ lies in K and satisfies

$$x_i + \lambda y \in K, \qquad \lambda \geq 0, \qquad y \in \mathcal{C}. \tag{9}$$

Now since $b_j^T x^* < \beta_j$ for $k < j \leq m$, we may choose $\lambda > 0$ so that

$$\|z - x^*\| < 2\lambda \Longrightarrow b_j^T z \leq \beta_j, \qquad k < j \leq m. \tag{10}$$

Thus if N is chosen sufficiently large so that

$$i > N \Longrightarrow \|x_i - x^*\| < \lambda,$$

then for every $y \in \mathcal{C}$,

$$\left\| x_i + \lambda \frac{y}{\|y\|} - x^* \right\| \leq \|x_i - x^*\| + \lambda < 2\lambda.$$

It now follows from (10) that

$$b_j^T \left(x_i + \lambda \frac{y}{\|y\|} \right) \leq \beta_j, \qquad k < j \leq m.$$

Furthermore by (9),

$$b_j^T\left(x_i + \lambda\frac{y}{\|y\|}\right) \le \beta_j, \qquad 1 \le j \le k,$$

and so $x_i + \lambda\frac{y}{\|y\|} \in G$. □

We now apply the above lemmas to prove Lemma 3.9. The proof proceeds by showing that for an appropriately chosen convergent subsequence $\{x_{\sigma(i)}\}$, the projection of $\pi_{\sigma(i)}^T T$ in the direction of $x_{\sigma(i+1)} - x_{\sigma(i)}$ is uniformly bounded. Once this is established the conclusion of Lemma 3.9 is immediate.

LEMMA 3.9. *Suppose $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ is a subsequence of the sequence of solutions generated by the inexact cut algorithm, and let $\{\pi_{\tau(i)}\}$ be the corresponding approximately optimal solutions to the dual of SP$(x_{\tau(i)})$. Then there exists a subsequence of $\{x_{\tau(i)}\}$ (indexed by $\sigma(i)$) such that $x_{\sigma(i)} \to x^*$ and*

$$\liminf -\pi_{\sigma(i)}^T T(x_{\sigma(i+1)} - x_{\sigma(i)}) \ge 0.$$

Proof. Since X is bounded, convex, and polyhedral, the (finite) collection of all relative interiors of the faces of X partition it [16, Theorem 18.2]. Hence there is a subsequence of $\{x_{\tau(i)}\}$, indexed by $\gamma(i)$, such that $\{x_{\gamma(i)}\}$ lies in the relative interior of a face G of X and converges to a point $x^* \in G$. (We shall henceforth denote the relative interior of G by $\operatorname{ri} G$.) Since G is polyhedral we may represent it by

$$G = \{x \mid b_i^T x \le \beta_i,\ 1 \le i \le m\}.$$

If x^* is in the interior of G, then define $\mathcal{C}$ to be $\mathbb{R}^n$. In this case there is clearly some $\lambda > 0$ such that for every $y \in \mathcal{C}$, and i sufficiently large, $x_{\gamma(i)} + \lambda\frac{y}{\|y\|} \in G$.

Otherwise, without loss of generality define k to be such that

$$b_i^T x^* = \beta_i, \qquad 1 \le i \le k, \qquad b_i^T x^* < \beta_i, \qquad k < i \le m,$$

and define $\mathcal{C}$ to be the recession cone of $\{x \mid b_i^T x \le \beta_i,\ 1 \le i \le k\}$. By Lemma 3.8 there is some $\lambda > 0$ such that for every $y \in \mathcal{C}$ and for i sufficiently large,

$$x_{\gamma(i)} + \lambda\frac{y}{\|y\|} \in G. \tag{11}$$

Since we are concerned here with the limiting behavior of $\{x_{\gamma(i)}\}$ we shall henceforth assume that (11) holds for all members of $\{x_{\gamma(i)}\}$.

We now show that we can choose a subsequence $\{x_{\sigma(i)}\}$ of $\{x_{\gamma(i)}\}$ such that $x_{\sigma(i-1)} - x_{\sigma(i)} \in \mathcal{C}$. When $\mathcal{C} = \mathbb{R}^n$ this is trivial. Otherwise we construct the subsequence by choosing $x_{\sigma(k)}$ given $x_{\sigma(k-1)}$ in the following manner. Since $x_{\sigma(k-1)} \in \operatorname{ri} G$, there exists $\epsilon > 0$ such that

$$(\{x_{\sigma(k-1)}\} + \epsilon B) \cap \operatorname{aff} G \subseteq G,$$

where B is the open unit ball and $\operatorname{aff} G$ is the affine hull of G. Now for $\gamma(i)$ large enough we have that $x^* - x_{\gamma(i)} \in \epsilon B$, and so if we choose $\sigma(k) = \gamma(i)$, then

$$x^* + (x_{\sigma(k-1)} - x_{\sigma(k)}) = x_{\sigma(k-1)} + x^* - x_{\gamma(i)} \in G,$$

since $x_{\sigma(k-1)} + x^* - x_{\gamma(i)}$ is also in $\operatorname{aff} G$. Therefore

$$x^* + (x_{\sigma(k-1)} - x_{\sigma(k)}) \in \{x \mid b_i^T x \le \beta_i,\ 1 \le i \le k\},$$

and by Lemma 3.7 we deduce that

$$x_{\sigma(k-1)} - x_{\sigma(k)} \in \mathcal{C}. \tag{12}$$

Since $x_{\sigma(i)} \in \operatorname{ri} G$, this construction may be repeated to yield an infinite sequence.

Applying Lemma 2.1 to members of $\{x_{\sigma(i)}\}$, we have for any x that

$$\mathcal{Q}(x) \geq \mathcal{Q}(x_{\sigma(i-1)}) - \pi^T_{\sigma(i-1)} T(x - x_{\sigma(i-1)}) - \epsilon_{\sigma(i-1)}.$$

If we choose

$$x = x_{\sigma(i-1)} + \frac{x_{\sigma(i-1)} - x_{\sigma(i)}}{\left\|x_{\sigma(i-1)} - x_{\sigma(i)}\right\|}\lambda,$$

then by Lemma 3.8, (12) and (11) yield $x \in G$ and give

$$\begin{aligned} -\lambda \pi^T_{\sigma(i-1)} T \frac{x_{\sigma(i-1)} - x_{\sigma(i)}}{\left\|x_{\sigma(i-1)} - x_{\sigma(i)}\right\|} &\leq \mathcal{Q}(x) - \mathcal{Q}(x_{\sigma(i-1)}) + \epsilon_{\sigma(i-1)} \\ &\leq \sup_{x\in G} \mathcal{Q}(x) - \inf_{x \in G} \mathcal{Q}(x) + \epsilon_{\sigma(i-1)}. \end{aligned}$$

If we set $M = \sup_{x\in G} \mathcal{Q}(x) - \inf_{x\in G} \mathcal{Q}(x) + \epsilon_1$, then since $\{\epsilon_i\}$ is decreasing we obtain

$$-\pi^T_{\sigma(i-1)} T \frac{x_{\sigma(i-1)} - x_{\sigma(i)}}{\left\|x_{\sigma(i-1)} - x_{\sigma(i)}\right\|} \leq \frac{M}{\lambda}.$$

Therefore

$$-\pi^T_{\sigma(i-1)} T(x_{\sigma(i)} - x_{\sigma(i-1)}) \geq -\frac{M}{\lambda} \left\|x_{\sigma(i-1)} - x_{\sigma(i)}\right\|,$$

which implies

$$\liminf -\pi^T_{\sigma(i-1)} T(x_{\sigma(i)} - x_{\sigma(i-1)}) \geq 0. \qquad \square$$

THEOREM 3.10. *If $X = \{x \geq 0 \mid Ax = b\}$ is bounded and $X \subseteq \operatorname{dom} \mathcal{Q}$, the inexact cut algorithm terminates in a finite number of iterations with a δ-optimal solution of P.*

Proof. The proof is similar to that of Theorem 3.6. We will start by showing $U_i - L_i \downarrow 0$. If there exists m such that $\theta_i \geq v_i$ for all $i \geq m$, then Lemma 3.3 delivers the conclusion. Otherwise, there exists a subsequence $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ such that $\theta_{\tau(i)} < v_{\tau(i)}$, and since X is bounded, without loss of generality we may assume that $\{(x_{\tau(i)}, \theta_{\tau(i)})\}$ converges to (x^*, θ^*), say. Then by Lemma 3.4,

$$\liminf \pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)}) \geq 0.$$

Thus

$$\limsup -\pi^T_{\tau(i-1)} T(x_{\tau(i)} - x_{\tau(i-1)}) \leq 0. \tag{13}$$

Now we can apply Lemma 3.9 to extract a subsequence $\{x_{\sigma(i)}\}$ of $\{x_{\tau(i)}\}$ such that

$$\liminf -\pi^T_{\sigma(i)} T(x_{\sigma(i+1)} - x_{\sigma(i)}) \geq 0. \tag{14}$$

From (13) and (14) we have

$$-\pi^T_{\sigma(i)} T(x_{\sigma(i+1)} - x_{\sigma(i)}) \to 0.$$

This yields $U_{\sigma(i)} - L_{\sigma(i)} \to 0$, implying that the decreasing sequence $\{U_i - L_i\}$ tends to 0, which then gives the result as in the proof of Theorem 3.6. □

4. Dantzig–Wolfe decomposition.

It is well known that Benders decomposition is dual to Dantzig–Wolfe decomposition. Therefore some form of inexact optimization procedure should apply to the latter algorithm in a way that mirrors the steps of the inexact cut algorithm described in section 2. In fact such a scheme has been outlined in the literature by Kim and Nazareth [15], who discuss the computational advantages of using interior-point methods in such an approach. We digress briefly in this section to explore the asymptotic convergence properties of such an algorithm.

The dual problem of P can be formulated as

$$\begin{array}{lll} \text{D:} & \text{maximize} & b^T u + h^T v \\ & \text{subject to} & A^T u + T^T v \le c, \\ & & W^T v \le q. \end{array}$$

Suppose for the moment that the set $V = \{v \mid W^T v \le q\}$ is bounded with extreme points $\{v_i\}$. Then Dantzig–Wolfe decomposition solves a restricted master problem

$$\begin{array}{lll} \text{MD:} & \text{maximize} & b^T u + \sum_i \lambda_i h^T v_i \\ & \text{subject to} & A^T u + \sum_i \lambda_i T^T v_i \le c, \\ & & \sum_i \lambda_i = 1, \\ & & \lambda \ge 0, \end{array}$$

where the summations are taken over a subset of $\{v_i\}$. New extreme points are added iteratively to this subset by solving MD, obtaining optimal dual variables (x, θ), and then solving the subproblem

$$\begin{array}{lll} \text{SD}(x): & \text{maximize} & (h^T - x^T T^T)v \\ & \text{subject to} & W^T v \le q, \end{array}$$

to give a new column $\left[\begin{smallmatrix} T^T v_i \\ 1 \end{smallmatrix}\right]$ to be added to the restricted master problem, in the event that this column has a positive reduced cost defined by

$$(h^T - x^T T^T)v_i - \theta.$$

In our inexact Dantzig–Wolfe decomposition algorithm we first choose a convergence tolerance δ, set an iteration counter $i := 0$, and choose some decreasing sequence $\{\epsilon_i\}$ that converges to 0. We do not require that V be bounded, but following [15] we require an initial set of (not necessarily extreme) points $\{v_1, v_2, \dots, v_N\} \subseteq V$ such that MD has a feasible solution. The algorithm then proceeds as follows.

Inexact Dantzig–Wolfe decomposition algorithm.

```
While $U_i - L_i > \delta$
    (1) Set $i := i + 1$.
    (2) Solve MD to obtain $(u_i, \lambda)$ and dual variables $x_i$ and $\theta_i$.
    (3) Set $L_i := b^T u_i + \sum_i \lambda_i h^T v_i$.
    (4) Perform an inexact optimization to generate a vector $v_i$ feasible for SD($x_i$)
        such that
(15)        $v_i^T(h - Tx_i) + \epsilon_i > V(\text{SD}(x_i))$.
    (5) Set $U_i := \min\{U_{i-1}, c^T x_i + v_i^T(h - Tx_i) + \epsilon_i\}$.
```

(6) If $v_i^T(h - Tx_i) > \theta_i$, then add the column $\begin{bmatrix} T^T v_i \\ 1 \end{bmatrix}$ to MD,
else set $i := i + 1$, $x_{i+1} := x_i$, $\theta_{i+1} := \theta_i$, $L_{i+1} := L_i$, $U_{i+1} := U_i$ and go to step 4.

Here $V(\mathrm{SD}(x_i))$ is the optimal value of $\mathrm{SD}(x_i)$. Since the dual of $\mathrm{SD}(x_i)$ is easily seen to be $\mathrm{SP}(x_i)$, $V(\mathrm{SD}(x_i)) = \mathcal{Q}(x_i)$, and so step 4 of this algorithm is identical to the same step of the inexact cut algorithm of section 2.

In classical Dantzig–Wolfe decomposition, each solution v_i obtained for SD is an extreme point, of which there is a finite number, thus guaranteeing finite termination. In the inexact algorithm, this is no longer true. However, Theorem 3.10 may be invoked to yield the following corollary.

COROLLARY 4.1. *If $X = \{x \geq 0 \mid Ax = b\}$ is a bounded set and for every $x \in X$ the problem SD(x) is bounded, then the inexact Dantzig–Wolfe algorithm terminates in a finite number of iterations with a δ-optimal solution of D.*

Since $\mathrm{SD}(x)$ will always have a feasible solution (if D does), the boundedness condition on $\mathrm{SD}(x)$ is equivalent to $\mathrm{SP}(x)$ being feasible, which is the relatively complete recourse assumption of the previous section. The other assumption, that X is bounded, appears to be rather restrictive in the current context, and it fails to hold in the case when A and b are both absent, a typical situation in many applications of Dantzig–Wolfe decomposition. The convergence proof requires X to be bounded to enable the extraction of convergent subsequences. Even when A and b fail to bound X, we can still extract convergent subsequences as long as we have a guarantee that the sequence $\{x_i\}$ lies in a bounded set. In Benders decomposition we can enforce this condition in practice by placing a priori bounds on the components of x. Similarly, in inexact Dantzig–Wolfe decomposition we can impose a priori bounds on the optimal dual variables for the master problem constraints (by placing a priori penalties on infeasibilities in these constraints).

5. Computational results. We conclude by presenting some computational results of applying the inexact cut algorithm to a set of problems that arise in the planning of hydroelectric power generation. The problems are all based on a multistage stochastic programming model developed by Broad [4], in which the New Zealand electricity system is represented as a side-constrained network model with nodes representing hydroelectric reservoirs, hydroelectric generation facilities, thermal generation facilities, and demand points and arcs with constant losses representing the transmission network. The model consists of six reservoirs, six thermal stations, and 22 hydrostations.

Each stage is a week long, and demand in each week is represented by a piecewise linear load duration curve with three linear sections. At each stage several random outcomes are possible for the inflows into the reservoirs in the current week. We impose a lower bound on the final level of the reservoirs at the end of the final stage. This lower bound is a fixed fraction of the original initial level of the reservoirs in the very first stage. Additional side constraints include DC load flow constraints that govern the transmission flows and conservation of water flow equations in hydroelectric systems. The linear program for each stage has 273 variables and 120 constraints. The objective in each stage is to minimize the cost of thermal electricity generation over the current week plus the expected future cost of thermal generation.

The multistage models described above were converted into two-stage and three-stage problems by aggregating consecutive stages into larger problems. For example, to obtain a two-stage problem from a multistage problem we aggregate each second-stage problem and its descendants into a single deterministic equivalent linear program.

Table 1
Problem sizes.

Problem	*# agg stg*	*P*	*Subproblem*	*# stg*	*# scen/stg*
P1	2	10,920 × 24,843	1,200 × 2,730	3	9
P2	2	10,920 × 24,843	1,200 × 2,730	3	9
P3	2	14,520 × 33,033	4,800 × 10,920	5	3
P4	2	14,520 × 33,033	4,800 × 10,920	5	3
P5	2	43,680 × 99,372	14,520 × 33,033	6	3
P6	2	43,680 × 99,372	14,520 × 33,033	6	3
P7	3	14,520 × 33,033	1,560 × 3,549	5	3
P8	3	14,520 × 33,033	1,560 × 3,549	5	3
P9	3	43,680 × 99,372	4,800 × 10,920	6	3
P10	3	43,680 × 99,372	4,800 × 10,920	6	3
P11	3	35,154 × 42,966	1,404 × 1,560	5	5

Similarly, to obtain a three-stage problem from a multistage problem, we aggregate each third-stage problem and its descendants into a single deterministic equivalent linear program. Table 1 presents the size and characteristics of the resulting problems. Although the problems in each pair have the same size, they differ in the lower bounds imposed on the final levels of the reservoirs. Column 1 of Table 1 gives the problem identifiers, column 2 presents the number of stages in the problem (after aggregation), and column 3 contains the size of the deterministic equivalent problem. Column 4 contains the size of each subproblem after aggregation. Column 5 contains the number of stages in the problem before aggregation. For example, problem P5 is a six-stage problem, in which we have aggregated the last five stages to produce a two-stage problem. The last column contains the number of random outcomes (inflows) at each stage.

When applied to stochastic programs, Benders decomposition and the inexact cut algorithm must solve a number of subproblems in each iteration. The resulting cut has as coefficients the expectation of the subproblem coefficients. In the case of three-stage problems we traverse the scenario tree depth first using the *fast pass* procedure (see [12, 18]).

Benders decomposition and the inexact cut algorithm were both implemented using CPLEX 4.0's primal-dual interior-point solver `baropt` to solve the subproblems and the simplex solver `optimize` to solve the first-stage problems. We do not apply the crossover operation (`hybbaropt`) in solving the subproblems. For the inexact cut algorithm we terminate optimizing the last stage problems once an ϵ-optimal solution has been achieved. (All but last-stage problems are solved to optimality.) We start with $\epsilon = 10,000$ and reduce it by a factor of 10 at each iteration; we terminate `baropt` when both primal and dual feasibility are attained in the subproblem and the dual objective is at most ϵ away from the primal objective.

Observe that obtaining a primal feasible solution is not a key requirement of the algorithm but gives a convenient means for bounding how far our dual solution is from optimality; there is potential for efficiency improvements if a bound can be found that requires less computation. Indeed it is easy to see that since the proof of convergence works with a subsequence of the iterates, the requirement that ϵ_i decreases monotonically is not necessary, as long as $\epsilon_i \to 0$. This raises the (unexplored) possibility of ignoring ϵ_i, at least in the early stages of the algorithm, and interrupting `baropt` in step 4 as soon as dual feasibility is attained, then restarting it only if $\pi_i^T(h-Tx_i) \leq \theta_i$ (i.e., the cut is not exact enough to change x_i).

Table 2
Performance comparison.

Problem	*# BD cuts*	*# inex cuts*	*BD time*	*inex time*	*% improvement*
P1	22	9	170	68	60%
P2	33	20	261	159	39%
P3	5	5	124	109	12%
P4	24	17	640	398	38%
P5	4	4	594	546	8%
P6	4	4	626	585	7%
P7	30	14	324	150	54%
P8	33	27	376	304	19%
P9	17	15	1207	1087	10%
P9	14	11	979	780	20%
P11	4	4	150	134	11%

Table 2 contains a comparison of the computational results for the two methods. The termination criterion for both algorithms requires a relative gap of 10^{-5} between the upper and the lower bounds (i.e., we stop when $\frac{U-L}{U} < 10^{-5}$). All times are reported on an SGI Power Challenge. Column 1 contains the problem identifiers. Columns 2 and 3 contain the number of cuts under the exact and inexact cut algorithms, respectively. Columns 4 and 5 contain the timing in seconds for the exact and inexact methods, respectively. The last column contains the percentage of improvement of the inexact cut algorithm over the exact Benders decomposition algorithm. The entries in this column are calculated as $(\frac{\text{exact time} - \text{inexact time}}{\text{exact time}}) \times 100\%$.

Note that traditionally the subproblems are not aggregated and they are solved using the (dual) simplex method with warm starting. For some problems this is more efficient than using an interior-point method on an aggregated subproblem, although in other cases (e.g., P3, P7, and P11) we experienced significant speed-up by aggregating and using the interior-point method versus Benders decomposition with warm starting simplex. It may be possible to warm start the interior-point method effectively when solving the subproblems, using recent research developed to this end (see, for example, [19, 9]).

6. Conclusions. In every one of our problems the inexact cut algorithm improved the time to obtain a solution with the same accuracy as that of the Benders decomposition algorithm. In our experiments, the choice of $\{\epsilon_i\}$ is made independently of the problem. Further improvements in speed can be achieved by making a problem-dependent choice of $\{\epsilon_i\}$. In Table 2 the greatest improvements were obtained in cases where the Benders decomposition required a large number of cuts. In these cases we observed that often during the course of the exact algorithm the lower bounds did not change over the course of several iterations. The inexact cut algorithm does not display this behavior, and it reaches an approximately optimal solution with fewer cuts. This suggests that computing cuts inexactly is a promising and simple improvement strategy for operations research practitioners who observe similar behavior in Benders decomposition applied to their stochastic linear programming models.

Acknowledgments. We thank Michael Saunders, Suvrajeet Sen, and the anonymous referees for their insightful comments, which significantly improved the exposition of this paper.

REFERENCES

[1] K. T. Au, J. L. Higle, and S. Sen, *Inexact subgradient methods with applications in stochastic programming*, Math. Programming, 63 (1994), pp. 65–82.

[2] O. Bahn, O. Du Merle, J.-L. Goffin, and J.-P. Vial, *A cutting plane method from analytic centers for stochastic programming*, Math. Programming Ser. B, 69 (1995), pp. 45–73.

[3] J. F. Benders, *Partitioning procedures for solving mixed-variables programming problems*, Numer. Math., 4 (1962), pp. 238–252.

[4] K. P. Broad, *Power Generation Planning Using Scenario Aggregation*, M.S. thesis, University of Auckland, Auckland, New Zealand, 1996.

[5] G. B. Dantzig and P. Wolfe, *Decomposition principle for linear programs*, Oper. Res., 8 (1960), pp. 101–111.

[6] G. B. Dantzig and P. Wolfe, *The decomposition algorithm for linear programs*, Econometrica, 29 (1961), pp. 767–778.

[7] E. Flippo and A. Rinnooy Kan, *Decomposition in general mathematical programming*, Math. Programming, 60 (1993), pp. 361–382.

[8] A. M. Geoffrion, *Generalized Benders decomposition*, J. Optim. Theory Appl., 10 (1972), pp. 237–260.

[9] J. Gondzio, *Warm start of the primal-dual method applied in the cutting-plane scheme*, Math. Programming Ser. A, 83 (1998), pp. 125–143.

[10] J. L. Higle and S. Sen, *On the convergence of algorithms with implications for stochastic and nondifferentiable optimization*, Math. Oper. Res., 17 (1992), pp. 112–131.

[11] W. Hogan, *Application of general convergence theory for outer approximation algorithms*, Math. Programming, 5 (1973), pp. 151–168.

[12] J. Jacobs, G. Freeman, J. Grygier, D. Morton, G. Schultz, K. Staschus, and J. Stedinger, *SOCRATES: A system for scheduling hydro-electric generation under uncertainty*, Ann. Oper. Res., 59 (1995), pp. 99–133.

[13] P. Kall and S. W. Wallace, *Stochastic Programming*, John Wiley, New York, 1994.

[14] J. E. Kelley, Jr., *The cutting-plane method for convex programs*, J. Soc. Indust. Appl. Math., 8 (1960), pp. 703–712.

[15] K. Kim and J. L. Nazareth, *The decomposition principle and algorithm for linear programming*, Linear Algebra Appl., 152 (1991), pp. 119–133.

[16] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[17] R. M. Van Slyke and R. Wets, *L-shaped linear programs with applications to optimal control and stochastic programming*, SIAM J. Appl. Math., 17 (1969), pp. 638–663.

[18] R. J. Wittrock, *Advances in a Nested Decomposition Algorithm for Solving Staircase Linear Programs*, Report SOL 83-2, Systems Optimization Laboratory, Department of Operations Research, Stanford University, Stanford, CA, 1983.

[19] G. Zakeri, D. M. Ryan, and A. B. Philpott, *Techniques for Solving Large Scale Set Partitioning Problems*, Technical report, University of Auckland, New Zealand, 1996.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 658–672

MINIMIZING SEPARABLE CONVEX FUNCTIONS SUBJECT TO SIMPLE CHAIN CONSTRAINTS[*]

MICHAEL J. BEST[†], NILOTPAL CHAKRAVARTI[‡], AND VASANT A. UBHAYA[§]

Abstract. Minimizing a separable convex objective subject to an ordering restriction on its variables is a generalization of a class of problems in statistical estimation and inventory control. It is shown that a pool adjacent violators (PAV) algorithm can be used to compute an optimal solution of this problem as well as the minimal and maximal extended solutions, which provide lower and upper bounds on all optimal solutions and solve certain subproblems. These results unify and extend several previously known results. In addition, it is shown that a PAV algorithm can be applied to solving the problem with integer constraints on the variables.

Key words. isotonic regression, median regression, pool adjacent violators algorithm, convex functions, computational complexity

AMS subject classifications. 90C25, 26A48, 68Q25

PII. S1052623497314970

1. Introduction.

Let $I = (a, b)$ be a real interval, where $-\infty \leq a, b \leq \infty$. Let $N = \{1, 2, \ldots, n\}$ and $f_i, i \in N$, be real-valued convex functions defined on I. We consider the following problem.

PROBLEM P. Find $x_i \in I$ for $i \in N$, so as to

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^{n} f_i(x_i) \\ &\text{subject to} \quad x_1 \leq x_2 \leq \cdots \leq x_n. \end{aligned}$$

The constraint $x_1 \leq x_2 \leq \cdots \leq x_n$ in the Problem P is called the simple chain or monotonicity constraint. A vector $x = (x_1, x_2, \ldots, x_n)$ with $x_i \in I$, satisfying the monotonicity constraint, is called a monotone vector. Since I is open, convex functions such as f_i are continuous on I but they may not have minimizers in I. Unless otherwise stated, we assume throughout that the following Condition A holds.

CONDITION A. Each $f_i, i \in N$, has a minimizer μ_i in I. □

Note that μ_i is in the interior of I since I is open.

Several special cases of Problem P have been previously considered in the literature, for example, in statistical applications, estimation, inventory control, and curve fitting. We cite three cases. The first is the isotonic regression problem; in this case $I = (-\infty, \infty)$, $f_i(x_i) = w_i(c_i - x_i)^2$, with $w_i > 0$ for each i. See, e.g., Robertson, Wright, and Dykstra [14] or Best and Chakravarti [2]. The second is the isotonic median regression; in this case $I = (-\infty, \infty)$, $f_i(x_i) = |c_i - x_i|$ for each i. See, e.g., Robertson and Wright [13] or Chakravarti [4]. In the third case, $I = (0, \infty)$, $f_i(x_i) = a_i/x_i + b_i x_i$, with $a_i > 0$, $b_i > 0$ for each i. See, e.g., Schwarz

[*] Received by the editors January 14, 1997; accepted for publication (in revised form) March 26, 1999; published electronically February 29, 2000.

http://www.siam.org/journals/siopt/10-3/31497.html

[†] Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada.

[‡] Indian Institute of Management, Calcutta, Diamond Harbour Road, P.O. Box 16757, P.O. Alipore, Calcutta 700027, India.

[§] Department of Computer Science and Operations Research, 258 IACC Building, North Dakota State University, Fargo, ND 58105 (ubhaya@plains.nodak.edu).

and Schrage [16]. Each of these cases can be solved by a pool adjacent violators (PAV) algorithm. Strömberg [17] has also applied a PAV algorithm to a problem with a convex distance function defined in a special way that entails conditions stronger than Condition A. The algorithm computes an optimal solution and the minimal and maximal optimal solutions. This problem is also a special case of our problem. The PAV algorithm appearing in these works or elsewhere is essentially the same. It maintains a partition of the set of indexes N into sets of consecutive integers, called blocks. Adjacent blocks are pooled in the event of a "violation," suitably defined in each case. Our Problem P is the most general considered up to this time in that each f_i is arbitrary and is allowed to have more than one minimizer, and hence a bounded or even an unbounded interval of minimizers. In this article we show that a PAV algorithm can be used to solve the more general Problem P. The article therefore extends and unifies previous results. Our contribution is in enlarging the class of problems that can be solved by a PAV algorithm. (Although the PAV algorithm is essentially the same, for the least squares objective it may be cast into a somewhat different form using greatest convex minorants. See Preparata and Shamos [12, p. 168] and Ubhaya [20].)

We also show that the PAV algorithm can be suitably used to obtain what we call the extremal (minimal and maximal) extended solutions. These solutions give lower and upper bounds on all optimal solutions to P. Furthermore, subvectors formed by their components provide extremal optimal solutions to certain subproblems of P. In the case of certain optimization and approximation problems, it is known that there exist extremal (a maximal and/or a minimal) optimal solutions so that the set of all optimal solutions is appropriately bounded above and/or below by the respective extremal solution. Under certain conditions, the converse is also true, i.e., that any feasible solution to the problem that is enclosed between the two extremal solutions is optimal. See, e.g., Landers and Rogge [8, 9], Liu and Ubhaya [10], and Ubhaya [18, 19, 21]. However, these conditions do not apply to our very general problem. The extremal extended solutions mentioned above fulfill the role of the extremal solutions.

We now describe the significance of multiple and extremal solutions. If the set S of optimal solutions to P has more than one element, then we may be able to find one solution that optimizes another objective function of interest over S, e.g., minimize (respectively, maximize) $\Sigma_{i=1}^{n} c_i x_i$, where $c_i > 0$ represent cost (respectively, profit) per unit of $i$the commodity. Roughly speaking, the required optimal solution will be the minimal (respectively, maximal) solution. When c_i have mixed signs, a solution in between these two may result. Another problem is sensitivity analysis, i.e., the effect of perturbations in the constants z appearing in f_i on the optimal solution. For example, if $f_i(x_i) = |c_i - x_i|$, then $z = (c_1, c_2, \ldots, c_n)$, and if $f_i(x_i) = a_i/x_i + b_i x_i$, then $z = (a_1, b_1, a_2, b_2, \ldots, a_n, b_n)$. For each z, let S_z denote the set of optimal solutions to P and consider the "selection problem" of selecting one solution x out of S_z for each z so that the mapping $z \to x$ has certain properties such as stability. Ubhaya [21] has shown that such selections can be made in certain optimization problems and that the extremal optimal solutions play a significant part. In the case of Problem P, further investigations are needed.

We also show that a PAV algorithm can be applied to obtain a solution of Problem P with integer constraints on its variables. Goldstein and Kruskal [7] have shown that the isotonic regression problem with integer constraints may be solved by first solving the continuous problem and then simply rounding the solution. In our more general framework, such a strategy does not work. We show, however, that this strategy can be used if certain conditions, which are satisfied by the integer isotonic regression, are imposed on P. Integer constrained problems arise when x_i represent quantities taking integer values such as number of units produced, number of workers assigned, ranks, etc. Examples appear in [7] and in Liu and Ubhaya [10] and will not be repeated here. In [10], polynomial algorithms are developed for solving a related regression problem with a convex but nonseparable distance function, and monotonicity and integer constraints on the variables.

The rest of this article is organized as follows. Section 2 mainly consists of a description and validation of the PAV algorithm for Problem P. It also contains the discussion of some special known cases of P in the light of our results. In section 3 we establish the existence of extremal extended solutions for P and present an algorithm to compute them. In section 4 we develop algorithms for obtaining an optimal solution and extremal extended solutions to the integer version of P.

2. PAV algorithm. In this section we develop a PAV algorithm for solving Problem P. Let us start by briefly reviewing some basic facts about subgradients of convex functions. See Clarke [5] and Rockafellar [15].

Let $I = (a, b)$ be as in section 1 and $f : I \to R$ be a convex function. A number ξ is said to be a subgradient of f at x in I if $f(y) - f(x) \geq \xi(y - x)$ for all $y \in I$. We denote by $\partial f(x)$ the set of all the subgradients at $x \in I$. It is known that $\partial f(x)$ is a nonempty compact interval for each x in I and that it has the following monotonicity properties. If $\xi_1 \in \partial f(x_1)$ and $\xi_2 \in \partial f(x_2)$, then

$$x_1 < x_2 \Rightarrow \xi_1 \leq \xi_2 \quad \text{and} \quad \xi_1 < \xi_2 \Rightarrow x_1 \leq x_2. \tag{2.1}$$

Furthermore, $x \in I$ minimizes f over I if and only if $0 \in \partial f(x)$. It is also known that if $f_i : I \to (-\infty, \infty)$, $i \in N$, are convex functions, then

$$\delta\left(\sum_{i=1}^{n} f_i\right)(x) = \sum_{i=1}^{n} \partial f_i(x), \quad x \in I. \tag{2.2}$$

We now state the optimality conditions for Problem P needed for developing the PAV algorithm. These may be derived easily by elementary methods from Theorem 6.1.1 of [5] or Theorem 28.3 of [15].

PROPOSITION 2.1. *$x = (x_1, x_2, \ldots, x_n)$ is an optimal solution for Problem P if and only if there exist dual variables $\xi_j \in \partial f_j(x_j)$, $j \in N$, and $\nu_i, 1 \leq i \leq n-1$, such that*

$$x_1 \leq x_2 \leq \cdots \leq x_n, \tag{2.3}$$

$$\xi_1 = -\nu_1, \qquad \xi_2 = \nu_1 - \nu_2, \ldots, \qquad \xi_{n-1} = \nu_{n-2} - \nu_{n-1}, \qquad \xi_n = \nu_{n-1}, \tag{2.4}$$

$$\nu_i \geq 0, \quad 1 \leq i \leq n-1, \tag{2.5}$$

$$x_i < x_{i+1} \Rightarrow \nu_i = 0, \quad 1 \leq i \leq n-1. \qquad \square \tag{2.6}$$

For any subset B of N, define a function F_B on I by

$$F_B(x) = \sum_{i \in B} f_i(x).$$

Clearly, $F_{\{i\}} = f_i$ for all $i \in N$. Note that F_B is convex. Recall that, by Condition A of section 1, each f_i has a minimizer μ_i in I. Clearly, each f_i has the following Property A.

Property A. Each f_i is nonincreasing on (a, μ_i) and nondecreasing on (μ_i, b). □

The above property may be used to prove the following proposition. The proof is left to the reader.

Proposition 2.2. *For any $B \subset N$, F_B has a minimizer μ_B in $[\alpha, \beta]$, where $\alpha = \min\{\mu_i : i \in B\} > a$ and $\beta = \max\{\mu_i : i \in B\} < b$. Furthermore, Problem P has an optimal solution $x = (x_1, x_2, \ldots, x_n)$ with $x_i \in [\gamma, \delta]$, where $\gamma = \min\{\mu_i : i \in N\} > a$ and $\delta = \max\{\mu_i : i \in N\} < b$.* □

A partition J of N is a decomposition of N into disjoint sets of consecutive integers whose union is N. Each member of the partition is called a block of J, generally denoted by B. We define $x(J)$ to be any n-vector whose $i$the coordinate $x_i(J)$, $i \in N$, is given by $x_i(J) = \mu_B$, where B is the unique block of J containing i and μ_B is a minimizer of F_B, which exists by Proposition 2.2. Thus $x_i(J)$ has identical values for all i in a block.

We next present a PAV algorithm for Problem P. Beginning with the finest partition whose blocks are single integers in N and an initial infeasible solution x (violating constraint (2.3)), the algorithm successively merges blocks to reduce infeasibility, obtaining a new, coarser partition J and an infeasible solution $x(J)$. It terminates when $x(J)$ becomes feasible, giving an optimal solution and partition. Let $B = \{p, \ldots, q\}$, $1 \le p \le q \le n$, denote a block of a partition J during any iteration of the algorithm. The predecessor and successor blocks of B, denoted, respectively, by B^- and B^+, of the same partition J, are defined as follows: if $p > 1$, then B^- is the block containing $p - 1$, otherwise $B^- = \emptyset$. Similarly, if $q < n$, then B^+ is the block containing $q + 1$, otherwise $B^+ = \emptyset$.

Algorithm 2.3 (the PAV algorithm for computing an optimal solution $x = x(J)$ to Problem P). Note that, by Proposition 2.2, F_B has a minimizer in I. In the following, the minimizers μ_B of F_B are not necessarily unique. Choose any minimizer μ_B, but once chosen, use this minimizer for all subsequent occurrences of that B.

```
Algorithm PAV
    initialization:
        Set $J = \{\{i\} : i \in N\}$; compute a minimizer $\mu_B$ of $f_B, B \in J$;
        Set $B = \{1\},\ B^+ = \{2\},\ B^- = \emptyset$;
    while $B^+ \neq \emptyset$
        if $\mu_B > \mu_{B^+}$ then
            merge $B$ and $B^+$ (i.e., set $J = J \backslash \{B, B^+\} \cup \{B \cup B^+\}$ and $B = B \cup B^+$);
            compute new $\mu_B$ and set $B^+$ appropriately ($B^-$ remains unchanged);
            while $B^- \neq \emptyset$ and $\mu_{B^-} > \mu_B$
                merge $B$ and $B^-$ (i.e., set $J = J \backslash \{B, B^-\} \cup \{B \cup B^-\}$ and $B = B \cup B^-$);
                compute new $\mu_B$ and set $B^-$ appropriately ($B^+$ remains unchanged);
            end while
        else
            set $B = B^+$; set $B^-$ and $B^+$ appropriately;
        end if
    end while
    $x_i(J) = \mu_B,\ i \in B \in J$ is an optimal solution;
end algorithm PAV □
```

We now analyze the worst-case complexity of the algorithm. The initial partition is of size n, i.e., it has n blocks. Whenever two blocks are merged, a new block is formed and the number of blocks in the partition is decreased by 1. In the worst case, therefore, there are n different partitions having decreasing sizes $n, n - 1, \ldots, 1$.

Each time a new block is formed, there are at most two comparisons, giving a total of $2(n-1)$. The algorithm also requires the computation of μ_B for subsets B of N at most $2n-1$ times, n times at the initialization step, and once every time a new block is formed by merging. The algorithm therefore has linear time worst-case complexity if we assume that each comparison and each computation of μ_B incurs unit cost. In general, it may take more than unit cost to compute μ_B (see below) or to set $B = B \cup B^+$, etc. Suitable data structures may have to be used to improve the complexity, as in some special cases of Problem P given later in the paper.

If each f_i is differentiable, then $\partial F_B(X)$ is a singleton and equals the derivative of F_B at x. In this case μ_B is determined by setting this derivative equal to zero. In case $f_i(x_i) = w_i(c_i - x_i)^2$ (isotonic regression), μ_B turns out to be the "block average" $(\Sigma_{i \in B} w_i c_i)/(\Sigma_{i \in B} w_i)$. Each computation of μ_B during the PAV algorithm requires a constant number of elementary arithmetic operations with the use of an appropriate data structure. The PAV algorithm therefore has linear time complexity in the usual unit cost random access machine model (Aho, Hopcroft, and Ullman [1]). For more details, see, e.g., Preparata and Shamos [12], Best and Chakravarti [2], or Grotzinger and Witzgall [6]. It has been observed by Ubhaya [20] that the problem of quasi-convex regression (with the above quadratic distance function) can be solved by solving n isotonic regression problems giving the worst-case complexity of $O(n^2)$. However, Ubhaya has shown that this complexity can be reduced to $O(n)$ by using special computations and a data structure. Such an improvement does not seem possible for quasi-convex regression with an arbitrary function $f_i(x_i)$. For the case $f_i(x_i) = |c_i - x_i|$ (isotonic median regression), μ_B is given by the median of the set $\{c_i : i \in B\}$. Each computation of μ_B requires $O(|B|)$ comparisons, and the time complexity of the algorithm is $O(n^2)$. An interesting feature of the algorithm in this case is that it requires no arithmetic operations other than comparisons and so is free of rounding error (Chakravarti [4]). Finally, let $f_i(x_i) = a_i/x_i + b_i x_i$, with $a_i > 0$, $b_i > 0$. In this case μ_B is given by $((\Sigma_{i \in B} a_i)/(\Sigma_{i \in B} b_i))^{1/2}$. If an appropriate data structure is used, each computation of μ_B requires a constant number of elementary arithmetic operations and a single square-root extraction. The PAV algorithm has linear time complexity provided that each square-root extraction incurs unit cost. In what follows we prove the correctness of the algorithm.

LEMMA 2.4. *Let J be any partition obtained during the course of Algorithm* 2.3, *and let $B = \{p, p+1, \ldots, q\}$ be any block of J. Let $x = x(J)$ be the solution corresponding to J. Then, there exist dual variables $\xi_i \in \partial f_i(x_i)$, $p \le i \le q$, such that $\Sigma_{i=p}^{q} \xi_i = 0$.*

Proof. Let Σ denote $\Sigma_{i=p}^{q}$. By Algorithm 2.3, we have $x_i = x_i(J) = \mu_B$ for all $i \in B$. By (2.2) we have $\Sigma \partial f_i(x_i) = \Sigma \partial f_i(\mu_B) = \partial(\Sigma f_i)(\mu_B) = \partial F_B(\mu_B)$. Since μ_B minimizes F_B and, by Condition A, $\mu_B \in I$, we have $0 \in \partial F_B(\mu_B)$. Hence ξ_i, $p \le i \le q$, may be so chosen so that $\xi_i \in \partial f_i(x_i)$ and $\Sigma \xi_i = 0$. □

THEOREM 2.5. *The PAV Algorithm* 2.3 *computes an optimal solution to Problem P.*

Proof. The proof of the algorithm proceeds by showing that for each partition J, there exist $\xi_i = \xi_i(J)$ and $\nu_i = \nu_i(J)$ satisfying (2.4), (2.5), and (2.6). Hence, at termination, all the optimality conditions (2.3)–(2.6) are satisfied and the solution $x(J)$ obtained is optimal. Throughout the proof, the dual variables ξ_i and ν_i, $1 \le i \le n$, stated in Proposition 2.1 will be chosen according to the following selection criterion, which ensures that they will automatically satisfy (2.4) and (2.6) but not necessarily (2.5). Let J be any partition and $B = \{p, p+1, \ldots, q\}$ be any block of

J. Then $x_i = x_i(J) = \mu_B$, $p \le i \le q$. We choose $\xi_i \in \partial f_i(x_i)$, $p \le i \le q$, such that $\Sigma_{i=p}^{q}\xi_i = 0$. By Lemma 2.4 such ξ_i exist. We also let $\nu_i = -\Sigma_{j=p}^{i}\xi_j = \Sigma_{j=i+1}^{q}\xi_j$, $p \le i \le q$. Clearly, ξ_i satisfy (2.4). Then, $\nu_q = -\Sigma_{j=p}^{q}\xi_j = 0$. Also, since $\nu_q = 0$, (2.6) is automatically satisfied for $i = q$ if $q < n$. However, (2.5) may not be satisfied.

We prove the theorem by induction on distinct partitions. For the initial partition each $\{i\}$ is a block, $x_i = \mu_i$, and $0 \in \partial f(x_i)$. We let $\xi_i = 0$, $i \in N$, and, by the selection criterion, $\nu_i = 0$ for all $1 \le i \le n-1$. Thus (2.4)–(2.6) hold. During any step with the (current) partition J, ξ_i and ν_i satisfy (2.4) and (2.6) by their selection criterion. We therefore assume that (2.5) holds for ν_i for partition J, and, if J' is the next partition, we produce $\xi_i' = \xi_i(J')$ and $\nu_i' = \nu_i(J')$ so that (2.5) also holds for ν_i'. Let $B_1 = \{p, p+1, \dots, q\}$ and $B_2 = \{q+1, q+2, \dots, r\}$ be two consecutive blocks of J. If $\mu_{B_1} \le \mu_{B_2}$, then $B_1, B_2 \in J' = J$, and we let $\xi_i' = \xi_i$ and $\nu_i' = \nu_i$. Then (2.5) holds for ν_i'. Now suppose that $\mu_{B_1} > \mu_{B_2}$. Then, $J' = J \backslash \{B_1, B_2\} \cup \{B_1 \cup B_2\}$ is obtained from J by merging B_1 and B_2, and $B = B_1 \cup B_2$. We need to show the existence of ξ_i' and ν_i' satisfying (2.5). We let $\xi_i' = \xi_i$ unless $p \le i \le r$. This gives $\nu_i' = \nu_i$ unless $p \le i \le r$, by the selection criterion. Hence we need only produce ξ_i', $p \le i \le r$, such that $\nu_i' \ge 0$ for $p \le i \le r$. We consider several cases. For convenience, we use the notation F, F_1, F_2, M, M_1, and M_2 instead of $F_B, F_{B_1}, F_{B_2}, \mu_B, \mu_{B_1}$, and μ_{B_2}, respectively. Then, by assumption, $M_1 > M_2$.

Case i. $M_1 > M > M_2$. We choose any ξ_i', $p \le i \le r$, satisfying its definition. Since $M_1 > M$, by (2.1), we have $\xi_k' \le \xi_k$, $p \le k \le q$. Hence, $\nu_i' = -\Sigma_{k=p}^{i}\,\xi_k' \ge -\Sigma_{k=p}^{i}\,\xi_k = \nu_i \ge 0$ for $p \le i \le q$. Again, since $M > M_2$, we have $\xi_k' \ge \xi_k$ for $q < k \le r$. Hence, $\nu_i' = \Sigma_{k=i+1}^{r}\,\xi_k' \ge \Sigma_{k=i+1}^{r}\xi_k = \nu_i \ge 0$, $q < i \le r$. ($\nu_r' = \nu_r = 0$.)

Case ii. $M_1 = M > M_2$. Since $\partial f_k(M_1) = \partial f_k(M)$ is a compact interval, we denote it by $[\underline{\delta}_k, \overline{\delta}_k]$, $p \le k \le r$. Then $\xi_k \ge \underline{\delta}_k$, $p \le k \le q$, since $\xi_k \in \partial f_k(M_1)$. Again, since $M_1 > M_2$, we have $\underline{\delta}_k \ge \xi_k, q < k \le r$. Hence, $\Sigma_{k=q+1}^{r}\underline{\delta}_k \ge \Sigma_{k=q+1}^{r}\,\xi_k = 0$. Now $0 \in \partial F(M)$, and hence $\Sigma_{k=p}^{r}\underline{\delta}_k \le 0$, which gives $\Sigma_{k=p}^{q}\underline{\delta}_k \le -\Sigma_{k=q+1}^{r}\underline{\delta}_k \le 0$. Again, $\Sigma_{k=p}^{q}\,\xi_k = 0$. Hence, clearly, we can find $\delta_k, p \le k \le q$, satisfying $\underline{\delta}_k \le \delta_k \le \xi_k$ and $\Sigma_{k=p}^{q}\delta_k = -\Sigma_{k=q+1}^{r}\underline{\delta}_k$. We choose $\xi_k' = \delta_k$, $p \le k \le q$, and $\xi_k' = \underline{\delta}_k$, $q < k \le r$. Then, for $p \le i \le q$, we have $\nu_i' = -\Sigma_{k=p}^{i}\xi_k' = -\Sigma_{k=p}^{i}\delta_k \ge -\Sigma_{k=p}^{i}\,\xi_k = \nu_i \ge 0$. Since $M > M_2$, we have $\underline{\delta}_k \ge \xi_k$, $q < k \le r$. Hence, as in Case i, for $q < i \le r$ we have $\nu_i' = \Sigma_{k=i+1}^{r}\,\xi_k' = \Sigma_{k=i+1}^{r}\underline{\delta}_k \ge \Sigma_{k=i+1}^{r}\,\xi_k = \nu_i \ge 0$.

Case iii. $M > M_1 > M_2$. Let $\lambda \in \partial F_2(M_1)$. Since $0 \in \partial F_2(M_2)$ and $M_1 > M_2$, by (2.1) we have $\lambda \ge 0$. Again, since $0 \in \partial F_1(M_1)$, by (2.2) we have $\lambda = \lambda + 0 \in \partial F(M_1)$. Now $0 \in \partial F(M)$ and $M > M_1$, hence $\lambda \le 0$. Thus $\lambda = 0$ and $0 \in \partial F_2(M_1)$. Hence M_1 is a minimizer of F_2. By (2.2), there exist $\eta_k \in \partial f_k(M_1)$, $q < k \le r$, such that $\Sigma_{k=q+1}^{r}\,\eta_k = 0$. Since $M > M_1$, for any ξ' consistent with the definition, we have $\xi_k' \ge \xi_k$, $p \le k \le q$. Similarly, $M > M_1$ yields $\xi_k' \ge \eta_k$, $q < k \le r$. Consequently, we have the inequality

$$\sum_{k=p}^{r}\xi_k' \ge \sum_{k=p}^{q}\xi_k + \sum_{k=q+1}^{r}\eta_k.$$

Since $\Sigma_{k=p}^{q}\,\xi_k = \nu_q = 0$, $\Sigma_{k=q+1}^{r}\eta_k = 0$ and $\Sigma_{k=p}^{r}\xi_k' = \nu_r' = 0$, we conclude that both sides of the above inequality are zero. Hence, we must have $\xi_k' = \xi_k$, $p \le k \le q$, and $\xi_k' = \eta_k$, $q < k \le r$. Now the first set of equations at once shows that $\nu_i' = \nu_i \ge 0$, $p \le i \le q$. Since $M > M_2$, we have $\xi_k' = \xi_k$, $q < k \le r$. Then, as in Case i, we obtain $\nu_i' \ge \nu_i \ge 0$, $q < i \le r$.

The remaining cases, $M_1 > M_2 = M$ and $M_1 > M_2 > M$, may be treated similarly. □

3. Minimal and maximal extended solutions. Let $\overline{I} = [a, b]$, which is a closed interval in the extended real line $[-\infty, \infty]$. A vector $x = (x_1, x_2, \ldots, x_n)$ is called an extended monotone vector if $x_i \in \overline{I}$ and $x_1 \le x_2 \le \cdots < x_n$. Note that a monotone vector is extended monotone. In this section, we obtain two extended monotone vectors u and v which provide the lower and upper bounds on all optimal solutions to P. Furthermore, subvectors formed by components of u and v give extremal optimal solutions to certain subproblems of P. We call u and v, respectively, the minimal and maximal extended solutions to P and develop PAV algorithms to compute them.

Let $B \subset N$. If S_B is the set of all minimizers of convex function $F_B(x) = \Sigma_{i \in B} f_i(x)$ defined for $x \in I = (a, b)$, then, by Proposition 2.2, $S_B \ne \emptyset$. Define $\underline{\mu}_B = \inf S_B$ and $\overline{\mu}_B = \sup S_B$. Note that $\underline{\mu}_B$ (respectively, $\overline{\mu}_B$) may equal the endpoint a (respectively, b) of I and, thus, not be an element of I. By the convexity of F_B, we have $S_B = [\underline{\mu}_B, \overline{\mu}_B] \cap I$. When $B = \{i\}$, $i \in N$, the above defines the corresponding quantities $\underline{\mu}_i$ and $\overline{\mu}_i$ for $f_i = F_{\{i\}}$. If $\underline{\mu}_B$ (respectively, $\overline{\mu}_B$) is in I, then it is the smallest (respectively, largest) minimizer of F_B. The proof of the following proposition uses Condition A and is similar to that of Proposition 2.2. It is left to the reader.

Proposition 3.1 (bounds on optimal solutions). *Let $B \subset N$. Then,*

$$a \le \min \left\{ \underline{\mu}_i : i \in B \right\} \le \underline{\mu}_B \le \max \left\{ \underline{\mu}_i : i \in B \right\} < b,$$
$$a < \min \left\{ \overline{\mu}_i : i \in B \right\} \le \overline{\mu}_B \le \max \left\{ \overline{\mu}_i : i \in B \right\} \le b.$$

Furthermore, Problem P has at least one optimal solution, and any optimal solution $x = (x_1, x_2, \ldots, x_n)$ satisfies

$$a \le \min \left\{ \underline{\mu}_i : i \in N \right\} \le x_i \le \max \left\{ \overline{\mu}_i : i \in N \right\} \le b, \quad \textit{for all } 1 \le i \le n. \qquad \square$$

If $x = (x_1, x_2, \ldots, x_n)$ is an extended monotone vector, we define $\underline{\text{ind}}(x)$ (respectively, $\overline{\text{ind}}(x)$) to be the largest (smallest) index $i \in N$ such that $x_i = a$ (respectively, $x_i = b$) if such an index exists, and 0 (respectively, $n + 1$), otherwise. We define the minimal (respectively, maximal) optimal solution to a problem to be a solution that is the smallest (respectively, largest) in each coordinate among all the optimal solutions to the problem. Clearly, such solutions, if they exist, are unique. We define a Subproblem $\text{SP}(c, d)$, indexed by two integers c and d in N, as follows.

Subproblem $\text{SP}(c, d)$. Find $x_i \in I,\ c \le i \le d$, so as to

$$\begin{aligned} &\text{minimize} && \sum_{i=c}^{d} f_i(x_i) \\ &\text{subject to} && x_c \le x_{c+1} \le \cdots \le x_d. \end{aligned}$$

Clearly, $P = \text{SP}(1, n)$. The following theorem establishes existence and properties of extremal extended solutions. These results and some others to follow are intuitively obvious but their proofs, requiring essentially basic convexity arguments, are tedious. We leave them to the reader. A sample proof of Theorem 3.2, part 3, appears in the appendix. (The third author may be contacted for a copy of the original technical report [3], which includes all the proofs.) If X is the set of all optimal solutions to

P, then u and v in the following theorem are defined by $u_i = \inf\{x_i : x \in X\}$ and $v_i = \sup\{x_i : x \in X\}$.

THEOREM 3.2 (minimal and maximal extended solutions). *There exist two unique extended vectors $u = (u_1, u_2, \ldots, u_n)$ and $v = (v_1, v_2, \ldots, v_n)$ for Problem P (which are called, respectively, the minimal and maximal extended solutions) with the following properties, where $p = \underline{\operatorname{ind}}(u)$ and $q = \overline{\operatorname{ind}}(v)$:*

1. *$a \leq u_i < b,\ a < v_i \leq b$, and $u_i \leq v_i$ for all i.*
2. *If $x = (x_1, x_2, \ldots, x_n)$ is any optimal solution to P, then $u_i \leq x_i \leq v_i$ for all i.*
3. i. *$0 \leq p \leq n$, $\underline{\mu}_i = u_i = a$ for $1 \leq i \leq p$, and $\underline{\mu}_{p+1} \geq u_{p+1} > a$ if $p < n$.*
   ii. *$1 \leq q \leq n+1$, $\overline{\mu}_i = v_i = b$ for $q \leq i \leq n$, and $\overline{\mu}_{q-1} \leq v_{q-1} < b$ if $q > 1$.*
4. i. *$(u_{p+1}, u_{p+2}, \ldots, u_n)$ is the minimal optimal solution to Subproblem* SP$(p+\ 1, n)$*. If $p = 0$ (equivalently, $u_i \in I,\ i \in N$), then u is the minimal optimal solution to P.*
   ii. *$(v_1, v_2, \ldots, v_{q-1})$ is the maximal optimal solution to Subproblem* SP$(1, q-1)$*. If $q = n+1$ (equivalently, $v_i \in I,\ i \in N$), then v is the maximal optimal solution to P.*
5. i. *u_i minimizes f_i for* $\max\{p+1, q\} \leq i \leq n$.
   ii. *v_i minimizes f_i for* $1 \leq i \leq \min\{p, q-1\}$.
   iii. *If $p+1 \leq q-1$, then $(u_{p+1}, u_{p+2}, \ldots, u_{q-1})$ and $(v_{p+1}, v_{p+2}, \ldots, v_{q-1})$ are, respectively, the minimal and maximal optimal solutions to Subproblem* SP$(p+1, q-1)$.
6. *A monotone vector $x = (x_1, x_2, \ldots, x_n)$ is an optimal solution to P if and only if the following three conditions are satisfied: x_i minimizes f_i (equivalently, $a < x_i \leq \overline{\mu}_i$) for $1 \leq i \leq \min\{p, q-1\}$; if $p+1 \leq q-1$, then $(x_{p+1}, x_{p+2}, \ldots, x_{q-1})$ is an optimal solution* SP$(p+1, q-1)$*; and x_i minimizes f_i (equivalently, $\underline{\mu}_i \leq x_i < b$) for* $\max\{p+1, q\} \leq i \leq n$. □

Note that, by convexity of f_i, the limits stated in part 1 of the next proposition exist. We leave its simple proof to the reader.

PROPOSITION 3.3. *In the following four statements, $1 \Rightarrow 2 \Leftrightarrow 3 \Rightarrow 4$.*

1. $\lim\{f_i(x) : x \downarrow a\} = \lim\{f_i(x) : x \uparrow b\} = \infty$ *for all i.*
2. *$\underline{\mu}_i, \overline{\mu}_i \in I$, and are, respectively, the minimal and maximal minimizers of $f_i = F_{\{i\}}$ for all i.*
3. *$\underline{\mu}_B, \overline{\mu}_B \in I$, and are, respectively, the minimal and maximal minimizers of F_B for all $B \subset N$.*
4. *$u_i, v_i \in I$ for all i, and u and v are, respectively, the minimal and maximal optimal solutions to P.* □

A special case of the statement $3 \Rightarrow 4$ of the above proposition is established by Strömberg [17] for his distance function. We now state another PAV algorithm.

ALGORITHM 3.4 (the PAV algorithm for determining $u = u(J)$ and $v = v(J)$).

1. The steps of the algorithm to compute $u(J)$ are identical to those of the PAV algorithm (Algorithm 2.3) with the following changes: Replace $\mu_i, \mu_{\{i\}}, \mu_B$, μ_{B^-}, μ_{B^+}, $x(J)$, and $x_i(J)$ by $\underline{\mu}_i, \underline{\mu}_{\{i\}}, \underline{\mu}_B, \underline{\mu}_{B^-}, \underline{\mu}_{B^+}, u(J)$, and $u_i(J)$, respectively. (Note that some of the values of $\underline{\mu}_i, \underline{\mu}_B$, etc., may equal the endpoint a of I. If $a = -\infty$, use a sufficiently small number for a in computer implementation.)
2. The steps of the algorithm to compute $v(J)$ are identical to those of the PAV algorithm (Algorithm 2.3) with the following changes: Replace $\mu_i, \mu_{\{i\}}, \mu_B$, $\mu_{B^-}, \mu_{B^+}, x(J)$, and $x_i(J)$ by $\overline{\mu}_i, \overline{\mu}_{\{i\}}, \overline{\mu}_B, \overline{\mu}_{B^-}, \overline{\mu}_{B^+}, v(J)$, and $v_i(J)$, respec-

tively. (Note that some of the values of $\overline{\mu}_i, \overline{\mu}_B$, etc., may equal the endpoint b of I. If $b = \infty$, use a sufficiently large number for b in computer implementation.)

Note that the final (optimal) partitions J obtained in statements 1 and 2 above are not necessarily identical. □

The remainder of this section is devoted to validating the algorithm. We omit the proof of the next proposition, as it is similar to that Proposition 3.1.

PROPOSITION 3.5. *Let B_i, $1 \leq i \leq m$, by any nonempty disjoint subsets of N, and let $B = \cup\{B_i : 1 \leq i \leq m\}$. Then*

$$a \leq \min\left\{\underline{\mu}_{B_i} : 1 \leq i \leq m\right\} \leq \underline{\mu}_B \leq \max\left\{\underline{\mu}_{B_i} : 1 \leq i \leq m\right\} < b,$$
$$a < \min\left\{\overline{\mu}_{B_i} : 1 \leq i \leq m\right\} \leq \overline{\mu}_B \leq \max\left\{\overline{\mu}_{B_i} : 1 \leq i \leq m\right\} \leq b. \qquad \square$$

We say that (L, U) is a split of a block B if L and U are nonempty disjoint sets of consecutive indices in B such that $L \cup U = B$. Clearly, there exists a unique index p in B such that $L = \{i \in B : i \leq p\}$ and $U = \{i \in B : i > p\}$; L and U are, indeed, subblocks of B with $p \in L$ and $p + 1 \in U$.

LEMMA 3.6. *Let (L, U) be a split of a block B in any partition of the PAV Algorithm* 3.4. *Then $\underline{\mu}_L > \underline{\mu}_U$ and $\overline{\mu}_L > \overline{\mu}_U$.*

Proof. There exists $p \in B$ such that $L = \{i \in B : i \leq p\}$ and $U = \{i \in B : i > p\}$. We prove the first inequality by induction on successive distinct partitions; the proof for the second is similar. We fix the index p and consider a succession of blocks to which p belongs as the partitions change. Initially, each element of N forms a block. Hence, there exists a first partition J_0 in which p and $p + 1$ belong to the same block B_0, say. Let (L_0, U_0) be the split of B_0 with $p \in L_0$ and $p + 1 \in U_0$. Then L_0 and U_0 were merged to obtain B_0 when J_0 was formed. According to the PAV criterion, $\underline{\mu}_{L_0} > \underline{\mu}_{U_0}$ holds. For any sequence of subsequent partitions which have B_0 as a block, the above strict inequality, of course, continues to hold.

Now suppose that J_1 and J_2 are any two consecutive partitions with corresponding distinct blocks B_1 and B_2 to which p belongs. Then B_2 is obtained from B_1 by a merge operation. Let (L_1, U_1) be the split of B_1 such that $p \in L_1$ and $p + 1 \in U_1$. Let (L_2, U_2) denote the corresponding split of B_2. To prove by induction, we assume that $\underline{\mu}_{L_1} > \underline{\mu}_{U_1}$ holds and show that $\underline{\mu}_{L_2} > \underline{\mu}_{U_2}$. Suppose that $B_2 = B_1 \cup C$, where C (B^+ in the algorithm) is the successor block of B_1 merged with B_1. (The proof when C is the predecessor block of B_1, i.e., when B^- in the algorithm, is similar.) Then clearly, $L_2 = L_1$ and $U_2 = U_1 \cup C$. Also, $\underline{\mu}_{B_1} > \underline{\mu}_C$ by the merge criterion of the algorithm. By Proposition 3.5, we have $\underline{\mu}_{B_1} \leq \max\{\underline{\mu}_{L_1}, \underline{\mu}_{U_1}\}$. By hypothesis, $\underline{\mu}_{L_1} > \underline{\mu}_{U_1}$, hence, the above inequality gives $\underline{\mu}_{B_1} \leq \underline{\mu}_{L_1}$. Hence, $\underline{\mu}_C < \underline{\mu}_{B_1} \leq \underline{\mu}_{L_1}$. Again, since $U_2 = U_1 \cup C$, Proposition 3.5 shows that

$$\underline{\mu}_{U_2} \leq \max\left\{\underline{\mu}_{U_1}, \underline{\mu}_C\right\} < \underline{\mu}_{L_1} = \underline{\mu}_{L_2},$$

which is the required result. □

The simple proof of the next lemma is left to the reader.

LEMMA 3.7. *There exist $\underline{\sigma}, \overline{\sigma} \in I$ such that, for any $B \subset N$ with $\underline{\mu}_B = a$ (respectively, $\overline{\mu}_B = b$), every point in $(a, \underline{\sigma}]$ (respectively, $[\overline{\sigma}, b)$) is a minimizer of F_B.* □

THEOREM 3.8. *The PAV Algorithm* 3.4 *computes the extended minimal and maximal solutions $u = u(J)$ and $v = v(J)$ to Problem P as obtained in Theorem* 3.2.

Proof. We first show that $u \leq x$ for any optimal solution x of P. Let J be the final partition obtained by Algorithm 3.4 when computing u. Suppose that, for some index

k, we have $x_k < u_k$. Let $B = \{p, p+1, \ldots, q\}$ be the last block of J such that for some $j \in B$ we have $x_j < u_j$. Then, by the PAV algorithm, $u_i = u_p = \underline{\mu}_B$ for all $i \in B$ and $x_i \leq x_j < u_j = u_p$ for all i with $p \leq i \leq j$. Let L be the set of all indices i such that $p \leq i \leq n$ and $x_i = x_p$. We then have $u_{q+1} \geq u_p$ if $q < n$. (Note that $u_{q+1} = u_p$ is a possibility.) Again, by the choice of B, we have $x_{q+1} \geq u_{q+1}$ if $q < n$. Consequently, $x_{q+1} \geq u_p > x_p$, if $q < n$, and hence, L has the form $L = \{p, p+1, \ldots, r\}$, where $r \leq q$. Thus $L \subset B$. Clearly, $x_{r+1} > x_r = x_p < u_p$.

Define $y = (y_1, y_2, \ldots, y_n)$ by $y_i = x_i$ for $1 \leq i < p$ and $r < i \leq n$, and $y_i = \min\{u_p, x_{r+1}\} = \lambda$, say, for $p \leq i \leq r$. Then y is a monotone vector and is feasible to P. Also, $x_p < \lambda \leq u_p = \underline{\mu}_B$. If $L \neq B$, then (L, U), where $U = B \backslash L$, is a split of B. By Lemma 3.6, we have $\underline{\mu}_L > \underline{\mu}_U$. By Proposition 3.5, we have $\underline{\mu}_L = \max\{\underline{\mu}_L, \underline{\mu}_U\} \geq \underline{\mu}_B$. Hence $x_p < \lambda \leq \underline{\mu}_L$. Since $a < x_p$, this shows that $\underline{\mu}_L \in I$. Hence, $\underline{\mu}_L$ is the smallest minimizer of F_L, and, by convexity of F_L, we conclude that $F_L(\lambda) - F_L(x_p) < 0$. Similarly, if $L = B$, then $x_p < \lambda \leq \underline{\mu}_B$. Again, since $\underline{\mu}_B$ is the smallest minimizer of F_B, by convexity of F_B, we find that $F_B(\lambda) - F_B(x_p) < 0$. Now

$$\sum_{i=1}^{n} f_i(y_i) - \sum_{i=1}^{n} f_i(x_i) = F_L(\lambda) - F_L(x_p) < 0,$$

in both the cases, $L \neq B$ and $L = B$. It follows that x is not an optimal solution to P. This contradiction shows that $x_i \geq u_i$ for all i.

Let u' denote the u of Theorem 3.2. We wish to show that $u' = u$, where u is as in this theorem, obtained by Algorithm 3.4. As shown above, $u \leq x$ for any optimal x to P. Hence, $u \leq u'$. Let $r = \underline{\text{ind}}(u')$. Then, by Theorem 3.2, we have $\underline{\mu}_i = u'_i = a$, $1 \leq i \leq r$. Since $u \leq u'$, we have $u_i = u'_i = a$, $1 \leq i \leq r$. Now we show that $u_i = u'_i$, $r+1 \leq i \leq n$. Now let J be the final partition obtained by Algorithm 3.4 as applied to the computation of u. As pointed out earlier, some of $\underline{\mu}_i, \underline{\mu}_B$, etc. used in Algorithm 3.4 may equal a. Since $a \notin I$, these quantities are not a minimizer of f_i, F_B, etc., as in that algorithm. Hence, the conclusions of Theorem 2.5 cannot be applied. We now show that the same partition J is obtained by using suitable minimizers of f_i, F_B, etc. in the algorithm so that Theorem 2.5 can be applied. Let B be the set of all blocks of N, including $\{i\}$ for $i \in N$. Define $\underline{\theta} = \min\{\underline{\mu}_B : \mathrm{B} \in B,\ \underline{\mu}_B > a\}$. Then $\underline{\theta} > a$. Now let $\underline{\sigma}$ be as in Lemma 3.7 and $a < \underline{c} < \min\{\underline{\sigma}, \underline{\theta}\}$. Then, by that lemma, $\underline{c}$ is a minimizer of F_B when $\underline{\mu}_B = a$. In Algorithm 3.4 we replace $\underline{\mu}_B$ by $\underline{c}$ whenever $\underline{\mu}_B = a$. In particular, we replace each $\underline{\mu}_i = a$, $1 \leq i \leq r$, by $\underline{c}$. Note that $\underline{\mu}_B > a$ if and only if $\underline{\mu}_B > \underline{c}$. It follows that the conditions such as $\underline{\mu}_{B_1} \leq \underline{\mu}_{B_2}$ or $\underline{\mu}_{B_1} \geq \underline{\mu}_{B_2}$ in the algorithm hold with the original values of $\underline{\mu}_B$ if and only if they hold after replacement by $\underline{c}$ as stated. Hence, in the latter case the algorithm gives the same terminal partition J and, by Theorem 2.5, gives the optimal solution $s = (s_1, s_2, \ldots, s_r, u_{r+1}, \ldots, u_n)$ to P, where $s_i = \underline{c}, 1 \leq i \leq r$. By the definition of u' as an infimum, we conclude that $u' \leq s$. This, together with $u \leq u'$ obtained earlier, gives $u'_i = u_i$, $r+1 \leq i \leq n$. Thus $u = u'$. The proof for v is similar. ☐

4. Problem P with integer constraint. In this section, we consider Problem P with the integer restriction on the variables. We develop PAV algorithms to compute its optimal solution and also its extremal extended integer solutions. It will be seen that all the results are applicable when the variables are restricted to take values in a mesh (discrete set of points) in I. Due to integer constraints, the optimality conditions of Proposition 2.1 cannot be applied; hence, we adopt a different approach. We approximate each convex function f_i by a piecewise linear convex function f_i^0, which agrees with f_i at integer points. Similar ideas have been used earlier by Minoux [11].

Let D denote the set of all integers in I. We consider the following problem.

Problem Q. Find $y_i \in D$, $i \in N$, so as to

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^{n} f_i(y_i) \\ &\text{subject to} \quad y_1 \leq y_2 \leq \cdots \leq y_n. \qquad \Box \end{aligned}$$

If f is convex on I, then an element λ in D that minimizes f over D is called an integer minimizer of f over I. Without the qualifier "integer," a minimizer or a continuous minimizer of f is simply some μ in I that minimizes f over I. An optimal solution to Problem Q is referred to as an integer optimal solution.

Our first proposition links the existence of an integer minimizer of a convex function with its (continuous) minimizer. Its simple proof is left to the reader.

Proposition 4.1. *Suppose that f is a real convex function over $I = (a, b)$. If $D \neq \emptyset$ and I is bounded, then f has an integer minimizer. If I is unbounded, then $D \neq \emptyset$, and in the following two statements,* $1 \Rightarrow 2$. *If $I = (-\infty, \infty)$, then* $2 \Rightarrow 1$. *Thus, if $I = (-\infty, \infty)$, then* $1 \Leftrightarrow 2$.

1. *f has a (continuous) minimizer in I.*
2. *f has an integer minimizer in I.* $\Box$

Note that 2 in the above proposition does not necessarily imply 1. As an example, consider $I = (0, 2)$ or $(0, \infty)$, and $f(x) = x$ on I. Then f has no continuous minimizers in I but has the integer minimizer 1.

Throughout this section we assume that the following Condition B holds.

Condition B. $D \neq \emptyset$, and each $f_i, i \in N$, has an integer minimizer λ_i in I. $\Box$

Analogous to μ_B, F_B has an integer minimizer λ_B as in the following proposition, which may be proved just like Proposition 2.2.

Proposition 4.2. *For any $B \subset N$, F_B has an integer minimizer λ_B in I with*

$$\min\{\lambda_i : i \in B\} \leq \lambda_B \leq \max\{\lambda_i : i \in B\}.$$

Furthermore, Problem Q has an integer optimal solution $y = (y_1, y_2, \ldots, y_n)$ with

$$\min\{\lambda_i : i \in N\} \leq y_i \leq \max\{\lambda_i : i \in N\}, \quad i \in N. \qquad \Box$$

Let I^0 be the smallest interval containing D. Then $I^0 \subset I$. For each function f on I, define a function f^0 on I^0 by $f^0(y) = f(y)$ for all $y \in D$ and by linear interpolation between every two consecutive integers in D. Since f_i is convex, so is f_i^0. Now $F_B = \Sigma_{i \in B} f_i$, for any $B \subset N$. It is easy to see that $F_B^0 = \Sigma_{i \in B} f_i^0$. Since each f_i^0 is convex and linear between any two consecutive integers in D, so is F_B^0. Clearly, if $\lambda \in D$, then λ is an integer minimizer of F_B on I if and only if λ is a continuous minimizer of F_B^0 on I^0. Now we state our algorithm for solving Problem Q.

Algorithm 4.3 (the PAV algorithm for determining an integer optimal solution $y = y(J)$ to Problem Q). The steps of the algorithm are identical to those of the PAV Algorithm 2.3 with the following changes: substitute integer minimizer(s) for minimizers(s) everywhere in the algorithm. Also substitute λ for μ and y for x everywhere (thus, $\mu_i, \mu_B, x_i(J)$, etc. become $\lambda_i, \lambda_B, y_i(J)$, etc.). $\Box$

Theorem 4.4. *The PAV Algorithm* 4.3 *computes an integer optimal solution $y = y(J)$ to Problem Q.*

Proof. In what follows, we consider f_i as a special case $F_{\{i\}}$ of F_B. Consider Problem P^0: Find $x_i \in I^0, i \in N$, so as to minimize $\Sigma_{i=1}^n f_i^0(x_i)$ subject to $x_1 \leq x_2 \leq \cdots \leq x_n$. Algorithm 4.3 as applied to Q and P^0 give the same results because functions F_B and F_B^0 agree on D. Since integer λ_B used in the algorithm is a continuous minimizer of F_B^0 on I^0, by Theorem 2.5, the algorithm computes a continuous optimal solution y to P^0, which has integer components. Clearly, it is an optimal solution to Q. □

Now we establish a result that can be applied to the least squares distance function.

THEOREM 4.5. *Suppose that each F_B is symmetric around some continuous minimizer μ_B and, in particular, so is each $f_i = F_{\{i\}}$ around some minimizer μ_i. Then an integer optimal solution to Problem Q is obtained by rounding (to the nearest integer) a continuous optimal solution to Problem P given by Algorithm* 2.3 *when minimizers μ_B and μ_i stated above are used for computation in that algorithm.*

Proof. Let λ_B denote an integer minimizer of F_B. Clearly, under the assumption of symmetry, λ_B is obtained by rounding μ_B to the nearest integer. We now show that the final (optimal) partition J obtained by Algorithm 2.3 is also an optimal partition for the integer Problem Q. We do this by comparing the steps of Algorithms 2.3 and 4.3. The condition $\mu_{B_1} > \mu_{B_2}$ in Algorithm 2.3 implies the condition $\lambda_{B_1} \geq \lambda_{B_2}$, since λ_B is obtained by rounding μ_B. Now Algorithm 4.3 merges blocks B_1 and B_2 if $\lambda_{B_1} > \lambda_{B_2}$. But, if it also unnecessarily merged these blocks when $\lambda_{B_1} = \lambda_{B_2}$, that will still lead to optimality. Also, $\mu_{B_1} \leq \mu_{B_2}$ implies $\lambda_{B_1} \leq \lambda_{B_2}$. Thus the steps of Algorithm 2.3 imply those of Algorithm 4.3 with an insignificant change when $\lambda_{B_1} = \lambda_{B_2}$. This proves that J is optimal for Q. Now, Algorithm 4.3 computes the final (optimal) solution by letting $y_i(J) = \lambda_B$, $i \in B$, where B is a block of J. Since λ_B is rounded μ_B, the conclusion of the theorem follows. □

If $f_i(x_i) = w_i(c_i - x_i)^2$, $w_i > 0$, then f_i is symmetric around $\mu_i = c_i$. Also, $F_B(x) = W_B(\mu_B - x)^2$, where $W_B = \Sigma_{i \in B} w_i$ and $\mu_B = \Sigma_{i \in B}(w_i \mu_i)/W_B$. Thus F_B is symmetric around μ_B. Thus Theorem 4.6 applies to give a result of Goldstein and Kruskal [7]: an optimal solution of the integer isotonic regression problem is obtained by rounding the unique optimal solution of the continuous isotonic regression problem. Note that the integer problem does not, in general, have a unique solution.

As in section 3, we now obtain the extended minimal and maximal integer solutions, z and t, respectively, to Problem Q. These provide the lower and upper bounds on all optimal solutions to Problem Q, and subvectors formed by components of z and t give optimal solutions to certain subproblems of Q.

If T_B is the set of all integer minimizers of the convex function F_B, then, by Proposition 4.2, $T_B \neq \emptyset$. Define $\underline{\lambda}_B = \inf T_B$ and $\overline{\lambda}_B = \sup T_B$. Then $\underline{\lambda}_B$ (respectively, $\overline{\lambda}_B$) is either an integer or $-\infty$ (respectively, $+\infty$). Note that $\underline{\lambda}_B$ (respectively, $\overline{\lambda}_B$) equals the endpoint a (respectively, b) of I if and only if $a = -\infty$ (respectively, $b = \infty$). By convexity of F_B, we have $T_B = [\underline{\lambda}_B, \overline{\lambda}_B] \cap D$. When $B = \{i\}$, $i \in N$, the above defines the corresponding quantities $\underline{\lambda}_i$ and $\overline{\lambda}_i$ for $f_i = F_{\{i\}}$. If $\underline{\lambda}_B$ (respectively, $\overline{\lambda}_B$) is in I, then it is the smallest (respectively, largest) integer minimizer of F_B. A proposition analogous to Proposition 3.1 may be stated and proved for $\underline{\lambda}_B$, $\overline{\lambda}_B$, and an integer solution to Q. A vector $y = (y_1, y_2, \ldots, y_n)$ is called an extended integer monotone vector if $y_1 \leq y_2 \leq \cdots \leq y_n$ and each y_i is an integer, $-\infty$ or $+\infty$. Just like Subproblem SP(c, d), we define a Subproblem SQ(c, d), indexed by $c, d \in N$, as follows.

Subproblem SQ(c, d). Find $y_i \in D$, $c \le i \le d$, so as to

$$\begin{aligned} &\text{minimize} \quad \sum_{i=c}^{d} f_i(y_i) \\ &\text{subject to} \quad y_c \le y_{c+1} \le \cdots \le y_d. \qquad \square \end{aligned}$$

Clearly, $Q = \text{SQ}(1, n)$. The following theorem is similar to Theorem 3.2, but note the one difference regarding p and q. If a (respectively, b) is finite, then $p = 0$ (respectively, $q = n + 1$).

Theorem 4.6 (minimal and maximal extended integer solutions). *There exist two unique extended integer vectors $z = (z_1, z_2, \ldots, z_n)$ and $t = (t_1, t_2, \ldots, t_n)$ for Problem Q with the following properties, where $p = \underline{\text{ind}}(z)$, if $a = -\infty$, and 0 otherwise, and where $q = \overline{\text{ind}}(t)$, if $b = \infty$, and $n + 1$ otherwise.*

1. *$a \le z_i < b$, $a < t_i \le b$, and $z_i \le t_i$ for all i.*
2. *If $y = (y_1, y_2, \ldots, y_n)$ is any optimal solution to Q, then $z_i \le y_i \le t_i$ for all i.*
3. i. *$0 \le p \le n$, $\underline{\lambda}_i = z_i = a = -\infty$ for $1 \le i \le p$, and $\underline{\lambda}_{p+1} \ge z_{p+1} > a$ if $p < n$.*
   ii. *$1 \le q \le n + 1$, $\overline{\lambda}_i = t_i = b = \infty$ for $q \le i \le n$ and $\overline{\lambda}_{q-1} \le t_{q-1} < b$ if $q > 1$.*
4. i. *$(z_{p+1}, z_{p+2}, \ldots, z_n)$ is the minimal optimal solution to* SQ$(p + 1, n)$. *If $p = 0$ (equivalently, $z_i \in D$, $i \in N$), then z is the minimal optimal solution to Q.*
   ii. *$(t_1, t_2, \ldots, t_{q-1})$ is the maximal optimal solution to* SQ$(1, q-1)$. *If $q = n+1$ (equivalently, $t_i \in D, i \in N$), then t is the maximal optimal solution to Q.*
5. i. *z_i is a continuous minimizer of f_i for $\max\{p + 1, q\} \le i \le n$.*
   ii. *t_i is a continuous minimizer of f_i for $1 \le i \le \min\{p, q - 1\}$.*
   iii. *If $p + 1 \le q - 1$, then $(z_{p+1}, z_{p+2}, \ldots, z_{q-1})$ and $(t_{p+1}, t_{p+2}, \ldots, t_{q-1})$ are, respectively, the minimal and maximal optimal solutions to Subproblem* SQ$(p + 1, q - 1)$.
6. *A monotone integer vector $y = (y_1, y_2, \ldots, y_n)$ is an optimal solution to Q if and only if the following three conditions are satisfied: y_i is a continuous minimizer of f_i (equivalently, $-\infty = a < y_i \le \overline{\lambda}_i$) for $1 \le i \le \min\{p, q - 1\}$; if $p + 1 \le q - 1$, then $(y_{p+1}, y_{p+2}, \ldots, y_{q-1})$ is an integer optimal solution to* SQ$(p + 1, q - 1)$*; and y_i is a continuous minimizer of f_i (equivalently, $\underline{\lambda}_i \le y_i < b = \infty$) for $\max\{p + 1, q\} \le i \le n$.* □

Both z and t may be easily computed by an algorithm similar to Algorithm 3.4, which uses $\underline{\lambda}_B$ and $\overline{\lambda}_B$ instead of $\underline{\mu}_B$ and $\overline{\mu}_B$ in computations. The proof of Theorem 4.6 and the algorithm is similar to that in section 3.

Appendix. *Proof of Theorem* 3.2*, part* 3.

For convenience, we let Σ stand for $\Sigma_{i=1}^{n}$ below unless otherwise stated. Suppose that $x = (x_1, x_2, \ldots, x_n)$ and $y = (y_1, y_2, \ldots, y_n)$ are in X. Let $x \wedge y = (x_1 \wedge y_1, x_2 \wedge y_2, \ldots, x_n \wedge y_n)$ and $x \vee y = (x_1 \vee y_1, x_2 \vee y_2, \ldots, x_n \vee y_n)$, where $x_i \wedge y_i = \min\{x_i, y_i\}$ and $x_i \vee y_i = \max\{x_i, y_i\}$ denote, respectively, the pointwise minimum and maximum of x and y. Since x and y are monotone vectors, so are $x \wedge y$ and $x \vee y$, as may be easily verified. We first show that both $x \wedge y$ and $x \vee y$ are in X. We have

$$f_i(x_i \wedge y_i) + f_i(x_i \vee y_i) = f_i(x_i) + f_i(y_i), \quad i \in N.$$

This follows by considering the two cases $x_i \le y_i$ and $x_i > y_i$. Hence, $\Sigma f_i(x_i \wedge y_i) + \Sigma f_i(x_i \vee y_i) = 2\theta$, where $\theta = \Sigma f_i(x_i) = \Sigma f_i(y_i)$ is the optimal objective value of P.

But $\Sigma f_i(x_i \wedge y_i) \geq \theta$ and $\Sigma f_i(x_i \vee y_i) \geq \theta$, since $x \wedge y$ and $x \vee y$ are feasible to P. We conclude that each of these sums equals θ, and thus $x \wedge y$ and $x \vee y$ are in X. It follows by induction that the pointwise minimum and maximum of a finite number of elements in X are also in X.

From part 1 of this theorem, $u_i < b$ for all i, and hence $0 \leq p \leq n$. Since $a < u_{p+1}$, we can choose numbers α, β so that $a < \alpha \leq u_i < \beta < b$ for $p+1 \leq i \leq n$. Let $\varepsilon > 0$. Then, by continuity of f_i on $[\alpha, \beta]$, there exists $\rho > 0$, sufficiently small, such that $u_i + \rho \leq \beta$ and $|f_i(w) - f_i(t)| < \varepsilon/n$ for all $p+1 \leq i \leq n$, whenever $w, t \in [\alpha, \beta]$ and $|w - t| \leq \rho$. Now let $s = (s_1, s_2, \ldots, s_p, u_{p+1}, \ldots, u_n)$ be a monotone vector with $a < s_i \leq \overline{\mu}_i$, $1 \leq i \leq p$. Then there exists $x^i \in X$ such that $a = u_i < x_i^i \leq s_1, 1 \leq i \leq p$, and $a < u_i \leq x_i^i \leq u_i + \rho$, $p+1 \leq i \leq n$. Let $z = x^1 \wedge x^2 \wedge \cdots \wedge x^n$. As shown above, $z \in X$. Clearly, then, $a < z_i \leq x_i^i \leq s_1$ for $1 \leq i \leq p$, and $u_i \leq z_i \leq x_i^i \leq u_i + \rho$ for $p+1 \leq i \leq n$. Since $z_i \leq s_1 \leq s_i \leq \overline{\mu}_i$, $1 \leq i \leq p$, by convexity of f_i, we have $f_i(z_i) \geq f_i(s_i)$, $1 \leq i \leq p$. Again, since $z_i, u_i \in [\alpha, \beta]$ and $|z_i - u_i| \leq \rho$ for $p+1 \leq i \leq n$, we have $f_i(u_i) \leq f_i(z_i) + \varepsilon/n$, $p+1 \leq i \leq n$. Hence $A = \Sigma_{i=1}^p f_i(s_i) \leq \Sigma_{i=1}^p f_i(z_i)$ and $B = \Sigma_{i=p+1}^n f_i(u_i) \leq \Sigma_{i=p+1}^n f_i(z_i) + \varepsilon$. Thus, $A + B \leq \Sigma f_i(z_i) + \varepsilon \leq \theta + \varepsilon$. Since ε is arbitrary, we conclude that $s \in X$. Let $t = (t_1, t_2, \ldots, t_p, u_{p+1}, \ldots, u_n)$ be another monotone vector with $a < t_i < s_i \leq \overline{\mu}_i$, $1 \leq i \leq p$. Then, as shown above, $t \in X$. Consequently, $\Sigma_{i=1}^p f_i(t_i) = \Sigma_{i=1}^p f_i(s_i)$. Again, since $t_i < s_i \leq \overline{\mu}_i$, by convexity, we must have $f_i(t_i) \geq f_i(s_i)$, $1 \leq i \leq p$. It follows that $f_i(t_i) = f_i(s_i), 1 \leq i \leq p$, which in turn equals the minimum value of f_i, showing that $(a, \overline{\mu}_i] \cap I$ is the set of minimizers of f_i, $1 \leq i \leq p$. Thus, $\underline{\mu}_i = u_i = a$ and s_i minimizes f_i for $1 \leq i \leq p$.

To show that $\underline{\mu}_{p+1} \geq u_{p+1}$, assume the contrary: that $\underline{\mu}_{p+1} < u_{p+1}$. Define a monotone vector $\hat{s} = (s_1, s_2, \ldots, s_{p+1}, u_{p+2}, \ldots, u_n)$, where $a < s_i \leq \overline{\mu}_i$, $1 \leq i \leq p$, $\underline{\mu}_{p+1} \leq s_{p+1} < u_{p+1}$, and $s_{p+1} \leq \overline{\mu}_{p+1}$. This is clearly possible, since $\underline{\mu}_{p+1} \leq \overline{\mu}_{p+1}$. Again, since $\underline{\mu}_{p+1} \leq s_{p+1} \leq \overline{\mu}_{p+1}$, we find that s_{p+1} minimizes f_{p+1}, and, thus, $f_{p+1}(s_{p+1}) \leq f_{p+1}(u_{p+1})$. Since $s_p \leq s_{p+1} < u_{p+1}$, we have that $s = (s_1, s_2, \ldots, s_p, u_{p+1}, \ldots, u_n)$ is a monotone vector. Again, as shown earlier, s is optimal to P, and consequently, $\hat{s}$ is also optimal to P. Hence, by the definition of u_{p+1} as an infimum, we have $u_{p+1} \leq s_{p+1}$, a contradiction. We proved part i; proof for part ii is similar. □

REFERENCES

[1] A. V. Aho, J. E. Hopcroft, and J. D. Ullman, *Design and Analysis of Computer Algorithms*, Addison–Wesley, Reading, MA, 1974.

[2] M. J. Best and N. Chakravarti, *Active set algorithms for isotonic regression: A unifying framework*, Math. Programing, 47 (1990), pp. 425–439.

[3] M. J. Best, N. Chakravarti, and V. A. Ubhaya, *Minimizing Separable Convex Functions Subject to Simple Chain Constraints*, Technical Report, Department of Computer Science and Operations Research, 258 IACC Building, North Dakota State University, Fargo, ND, 1996.

[4] N. Chakravarti, *Isotonic median regression: A linear programming approach*, Math. Oper. Res., 14 (1989), pp. 303–308.

[5] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983; reprinted as Classics in Appl. Math. 5, SIAM, Philadelphia, 1990.

[6] S. J. Grotzinger and C. Witzgall, *Projection onto order simplexes*, Appl. Math. Optim., 12 (1984), pp. 247–270.

[7] A. J. Goldstein and J. B. Kruskal, *Least square fitting for monotonic functions having integer values*, J. Amer. Statist. Assoc., 71 (1976), pp. 370–373.

[8] D. Landers and L. Rogge, *Best approximants in L_Φ-spaces*, Z. Wahrsch. verw. Gebiete, 51 (1980), pp. 215–237.

[9] D. Landers and L. Rogge, *Natural choice of L_1-approximants*, J. Approx. Theory, 33 (1981), pp. 268–280.

[10] M.-H. Liu and V. A. Ubhaya, *Integer isotone optimization*, SIAM J. Optim., 7 (1997), pp. 1152–1159.

[11] M. Minoux, *Solving integer minimum cost flows with separable convex cost objective polynomially*, Math. Programming Stud., 26 (1986), pp. 237–239.

[12] F. P. Preparata and M. I. Shamos, *Computational Geometry*, Springer-Verlag, New York, 1985.

[13] T. Robertson and F. T. Wright, *Algorithms in order restricted statistical inference and the Cauchy mean value property*, Ann. Statist., 8 (1980), pp. 645–651.

[14] T. Robertson, F. T. Wright, and R. L. Dykstra, *Order Restricted Statistical Inference*, John Wiley, New York, 1988.

[15] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[16] L. B. Schwarz and L. Schrage, *Optimal and system-myopic policies for multi-echelon production/inventory assembly systems*, Management Sci., 21 (1975), pp. 1285–1294.

[17] U. Strömberg, *An algorithm for isotonic regression with arbitrary convex distance function*, Comput. Statist. Data Anal., 11 (1991), pp. 205–219.

[18] V. A. Ubhaya, *Isotone optimization*, I, II, J. Approx. Theory, 12 (1974), pp. 146–159, 315–331.

[19] V. A. Ubhaya, *An O(n) algorithm for discrete n-point convex approximation with applications to continuous case*, J. Math. Anal. Appl., 72 (1979), pp. 338–354.

[20] V. A. Ubhaya, *An O(n) algorithm for least squares quasi-convex approximation*, Comput. Math. Appl., 14 (1987), pp. 583–590.

[21] V. A. Ubhaya, *Lipschitzian selections in best approximation by continuous functions*, J. Approx. Theory, 61 (1990), pp. 40–52.

SIAM J. Optim.
Vol. 10, No. 3, pp. 673–696

A SPECTRAL BUNDLE METHOD FOR SEMIDEFINITE PROGRAMMING[*]

C. HELMBERG[†] AND F. RENDL[‡]

Abstract. A central drawback of primal-dual interior point methods for semidefinite programs is their lack of ability to exploit problem structure in cost and coefficient matrices. This restricts applicability to problems of small dimension. Typically, semidefinite relaxations arising in combinatorial applications have sparse and well-structured cost and coefficient matrices of huge order. We present a method that allows us to compute acceptable approximations to the optimal solution of large problems within reasonable time.

Semidefinite programming problems with constant trace on the primal feasible set are equivalent to eigenvalue optimization problems. These are convex nonsmooth programming problems and can be solved by bundle methods. We propose replacing the traditional polyhedral cutting plane model constructed from subgradient information by a semidefinite model that is tailored to eigenvalue problems. Convergence follows from the traditional approach but a proof is included for completeness. We present numerical examples demonstrating the efficiency of the approach on combinatorial examples.

Key words. eigenvalue optimization, convex optimization, semidefinite programming, proximal bundle method, large-scale problems

AMS subject classifications. Primary, 65F15, 90C25; Secondary, 52A41, 90C06

PII. S1052623497328987

1. Introduction.

The development of interior point methods for semidefinite programming [19, 31, 1, 46] has increased interest in semidefinite modeling techniques in several fields such as control theory, eigenvalue optimization, and combinatorial optimization. In fact, interior point methods proved to be very useful and reliable solution methods for semidefinite programs of moderate size. However, if the problem is defined over large matrix variables or a huge number of constraints, interior point methods grow terribly slow and consume huge amounts of memory. The most efficient methods of today [15, 23, 2, 32, 45, 29] are primal-dual methods that require, in each iteration of the interior point method, the factorization of a dense matrix of order equal to the number of constraints and one to three factorizations of the positive semidefinite matrix variables within the line search. For a typical workstation this restricts the number of constraints to 2,000 and the size of the matrix variables to 500 if reasonable performance is required. For larger problems time and memory requirements are prohibitive. It is important to realize that either the primal or the dual matrix is generically dense even if cost and coefficient matrices are very sparse. Very recently, a pure dual approach was proposed in [4] which offers some possibilities to exploit sparsity. It is too early to judge the potential of this method.

In combinatorial optimization, semidefinite relaxations were introduced in [27]. At that time they were mainly considered a theoretical tool for obtaining strong bounds [11, 28, 40]. With the development of interior point methods, hopes soared high that these relaxations could be of practical value. Within a short time several ap-

[*] Received by the editors October 20, 1997; accepted for publication (in revised form) January 25, 1999; published electronically February 29, 2000.

http://www.siam.org/journals/siopt/10-3/32898.html

[†] Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustraße 7, 14195 Berlin, Germany (helmberg@zib.de, http://www.zib.de /helmberg).

[‡] Universität Klagenfurt, Institut für Mathematik, Universitätsstr. 65-67, A 9020 Klagenfurt, Austria (franz.rendl@uni-klu.ac.at). This author was supported by Austrian FWF project P12660-MAT.

proximation algorithms relying on semidefinite programming were published, most of them based on the approach by Goemans and Williamson [8]. On the implementation side [14, 16, 20] cutting plane approaches for semidefinite relaxations of constrained quadratic 0-1 programming problems proved to yield solutions of high quality. However, as mentioned above, they were very expensive to compute even for problems of small size (a few hundred 0-1 variables). Problems arising in practical applications (starting with a few thousand 0-1 variables) were out of reach. We believe that the method proposed in this paper will open the door to problems of this size.

Although combinatorial applications are our primary concern we stress that the method is not restricted to this kind of problem. In fact, it will be a useful alternative to interior point methods whenever the number of constraints or the order of the matrices is quite large.

We transform a standard dual semidefinite program into an eigenvalue optimization problem by reformulating the semidefinite constraint as a nonnegativity constraint on the minimal eigenvalue of the slack matrix variable and lifting this constraint into the cost function by means of a Lagrange multiplier. The correct value of the Lagrange multiplier is known in advance if the primal feasible matrices have constant trace. (This is the case for the combinatorial applications we have in mind.)

In this paper we develop a bundle method for solving the problem of minimizing the maximal eigenvalue of an affine matrix function with an additional linear objective term. These functions are well known to be convex and nonsmooth. A very general method for optimizing nonsmooth convex functions is the bundle method; see, e.g., [21, 42, 17, 18]. In each step the function value and a subgradient of the function is computed for some specific point. A cutting plane model of the function is formed using the collected subgradients. The minimizer of the cutting plane model, augmented by a regularization term, yields the new point. In the case of eigenvalue optimization, the subgradient is formed by means of an eigenvector for the maximal eigenvalue. Extremal eigenvalues and associated eigenvectors of large symmetric matrices can be computed efficiently by Lanczos methods (see, e.g., [9]). Lanczos methods need a subroutine that computes the product of the matrix with a vector. This allows exploitation of any kind of structure present in the matrix.

The polyhedral cutting plane model used in traditional bundle algorithms is updated by new subgradient information so as to approximate well the subdifferential, and thus the function itself, in the vicinity of the current point. For eigenvalue optimization problems the subdifferential is generated by a semidefinite set, in particular by the intersection of a simple affine constraint and a face of the semidefinite cone. This suggests using, instead of the traditional polyhedral cutting plane model, a semidefinite cutting plane model that works with an approximation of this face of the semidefinite cone. This specialization of the cutting plane model is the main contribution of the paper.

The semidefinite bundle approach allows for an intriguing interpretation in terms of the original semidefinite program. The cutting plane model requires that the dual slack matrix of the semidefinite program is positive semidefinite only with respect to a subspace of vectors; thus it may be interpreted as a relaxation of the dual semidefinite program. In general the optimal solution of this relaxed semidefinite problem will produce an indefinite dual slack matrix. One or more of the negative eigenvalues and corresponding eigenvectors of the slack matrix are used to update the subspace in order to improve the relaxation, and the process is iterated.

This process trivially provides the optimal solution if the subspace grows to the full space. However, we show that, during the algorithm, generically the dimension of the subspace is bounded by (roughly) the square root of the number of constraints. If this is still considered too large, the introduction of an aggregate subgradient guarantees convergence for restricted bundle sizes. In the extreme the bundle may consist of one new eigenvector for the maximal eigenvalue only.

In contrast, the "classical" algorithms of Cullum, Donath, and Wolfe [6] and Polak and Wardi [38] require in each iteration the computation of all eigenvectors to eigenvalues within an ε-distance of the maximal eigenvalue. Thus, close to the optimal solution, this number is at least as large as the multiplicity of the maximal eigenvalue in the optimal solution. In the quadratically convergent algorithm of Overton [35], each step is computed from a complete spectral decomposition of the matrix and a guess of the exact multiplicity of the maximal eigenvalue in the optimal solution. In recent work [33, 34] Oustry reinterprets the algorithm of Overton within the framework of the $\mathcal{U}$-Lagrangian introduced in [26] and embeds it in a first-order method to ensure global convergence. Again, for global convergence the approach relies on the spectrum of all eigenvalues within ε-distance of the maximal eigenvalue and makes use of the entire spectral information to obtain local quadratic convergence.

Because of the restricted bundle size quadratic convergence is out of reach for our algorithm; it is a first-order method only. In principle, convergence follows from the traditional approach (see, e.g., [21]) but we include a proof for completeness. We also present a primal-dual interior point code for solving the quadratic semidefinite programming problems associated with the semidefinite cutting plane models and discuss efficiency aspects. The properties of the algorithm are illustrated on several combinatorial examples.

In section 2 some basic properties of semidefinite programs are stated. Then we transform semidefinite programs into eigenvalue optimization problems. Section 3 introduces the bundle method. The algorithm and the proof of convergence is given in section 4. The quadratic semidefinite subproblems arising in the bundle method can be solved by interior point methods, as explained in section 5. Section 6 gives an outline of the implementation and briefly discusses the computation of the maximal eigenvalue and an associated eigenvector. Numerical examples for combinatorial problems are presented in section 7. We conclude the paper with a summary and possible extensions and improvements in section 8. For the convenience of the reader, an appendix explaining the notation and the symmetric Kronecker product is included at the end of the paper.

2. Semidefinite programs and eigenvalue optimization. We denote the set of symmetric matrices of order n by S_n, which we regard as a space isomorphic to $\mathbb{R}^{\binom{n+1}{2}}$. As a scalar product of $A, B \in S_n$ (or more general, $A, B \in \mathbb{R}^{m\times n}$) we use $\langle A, B\rangle = \operatorname{tr}(B^T A)$, where the trace is the sum of the diagonal elements of a square matrix. We will often use the same symbol for the canonical scalar product of vectors $a, b \in \mathbb{R}^m$, $\langle a, b\rangle = b^T a$; the appropriate space will be clear from the context. The subset of positive semidefinite matrices S_n^+ is a full-dimensional, nonpolyhedral convex cone in S_n and defines a partial order on the symmetric matrices by$A \succeq B \iff (A - B) \in S_n^+$. Positive definite matrices are denoted by $A \in S_n^{++}$ or $A \succ 0$.

Consider the standard primal-dual pair of semidefinite programs,

$$\text{(P) } qquad \begin{array}{ll} \max & \langle C, X\rangle \\ \text{s.t.} & \mathcal{A}X = b \\ & X \succeq 0. \end{array} \qquad \text{(D)} \quad \begin{array}{ll} \min & b^T y \\ \text{s.t.} & Z = \mathcal{A}^T y - C \\ & Z \succeq 0. \end{array}$$

Here $\mathcal{A} : S_n \to \mathbb{R}^m$ is a linear operator and $\mathcal{A}^T : \mathbb{R}^m \to S_n$ is its adjoint operator, defined by $\langle \mathcal{A}X, y \rangle = \langle X, \mathcal{A}^T y \rangle$ for all $X \in S_n$ and $y \in \mathbb{R}^m$. They are of the form

$$\mathcal{A}X = \begin{bmatrix} \langle A_1, X \rangle \\ \vdots \\ \langle A_m, X \rangle \end{bmatrix} \quad \text{and} \quad \mathcal{A}^T y = \sum_{i=1}^{m} y_i A_i$$

with $A_i \in S_n$, $i = 1, \ldots, m$. $C \in S_n$ is the cost matrix, $b \in \mathbb{R}^m$ the right-hand-side vector.

We assume some constraint qualification to hold, so that these problems satisfy strong duality in the sense that for any optimal solution X^* of (P) and any optimal solution (y^*, Z^*) of (D) we have

$$X^* Z^* = 0. \tag{2.1}$$

The following assumption allows a simple reformulation of the dual (D) as an eigenvalue optimization problem. We assume that

$$\mathcal{A}X = b \quad \text{implies} \quad \operatorname{tr} X = a \tag{2.2}$$

for some constant $a > 0$. In this case we can add $\operatorname{tr} X = a$ as a redundant constraint to the primal problem and obtain the following dual equivalent to (D)

$$\min \; a\lambda + b^T y \;\; \text{s.t.} \;\; Z = \mathcal{A}^T y + \lambda I - C \succeq 0.$$

Now $a > 0$ implies $X \neq 0$ at the optimum, so any optimal Z of this dual is singular. Therefore, all dual optimal solutions Z satisfy $0 = \lambda_{\max}(-Z)$, leading to

$$\lambda = \lambda_{\max}(C - \mathcal{A}^T y).$$

Thus, we have shown that (D) is equivalent to $\min_y a\lambda_{\max}(C - \mathcal{A}^T y) + b^T y$. For notational convenience we assume $a = 1$ and deal with the following problem:

$$\min_{y \in R^m} \lambda_{\max}(C - \mathcal{A}^T y) + b^T y. \tag{E}$$

The eigenvalue problem (E) is a convex, nonsmooth optimization problem. It is well studied in the literature. Here we only recall some basic facts. The function

$$\lambda_{\max}(X) = \max\{\langle W, X \rangle : \operatorname{tr} W = 1, W \succeq 0\}$$

is differentiable if and only if the maximal eigenvalue has multiplicity 1. When optimizing eigenvalue functions, the optimum is generically attained at matrices whose maximal eigenvalue has multiplicity larger than 1. In this case one has to consider the subdifferential of $\lambda_{\max}$ at X,

$$\partial\lambda_{\max}(X) = \{W \succeq 0 : \langle W, X \rangle = \lambda_{\max}(X), \operatorname{tr} W = 1\}$$

(see, e.g., [35]). In particular, for any $v \in \mathbb{R}^n$ belonging to the eigenspace of the maximal eigenvalue of X, $W = vv^T$ is contained in the subdifferential of $\lambda_{\max}$ at X. For the function of interest,

$$f(y) = \lambda_{\max}(C - \mathcal{A}^T y) + b^T y,$$

the subdifferential of f at y can be derived by standard rules (see [17]),

$$\partial f(y) = \left\{ b - \mathcal{A}W : \left\langle W, C - \mathcal{A}^T y \right\rangle = \lambda_{\max}(C - \mathcal{A}^T y), \operatorname{tr} W = 1, W \succeq 0 \right\}.$$

Observe that the set of all subgradients is bounded.

Remark 2.1. Even though our assumption (2.2) might look artificial, it does hold for semidefinite programs arising from quadratic 0-1 optimization. It also holds for many other semidefinite programs derived as relaxations of combinatorial optimization problems; see, for instance, [1, 12, 24].

3. The bundle method.

In this section we develop a new method for minimizing f. We use two classical ingredients, the *proximal point idea* and the *bundle concept*. The new contribution lies in the way that we derive the new iterate from the "bundle" of subgradient information collected from previous iterates. Since our approach builds on several subtle ideas, we proceed in small steps and explain first how we derive a minorant of f from local information.

3.1. Minorizing f by $\hat{f}$.

Our first goal is to obtain a minorant $\hat{f}$ of f which approximates f in the neighborhood of the current iterates reasonably well, and which is easier to handle than f. Introducing the function

$$L(W, y) := \left\langle C - \mathcal{A}^T y, W \right\rangle + b^T y$$

we can express $f(y)$ as

$$f(y) = \max\{L(W, y) : W \succeq 0, \operatorname{tr} W = 1\}.$$

This formulation shows that lower approximations of f can be obtained by constraining W to a subset of all semidefinite matrices with $\operatorname{tr} W = 1$.

We propose the following choice for this subset. Let $P \in \mathbb{R}^{n \times r}$ with $P^T P = I_r$, and $\overline{W} \in S_n^+$ with $\operatorname{tr} \overline{W} = 1$ be two matrices. We restrict W to be contained in the set

$$\widehat{\mathcal{W}} = \left\{ \alpha \overline{W} + P V P^T : \alpha + \operatorname{tr} V = 1, \alpha \geq 0, V \succeq 0 \right\}. \tag{3.1}$$

The (convex) minorant $\hat{f}$, defined through P and $\overline{W}$, now reads

$$\hat{f}(y) := \max\{L(W, y) : W \in \widehat{\mathcal{W}}\}.$$

By definition, we have $\hat{f}(y) \leq f(y) \quad \forall y$. If, for some $\hat{y} \in \mathbb{R}^n$, $vv^T \in \widehat{\mathcal{W}}$ for some eigenvector v to $\lambda_{\max}(C - \mathcal{A}^T \hat{y})$, then $\hat{f}(\hat{y}) = f(\hat{y})$. This is, e.g., the case if v is a column of P or v is contained in the range space of P.

The intuitive idea behind our specific choice of $\widehat{\mathcal{W}}$ is as follows: the matrix P contains subgradient information from the current point $\hat{y}$, and perhaps from previous iterates. We explain below, in detail, how we propose to select and update the matrix P. For computational efficiency, we would like to keep the number r of columns of P small, independent of the multiplicity of the largest eigenvalue. Therefore we collect indispensable subgradient information, which has to be removed from P, in an aggregate subgradient. This *aggregation* is the final ingredient of our model of f. The matrix $\overline{W}$ plays the role of an aggregate subgradient. Again, we will discuss below how $\overline{W}$ is updated during the algorithm. The main point here is that instead of optimizing over all semidefinite matrices W, we constrain ourselves to a small subset.

Remark 3.1. If we set $\overline{W} = 0$ and use for the matrix P a set of eigenvectors for the r largest eigenvalues at $\hat{y}$, we would end up with a model closely related to the approach from [6]. In this case it would be important to select r at least as large as the multiplicity of the largest eigenvalue. In our present approach this is not necessary.

3.2. Proximal point idea. The next goal is to minimize $\hat{f}$ instead of f. Since $\hat{f}$ is built from local information from a few previous iterates, this model function is unlikely to be reliable for points far from the current iterate. Therefore we use the *proximal point idea* and add a penalty term for the displacement from the current point. Thus, we determine a new candidate y from the current iterate $\hat{y}$ by solving the following convex problem, referred to as the augmented model. (Here $u > 0$ is some fixed real weight.)

$$\min_y \hat{f}(y) + \frac{u}{2} \|y - \hat{y}\|^2 .$$

We note that this minimization problem corresponds to the Lagrangian relaxation of $\min\{\hat{f}(y) : \|y - \hat{y}\|^2 \le s^2\}$. Thus we replace the original function f by its minorant $\hat{f}$ and minimize locally around $\hat{y}$. The weight u controls (indirectly) the radius s of the sphere around $\hat{y}$, over which we minimize. Substituting the definition of $\hat{f}$, this problem is the same as

$$\min_y \quad \max_{W \in \widehat{\mathcal{W}}} \quad L(W, y) + \frac{u}{2} \|y - \hat{y}\|^2 . \tag{3.2}$$

This problem can be simplified, because y is unconstrained. Note that

$$L(W, y) = \left\langle C - \mathcal{A}^T \hat{y}, W \right\rangle + b^T \hat{y} + \langle b - \mathcal{A}W, y - \hat{y} \rangle .$$

Therefore, we obtain

$$\begin{aligned} &\min_y \max_{W \in \widehat{\mathcal{W}}} L(W, y) + \frac{u}{2} \|y - \hat{y}\|^2 \\ &\qquad = \max_{W \in \widehat{\mathcal{W}},\ b - \mathcal{A}W + u(y - \hat{y}) = 0} L(W, y) + \frac{u}{2} \|y - \hat{y}\|^2 \\ &\qquad = \max_{W \in \widehat{\mathcal{W}}} \left\langle C - \mathcal{A}^T \hat{y}, W \right\rangle + b^T \hat{y} - \frac{1}{2u} \langle \mathcal{A}W - b, \mathcal{A}W - b \rangle . \end{aligned}$$

The first equality follows from interchanging min and max (see Corollary 37.3.2 of [41]) and using first-order optimality for the inner minimization with respect to y,

$$y = \hat{y} + \frac{1}{u}(\mathcal{A}W - b). \tag{3.3}$$

The final problem is a semidefinite program with (concave) quadratic cost function. We will discuss in section 5 how problems of this kind can be solved efficiently. Its optimal solution W^{k+1} gives the new trial point y by (3.3).

Remark 3.2. The choice of the weight u is somewhat of an art. There are several clever update strategies published in the literature; see, for instance, [21, 42].

3.3. One iteration of the algorithm. The main ingredients of our approach have now been explained, so we can give a formal description of a general iteration k of the algorithm. To be consistent with the notation of the algorithm given in section 4, let us denote by x^k what was called $\hat{y}$ in section 3.2. The algorithm may have to compute several trial points $y^{k+1}, y^{k+2}, \ldots$ while keeping the same $x^k = x^{k+1} = \cdots$ if progress is not considered satisfactory (*null step*). For each y^{k+1} the function is evaluated and a subgradient (eigenvector) is computed. This information is added to $\widehat{\mathcal{W}}^k$ to form an improved model $\widehat{\mathcal{W}}^{k+1}$. Therefore, we assume that the current "bundle" $P^k = P$ contains an eigenvector of $\lambda_{\max}(C - \mathcal{A}^T y^k)$ in its span (y^k may or may not be equal to x^k). Other than that, P need only satisfy $P^T P = I_r$. The minorant of f in iteration k is denoted by $\hat{f}^k$:

$$\hat{f}^k(y) := \max_{W \in \widehat{\mathcal{W}}^k} L(W, y).$$

Here $\widehat{\mathcal{W}}^k$ represents the current approximation to the set of all semidefinite matrices of trace one; see (3.1). It will be convenient to introduce also the regularized version of $\hat{f}^k$. We define

$$f^k(y) := \hat{f}^k(y) + \tfrac{u}{2} \left\| y - x^k \right\|^2.$$

The new trial point y^{k+1} is obtained by minimizing $f^k(y)$ with respect to y. As described above, this can be done as follows. First, solve by interior point methods (see section 5)

$$\max_{W \in \widehat{\mathcal{W}}^k} \left\langle C - \mathcal{A}^T x^k, W \right\rangle + b^T x^k - \tfrac{1}{2u} \left\langle \mathcal{A}W - b, \mathcal{A}W - b \right\rangle, \tag{3.4}$$

yielding a (not necessarily unique) maximizer $W^{k+1} = \alpha^* \overline{W}^k + P^k V^* (P^k)^T$. Next, use (3.3) to compute

$$y^{k+1} = x^k + \tfrac{1}{u}(\mathcal{A}W^{k+1} - b). \tag{3.5}$$

To finish an iteration, we have to decide whether enough progress is made to perform a *serious step* or not; i.e., whether we are going to set $x^{k+1} = y^{k+1}$ or $x^{k+1} = x^k$, and how to update P^k and $\overline{W}^k$.

If P^k does not yet use the maximum number of columns allowed, then the update process is simple: orthogonalize the new eigenvector with respect to P^k, add it as a new column to form P^{k+1}, and continue. In general, however, P^k will already use the maximum number of columns and so we have to make room for the new subgradient information. Instead of simply eliminating some columns of P^k we can do better by exploiting the information available in α^* and V^*.

Let $Q \Lambda Q^T$ be an eigenvalue decomposition of V^*. Then the "important" part of the spectrum of W^{k+1} (the important subspace within the space spanned by P^k) is spanned by the eigenvectors associated with the "large" eigenvalues of V^*. Thus, we split the eigenvectors of Q into two parts $Q = [Q_1 Q_2]$ (with corresponding spectra Λ_1 and Λ_2), Q_1 containing as columns the eigenvectors associated with "large" eigenvalues of V^* and Q_2 containing the remaining columns,

$$W^{k+1} = P^k Q_1 \Lambda_1 (P^k Q_1)^T + \alpha^* \overline{W}^k + P^k Q_2 \Lambda_2 (P^k Q_2)^T. \tag{3.6}$$

Now the next P^{k+1} is computed to contain $P^k Q_1$ and at least one eigenvector v^{k+1} for the maximal eigenvalue of $C - \mathcal{A}^T y^{k+1}$,

$$P^{k+1} = \operatorname{orth}(\left[P^k Q_1 \; v^{k+1}\right]). \tag{3.7}$$

(The operator orth(.) indicates that we take an orthonormal basis of $[P^k Q_1 \; v^{k+1}]$.)

The next aggregate matrix is built in such a way that $W^{k+1} \in \widehat{\mathcal{W}}^{k+1}$. Since P^{k+1} contains only the important part of P^k, given by $P^k Q_1$, we include the remaining part of P^k, given by $P^k Q_2$ in $\overline{W}^{k+1}$:

$$\overline{W}^{k+1} = \frac{1}{\alpha^* + \operatorname{tr} \Lambda_2} (\alpha^* \overline{W}^k + P^k Q_2 \Lambda_2 (P^k Q_2)^T). \tag{3.8}$$

Note that $\overline{W}^{k+1}$ is scaled to have trace equal to 1.

PROPOSITION 3.3. *Update rules* (3.7) *and* (3.8) *ensure that* $W^{k+1} \in \widehat{\mathcal{W}}^{k+1}$.

Proof. Let W^{k+1} be of the form (3.6). By (3.7) there is an orthonormal matrix $\bar{Q}$ such that $P^{k+1}\bar{Q} = P^k Q_1$. Let $V = \bar{Q}\Lambda_1\bar{Q}^T$ and $\alpha = \alpha^* + \operatorname{tr}\Lambda_2$; then $V \succeq 0$, $\alpha \geq 0, \alpha + \operatorname{tr} V = 1$, and $W^{k+1} = P^{k+1}V(P^{k+1})^T + \alpha\overline{W}^{k+1} \in \widehat{\mathcal{W}}^{k+1}$. □

We summarize some easy facts, which will be used in the convergence analysis of the algorithm.

$$f^k(y^{k+1}) \leq f^k(y) \quad \forall y,$$

since y^{k+1} is minimizer of f^k. Because $f^k(x^k) = \hat{f}^k(x^k)$ and $\hat{f}^k$ minorizes f, we obtain

$$f^k(y^{k+1}) \leq f^k(x^k) \leq f(x^k). \tag{3.9}$$

Next let

$$f^k_*(y) := L(W^{k+1}, y) + \tfrac{u}{2}\left\|y - x^k\right\|^2.$$

Using the definition of y^{k+1} from (3.5) it follows easily that

$$f^k_*(y) = f^k_*(y^{k+1}) + \tfrac{u}{2}\left\|y - y^{k+1}\right\|^2. \tag{3.10}$$

Since $W^{k+1} \in \widehat{\mathcal{W}}^{k+1}$, the augmented model of the next iteration will satisfy

$$f^{k+1}(y) \geq f^k_*(y) \quad \forall y. \tag{3.11}$$

Remark 3.4. While the choice for the update of P^k is fairly natural, we could use other update formulas, such as $\overline{W}^{k+1} = W^{k+1}$ and $P^{k+1} = v^{k+1}$. The main properties guiding the update are that $W^{k+1} \in \widehat{\mathcal{W}}^{k+1}$, ensuring (3.11), and that in y^{k+1} the model is now supported by a subgradient of f pushing the model towards f in the vicinity of the last minimizer.

4. Algorithm and convergence analysis.

In the previous section we focused on the question of doing one iteration of the bundle method. Now we provide a formal description of the method and point out that except for the choice of the bundle, the nature of the subproblem, and some minor changes in parameters, the algorithm and its proof are identical to the algorithm of Kiwiel as presented in [21]. To keep the paper self-contained we present and analyze a simplified variant for fixed u. We refer the reader to [21] for an algorithm with variable choice of u.

ALGORITHM 4.1.

Input: *An initial point* $y^0 \in \mathbb{R}^m$, *a (normalized) eigenvector* v^0 *for the maximal eigenvalue of* $C - \mathcal{A}^T y^0$, *an* $\varepsilon > 0$ *for termination, an improvement parameter* $m_L \in (0, \frac{1}{2})$, *a weight* $u > 0$, *an upper bound* $R \geq 1$ *on the number of columns of* P.

(1) *(Initialization)* $k = 0$, $x^0 = y^0$, $P^0 = v^0$, $\overline{W}^0 = v^0(v^0)^T$.

(2) *(Direction finding) Solve* (3.4) *to get* y^{k+1} *from* (3.5). *Decompose* V^* *into* $V^* = Q_1\Lambda_1Q_1^T + Q_2\Lambda_2Q_2^T$ *with* $\operatorname{rank}(Q_1) \leq R - 1$. *Compute* $\overline{W}^{k+1}$ *using* (3.8).

(3) *(Evaluation) Compute* $\lambda_{\max}(C - \mathcal{A}^T y^{k+1})$ *and an eigenvector* v^{k+1}. *Compute* P^{k+1} *by* (3.7).

(4) *(Termination) If* $f(x^k) - \hat{f}^k(y^{k+1}) \leq \varepsilon$ *then* **stop**.

(5) *(Serious step) If*

$$f(y^{k+1}) \le f(x^k) - m_L(f(x^k) - \hat{f}^k(y^{k+1})) \tag{4.1}$$

then set $x^{k+1} = y^{k+1}$*, continue with step* 7*. Otherwise continue with step* 6*.*

(6) *(Null step) Set* $x^{k+1} = x^k$*.*

(7) *Increase* k *by* 1 *and go to step* 2*.*

We prove convergence of the algorithm for $\varepsilon = 0$. If the algorithm stops after a finite number of iterations, then by (3.9) $f^k(y^{k+1}) = f(x^k)$, which implies $y^{k+1} = x^k$, and thus by (3.5) $0 \in \partial f(x^k)$, so x^k is optimal. Assume in the following that the algorithm does not stop. First consider the case when only null steps occur after some iteration K.

Lemma 4.2. *If there is a* $K \ge 0$ *such that* (4.1) *is violated for all* $k \ge K$*, then* $\lim_{k\to\infty} \hat{f}^k(y^{k+1}) = f(x^K)$ *and* $0 \in \partial f(x^K)$*.*

Proof. For convenience, we set $x = x^K = x^{K+1} = \cdots$. Using the relations (3.10), (3.11), and (3.9), we obtain for all $k \ge K$

$$f_*^k(y^{k+1}) + \frac{u}{2}\left\|y^{k+2} - y^{k+1}\right\|^2 = f_*^k(y^{k+2}) \le f^{k+1}(y^{k+2}) \le f^{k+1}(x^{k+1}) \le f(x).$$

Therefore, the $f^k(y^{k+1})$ converge to some $f^* \le f(x)$ and $\left\|y^{k+2} - y^{k+1}\right\| \to 0$ (y_k is bounded by (3.5)). Let $g^{k+1} = b - \mathcal{A}(v^{k+1}(v^{k+1})^T)$ denote the computed gradient of f in y^{k+1} and observe that the linearization $\bar{f}$ of f in y^{k+1},

$$\bar{f}(y; y_{k+1}) = f(y^{k+1}) + \left\langle g^{k+1}, y - y^{k+1} \right\rangle,$$

minorizes $\hat{f}^{k+1}$, because $(v^{k+1}(v^{k+1})^T) \in \widehat{\mathcal{W}}^{k+1}$. Thus

$$\begin{aligned}
0 &\le f(y^{k+1}) - \hat{f}^k(y^{k+1}) \\
&= \bar{f}(y^{k+1}; y^{k+1}) - \hat{f}^k(y^{k+1}) \\
&= \bar{f}(y^{k+2}; y^{k+1}) - \hat{f}^k(y^{k+1}) - \left\langle g^{k+1}, y^{k+2} - y^{k+1} \right\rangle \\
&\le \hat{f}^{k+1}(y^{k+2}) - \hat{f}^k(y^{k+1}) + \left\|g^{k+1}\right\| \cdot \left\|y^{k+2} - y^{k+1}\right\| \\
&= f^{k+1}(y^{k+2}) - f^k(y^{k+1}) - u\left\|y^{k+2} - x\right\|^2 + u\left\|y^{k+1} - x\right\|^2 \\
&\quad + \left\|g^{k+1}\right\| \cdot \left\|y^{k+2} - y^{k+1}\right\|.
\end{aligned}$$

The convergence of the $f^k(y^{k+1})$, the boundedness of the gradients, and the fact that $\left\|y^{k+2} - y^{k+1}\right\| \to 0$ imply that the last term goes to zero for $k \to \infty$. So for all $\delta > 0$, there is an $M \in \mathbb{N}$ such that for all $k > M$

$$f(y^{k+1}) - \delta \le \hat{f}^k(y^{k+1}) \le f(x) < \frac{f(y^{k+1}) - m_L\hat{f}^k(y^{k+1})}{1 - m_L} \le f(y^{k+1}) + \frac{m_L}{1 - m_L}\delta,$$

where "<" follows from (4.1) being violated for all $k > K$. Thus, the sequences $f(y^{k+1})$ and $\hat{f}^k(y^{k+1})$ both converge to $f(x)$. y^{k+1} is the minimizer of the regularized function f^k. On the one hand, this implies that $y^{k+1} \to x$. On the other hand, 0 must be contained in the subgradient $\partial f^k(y^{k+1}) = \partial\hat{f}^k(y^{k+1}) + u(y^{k+1} - x)$; see (3.3). Therefore, there is a sequence $h^k \in \partial\hat{f}^k(y^{k+1})$ of subgradients converging to zero. The $\hat{f}^k$ minorize f, the $\hat{f}^k(y^{k+1})$ converge to $f(x)$, and the y^{k+1} converge to x; hence zero must be contained in $\partial f(x)$. □

We may concentrate on serious steps in the following. In order to simplify notation we will speak of x^k as the sequence generated by serious steps with all duplicates eliminated. By f^k (and the corresponding $\hat{f}^k$) we will refer to the function whose minimization gives rise to x^{k+1}.

The next lemma investigates the case when the $f(x^k)$ remain above some value $f(\tilde{x})$ for some fixed $\tilde{x}$.

Lemma 4.3. *If*

$$f(x^k) > f(\tilde{x}) \quad \textit{for some fixed } \tilde{x} \in \mathbb{R}^m \textit{ and all } k, \tag{4.2}$$

then the x^k converge to a minimizer of f.

Proof. First we prove the boundedness of the x^k. To this end denote by $g^{k+1} \in \partial \hat{f}^k(x^{k+1})$ the subgradient arising from the optimal solution of the minimization problem for $f^k, g^{k+1} = b - \mathcal{A}W^{k+1}$, and observe that by (3.3)

$$x^{k+1} - x^k = -\frac{g^{k+1}}{u}. \tag{4.3}$$

Since $\hat{f}^k$ minorizes f we obtain

$$f(x^k) \geq f(\tilde{x}) \geq \hat{f}^k(x^{k+1}) + \left\langle g^{k+1}, \tilde{x} - x^{k+1} \right\rangle.$$

Therefore, the distance of x^{k+1} to $\tilde{x}$ can be bounded by

$$\begin{aligned}
\left\| \tilde{x} - x^{k+1} \right\|^2 &= \left\| \tilde{x} - x^k + x^k - x^{k+1} \right\|^2 \\
&\leq \left\| \tilde{x} - x^k \right\|^2 + 2 \left\langle \tilde{x} - x^k, x^k - x^{k+1} \right\rangle + 2 \left\langle x^k - x^{k+1}, x^k - x^{k+1} \right\rangle \\
&= \left\| \tilde{x} - x^k \right\|^2 + 2 \left\langle \tilde{x} - x^{k+1}, g^{k+1}/u \right\rangle \\
&\leq \left\| \tilde{x} - x^k \right\|^2 + \tfrac{2}{u}(f(x^k) - \hat{f}^k(x^{k+1})).
\end{aligned}$$

For any $k > K$, a recursive application of the bound above yields

$$\left\| \tilde{x} - x^k \right\|^2 \leq \left\| \tilde{x} - x^K \right\|^2 + \tfrac{2}{u} \sum_{i=K}^{\infty} (f(x^k) - \hat{f}^k(x^{k+1})). \tag{4.4}$$

By (4.1) the progress of the algorithm in each serious step is at least $m_L(f(x^k) - \hat{f}^k(x^{k+1}))$, and together with (4.2) we obtain

$$\sum_{i=0}^{\infty} (f(x^i) - \hat{f}^i(x^{i+1})) \leq \tfrac{1}{m_L}(f(x^0) - f(\tilde{x})).$$

Therefore, the sequence of the x^k remains bounded and has an accumulation point $\bar{x}$. By replacing $\tilde{x}$ by $\bar{x}$ in (4.4) and choosing K sufficiently large, the remaining sum can be made smaller than an arbitrary small $\delta > 0$, thus proving the convergence of the x^k to $\bar{x}$. As the x^{k+1} converge to $\bar{x}$, the g^{k+1} converge to zero by (4.3), and since the sequence $(f(x^k) - \hat{f}^k(x^{k+1}))$ must converge to zero as well, we conclude that $0 \in \partial f(\bar{x})$, i.e., that $\bar{x}$ is a minimizer of f. □

The lemma also implies that $f(x^k) \to \inf f$ if there are no minimizers. We summarize the discussion in the following theorem.

Theorem 4.4 (see [21]). *If the set of minimizers of f is not empty, then the x^k converge to a minimizer of f. In any case $f(x^k) \to \inf f$.*

Remark 4.5. We have just seen that the bundle algorithm works correctly even if P contains only one column. In this case the use of the aggregate subgradient is crucial.

To achieve correctness of the bundle algorithm without aggregate subgradients, it suffices to store in P only the subspace spanning the eigenvectors corresponding to nonzero eigenvalues of an optimal solution W^{k+1} of (3.2). Using the bound of [36] it is not too difficult to show that in this case the maximal number of columns one has to provide is the largest $\bar{r} \in \mathbb{N}$ satisfying $\binom{\bar{r}+1}{2} \leq m+1$ plus the number of eigenvectors to be added in each iteration (this is at least one). In our computational experiments we found that this upper bound is hardly ever reached. In fact, typical values for the maximal rank are around half this upper bound.

5. Solving the subproblem. In this section we concentrate on how the minimizer of f^k can be computed efficiently. We have already seen in section 3 that this task is equivalent to solving the quadratic semidefinite program (3.4). Problems of this kind can be solved by interior point methods; see, e.g., [7, 23]. Dropping the iteration index k and the constants in (3.4) we obtain for $y = x^k$

$$\begin{array}{rl} \min & \frac{1}{2u}\left\langle \mathcal{A}W, \mathcal{A}W \right\rangle - \frac{1}{u}\left\langle b, \mathcal{A}W \right\rangle - \left\langle C - \mathcal{A}^T(y), W \right\rangle \\ \text{s.t.} & W = \alpha\overline{W} + PVP^T, \\ & \alpha + \operatorname{tr} V = 1, \\ & \alpha \geq 0, V \succeq 0. \end{array}$$

Expanding $W = \alpha\overline{W} + PVP^T$ into the cost function yields

$$\begin{array}{rl} \min & \frac{1}{2u}\left[\left\langle \mathcal{A}(PVP^T), \mathcal{A}(PVP^T) \right\rangle + 2\alpha\left\langle \mathcal{A}(PVP^T), \mathcal{A}\overline{W} \right\rangle + \alpha^2\left\langle \mathcal{A}\overline{W}, \mathcal{A}\overline{W} \right\rangle\right] \\ & -\left\langle \frac{1}{u}\mathcal{A}^T b + C - \mathcal{A}^T y, P^T V P \right\rangle - \alpha\left\langle \frac{1}{u}\mathcal{A}^T b + C - \mathcal{A}^T y, \overline{W} \right\rangle \\ \text{s.t.} & \alpha + \operatorname{tr} V = 1, \\ & \alpha \geq 0, V \succeq 0. \end{array}$$

Using the svec-operator (see the appendix for a definition and important properties of svec and the symmetric Kronecker product $\otimes_s$) to expand symmetric matrices from S_r into column vectors of length $\binom{r+1}{2}$ we obtain the quadratic program (recall that, for $A, B \in S_r$, $\langle A, B \rangle = \operatorname{svec}(A)^T \operatorname{svec}(B)$ and that $\operatorname{tr} V = \langle I, V \rangle$)

$$\begin{array}{rl} \min & \frac{1}{2}\operatorname{svec}(V)^T Q_{11} \operatorname{svec}(V) + \alpha q_{12}^T \operatorname{svec}(V) + \frac{1}{2} q_{22}\alpha^2 + c_1^T \operatorname{svec}(V) + c_2\alpha, \\ & \alpha + s_I^T \operatorname{svec}(V) = 1, \\ & \alpha \geq 0, V \succeq 0, \end{array} \tag{5.1}$$

where (after some technical linear algebra)

$$Q_{11} = \frac{1}{u}\sum_{i=1}^{m} \operatorname{svec}(P^T A_i P) \operatorname{svec}(P^T A_i P)^T, \tag{5.2}$$

$$q_{12} = \frac{1}{u}\operatorname{svec}(P^T \mathcal{A}^T(\mathcal{A}\overline{W})P), \tag{5.3}$$

$$q_{22} = \frac{1}{u}\left\langle \mathcal{A}\overline{W}, \mathcal{A}\overline{W} \right\rangle, \tag{5.4}$$

$$c_1 = -\operatorname{svec}(P^T(\mathcal{A}^T(\tfrac{1}{u}b - y) + C)P), \tag{5.5}$$

$$c_2 = -(\left\langle \tfrac{1}{u}b - y, \mathcal{A}\overline{W} \right\rangle + \left\langle C, \overline{W} \right\rangle), \tag{5.6}$$

$$s_I = \operatorname{svec}(I).$$

At this point it is advisable to spend some thought on $\overline{W}$. The algorithm is designed for very large and sparse cost matrices C. $\overline{W}$ is of the same size as C. Initially it might be possible to exploit the low-rank structure of $\overline{W}$ for efficient representations, but as the algorithm proceeds, the rank of $\overline{W}$ inevitably grows. Thus, it is impossible to store all the information of $\overline{W}$. However, as we can see in (5.2) to (5.6), it suffices to have available the vector $\mathcal{A}\overline{W} \in \mathbb{R}^m$ and the scalar $\langle C, \overline{W} \rangle$ to construct the quadratic program. Furthermore, by the linearity of $\mathcal{A}(\cdot)$ and $\langle C, \cdot \rangle$, these values are easily updated whenever $\overline{W}$ is changed.

To solve (5.1) we employ a primal-dual interior point strategy. To formulate the defining equations for the central path we introduce a Lagrange multiplier t for the equality constraint, a dual slack matrix $U \succeq 0$ as complementary variable to V, a dual slack scalar $\beta \geq 0$ as complementary variable to α, and a barrier parameter $\mu > 0$. The system reads

$$\begin{array}{rcrcl} F_U & = & Q_{11}\,\mathrm{svec}(V) + \alpha q_{12} + c_1 - t s_I - \mathrm{svec}(U) & = & 0, \\ F_\beta & = & \alpha q_{22} + q_{12}^T\,\mathrm{svec}(V) + c_2 - t - \beta & = & 0, \\ F_1 & = & 1 - \alpha - s_I^T\,\mathrm{svec}(V) & = & 0, \\ & & UV & = & \mu I, \\ & & \alpha\beta & = & \mu. \end{array}$$

The step direction $(\Delta\alpha, \Delta\beta, \Delta U, \Delta V, \Delta t)$ is determined via the linearized system

$$\begin{aligned} Q_{11}\,\mathrm{svec}(\Delta V) + \Delta\alpha q_{12} - \Delta t s_I - \mathrm{svec}(\Delta U) &= -F_U, \\ q_{12}^T\,\mathrm{svec}(\Delta V) + q_{22}\Delta\alpha - \Delta t - \Delta\beta &= -F_\beta, \\ -\Delta\alpha - s_I^T\,\mathrm{svec}(\Delta V) &= -F_1, \\ (U \otimes_s V^{-1})\,\mathrm{svec}(\Delta V) + \mathrm{svec}(\Delta U) &= \mu\ \mathrm{svec}(V^{-1}) - \mathrm{svec}(U), \\ (\beta/\alpha)\Delta\alpha + \Delta\beta &= \mu\alpha^{-1} - \beta. \end{aligned}$$

In the current context we prefer the linearization $(U \otimes_s V^{-1})\,\mathrm{svec}(\Delta V) + \mathrm{svec}(\Delta U)$ because it makes the system easy to solve for ΔV with relatively little computational work per iteration. The final system for ΔV reads

$$\begin{aligned} (5.7)\quad & \left(Q_{11} + U \otimes_s V^{-1} + \left(\frac{\beta}{\alpha} + q_{22}\right) s_I s_I^T - q_{12} s_I^T - s_I q_{12}^T\right) \mathrm{svec}(\Delta V) \\ & = \mu\ \mathrm{svec}(V^{-1}) - \mathrm{svec}(U) - F_U - F_1 q_{12} - \left(\mu\alpha^{-1} - \beta - \frac{\beta}{\alpha} F_1 - q_{22} F_1\right) s_I. \end{aligned}$$

It is not too difficult to see that the system matrix is positive definite—because $U \otimes_s V^{-1} \succ 0$, it suffices to show that $Q_{11} + q_{22} s_I s_I^T - q_{12} s_I^T - s_I q_{12}^T \succeq 0$ using $\left[\begin{smallmatrix} Q_{11} & q_{12} \\ q_{12} & q_{22} \end{smallmatrix}\right] \succeq 0$. The main work per iteration is the factorization of this matrix (with $v \in S_r$ this is $\binom{r+1}{2}^3/3)$ and it is not possible to do much better since Q_{11} has to be inverted at some point. Because of the strong dominance of the factorization it pays to employ a predictor corrector approach, but we will not delve into this here.

For $V \in S_r$ a strictly feasible primal starting point is

$$\begin{aligned} V^0 &= I/(r+1), \\ \alpha^0 &= 1/(r+1), \end{aligned}$$

and a strictly feasible dual starting point can be constructed by choosing t^0 sufficiently negative such that

$$\begin{aligned} U^0 &= \overset{-1}{\text{svec}}(Q_{11}\,\text{svec}(V^0) + \alpha^0 q_{12} + c_1) - t^0 I \ \succ 0, \\ \beta^0 &= \alpha q_{22} + q_{12}^T\,\text{svec}(V) + c_2 - t^0 \ > 0. \end{aligned}$$

Starting from this strictly feasible primal-dual pair we compute the first μ by $\mu = (\langle U, V\rangle + \alpha\beta)/(r+1)$, compute the step direction $(\Delta\alpha, \Delta\beta, \Delta U, \Delta V, \Delta t)$ as indicated above, perform a line search with line search parameter $0 < \delta \leq 1$ such that $(\alpha + \delta\Delta\alpha, \beta + \delta\Delta\beta, U + \delta\Delta U, V + \delta\Delta V, t + \delta\Delta t)$ is again strictly feasible, move to this new point, compute a new μ by

$$\mu = \min\left\{\mu_{\text{old}}, \gamma\frac{\langle U, V\rangle + \alpha\beta}{r+1}\right\} \quad \text{with} \quad \gamma = \begin{cases} 1 & \text{if } \delta \leq \frac{1}{5}, \\ \frac{5}{10} - \frac{4}{10}\delta^2 & \text{if } \delta > \frac{1}{5}, \end{cases}$$

and iterate. We stop if $(\langle U, V\rangle + \alpha\beta)/(r+1) < 10^{-7}$.

6. Implementation.

In our implementation of the algorithm we largely follow the rules outlined in [21]. In particular u is adapted during the algorithm. The first guess for u is equal to the norm of the first subgradient determined by v^0. The scheme for adapting u is the same as in [21] except for a few changes in parameters. For example the parameter m_L, for accepting a step as serious, is set to $m_L = 0.2$ and the parameter m_R, indicating that the model is so good (progress by the serious step is larger than $m_R[f(x^k) - \hat{f}^k(y^{k+1})]$) that u can be decreased, is set to $m_R = 0.7$.

The stopping criterion is formulated in relative precision,

$$f(x^k) - \hat{f}^k(y^{k+1}) \leq \varepsilon \cdot (|f(x^k)| + 1)$$

with $\varepsilon = 10^{-5}$ in the implementation.

The choice of the upper bound R on the number of columns r of P and the selection of the subspace merits some additional remarks. Observe that by Remark 4.5 it is highly unlikely that r violates the bound $\binom{r+1}{2} \leq m$ even if the number of columns of P is not restricted. $\binom{r+1}{2}$ is also the order of the system matrix in (5.7) and is usually considerably smaller than the size of the system matrix in traditional interior point codes for semidefinite programming which is of order m. Furthermore, the order of the matrix variables is r as compared to n for traditional interior point codes. Thus, if the number of constraints m is roughly of the same size as n and a matrix of order m is still considered factorizable, then running the algorithm without bounding the number of columns of P may turn out to be considerably faster than running an interior point method. This can be observed in practice; see section 7.

For huge n and m primal-dual interior point methods are not applicable, because X, Z^{-1}, and the system matrix are dense. In this case the proposed bundle approach allows application of the powerful interior point approach at least on an important subspace of the problem. The correct identification of the relevant subspace in V^* is facilitated by the availability of the complementary variable U^*. The matrix U^* helps to discern between the small eigenvalues of V^* (because of the interior point approach we have $V^* \succ 0$!). Eigenvectors v of V^* that are of no importance for the optimal solution of the subproblem will have a large value $v^T U^* v$, whereas eigenvectors that are ambiguous will have both a small eigenvalue $v^T V^* v$ and a small value $v^T U^* v$.

In practice we restrict the number of columns of P to 25 and provide room for at least five new vectors in each iteration (see below). Eigenvectors v that correspond to small but important eigenvalues λ of $V^*(\lambda < 10^{-3}\lambda_{\max}(V^*)$ and $\lambda > 10^{-2}v^TU^*v)$ are added to $\overline{W}$; important eigenvectors $(\lambda > 10^{-3}\lambda_{\max}(V^*))$ are added to $\overline{W}$ only if more room is needed for new vectors.

For large m the computation of (5.2) to (5.6) is quite involved. A central object appearing in all constants is the projection of the constraint A_i on the space spanned by P, i.e., P^TA_iP. Since the A_i are of the same size as X, which we assume to be huge, it is important to exploit whatever structure is present in A_i to compute this projection efficiently. In combinatorial applications the A_i are of the form vv^T, with v sparse, and the projection can be computed efficiently. In the projection step, and in particular in forming Q_{11}, the size of r is again of strong influence. If we neglect the computation of $\operatorname{svec}(P^TA_iP)$, the computation of Q_{11} still requires $2m\binom{\binom{r+1}{2}+1}{2}$ flops. Indeed, if m is large, then for small r the construction of Q_{11} takes longer than solving the associated quadratic semidefinite program.

The large computational costs involved in the construction and solution of the semidefinite subproblems may lead to the conviction that this model may not be worth the trouble. However, the evaluation of the eigenvalue-function is in fact much more expensive. There has been considerable work on computing eigenvalues of huge, sparse matrices; see, e.g., [9] and the references therein. For extremal eigenvalues of symmetric matrices there seems to be a general consensus that Lanczos-type methods work best. Iterative methods run into difficulties if the eigenvalues are not well separated. In our context it is to be expected that in the course of the algorithm the largest eigenvalues will get closer and closer till all of them are identical in the limit. For reasonable convergence, block Lanczos algorithms, with block size corresponding to the largest multiplicity of the eigenvalues, must be employed. During the first 10 iterations the largest eigenvalue is usually well separated and the algorithm is fast. But soon the eigenvalues start to cluster, larger and larger block sizes must be used, and the eigenvalue problem gets more and more difficult to solve. In order to reduce the number of evaluations, it seems worthwhile to employ powerful methods in the cutting plane model. The increase in computation time required to solve the subproblem goes hand in hand with the difficulty of the eigenvalue problem because of the correspondence of the rank of P and the number of clustered eigenvalues.

Iterative methods for computing maximal eigenvectors generically offer approximate eigenvectors for several other large eigenvalues, as well. The space spanned by these approximate eigenvectors is likely to be a good approximation of the true eigenspace. If the maximal number of columns for P is not yet attained it may be worthwhile to include several of these approximate eigenvectors as well.

In our algorithm we use a block Lanczos code of our own that is based on a Fortran code of Hua (we guess that this is Hua Dai of [47]). It works with complete orthogonalization and employs Chebyshev iterations for acceleration. The choice of the block size is based on the approximate eigenvalues produced by previous evaluations but is at most 30. Four block Lanczos steps are followed by 20 Chebyshev iterations. This scheme is repeated till the maximal eigenvalue is found to the required relative precision. The relative precision depends on the distance of the maximal to the second largest eigenvalue but is bounded below by 10^{-6}. As starting vectors, we use the complete block of eigenvectors and Lanczos-vectors from the previous evaluation.

7. Combinatorial applications. The combinatorial problem we investigate is quadratic programming in $\{-1,1\}$ variables,

$$\text{(MC)} \qquad \max\ x^TCx \ \text{ s.t. } x \in \{-1,1\}^n .$$

Table 7.1

Comparison of the interior point (PDIP) *and the bundle* (B) *approach.* sol *gives the computed solution value and* time *gives the computation time.*

	PDIP-sol	PDIP-time	B-sol	B-time
G_1	12083.20	1:18:42	12083.22	4:11
G_2	12089.43	1:19:14	12089.45	5:19
G_3	12084.33	1:25:30	12084.34	4:38
G_4	12111.45	1:23:16	12111.46	3:37
G_5	12099.89	1:27:09	12099.91	4:38
G_6	2656.16	1:24:53	2656.18	3:57
G_7	2489.26	1:32:34	2489.29	7:54
G_8	2506.93	1:21:47	2506.95	3:38
G_9	2528.73	1:30:36	2528.75	3:51
G_{10}	2485.06	1:24:30	2485.08	3:56
G_{11}	629.16	1:28:41	629.21	45:26
G_{12}	623.87	1:34:55	623.89	31:14
G_{13}	647.13	1:37:52	647.14	18:44
G_{14}	3191.57	2:05:24	3191.58	14:30
G_{15}	3171.56	2:25:53	3171.56	23:20
G_{16}	3175.02	2:18:21	3175.04	16:59
G_{17}	3171.32	2:13:10	3171.35	16:32
G_{18}	1166.01	2:58:22	1166.02	18:32
G_{19}	1082.01	3:07:58	1082.04	12:27
G_{20}	1111.39	3:12:41	1111.40	11:59
G_{21}	1104.28	3:13:53	1104.29	13:35

In the case that C is the Laplace matrix of a (possibly weighted) graph, the problem is known to be equivalent to the max-cut problem.

The standard semidefinite relaxation is based on the identity $x^T Cx = \langle C, xx^T \rangle$. For all $\{-1,1\}^n$ vectors, xx^T is a positive semidefinite matrix with all diagonal elements equal to 1. We relax xx^T to $X \succeq 0$ and $\operatorname{diag}(X) = e$ and obtain the following primal-dual pair of semidefinite programs,

$$\begin{array}{rrl@{\qquad\qquad}rrl} & \max & \langle C, X \rangle & & \min & e^T y \\ \text{(PMC)} & \text{s.t.} & \operatorname{diag}(X) = e & \text{(DMC)} & \text{s.t.} & C + Z - \operatorname{Diag}(y) = 0 \\ & & X \succeq 0 & & & Z \succeq 0. \end{array}$$

For nonnegatively weighted graphs, a celebrated result of Goemans and Williamson [8] says that there is always a cut within .878 of the optimal value of the relaxation.

One of the first attempts to approximate (DMC) using eigenvalue optimization is contained in [39]. The authors use the bundle code of Schramm and Zowe [42] with a limited number of bundle iterations, and so do not solve (DMC) exactly. Until now, the only practical algorithms for computing the optimal value were primal-dual interior point algorithms. However, these are not able to exploit the sparsity of the cost function and have to cope with dense matrices X and Z^{-1}. An alternative approach based on a combination of the power method with a generic optimization scheme of Plotkin, Shmoys, and Tardos [37] was proposed in [22], but seems to be purely theoretical.

In Table 7.1 we compare the proposed bundle method to our semidefinite primal-dual interior point code of [14] (called PDIP in what follows) for graphs on $n = m = 800$ nodes that were generated by `rudy`, a machine independent graph generator written by G. Rinaldi. Table 7.7 contains the command lines specifying the graphs. Graphs G_1 to G_5 are unweighted random graphs with a density of 6% (approximately 19,000 edges). G_6 to G_{10} are the same graphs with random edge weights from $\{-1, 1\}$. G_{11} to G_{13} are toroidal grids with random edge weights from $\{-1, 1\}$ (1,600 edges). G_{14} to G_{17} are unweighted "almost" planar graphs having as edge set the union of two (almost maximal) planar graphs (approximately 4,500 edges). G_{18} to G_{21} are the same almost planar graphs with random edge weights from $\{-1, 1\}$. In all cases the cost matrix C is the Laplace matrix of the graph divided by 4; i.e., let A denote the (weighted) adjacency matrix of G; then

$$C = \frac{1}{4}(\mathrm{Diag}(Ae) - A).$$

For a description of the code PDIP see [14]. The termination criterion requires the gap between primal and dual optimal solution to be closed to a relative accuracy of $5 \cdot 10^{-6}$.

For the bundle algorithm, (DMC) is transformed into an eigenvalue optimization problem as described in section 2. In addition, the diagonal of C is removed so that, in fact, the algorithm works on the problem

$$\min_{y \in \mathbb{R}^n} n\lambda_{\max}(\bar{C} - \mathrm{Diag}(y)) + e^T y$$

with $\bar{C} = \frac{1}{4}(\mathrm{Diag}(\mathrm{diag}(A)) - A)$. This does not change problem (PMC) because the diagonal elements of X are fixed to 1. The offset $\frac{1}{4}e^T(Ae - \mathrm{diag}(A))$ is added to the output only and has no influence on the algorithm whatsoever; in particular, it has no influence on the stopping criterion. As starting vector y^0 we choose the zero vector. All other parameters are as described in section 6.

All computation times, for the interior point code PDIP as well as for the bundle code, refer to the same machine, a Sun Sparc Ultra 1 with a Model 140 UltraSPARC CPU and 64 MB RAM. The time measured is elapsed user time and it is given in the format *hh*:*mm*:*ss*, hours:minutes:seconds. Leading zeros are dropped.

The first column of Table 7.1 identifies the graphs. The second and third refer to PDIP and contain the optimal objective value produced (these can be regarded as highly accurate solutions) and the computation time. The fourth and fifth columns give the same numbers for the bundle code.

On these examples the bundle code is superior to PDIP. Although the examples do belong to the favorable class of instances having small m and relatively large n, the difference in computation time is astonishing. Note that the termination criterion used in the bundle code is quite accurate, except for G_{11}, which seems to be a difficult problem for the bundle method. This deviation in accuracy is *not* caused by cancellations in connection with the offset. The difficulty of an example does not seem to depend on the number of nonzeros but rather on the shape of the objective function. For toroidal grid graphs the maximum cut is likely to be not unique, thus the objective function will be rather flat. This flatness has its effect on the distribution of the eigenvalues in the optimal solution. Indeed, for G_{11} more eigenvalues cluster around the maximal eigenvalue than for the other problems. We illustrate this in Table 7.2, which gives the 30 largest eigenvalues of the solution at termination for problems G_1, G_6, G_{11}, G_{14}, and G_{18}.

Table 7.3 provides additional information on the performance of the bundle algorithm on the examples of Table 7.1. The second column gives the accumulated time spent in the eigenvalue computation, accounting for roughly 90% of the computation time. *Serious* displays the number of serious steps, *iter* gives the total number of iterations, including both serious and null steps. $\|g\|$ is the norm of the subgradient arising

Table 7.2
The 30 *maximal eigenvalues after termination of examples* G_1, G_6, G_{11}, G_{14}, *and* G_{18}.

	G_1	G_6	G_{11}	G_{14}	G_{18}
1	3.1190	3.2240	0.7653	1.0557	1.4175
2	·	3.2239	·	·	·
3	·	·	·	·	·
4	·	·	·	·	·
5	·	·	·	·	·
6	·	·	·	·	·
7	·	·	0.7653	·	·
8	·	·	0.7652	·	·
9	·	·	0.7652	·	·
10	·	·	0.7652	·	1.4175
11	·	·	0.7652	·	1.4155
12	·	·	0.7651	·	1.4090
13	3.1190	3.2239	0.7651	1.0557	1.4047
14	3.1135	3.2181	0.7650	1.0515	1.4007
15	3.1020	3.1968	0.7650	1.0500	1.3905
16	3.0928	3.1774	0.7649	1.0490	1.3834
17	3.0772	3.1556	0.7649	1.0450	1.3814
18	3.0594	2.7886	0.7648	1.0432	1.3798
19	2.7214	2.7716	0.7647	1.0398	1.3725
20	2.6964	2.7681	0.7646	1.0379	1.3709
21	2.6858	2.7269	0.7645	1.0358	1.3652
22	2.6834	2.6756	0.7644	1.0341	1.3583
23	2.6682	2.5851	0.7641	1.0331	1.3555
24	2.6649	2.5239	0.7639	1.0284	1.3510
25	2.6468	2.2357	0.7638	1.0266	1.3495
26	2.6274	1.8722	0.7636	1.0239	1.3480
27	2.6035	1.8473	0.7634	1.0231	1.3417
28	2.5137	1.7974	0.7633	1.0211	1.3397
29	2.4840	1.4859	0.7630	1.0180	1.3345
30	2.3281	1.4411	0.7629	1.0175	1.3291

from the last optimal W^{k+1} before termination. For G_{11} the norm is considerably higher than for all other examples. Since the desired accuracy was not achieved for G_{11} by the standard stopping criterion it may be worthwhile considering an alternative stopping criterion taking into account the norm of the subgradient as well. Column *max-r* gives the maximal rank of P attained over all iterations. The rank of P was limited to 25, but this bound never came into effect for any of these examples. Aggregation was not necessary. Observe that the theoretic bound allows for r up to 39, yet the maximal rank is only half this number. The last column gives the time when the objective value was first within 10^{-3} of the optimum.

For combinatorial applications, high accuracy of the optimal solution is of minor importance. An algorithm should deliver a reasonable bound fast and its solution should provide some hint on how a good feasible solution can be constructed. The bundle algorithm offers both. With respect to computation time, the bundle algorithm displays the usual behavior of subgradient algorithms. Initially progress is very fast, but as the bound approaches the optimum there is a strong tailing-off effect. We illustrate this by giving the objective values and computation times for the serious steps of example G_6 (the diagonal offset is +77 in this example) in Table 7.4. After one minute the bound is within 0.1% of the optimum. For the other examples see the last column of Table 7.3.

With respect to a primal feasible solution observe that $P^k V_*^k (P^k)^T$ is a successively better and better approximation to the primal optimal solution X^*. In case

Table 7.3

Additional information about the bundle algorithm for the examples of Table 7.1. λ-time *gives the total amount of time spent for computing the eigenvalues and eigenvectors,* serious *gives the number of serious steps,* iter *the total number of iterations including null steps.* $\|g\|$ *refers to the norm of the gradient resulting from the optimal solution of the last semidefinite subproblem.* max-r *is the maximum number of columns used in P (the limit would have been* 25*).* 0.1%-time *gives the time when the bound is within* 10^{-3} *of the optimum in relative precision.*

	λ-time	serious	iter	$\|g\|$	max-r	0.1%-time
G_1	3:12	22	33	0.1639	18	48
G_2	4:04	21	36	0.05035	18	1:02
G_3	3:20	21	33	0.04107	19	57
G_4	2:38	19	27	0.08235	19	54
G_5	3:41	23	39	0.04425	17	46
G_6	2:42	21	35	0.0646	18	1:09
G_7	5:46	24	65	0.0854	17	57
G_8	2:39	23	38	0.1549	17	59
G_9	2:59	24	34	0.0711	17	1:04
G_{10}	2:56	23	37	0.02997	17	1:15
G_{11}	42:10	97	172	0.4696	15	17:04
G_{12}	28:47	50	130	0.2579	15	8:19
G_{13}	17:24	43	78	0.218	15	6:17
G_{14}	12:43	41	59	0.1682	18	3:09
G_{15}	20:16	44	89	0.1059	18	3:08
G_{16}	14:35	31	69	0.2246	19	3:11
G_{17}	14:38	41	65	0.2079	18	3:22
G_{18}	16:46	38	98	0.08161	15	4:15
G_{19}	11:24	41	71	0.1571	15	3:34
G_{20}	11:05	42	71	0.08226	15	3:50
G_{21}	12:23	44	80	0.09432	15	3:32

too much information is stored in the aggregate vector $\mathcal{A}\overline{W}^k$ (remember that it is not advisable to store $\overline{W}^k$ itself), P^k may be enriched with additional Lanczos-vectors from the eigenvalue computation. The solution of this enlarged quadratic semidefinite subproblem will be an acceptable approximation of X^*. It is not necessary to construct the whole matrix X^*. In fact, the factorized form $(P^k\sqrt{V_*^k})(P^k\sqrt{V_*^k})^T$ is much more convenient to work with. For example, the approximation algorithm of Goemans and Williamson [8] requires precisely this factorization. A particular x_{ij} element of X^* is easily computed by the inner product of row i and j of the $n \times r$ matrix $P^k\sqrt{V_*^k}$. In principle, this opens the door for branch-and-cut approaches to improve the initial relaxation. This will be the subject of further work.

Table 7.5 gives a similar set of examples for $n = m = 2,000$.

A last set of examples is devoted to the Lovász ϑ-function [27], which yields an upper bound on the cardinality of a maximal independent (or stable) set of a graph. For implementational convenience we use its formulation within the quadratic $\{-1, 1\}$ programming setting; see [24]. For a graph with k nodes and h edges we obtain a semidefinite program with matrix variables of order $n = k + 1$ and $m = k + 1 + h$ constraints. The examples we consider have more than one thousand nodes and more than six thousand edges. For these examples, interior point methods are not applicable because of memory requirements. It should be clear from the examples of Table 7.5 that there is also little hope for the bundle method to terminate within reasonable time. However, the most significant progress is achieved in the beginning and for the bundle method memory consumption is not a problem. We run these examples with a time limit of five hours. More precisely, the algorithm is terminated after the first serious step that occurs after five hours of computation time.

TABLE 7.4
Detailed account of the serious steps of example G_6.

iter	value	time	$\|g\|$	max-r
0	2861.20			
1	2821.75	3	42.02	3
2	2798.60	4	28.83	7
3	2782.99	6	25.94	8
4	2736.11	9	18.58	8
5	2704.98	13	12.69	10
6	2685.84	17	10.05	12
7	2679.95	23	7.46	14
8	2670.10	28	5.992	15
9	2666.07	34	4.173	16
10	2660.31	48	3.268	16
11	2658.65	1:09	0.6295	17
12	2656.66	1:32	0.7753	17
13	2656.42	1:41	0.6636	18
14	2656.31	1:57	0.431	18
15	2656.27	2:14	0.2997	18
16	2656.22	2:45	0.1609	18
17	2656.20	3:01	0.13	18
18	2656.19	3:17	0.09695	18
19	2656.19	3:33	0.07243	18
20	2656.17	3:49	0.05042	18
21	2656.17	3:57	0.0646	18

TABLE 7.5
Examples for $n = m = 2,000$.

	B-sol	B-time	λ-time	serious	iter	$\|g\|$	max-r
G_{22}	14135.98	38:11	28:00	26	52	0.0781	23
G_{23}	14145.58	1:19:29	1:01:06	32	107	0.3707	23
G_{24}	14140.88	28:04	21:11	25	40	0.1565	23
G_{25}	14144.30	43:44	32:26	27	59	0.3452	24
G_{26}	14132.93	34:45	26:37	31	48	0.3066	23
G_{27}	4141.68	24:56	18:54	23	37	0.1423	23
G_{28}	4100.81	29:41	21:08	23	39	0.1954	25
G_{29}	4208.94	48:16	37:53	27	65	0.1725	22
G_{30}	4215.42	1:02:39	48:16	27	83	0.1282	22
G_{31}	4116.70	26:11	19:02	24	38	0.1889	24
G_{32}	1567.80	6:20:54	6:09:52	144	312	0.892	15
G_{33}	1544.42	6:04:22	5:53:37	112	305	0.6887	15
G_{34}	1546.82	3:46:31	3:39:26	105	198	0.8369	15
G_{35}	8014.81	4:31:17	3:54:01	61	209	0.2397	22
G_{36}	8006.04	2:56:10	2:31:09	62	115	0.2634	24
G_{37}	8018.68	3:10:01	2:46:35	58	130	0.254	23
G_{38}	8015.01	4:03:53	3:39:24	58	155	0.1937	22
G_{39}	2877.71	1:20:24	1:12:23	50	85	0.2194	20
G_{40}	2864.96	2:42:02	2:30:21	59	158	0.2737	19
G_{41}	2865.29	1:41:33	1:32:50	51	108	0.1954	19
G_{42}	2946.29	1:32:45	1:24:12	59	93	0.1535	20

The graph instances are of the same type as above. The computational results are displayed in Table 7.6. The new columns n and m give the order of the matrix variable and the number of constraints, respectively. Observe that the toroidal grid graphs G_{48} and G_{49} are perfect with independence number 1,500; the independence number of G_{50} is 1,440 but G_{50} is not perfect. We do not know the independence number of the other graphs. Except for G_{48} and G_{49}, which have $\theta(G_{48}) = \theta(G_{49}) = 1,500$ by

Table 7.6
Upper bound on the ϑ-function after five hours of computation time.

	n	m	B-sol	B-time	λ-time	serious	iter	$\|g\|$	max-r
G_{43}	1001	10991	308.47	6:02:20	3:40:57	38	104	0.5098	25
G_{44}	1001	10991	310.13	5:06:31	3:14:25	39	88	0.5493	25
G_{45}	1001	10991	309.00	5:10:38	3:19:25	40	87	0.5532	25
G_{46}	1001	10991	309.21	5:35:56	3:31:42	42	99	0.5535	25
G_{47}	1001	10991	310.84	5:24:52	3:10:35	44	100	0.5558	25
G_{48}	3001	9001	1526.53	5:11:31	4:59:57	54	94	0.4062	15
G_{49}	3001	9001	1521.24	5:06:21	4:53:45	53	102	0.3751	15
G_{50}	3001	9001	1536.12	5:17:51	5:01:25	50	124	0.4728	15
G_{51}	1001	6910	455.21	5:07:40	4:29:00	32	118	2.556	25
G_{52}	1001	6917	465.12	5:09:02	4:38:30	41	108	2.683	25
G_{53}	1001	6915	463.86	5:08:36	4:36:34	41	104	2.593	25
G_{54}	1001	6917	466.04	5:02:21	4:32:21	40	98	2.590	25

perfectness, it is hard to judge the quality of the solutions. Tracing the development of the bounds the last serious steps of examples G_{43} to G_{47} and G_{51} to G_{54} still produced improvements of 0.5% to 1%. This and the rather large norm of the subgradient of G_{51} and G_{54} indicate that the values cannot be expected to be "good" approximations of the ϑ-function. Also note that the size of the subspace required for G_{48} to G_{50} is still well below 25. In examples G_{51} to G_{54} the value of α is almost negligible, but for G_{43} to G_{47} the value of α is roughly 1/3 at termination. Thus, for these examples the restriction to 25 columns became relevant.

The computational results of Table 7.6 demonstrate that the algorithm has its limits. Nonetheless, the bounds obtained are still useful and the primal approximation corresponding to the subgradient is a reasonable starting point for primal heuristics.

8. Conclusions and extensions. We have proposed a proximal bundle method for solving semidefinite programs with large sparse or strongly structured coefficient matrices. The semidefinite constraint is lifted into the objective function by means of a Lagrange multiplier a whose correct value is not known in general, except for problems with fixed primal trace. In the latter case a is precisely the value of the trace. The approach differs from previous bundle methods in that the subproblem is tailored for semidefinite programming. In fact the whole approach can be interpreted as semidefinite programming over subspaces where the subspace is successively corrected and improved till the optimal subspace is identified. The set of subgradients modeled by the semidefinite subproblem is a superset of the subgradients used in the traditional polyhedral cutting plane model. Therefore convergence of the new method is a direct consequence of previous proofs for traditional bundle methods. It is not yet clear whether the specialized model admits stronger convergence results. The choice of u is still very much an open problem of great practical importance.

For (constrained) quadratic $\{-1, 1\}$-programming the method offers a good bound within reasonable time and allows the construction of an approximate primal optimal solution (of the relaxation) in compact representation. To improve the bound by a cutting plane approach the algorithm must be able to deal with sign constraints on the y-variables. In principle it is not difficult to model the sign constraints in the semidefinite subproblem. However, as a consequence the influence of the sign-constrained y variables on the cost coefficients of the quadratic subproblem cannot be eliminated any longer, rendering the method impractical even for a moderate number of cutting planes. Alternatively, one might consider active set methods, but these risk destroying convergence. Together with K.C. Kiwiel, we are currently working on

Table 7.7

Arguments for generating the graphs by the graph generator `rudy`.

G_1	-rnd_graph 800 6 8001
G_2	-rnd_graph 800 6 8002
G_3	-rnd_graph 800 6 8003
G_4	-rnd_graph 800 6 8004
G_5	-rnd_graph 800 6 8005
G_6	-rnd_graph 800 6 8001 -random 0 1 8001 -times 2 -plus -1
G_7	-rnd_graph 800 6 8002 -random 0 1 8002 -times 2 -plus -1
G_8	-rnd_graph 800 6 8003 -random 0 1 8003 -times 2 -plus -1
G_9	-rnd_graph 800 6 8004 -random 0 1 8004 -times 2 -plus -1
G_{10}	-rnd_graph 800 6 8005 -random 0 1 8005 -times 2 -plus -1
G_{11}	-toroidal_grid_2D 100 8 -random 0 1 8001 -times 2 -plus -1
G_{12}	-toroidal_grid_2D 50 16 -random 0 1 8002 -times 2 -plus -1
G_{13}	-toroidal_grid_2D 25 32 -random 0 1 8003 -times 2 -plus -1
G_{14}	-planar 800 99 8001 -planar 800 99 8002 +
G_{15}	-planar 800 99 8003 -planar 800 99 8004 +
G_{16}	-planar 800 99 8005 -planar 800 99 8006 +
G_{17}	-planar 800 99 8007 -planar 800 99 8008 +
G_{18}	-planar 800 99 8001 -planar 800 99 8002 + -random 0 1 8001 -times 2 -plus -1
G_{19}	-planar 800 99 8003 -planar 800 99 8004 + -random 0 1 8002 -times 2 -plus -1
G_{20}	-planar 800 99 8005 -planar 800 99 8006 + -random 0 1 8003 -times 2 -plus -1
G_{21}	-planar 800 99 8007 -planar 800 99 8008 + -random 0 1 8004 -times 2 -plus -1
G_{22}	-rnd_graph 2000 1 20001
G_{23}	-rnd_graph 2000 1 20002
G_{24}	-rnd_graph 2000 1 20003
G_{25}	-rnd_graph 2000 1 20004
G_{26}	-rnd_graph 2000 1 20005
G_{27}	-rnd_graph 2000 1 20001 -random 0 1 20001 -times 2 -plus -1
G_{28}	-rnd_graph 2000 1 20002 -random 0 1 20002 -times 2 -plus -1
G_{29}	-rnd_graph 2000 1 20003 -random 0 1 20003 -times 2 -plus -1
G_{30}	-rnd_graph 2000 1 20004 -random 0 1 20004 -times 2 -plus -1
G_{31}	-rnd_graph 2000 1 20005 -random 0 1 20005 -times 2 -plus -1
G_{32}	-toroidal_grid_2D 100 20 -random 0 1 20003 -times 2 -plus -1
G_{33}	-toroidal_grid_2D 80 25 -random 0 1 20002 -times 2 -plus -1
G_{34}	-toroidal_grid_2D 50 40 -random 0 1 20001 -times 2 -plus -1
G_{35}	-planar 2000 99 20001 -planar 2000 99 20002 +
G_{36}	-planar 2000 99 20003 -planar 2000 99 20004 +
G_{37}	-planar 2000 99 20005 -planar 2000 99 20006 +
G_{38}	-planar 2000 99 20007 -planar 2000 99 20008 +
G_{39}	-planar 2000 99 20001 -planar 2000 99 20002 + -random 0 1 20001 -times 2 -plus -1
G_{40}	-planar 2000 99 20003 -planar 2000 99 20004 + -random 0 1 20002 -times 2 -plus -1
G_{41}	-planar 2000 99 20005 -planar 2000 99 20006 + -random 0 1 20003 -times 2 -plus -1
G_{42}	-planar 2000 99 20007 -planar 2000 99 20008 + -random 0 1 20004 -times 2 -plus -1
G_{43}	-rnd_graph 1000 2 10001
G_{44}	-rnd_graph 1000 2 10002
G_{45}	-rnd_graph 1000 2 10003
G_{46}	-rnd_graph 1000 2 10004
G_{47}	-rnd_graph 1000 2 10005
G_{48}	-toroidal_grid_2D 50 60
G_{49}	-toroidal_grid_2D 30 100
G_{50}	-toroidal_grid_2D 25 120
G_{51}	-planar 1000 100 10001 -planar 1000 100 10002 +
G_{52}	-planar 1000 100 10003 -planar 1000 100 10004 +
G_{53}	-planar 1000 100 10005 -planar 1000 100 10006 +
G_{54}	-planar 1000 100 10007 -planar 1000 100 10008 +

alternative methods for incorporating sign constraints on y [13].

The backbone of the method is an efficient routine for computing the maximal eigenvalue of huge structured symmetric matrices. Although our own implementation (based on the code of Hua) seems to work quite well there is certainly much room for improvement. A straightforward approach to achieve serious speed-ups is to implement the algorithm on parallel machines; see, for instance, [43]. Recently, there has been renewed interest in the Lanczos method; see [25, 3, 5, 10, 30] and references therein. Most of these papers are based on the concept of an implicit restart proposed in [44], which is a polynomial acceleration approach that does not require additional matrix vector multiplications. It will be interesting to test these new ideas within the bundle framework.

Appendix. Notation.

$\mathbb{R}^n$	real column vector of dimension n,
$M_{m,n}$	$m \times n$ real matrices,
S_n	$n \times n$ symmetric real matrices,
S_n^{++}	$n \times n$ symmetric positive definite matrices,
S_n^{+}	$n \times n$ symmetric positive semidefinite matrices,
$A \succ 0$	A is positive definite,
$A \succeq 0$	A is positive semidefinite,
I, I_n	identity of appropriate size or of size n,
e	vector of all ones of appropriate dimension,
$\lambda_{\max}(A)$	maximal eigenvalue of A,
$\operatorname{tr} A$	trace of $A \in M_{n,n}, \operatorname{tr} A = \sum_{i=1}^n a_{ii} = \sum_{i=1}^n \lambda_i(A)$,
$\langle A, B\rangle$	inner product in $M_{m,n}$, $\langle A, B\rangle = \operatorname{tr}(B^T A)$,
$\operatorname{svec}(A)$	$\binom{n+1}{2}$-dimensional vector representation of $A \in S_n$,
$A \otimes_s B$	symmetric Kronecker product of $A, B \in M_{n,n}$,
$\operatorname{diag}(A)$	the diagonal of $A \in M_{n,n}$ as a column vector,
$\operatorname{Diag}(v)$	diagonal matrix with v on its main diagonal.

S_n is isomorphic to $\mathbb{R}^{\binom{n+1}{2}}$ via the map $\operatorname{svec}(A)$ defined by stacking the columns of the lower triangle of A on top of each other and multiplying the off-diagonal elements with $\sqrt{2}$,

$$\operatorname{svec}(A) := \left[a_{11}, \sqrt{2}a_{21}, \dots , \sqrt{2}a_{n1}, a_{22}, \sqrt{2}a_{32}, \dots , a_{nn}\right]^T .$$

The factor $\sqrt{2}$ for off-diagonal elements ensures that, for $A, B \in S_n$,

$$\langle A, B\rangle = \operatorname{tr}(AB) = \operatorname{svec}(A)^T \operatorname{svec}(B).$$

The symmetric Kronecker product $\otimes_s$ is defined for arbitrary square matrices $A, B \in M_{n,n}$ by its action on a vector $\operatorname{svec}(C)$ for a symmetric matrix $C \in S_n$,

$$(A \otimes_s B) \operatorname{svec}(C) := \frac{1}{2} \operatorname{svec}(BCA^T + ACB^T).$$

Both concepts were first introduced in [2]. Here we use the notation introduced in [45]. From the latter paper we also cite some properties of the symmetric Kronecker product for the convenience of the reader.

(1) $A \otimes_s B = B \otimes_s A$.
(2) $(A \otimes_s B)^T = B^T \otimes_s A^T$.
(3) $A \otimes_s I$ is symmetric if and only if A is.
(4) $(A \otimes_s A)^{-1} = A^{-1} \otimes_s A^{-1}$.
(5) $(A \otimes_s B)(C \otimes_s D) = \frac{1}{2}(AC \otimes_s BD + AD \otimes_s BC)$.
(6) If $A \succ 0$ and $B \succ 0$ then $(A \otimes_s B) \succ 0$.
(7) $\operatorname{svec}(A)^T \operatorname{svec}(B) = \langle A, B\rangle = \operatorname{tr}(AB)$.

Acknowledgments. We would like to thank K. C. Kiwiel for fruitful discussions and C. Lemaréchal and an anonymous referee for their constructive criticism that helped to improve the presentation.

REFERENCES

[1] F. Alizadeh, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.

[2] F. Alizadeh, J.-P. A. Haeberly, and M. L. Overton, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.

[3] J. Baglama, D. Calvetti, and L. Reichel, *Iterative methods for the computation of a few eigenvalues of a large symmetric matrix*, BIT, 36 (1996), pp. 400–421.

[4] S. Benson, Y. Ye, and X. Zhang, *Solving large-scale sparse semidefinite programs for combinatorial optimization*, SIAM J. Optim., 10 (2000), pp. 443–461.

[5] D. Calvetti, L. Reichel, and D. Sorensen, *An implicitly restarted Lanczos method for large symmetric eigenvalue problems*, Electron. Trans. Numer. Anal., 2 (1994), pp. 1–21.

[6] J. Cullum, W. E. Donath, and P. Wolfe, *The minimization of certain nondifferentiable sums of eigenvalues of symmetric matrices*, Math. Programming Stud., 3 (1975), pp. 35–55.

[7] L. Faybusovich, *Semidefinite programming: A path-following algorithm for a linear-quadratic functional*, SIAM J. Optim., 6 (1996), pp. 1007–1024.

[8] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, J. Assoc. Comput. Mach., 42 (1995), pp. 1115–1145.

[9] G. H. Golub and C. F. van Loan, *Matrix Computations*, 2nd ed., The Johns Hopkins University Press, Baltimore, MD, 1989.

[10] R. G. Grimes, J. G. Lewis, and H. D. Simon, *A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems*, SIAM J. Matrix Anal. Appl., 15 (1994), pp. 228–272.

[11] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, 2nd ed., Algorithms Combin., Springer-Verlag, Berlin, 1988.

[12] C. Helmberg, *Fixing variables in semidefinite relaxations*, in Algorithms—ESA'97, R. Burkard and G. Woeginger, eds., Lecture Notes in Comput. Sci. 1284, Springer, New York, 1997, pp. 259–270.

[13] C. Helmberg, K. C. Kiwiel, and F. Rendl, *Incorporating inequality constraints in the spectral bundle method*, in Integer Programming and Combinatorial Optimization, R. E. Bixby, E. A. Boyd, and R. Z. Ríos-Mercado, eds., Lecture Notes in Comput. Sci. 1412, Springer, New York, 1998, pp. 423–435.

[14] C. Helmberg and F. Rendl, *Solving quadratic* $(0, 1)$*-problems by semidefinite programs and cutting planes*, Math. Programming, 82 (1998), pp. 291–315.

[15] C. Helmberg, F. Rendl, R. J. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[16] C. Helmberg, F. Rendl, and R. Weismantel, *Quadratic knapsack relaxations using cutting planes and semidefinite programming*, in Integer Programming and Combinatorial Optimization, W. H. Cunningham, S. T. McCormick, and M. Queyranne, eds., Lecture Notes in Comput. Sci. 1084, Springer, New York, 1996, pp. 175–189.

[17] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* I, Grundlehren Math. Wiss. 305, Springer-Verlag, Berlin, Heidelberg, 1993.

[18] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* II, Grundlehren Math. Wiss. 306, Springer-Verlag, Berlin, Heidelberg, 1993.

[19] F. Jarre, *An interior-point method for minimizing the maximum eigenvalue of a linear combination of matrices*, SIAM J. Control Optim., 31 (1993), pp. 1360–1377.

[20] S. E. Karisch, F. Rendl, and J. Clausen, *Solving Graph Bisection Problems with Semidefinite Programming*, Technical Report DIKU-TR-97/9, Department of Computer Science, University of Copenhagen, Denmark, July 1997; INFORMS J. Comput., to appear.

[21] K. C. Kiwiel, *Proximity control in bundle methods for convex nondifferentiable minimization*, Math. Programming, 46 (1990), pp. 105–122.

[22] P. Klein and H.-I. Lu, *Efficient approximation algorithms for semidefinite programs arising from MAXCUT and COLORING*, in Proceedings of the Symposium on the Theory of Computing, 1996, pp. 338–347.

[23] M. Kojima, S. Shindoh, and S. Hara, *Interior-point methods for the monotone linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.

[24] M. Laurent, S. Poljak, and F. Rendl, *Connections between semidefinite relaxations of the max-cut and stable set problems*, Math. Programming, 77 (1997), pp. 225–246.

[25] R. B. Lehoucq and K. J. Maschhoff, *Implementation of an Implicitly Restarted Block Arnoldi Method*, Technical Report MCS-P649-0297, Argonne National Laboratory, Argonne, IL, 1997.

[26] C. Lemaréchal, F. Oustry, and C. Sagastizábal, *The $\mathcal{U}$-Lagrangian of a Convex Function*, Trans. Amer. Math. Soc., 352 (2000), pp. 711–729.

[27] L. Lovász, *On the Shannon capacity of a graph*, IEEE Trans. Inform. Theory, IT-25 (1979), pp. 1–7.

[28] L. Lovász and A. Schrijver, *Cones of matrices and set-functions and* 0-1 *optimization*, SIAM J. Optim., 1 (1991), pp. 166–190.

[29] R. D. C. Monteiro, *Polynomial convergence of primal-dual algorithms for semidefinite programming based on Monteiro and Zhang family of directions*, SIAM J. Optim., 8 (1998), pp. 797–812.

[30] R. B. Morgan and D. S. Scott, *Preconditioning the Lanczos algorithm for sparse symmetric eigenvalue problems*, SIAM J. Sci. Comput., 14 (1993), pp. 585–593.

[31] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, Philadelphia, 1994.

[32] Y. Nesterov and M. J. Todd, *Self-scaled barriers and interior-point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.

[33] F. Oustry, *The $\mathcal{U}$-Lagrangian of the maximum eigenvalue function*, SIAM J. Optim., 9 (1999), pp. 526–549.

[34] F. Oustry, *A Second-Order Bundle Method to Minimize the Maximum Eigenvalue Function*, Technical Report, INRIA-ENSTA, France, November 1997; Math. Programming, submitted.

[35] M. L. Overton, *Large-scale optimization of eigenvalues*, SIAM J. Optim., 2 (1992), pp. 88–120.

[36] G. Pataki, *On the rank of extreme matrices in semidefinite programming and the multiplicity of optimal eigenvalues*, Math. Oper. Res., 23 (1998), pp. 339–358.

[37] S. A. Plotkin, D. B. Shmoys, and E. Tardos, *Fast approximation algorithms for fractional packing and covering problems*, Math. Oper. Res., 20 (1995), pp. 257–301.

[38] E. Polak and Y. Wardi, *Nondifferentiable optimization algorithm for designing control systems having singular value inequalities*, Automatica, 18 (1982), pp. 267–283.

[39] S. Poljak and F. Rendl, *Node and edge relaxations of the max-cut problem*, Computing, 52 (1994), pp. 123–137.

[40] S. Poljak and F. Rendl, *Nonpolyhedral relaxations of graph-bisection problems*, SIAM J. Optim., 5 (1995), pp. 467–487.

[41] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[42] H. Schramm and J. Zowe, *A version of the bundle idea for minimizing a nonsmooth function: Conceptual idea, convergence analysis, numerical results*, SIAM J. Optim., 2 (1992), pp. 121–152.

[43] D. S. Scott, *Implementing Lanczos-like algorithms on hypercube architectures*, Comput. Phys. Comm., 53 (1989), pp. 271–281.

[44] D. C. Sorensen, *Implicit application of polynomial filters in a k-step Arnoldi method*, SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.

[45] M. J. Todd, K. C. Toh, and R. H. Tütüncü, *On the Nesterov-Todd direction in semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 769–796.

[46] L. Vandenberghe and S. Boyd, *A primal-dual potential reduction method for problems involving matrix inequalities*, Math. Programming Ser. B, 69 (1995), pp. 205–236.

[47] S. Zhou and H. Dai, *The block Chebyshev-Lanczos method for solving large symmetric eigenvalue problems*, J. Nanjing Aeronaut. Inst., 21 (1989), pp. 22–28.

SIAM J. Optim.
Vol. 10, No. 3, pp. 697–721

DUAL APPLICATIONS OF PROXIMAL BUNDLE METHODS, INCLUDING LAGRANGIAN RELAXATION OF NONCONVEX PROBLEMS*

STEFAN FELTENMARK† AND KRZYSZTOF C. KIWIEL‡

Abstract. We exhibit useful properties of proximal bundle methods for finding $\min_S f$, where f and S are convex. We show that they asymptotically find objective subgradients and constraint multipliers involved in optimality conditions, multipliers of objective pieces for max-type functions, and primal and dual solutions in Lagrangian decomposition of convex programs. When applied to Lagrangian relaxation of nonconvex programs, they find solutions to relaxed convexified versions of such programs. Numerical results are presented for unit commitment in power production scheduling.

Key words. nondifferentiable optimization, convex programming, proximal bundle methods, Lagrangian relaxation, convexified relaxations, unit commitment

AMS subject classifications. 65K05, 90C25

PII. S1052623498332336

1. Introduction.

We consider the convex constrained minimization problem

$$f_* := \min\{f(x) : x \in S\}, \tag{1.1}$$

where S is a nonempty closed convex set in $\mathbb{R}^n$, $f : \mathbb{R}^n \to \mathbb{R}$ is a convex function, and for each $x \in S$ we can compute $f(x)$ and a subgradient $g_f(x) \in \partial f(x)$ of f at x. We assume that the *optimal set* $S_* := \operatorname{Arg\,min}_S f$ is nonempty.

We show that the proximal bundle method [CoL93, Kiw90, Kiw95, Lem77, Mif82, ScZ88], [HUL93, section XV.3] finds asymptotically not only some $x^\infty \in S_*$, but also objective subgradients and constraint multipliers involved in optimality conditions for $\min_S f$, and multipliers of objective pieces when f is a max-type function. Until now, similar results have been known [LPS98] only for subgradient methods with divergent series stepsizes whose convergence is always slow.

We also complement the results of [Kiw95], which show that the proximal bundle method applied to Lagrangian duals of convex programs may find primal solutions by combining partial Lagrangian solutions. In particular, we show that primal recovery is not harmed by the imposition of dual upper bounds, provided such bounds majorize a dual optimal solution. We note that "artificial" upper bounds are frequently used to stabilize other dual methods, and our analysis may justify such heuristic techniques. This has already been done in [FeK97] for cutting plane methods, i.e., generalized linear programming [Dan63, section 22]. Primal recovery for conjugate subgradient bundle methods [Lem75, Wol75] is discussed in [Rzh89, RzK85], and for the ϵ-steepest descent bundle method [LSB81] in [Rob86, Rob89] (see also Remark 5.4 of this paper). For subgradient methods, the earliest result of [Zhu77] (mentioned in [Sho79, section 4.4]) has several extensions [AnW93, LaL89, LPS99, ShC96].

*Received by the editors January 8, 1998; accepted for publication (in revised form) May 20, 1999; published electronically February 29, 2000. This research was supported by the Swedish Research Council for Engineering Sciences, the Göran Gustafsson Foundation, and the Polish Academy of Sciences.

http://www.siam.org/journals/siopt/10-3/33233.html

†Department of Mathematics, Royal Institute of Technology, SE-10044 Stockholm, Sweden (stefanf@math.kth.se).

‡Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01–447 Warsaw, Poland (kiwiel@ibspan.waw.pl).

The dual problem arising in Lagrangian relaxation [Sha79, section 5.3] of a quite arbitrary nonconvex program (cf. [Fis81, Sho79] for mixed integer programs) may be solved by several methods for nondifferentiable optimization [Ber95, sections 6.3–6.4], [HUL93, section XII.4]. We show that, under fairly general assumptions and without extra cost, the proximal bundle method also finds solutions to a relaxed convexified version of the primal program [LeR96, MSW76], [Sha79, section 5.3]. Such results are given in [FeK97] for generalized linear programming [Dan63, section 22], are thus far missing for the subgradient method, and may be expected (but have not been established) to hold for the exponential smoothing method of [Ber82, section 5.6] and [BLSP83], which is restricted to problems with piecewise linear or concave costs and constraints. For the applications solved via the smoothing method [Ber82, section 5.6], [BLSP83], the solution of the relaxed problem can be used to generate a good suboptimal solution to the original problem. Since the proximal bundle method can handle more general problems, we hope that our results may stimulate research on the use of relaxed solutions in other applications.

As a partial justification of our hope, we present numerical results for the unit commitment (UC) problem [Bar88, DGM$^+$97, Fel97, LPRS96, LJS97, MuK77, ShF94, ZhG88] in the daily operation of power systems. This large-scale mixed integer programming problem determines on/off schedules and power outputs of the generators so as to minimize the system operating cost over a planning horizon of 24 to 168 hours.

The paper is organized as follows. In section 2 we review briefly the proximal bundle method of [Kiw90] and its convergence properties. In section 3 we show how certain affine minorants of f and $\imath_S$ (the indicator of S) can be used in stopping criteria and to identify subgradients of f and $\imath_S$ involved in optimality conditions for $\min_S f$. Their uses for identifying multipliers of objective pieces when f is a max-type function and multipliers of constraint functions describing S are discussed in section 4. Applications to Lagrangian decomposition of convex programs are studied in section 5. Extensions to Lagrangian relaxations of nonconvex problems are given in section 6. Our results for the UC application are presented in section 7.

Our notation is fairly standard. $|\cdot|$ is the Euclidean norm. $\partial_\epsilon f(\cdot) := \{p : f(x) \geq f(\cdot) - \epsilon + \langle p, x - \cdot\rangle \ \forall x\}$ is the ϵ-subdifferential of f. $d_C(\cdot) := \inf_{y \in C} |\cdot - y|$ is the distance function of $C \subset \mathbb{R}^n$. "s.t." abbreviates "such that" or "subject to," depending on the context.

2. The proximal bundle method.

We may regard our constrained problem $f_* := \min_S f$ (cf. (1.1)) as the unconstrained problem $f_* = \min f_S$ with the *essential objective*

$$f_S := f + \imath_S, \tag{2.1}$$

where $\imath_S$ is the *indicator function* of the feasible set S ($\imath_S(x) = 0$ if $x \in S$, ∞ if $x \notin S$). Clearly, f_S is convex. Let $\mathcal{N}_S := \partial \imath_S$ denote the *normal cone* operator of S.

The proximal bundle method of [Kiw90] generates a sequence $\{x^k\}_{k=1}^\infty \subset S$ converging to some $x^\infty \in S_*$ and *trial points* $y^k \in S$ for evaluating subgradients $g_f^k := g_f(y^k)$ of f and its *linearizations*

$$f_k(\cdot) := f(y^k) + \langle g_f^k, \cdot - y^k \rangle \leq f(\cdot), \tag{2.2}$$

starting from an arbitrary point $x^1 = y^1 \in S$. Iteration k uses the *polyhedral model* of f

$$\check{f}_k := \max_{j \in J^k} f_j \quad \text{with} \quad k \in J^k \subset \{1\colon k\} \tag{2.3}$$

for finding

$$y^{k+1} := \arg\min \left\{ \check{f}_k(x) + \tfrac{1}{2} u_k |x - x^k|^2 : x \in S \right\}, \tag{2.4}$$

where $u_k > 0$ is a *proximity weight.* A *descent* step to $x^{k+1} = y^{k+1}$ occurs if $f(y^{k+1}) \le f(x^k) + \kappa v_k$, where $\kappa \in (0,1)$ is fixed and

$$v_k := \check{f}_k(y^{k+1}) - f(x^k) \le 0 \tag{2.5}$$

is the *predicted descent* (if $v_k = 0$, then $x^k \in S_*$ and the method may stop). Otherwise, a *null* step $x^{k+1} = x^k$ improves the next model $\check{f}_{k+1}$ with f_{k+1} (cf. (2.3)).

Concerning the proximity weights, we assume for simplicity that, for each $k \ge 1$,

$$u_k \in [u_{\min}, u_{\max}] \quad \text{for some fixed } 0 < u_{\min} \le u_{\max} < \infty, \tag{2.6}$$

and that $u_{k+1} \ge u_k$ if $x^{k+1} = x^k$; i.e., the weight cannot decrease after a null step. More refined u_k-updating techniques are discussed in [Kiw90, Kiw95, Kiw96, LeS97].

It remains to describe the choice of J^{k+1}. By the optimality condition for (2.4)

$$0 \in \partial \left[\check{f}_k(\cdot) + \tfrac{1}{2} u_k |\cdot - x^k|^2 + \imath_S(\cdot) \right] (y^{k+1}),$$

there exists

$$p_f^k \in \partial \check{f}_k(y^{k+1}) \tag{2.7}$$

such that

$$p_S^k := -u_k(y^{k+1} - x^k) - p_f^k \in \mathcal{N}_S(y^{k+1}) := \partial \imath_S(y^{k+1}), \tag{2.8}$$

whereas by (2.7), (2.3), and (2.2) there are multipliers ν_j^k, $j \in J^k$, such that

$$p_f^k = \sum_{j \in J^k} \nu_j^k g_f^j, \ \sum_{j \in J^k} \nu_j^k = 1, \ \nu_j^k \ge 0, \ \nu_j^k \left[\check{f}_k(y^{k+1}) - f_j(y^{k+1}) \right] = 0, \ j \in J^k. \tag{2.9}$$

Let

$$\hat{J}^k := \left\{ j \in J^k : \nu_j^k > 0 \right\}. \tag{2.10}$$

To save storage without impairing convergence, it suffices to choose $J^{k+1} \supset \hat{J}^k \cup \{k+1\}$ (i.e., we may drop linearizations f_j with $\nu_j^k = 0$ that do not contribute to p_f^k in (2.9), and hence have no influence on y^{k+1} and $\check{f}_k(y^{k+1})$; cf. (2.7)–(2.8)).

From now on, $\{x^k\}$, $\{v_k\}$, etc. denote the sequences generated by the above method, under the assumptions introduced below (1.1). Note that (cf. (2.1)) $f_S(x^k) = f(x^k)$ for each k, since by construction (cf. (2.4)) $\{y^k\}$ and $\{x^k\}$ lie in S.

The analysis of [Kiw90, Kiw95] yields the following global convergence result.

THEOREM 2.1. *The sequence $\{x^k\}$ converges to a solution x^∞ of problem* (1.1), *i.e., a point x^∞ in S_*. Moreover, $f(x^k) \downarrow f(x^\infty) = f_* := \min_S f$ and $v_k \to 0$ as $k \to \infty$.*

Remark 2.2.

(i) The assumption that f is finite on $\mathbb{R}^n$ may be weakened as follows. Inspection of the proofs of [Kiw90, Kiw95] reveals that Theorem 2.1 requires only convexity and finiteness of f on S and *local boundedness* of $g_f(\cdot) \in \partial f(\cdot)$ on S (boundedness of g_f on bounded subsets of S). Indeed, then f is locally Lipschitz continuous on S (since $f(x) - f(y) \le \langle g_f(x), x - y\rangle \le |g_f(x)||x - y|\ \forall x, y \in S$).

(ii) Note that y^{k+1} (cf. (2.4)) and $\xi_k := \check{f}_k(y^{k+1}) = f(x^k) + v_k$ (cf. (2.5)) solve

$$\text{(2.11a)} \qquad \min \quad \tfrac{1}{2}u_k|x - x^k|^2 + \xi \qquad \text{over all } (x, \xi) \in S \times \mathbb{R},$$

$$\text{(2.11b)} \qquad \text{s.t.} \quad f_j(x^k) + \left\langle g_f^j, x - x^k \right\rangle \le \xi \qquad \forall j \in J^k.$$

When S is polyhedral, the quadratic programming (QP) method of [Kiw94] finds (y^{k+1}, ξ_k) and multipliers ν_j^k of (2.11b) such that (2.9) holds and $|\hat{J}^k| \le n + 1$ (cf. (2.10)).

3. Optimal objective and constraint subgradients.

We shall show that the *aggregate subgradients* p_f^k and $-p_S^k$ (cf. (2.7)–(2.8)) converge to the *optimal subgradient set*

$$\text{(3.1)} \qquad \mathcal{G} := \partial f(x^\infty) \cap -\mathcal{N}_S(x^\infty)$$

of our problem $\min_S f$, where $x^\infty \in S_*$ is the limit point of $\{x^k\}$ (cf. Theorem 2.1).

Remark 3.1. It is a fact that $\mathcal{G}$ does not really depend on x^∞, since (cf. [BuF91, Lemma 2])

$$\text{(3.2)} \qquad \mathcal{G} = \partial f(x) \cap -\mathcal{N}_S(x) \quad \text{for every } x \text{ in } S_*.$$

Further, $\mathcal{G}$ is closed and convex (so are $\partial f(x^\infty)$ and $\partial \imath_S(x^\infty)$) and nonempty (as will be seen from Theorem 3.4 below). In general, if $\partial f(x) \cap -\mathcal{N}_S(x) \ne \emptyset$ then $x \in S_*$, since $0 \in \partial f(x) + \mathcal{N}_S(x) \subset \partial f_S(x)$, and $0 \in \partial f_S(x)$ iff $x \in S_*$, but we do not require that $\partial f_S = \partial f + \partial \imath_S$, in view of the weakened assumption of Remark 2.2(i). Thus, $\mathcal{G}$ describes subgradients of f and $\imath_S$ involved in the optimality condition $0 \in \partial f(x) + \mathcal{N}_S(x)$ that characterizes solutions x of $\min_S f$.

We shall employ the following *aggregate linearizations* of f, $\imath_S$, and f_S (cf. (2.1)):

$$\text{(3.3)} \quad \tilde{f}_k(\cdot) := \check{f}_k(y^{k+1}) + \left\langle p_f^k, \cdot - y^{k+1} \right\rangle, \quad \tilde{\imath}_S^k(\cdot) := \left\langle p_S^k, \cdot - y^{k+1} \right\rangle, \quad \tilde{f}_S^k := \tilde{f}_k + \tilde{\imath}_S^k,$$

stemming from (2.7)–(2.8) with $\imath_S(y^{k+1}) = 0$. They are described by their gradients

$$\text{(3.4)} \qquad \nabla \tilde{f}_k = p_f^k, \quad \nabla \tilde{\imath}_S^k = p_S^k, \quad \nabla \tilde{f}_S^k = p^k := p_f^k + p_S^k = -u_k(y^{k+1} - x^k)$$

(cf. (2.8) for the final equality), as well as their *linearization errors* at x^k

$$\text{(3.5)} \qquad \tilde{\epsilon}_f^k := f(x^k) - \tilde{f}_k(x^k), \quad \tilde{\epsilon}_S^k := \imath_S(x^k) - \tilde{\imath}_S^k(x^k), \quad \tilde{\epsilon}_k := f(x^k) - \tilde{f}_S^k(x^k).$$

The following preliminary technical result lists their well-known properties [Kiw95].

LEMMA 3.2.

(i) $\tilde{f}_k \le f$, $\tilde{\imath}_S^k \le \imath_S$, $\tilde{f}_S^k \le f_S$.

(ii) $\tilde{f}_k = \sum_{j \in J^k} \nu_j^k f_j$.

(iii) $p_f^k \in \partial_{\tilde{\epsilon}_f^k} f(x^k)$, $p_S^k \in \partial_{\tilde{\epsilon}_S^k} \imath_S(x^k)$, $p^k \in \partial_{\tilde{\epsilon}_k} f_S(x^k)$, *with* $\tilde{\epsilon}_f^k \ge 0$, $\tilde{\epsilon}_S^k \ge 0$, $\tilde{\epsilon}_k = \tilde{\epsilon}_f^k + \tilde{\epsilon}_S^k \ge 0$. *Further,*

$$\text{(3.6)} \qquad f_S(x) \ge \tilde{f}_S^k(x) = f(x^k) - \tilde{\epsilon}_k + \left\langle p^k, x - x^k \right\rangle \qquad \forall x.$$

(iv) $-v^k = u_k|y^{k+1} - x^k|^2 + \tilde\epsilon_k = |p^k|^2/u_k + \tilde\epsilon_k \ge 0$.

Proof. (i) (3.3), (2.7), (2.8) with $\imath_S(y^{k+1}) = 0$ (since $y^{k+1} \in S$ by (2.4)) and the subgradient inequality yield $\tilde f_k \le \check f_k$ and $\tilde\imath_S^k \le \imath_S$. By adding these inequalities and using (cf. (2.2)–(2.3)) $\check f_k \le f$, we get $\tilde f_S^k := \tilde f_k + \tilde\imath_S^k \le f + \imath_S =: f_S$.

(ii) Since $f_j(\cdot) := f(y^j) + \langle g_f^j, \cdot - y^j\rangle = f_j(y^{k+1}) + \langle g_f^j, \cdot - y^{k+1}\rangle$, (2.9) and (3.3) yield

$$\tilde f_k(\cdot) = \sum_{j\in J^k} \nu_j^k \left[f_j(y^{k+1}) + \left\langle g_f^j, \cdot - y^{k+1}\right\rangle\right] = \sum_{j\in J^k} \nu_j^k f_j(\cdot).$$

(iii) By (i) and (3.3)–(3.5), $f(\cdot) \ge \tilde f_k(\cdot) = f(x^k) - \tilde\epsilon_f^k + \langle p_f^k, \cdot - x^k\rangle$, $\imath_S(\cdot) \ge \tilde\imath_S^k(\cdot) = \imath_S(x^k) - \tilde\epsilon_S^k + \langle p_S^k, \cdot - x^k\rangle$, $f_S(\cdot) \ge \tilde f_S^k(\cdot) = f(x^k) - \tilde\epsilon_k + \langle p^k, \cdot - x^k\rangle$ with $\imath_S(x^k) = 0$ (since $x^k \in S$) and $\tilde\epsilon_k = f(x^k) - \tilde f_k(x^k) - \tilde\imath_S^k(x^k) = \epsilon_f^k + \epsilon_S^k$; set $\cdot = x^k$ to get $\tilde\epsilon_f^k, \tilde\epsilon_S^k, \tilde\epsilon_k \ge 0$.

(iv) Using (3.3) and the right-most equalities in (3.4), we have

$$\check f_k(y^{k+1}) = \tilde f_S^k(y^{k+1}) = \tilde f_S^k(x^k) + \left\langle p^k, y^{k+1} - x^k\right\rangle = \tilde f_S^k(x^k) - u_k|y^{k+1} - x^k|^2$$

and $u_k|y^{k+1} - x^k|^2 = |p^k|^2/u_k$, so $-v_k = f(x^k) - \check f_k(y^{k+1}) = \tilde\epsilon_k + |p^k|^2/u_k$ by (2.5) and (3.5), where $\tilde\epsilon_k \ge 0$ by (iii). □

We now begin our study of asymptotic properties of the aggregate linearizations $\tilde f_k$, $\tilde\imath_S^k$, and $\tilde f_S^k$ (cf. (3.3)). First, we show that their errors $\tilde\epsilon_f^k$, $\tilde\epsilon_S^k$, and $\tilde\epsilon_k$ (cf. (3.5)), as well as the gradient p^k of $\tilde f_S^k$ (cf. (3.4)), vanish asymptotically. In effect, the graph of $\tilde f_S^k$ becomes horizontal, thus confirming via (3.6) the optimality of $x^\infty := \lim_k x^k$ (cf. Theorem 2.1), whereas $\tilde f_k$ and $\tilde\imath_S^k$ converge to the set of linearizations of f and $\imath_S$ at x^∞, provided their gradients (cf. (3.4)) p_f^k and p_S^k are bounded (i.e., the graphs of $\tilde f_k$ and $\tilde\imath_S^k$ do not become vertical). Since p_f^k is a convex combination of the past subgradients $\{g_f^j\}_{j\in J^k}$ (cf. (2.9)), its boundedness, as well as that of $p_S^k = p^k - p_f^k$ (cf. (3.4)), will follow from the boundedness of g_f^k, which we establish next.

LEMMA 3.3.

(i) *In the notation of* (3.5) *and* (3.4), *we have*

$$\tilde\epsilon_f^k \to 0, \quad \tilde\epsilon_S^k \to 0, \quad \tilde\epsilon_k \to 0, \quad \text{and} \quad p^k \to 0 \quad \text{as} \quad k \to \infty.$$

(ii) $\lim_{k\to\infty} y^k = x^\infty$ $(:= \lim_{k\to\infty} x^k$; *cf. Theorem* 2.1), *and* $\{g_f^k\}$ *is bounded.*

Proof. (i) and (ii). By Lemma 3.2(iii), (iv), $0 \le \tilde\epsilon_f^k, \tilde\epsilon_S^k, \tilde\epsilon_k \le -v_k \to 0$ (cf. Theorem 2.1). Then $|p^k|^2/u_k = u_k|y^{k+1} - x^k|^2 \le -v_k$ (cf. Lemma 3.2(iv)) with $u_k \in [u_{\min}, u_{\max}]$ (cf. (2.6)) give $p^k \to 0$, $y^{k+1} - x^k \to 0$. Hence, $y^k \to x^\infty$, since $x^k \to x^\infty$. Thus $\{y^k\}$ is bounded, and so is $\{g_f^k := g_f(y^k)\}$, since g_f is locally bounded on S. □

We may now show that p_f^k and $-p_S^k$ are bounded and converge to the optimal subgradient set $\mathcal{G}$ (cf. (3.1)) as x^k approaches x^∞, whereas $\tilde f_k$ and $\tilde\imath_S^k$ converge to the corresponding set of "optimal" linearizations of f and $\imath_S$ at x^∞. This fairly abstract result will form the basis of the more concrete results of sections 4–6.

THEOREM 3.4.

(i) $\{p_f^k\}$ *is bounded and each cluster point of* $\{p_f^k\}$ *lies in* $\partial f(x^\infty)$.

(ii) *Let* p_f^∞ *be a cluster point of* $\{p_f^k\}$. *Let* $K \subset \{1, 2, \dots\}$ *be such that* $p_f^k \xrightarrow{K} p_f^\infty$. *Then* $p_f^\infty \in \mathcal{G}$. *Moreover,*

$$p_S^k \xrightarrow{K} p_S^\infty, \quad \tilde f_k(\cdot) \xrightarrow{K} \tilde f_\infty(\cdot), \quad \text{and} \quad \tilde\imath_S^k(\cdot) \xrightarrow{K} \tilde\imath_S^\infty(\cdot),$$

where

$$p_S^\infty := -p_f^\infty \in \mathcal{N}_S(x^\infty), \quad \tilde{f}_\infty(\cdot) := f(x^\infty) + \left\langle p_f^\infty, \cdot - x^\infty \right\rangle, \quad \tilde{\imath}_S^\infty(\cdot) := \left\langle p_S^\infty, \cdot - x^\infty \right\rangle.$$

(iii) $\{p_S^k\}$ *is bounded and each cluster point of* $\{p_S^k\}$ *lies in* $\mathcal{N}_S(x^\infty)$.

(iv) $d_{\mathcal{G}}(p_f^k) \to 0$ *and* $d_{\mathcal{G}}(-p_S^k) \to 0$ *as* $k \to \infty$.

Proof. (i) By (2.9), $p_f^k \in \operatorname{co}\{g_f^j\}_{j=1}^k$. Hence, $\{p_f^k\}$ is bounded (so is $\{g_f^k\}$ by Lemma 3.3(ii)). Next, $p_f^k \in \partial_{\tilde{\epsilon}_f^k} f(x^k)$ (Lemma 3.2(iii)) with $x^k \to x^\infty$ and $\tilde{\epsilon}_f^k \to 0$ (Lemma 3.3(i)) imply that each cluster point of $\{p_f^k\}$ lies in $\partial f(x^\infty)$, since the approximate subdifferential mapping $(x, \epsilon) \to \partial_\epsilon f(x)$ is closed [HUL93, section XI.4.1].

(ii) Using (3.3), the facts that $\tilde{\epsilon}_f^k := f(x^k) - \tilde{f}_k(x^k) \to 0$ (cf. (3.5) and Lemma 3.3(i)) and $f(x^k) \downarrow f(x^\infty)$ (cf. Theorem 2.1), and our assumption $p_f^k \xrightarrow{K} p_f^\infty$, we obtain

$$\tilde{f}_k(\cdot) = f(x^k) - \tilde{\epsilon}_f^k + \left\langle p_f^k, \cdot - x^k \right\rangle \xrightarrow{K} f(x^\infty) + \left\langle p_f^\infty, \cdot - x^\infty \right\rangle =: \tilde{f}_\infty(\cdot).$$

By (i), $p_f^\infty \in \partial f(x^\infty)$. Next, $p^k - p_f^k = p_S^k \in \partial_{\tilde{\epsilon}_S^k} \imath_S(x^k)$ (cf. (3.4) and Lemma 3.2(iii)) with $p^k \to 0$, $\tilde{\epsilon}_S^k \to 0$ (cf. Lemma 3.3(i)) yield $p_S^k \xrightarrow{K} -p_f^\infty \in \partial \imath_S(x^\infty)$ by the closedness of $\partial_\epsilon \imath_S(x)$. Since $\tilde{\epsilon}_S^k := -\tilde{\imath}_S^k(x^k) \to 0$ (cf. (3.5) and Lemma 3.3(i)) and $p_S^\infty := -p_f^\infty$, we have $\tilde{\imath}_S^k(\cdot) = \tilde{\imath}_S^k(x^k) + \left\langle p_S^k, \cdot - x^k \right\rangle \xrightarrow{K} \tilde{\imath}_S^\infty(\cdot)$.

(iii) By (i), (ii), $\{p_S^k = p^k - p_f^k\}$ is bounded, since $p^k \to 0$ and $\{p_f^k\}$ is bounded. If $\{p_S^k\}$ has a cluster point p_S^∞, then by (i), (ii), $\{p_f^k\}$ has a cluster point p_f^∞ s.t. $p_S^\infty = -p_f^\infty \in \mathcal{N}_S(x^\infty)$.

(iv) This follows from (i)–(iii) and the continuity of $d_{\mathcal{G}}$ (e.g., pick K s.t. $d_{\mathcal{G}}(p_f^k) \xrightarrow{K} \limsup_k d_{\mathcal{G}}(p_f^k)$ and, using (i), (ii), $p_f^k \xrightarrow{K} p_f^\infty \in \mathcal{G}$ to get $d_{\mathcal{G}}(p_f^k) \xrightarrow{K} d_{\mathcal{G}}(p_f^\infty) = 0$). □

The full strength of Theorem 3.4 will be exploited later. The remainder of this section is devoted to two simple, but quite useful, results related to Lemma 3.3 and Theorem 3.4.

The usual stopping criteria of proximal bundle methods (cf. [Kiw90, ScZ88], [HUL93, section XV.3]) tend to work quite well in most cases, but they do not *guarantee* that, for a given $\epsilon > 0$, $f(x^k) \le f_* + \epsilon$ upon termination. The following result may be used for developing alternative stopping criteria when S is bounded, as happens in many applications.

LEMMA 3.5. *Suppose the feasible set* S *is bounded. Let* $\tilde{f}_{\min}^k := \min_S \tilde{f}_S^k$ *for all* $k \ge 1$. *Then* $\tilde{f}_{\min}^k \le \min_S \tilde{f}_k \le f_*$ *for all* k, *and* $\tilde{f}_{\min}^k \to f_*$ *as* $k \to \infty$.

Proof. The inequalities $f_* \ge \min_S \tilde{f}_k \ge \tilde{f}_{\min}^k$ follow from (cf. Lemma 3.2(i))

$$f(x) \ge \tilde{f}_k(x) = \tilde{f}_S^k(x) - \tilde{\imath}_S^k(x) \ge \tilde{f}_S^k(x) \quad \text{for each } x \text{ in } S \tag{3.7}$$

(since $\tilde{\imath}_S^k(x) \le \imath_S(x) = 0\ \forall x \in S$). Let $\tilde{x}^k \in \operatorname{Arg\,min}_S \tilde{f}_S^k$ so that $\tilde{f}_S^k(\tilde{x}^k) = \tilde{f}_{\min}^k \le f_* \le f(x^k)$. Set $x = \tilde{x}^k$ in (3.6) and use the Cauchy–Schwarz inequality together with $\tilde{\epsilon}_k, |p^k| \to 0$ (cf. Lemma 3.3(i)) and boundedness of $\{x^k\}, \{\tilde{x}^k\} \subset S$ to get

$$0 \le f(x^k) - \tilde{f}_{\min}^k = f(x^k) - \tilde{f}_S^k(\tilde{x}^k) = \tilde{\epsilon}_k - \left\langle p^k, \tilde{x}^k - x^k \right\rangle \le \tilde{\epsilon}_k + |p^k||\tilde{x}^k - x^k| \to 0.$$

However, $f(x^k) \downarrow f_*$ (Theorem 2.1), so the preceding relation gives $\tilde{f}_{\min}^k \to f_*$. □

Remark 3.6. When S is bounded, we may compute the lower bounds on f_*:

$$\tilde f_{\rm low}^k := \max\left\{\min_S \tilde f_k, \tilde f_{\rm low}^{k-1}\right\} \quad \text{for} \quad k \geq 1, \quad \text{with} \quad \tilde f_{\rm low}^0 := -\infty. \tag{3.8}$$

Since $\tilde f_{\rm low}^k \uparrow f_*$ (cf. Lemma 3.5), whereas $f(x^k) \downarrow f_*$ (Theorem 2.1), for any $\epsilon > 0$ there is k s.t. $f(x^k) - \tilde f_{\rm low}^k \leq \epsilon$, implying $f(x^k) \leq f_* + \epsilon$. This validates a stopping criterion of the form $f(x^k) - \tilde f_{\rm low}^k \leq \epsilon$. Note that it is better to use $\tilde f_k$ instead of $\tilde f_S^k$ in (3.8), since $\tilde f_k \geq \tilde f_S^k$ on S (cf. (3.7)). Conversely, if the computation of $\min_S \tilde f_k$ is difficult, but it is easier to find $\min_{\tilde S} \tilde f_S^k$ for some "simpler" bounded set $\tilde S \supset S$, then $\min_{\tilde S} \tilde f_S^k$ may replace $\min_S \tilde f_k$ in (3.8) (since $\min_{\tilde S} \tilde f_S^k \leq f_*$ and $\min_{\tilde S} \tilde f_S^k \to f_*$ by the proof of Lemma 3.5 with S replaced by $\tilde S$); in fact it may be more efficient to use $\tilde f_{\rm low}^k := \max\{\min_{\tilde S} \tilde f_S^k, \min_{\tilde S} \tilde f_k, \tilde f_{\rm low}^{k-1}\}$.

Having the feasible set S bounded is useful both for stopping criteria (cf. Remark 3.6) and for preventing "too long" steps away from S_*, especially at early iterations, when $\check f_k$ is a poor model of f. In some applications (cf. Example 4.5 and Remarks 5.3(ii) and 6.3(ii)), one wants to find $\min_{\breve S} f$ for an unbounded set $\breve S$, but one can find a bounded set $\bar S$ that intersects $\operatorname{Arg}\min_{\breve S} f$. Then it is natural to solve, instead of the original problem $\min_{\breve S} f$, its restricted version $\min_S f$ with $S = \breve S \cap \bar S$, since $\operatorname{Arg}\min_S f \subset \operatorname{Arg}\min_{\breve S} f$ and S is bounded. In other words, one imposes "artificial" constraints given by $\bar S$ on the original problem $\min_{\breve S} f$ to ensure boundedness of the feasible set S. In view of Theorem 3.4, this raises the question about the relationship of the optimal subgradient set $\mathcal{G}$ (cf. (3.1)) for $\min_S f$ to the optimal subgradient set of $\min_{\breve S} f$. A simple answer is given below.

LEMMA 3.7. *Suppose* $\min_S f$ *is a restriction of the original problem* $\min_{\breve S} f$ *in the sense that* $S = \breve S \cap \bar S$ *for two convex sets* $\breve S$ *and* $\bar S$. *Let* $\breve S_* := \operatorname{Arg}\min_{\breve S} f$. *Suppose* $\breve S_* \cap \operatorname{int} \bar S \neq \emptyset$. *Then the solution sets* S_* *of* $\min_S f$ *and* $\breve S_*$ *of* $\min_{\breve S} f$ *satisfy* $\emptyset \neq S_* \subset \breve S_*$, *and together with* (3.2), *i.e.,*

$$\mathcal{G} = \partial f(x) \cap -\mathcal{N}_S(x) \quad \textit{for every } x \textit{ in } S_*,$$

we have

$$\mathcal{G} = \partial f(x) \cap -\mathcal{N}_{\breve S}(x) \quad \textit{for every } x \textit{ in } \breve S_*.$$

Thus $\mathcal{G}$ *is the common optimal subgradient set of both* $\min_S f$ *and* $\min_{\breve S} f$.

Proof. Clearly, $\breve S_* \cap \operatorname{int} \bar S \subset S_* \subset \breve S_*$. Let $\breve x \in \breve S_* \cap \operatorname{int} \bar S$ and $\breve{\mathcal{G}} := \partial f(\breve x) \cap -\mathcal{N}_{\breve S}(\breve x)$. By [BuF91, Lemma 2] applied to $\min_{\breve S} f$, $\breve{\mathcal{G}} = \partial f(x) \cap -\mathcal{N}_{\breve S}(x)$ $\forall x \in \breve S_*$. However, $\mathcal{N}_{\bar S}(\breve x) = \{0\}$ and $\mathcal{N}_{\breve S}(\breve x) = \mathcal{N}_{\breve S}(\breve x) + \mathcal{N}_{\bar S}(\breve x) = \mathcal{N}_S(\breve x)$, so $\breve{\mathcal{G}} = \partial f(\breve x) \cap -\mathcal{N}_S(\breve x)$. Hence, by [BuF91, Lemma 2] applied to $\min_S f$, $\breve{\mathcal{G}} = \partial f(x) \cap -\mathcal{N}_S(x)$ $\forall x \in S_*$. For $x = x^\infty$, we get $\breve{\mathcal{G}} = \mathcal{G}$ (cf. (3.1)). □

Remark 3.8. Under the assumptions of Lemma 3.7, $\mathcal{N}_{\breve S}$ may replace $\mathcal{N}_S$ in Theorem 3.4; then $\mathcal{G} = \partial f(x^\infty) \cap -\mathcal{N}_{\breve S}(x^\infty)$ characterizes "optimal" subgradients for both $\min_S f$ *and* $\min_{\breve S} f$. In general, if $\breve S_* \neq \emptyset$ then it suffices to choose $\bar S$ "large enough" but compact to have S bounded as well. A useful illustration is given in Example 4.5.

4. Particular cases. By using elementary subdifferential calculus, we now specialize the results of section 3 to the cases where we have explicit representations of f as a finite-max-type function and of S as the solution set of finitely many nonlinear inequalities and linear equalities. (The readers mainly interested in Lagrangian relaxation may skip this section.)

4.1. Minimax objective multipliers. In this subsection we assume that the objective f has the finite max form

$$f(x) = \max_{i \in I} h_i(x) \quad \forall x,$$

where $|I| < \infty$ and each $h_i : \mathbb{R}^n \to \mathbb{R}$ is convex. Let

$$\Lambda := \left\{ \lambda \in \mathbb{R}_+^{|I|} : \sum\nolimits_{i \in I} \lambda_i = 1 \right\}$$

and, for each x,

$$I(x) := \{ i \in I : h_i(x) = f(x) \},$$

$$\Lambda(x) := \{ \lambda \in \Lambda : \lambda_i = 0 \text{ if } i \notin I(x) \},$$

so that

$$\partial f(x) = \left\{ \sum\nolimits_{i \in I} \lambda_i \partial h_i(x) : \lambda \in \Lambda(x) \right\}. \tag{4.1}$$

The (possibly empty) set of *optimal multipliers* associated with any x

$$\Lambda^*(x) := \left\{ \lambda \in \Lambda(x) : \left(\sum\nolimits_{i \in I} \lambda_i \partial h_i(x) \right) \cap -\mathcal{N}_S(x) \neq \emptyset \right\} \tag{4.2}$$

has the following properties (cf. [LPS98, Prop. 5.8]).

LEMMA 4.1.

(i) *For each x, $x \in S_*$ iff $\Lambda^*(x) \neq \emptyset$, and $\Lambda^*(x)$ is compact and convex.*

(ii) *For each $\bar{x} \in S_*$ (e.g., $\bar{x} = x^\infty$; cf. Theorem 2.1), $\Lambda^*(\bar{x}) = \Lambda^*(x)$ $\forall x \in S_*$.*

(iii) *Under the assumptions of Lemma 3.7, define $\breve{\Lambda}^*(x)$ via (4.2) with $\breve{S}$ replacing S. Then $\Lambda^*(x) = \breve{\Lambda}^*(x)$ $\forall x \in S_*$ so that also $\breve{\Lambda}^*(x)$ is independent of $x \in S_*$.*

Proof. (i) $x \in S_* \Leftrightarrow \partial f(x) \cap -\mathcal{N}_S(x) \neq \emptyset \Leftrightarrow \Lambda^*(x) \neq \emptyset$, using (4.1) in (4.2). The compactness and convexity of $\Lambda^*(x)$ follow from those of $\Lambda(x)$ and $\partial h_i(x)$, $i \in I$.

(ii) Suppose $\lambda \in \Lambda^*(\bar{x})$. Using (4.1) in (4.2), we get $f_* = f(\bar{x}) = \sum_i \lambda_i h_i(\bar{x})$ and

$$f_* := \min_{x \in S} \{ f(x) := \max\nolimits_i h_i(x) \} = \min_{x \in S} \sum\nolimits_i \lambda_i h_i(x),$$

so for each x in $S_* := \operatorname{Arg\,min}_S f$, $\lambda_i = 0$ if $h_i(x) < f(x) = f_*$; thus $\lambda \in \Lambda(x)$ and x minimizes $\sum_i \lambda_i h_i$ over S, i.e., $(\sum_i \lambda_i \partial h_i(x)) \cap -\mathcal{N}_S(x) \neq \emptyset$, and hence $\lambda \in \Lambda^*(x)$. Therefore, $\Lambda^*(\bar{x}) \subset \Lambda^*(x)$. By a symmetric argument, $\Lambda^*(\bar{x}) \supset \Lambda^*(x)$, so $\Lambda^*(\bar{x}) = \Lambda^*(x)$.

(iii) Let $x \in S_*$. Since $\partial f(x) \cap -\mathcal{N}_S(x) = \partial f(x) \cap -\mathcal{N}_{\breve{S}}(x)$ (Lemma 3.7), (4.1)–(4.2) yield $\Lambda^*(x) \subset \breve{\Lambda}^*(x)$. Since $\mathcal{N}_{\breve{S}}(x) \subset \mathcal{N}_S(x)$ from $\breve{S} \supset S$, (4.2) gives $\breve{\Lambda}^*(x) \subset \Lambda^*(x)$. □

Even with limited access to the subdifferential ∂f, estimates of optimal multipliers may be produced as follows. By (4.1), for each k, the subgradient g_f^k of f obtained at y^k has the form

$$g_f^k = \sum_{i \in I} \lambda_i^k g_{h_i}^k \quad \text{with} \quad \lambda^k \in \Lambda(y^k),\ g_{h_i}^k \in \partial h_i(y^k),\ i \in I. \tag{4.3}$$

The corresponding linearizations of the component functions h_i at y^k are given by

$$h_i^k(\cdot) := h_i(y^k) + \left\langle g_{h_i}^k, \cdot - y^k \right\rangle \le h_i(\cdot), \quad i \in I. \tag{4.4}$$

Using the weights $\{\nu_j^k\}_{j\in J^k}$ (cf. (2.9)), we define the kth *aggregate multiplier*

$$\tilde\lambda^k := \sum_{j\in J^k} \nu_j^k \lambda^j. \tag{4.5}$$

This is similar to the aggregate linearization $\tilde f_k = \sum_{j\in J^k} \nu_j^k f_j$ (Lemma 3.2(ii)). Hence, to prepare for the convergence analysis of $\{\tilde\lambda^k\}$, it will be convenient to express each f_j as a combination of the component linearizations $\{h_i^j\}_{i\in I}$ and then $\tilde f_k$ as a combination of the following aggregate linearizations of the component functions h_i:

$$\tilde h_i^k(\cdot) := \left\{\begin{matrix} \sum_{j\in J^k} (\nu_j^k \lambda_i^j / \tilde\lambda_i^k) h_i^j(\cdot) & \text{if } \tilde\lambda_i^k > 0 \\ h_i^k(\cdot) & \text{otherwise} \end{matrix}\right\}, \quad i \in I. \tag{4.6}$$

Basic properties of such linearizations are given in the following.

LEMMA 4.2.

(i) $f_k = \sum_{i\in I} \lambda_i^k h_i^k$.

(ii) $\tilde\lambda^k \in \Lambda$, $\tilde h_i^k \in \operatorname{co}\{h_i^j\}_{j\in J^k}$, *and* $\tilde h_i^k \le h_i$ *for all* $i \in I$, *and* $\tilde f_k = \sum_{i\in I} \tilde\lambda_i^k \tilde h_i^k$.

(iii) $\nabla \tilde h_i^k \in \partial_{\tilde\epsilon_i^k} h_i(x^k)$, *where* $\tilde\epsilon_i^k := h_i(x^k) - \tilde h_i^k(x^k) \ge 0$, $i \in I$.

Proof. (i) Use (2.2) and (4.3)–(4.4) with $h_i(y^k) = f(y^k)$ if $\lambda_i^k > 0$.

(ii) Since (cf. (2.9)) $\sum_j \nu_j^k = 1$ with $\nu_j^k \ge 0$ and (cf. (4.3)) $\lambda^j \in \Lambda$, $\tilde\lambda^k \in \Lambda$ by (4.5) and the convexity of Λ. Next, $\tilde\lambda_i^k = \sum_j \nu_j^k \lambda_i^j$ in (4.6) yields $\tilde h_i^k \in \operatorname{co}\{h_i^j\}_{j\in J^k}$, and hence $\tilde h_i^k \le h_i$ by (4.4). Finally, use Lemma 3.2(ii), (i) and (4.6) to get

$$\tilde f_k = \sum_{j\in J^k} \nu_j^k f_j = \sum_{j\in J^k} \nu_j^k \sum_{i\in I} \lambda_i^j h_i^j = \sum_{i\in I} \sum_{j\in J^k} \nu_j^k \lambda_i^j h_i^j = \sum_{i\in I} \tilde\lambda_i^k \tilde h_i^k.$$

(iii) By (ii), $h_i(\cdot) \ge \tilde h_i^k(\cdot) = h_i(x^k) - \tilde\epsilon_i^k + \langle \nabla \tilde h_i^k, \cdot - x^k \rangle$, $i \in I$. □

We may now show that the sequence of aggregate multipliers $\{\tilde\lambda^k\}$, constructed via (4.5) and (4.3), converges to the optimal multiplier set $\Lambda^*(x^\infty)$ as $\{x^k\}$ converges to x^∞.

THEOREM 4.3.

(i) $\{\tilde\lambda^k\}$ *is bounded and all its cluster points lie in* Λ.

(ii) *Each cluster point of* $\{\tilde\lambda^k\}$ *lies in* $\Lambda^*(x^\infty)$.

(iii) $d_{\Lambda^*(x^\infty)}(\tilde\lambda^k) \to 0$ *as* $k \to \infty$.

Proof. (i) By Lemma 4.2(ii), $\{\tilde\lambda^k\} \subset \Lambda$, a compact set.

(ii) We first show that, for each $i \in I$, $\{\nabla \tilde h_i^k\}$ is bounded. Since $\tilde h_i^k \in \operatorname{co}\{h_i^j\}_{j\in J^k}$ by Lemma 4.2(ii), we have $\nabla \tilde h_i^k \in \operatorname{co}\{\nabla h_i^j\}_{j\in J^k}$ with $\nabla h_i^j = g_{h_i}^j \in \partial h_i(y^j)$ by (4.3) and (4.4). Hence $\{\nabla \tilde h_i^k\}$ is bounded (so is $\{y^k\}$ by Lemma 3.3(ii), and ∂h_i is locally bounded).

Next, let $\tilde\lambda^\infty$ be a cluster point of $\{\tilde\lambda^k\}$. Then, by the boundedness of $\{\nabla \tilde h_i^k\}$, there are $K \subset \{1, 2, \dots\}$ and $\nabla \tilde h_i^\infty$, $i \in I$, such that $\tilde\lambda^k \xrightarrow{K} \tilde\lambda^\infty$, $\nabla \tilde h_i^k \xrightarrow{K} \nabla \tilde h_i^\infty$, $i \in I$. By (i), $\tilde\lambda^\infty \in \Lambda$. Using (3.3), Lemma 4.2(ii) and Theorem 3.4(ii), we get

$$p_f^k = \nabla \tilde f_k = \sum\nolimits_i \tilde\lambda_i^k \nabla \tilde h_i^k \xrightarrow{K} \sum\nolimits_i \tilde\lambda_i^\infty \nabla \tilde h_i^\infty =: p_f^\infty \in -\mathcal{N}_S(x^\infty). \tag{4.7}$$

Next, by Lemma 4.2(ii), since $\sum_i \tilde\lambda_i^k = 1$ and $\tilde f_k = \sum_i \tilde\lambda_i^k \tilde h_i^k$, we have

$$\begin{aligned} f(x^k) - \tilde f_k(x^k) &= \sum\nolimits_i \tilde\lambda_i^k [f(x^k) - \tilde h_i^k(x^k)] \\ &= \sum\nolimits_i \tilde\lambda_i^k \{[f(x^k) - h_i(x^k)] + [h_i(x^k) - \tilde h_i^k(x^k)]\}, \end{aligned} \tag{4.8}$$

where $\tilde\lambda_i^k \geq 0$ and $f \geq h_i \geq \tilde h_i^k$, $i \in I$, so that all the bracketed terms of (4.8) are nonnegative. Further, we have (cf. Theorem 2.1) $x^k \to x^\infty$, $f(x^k) \downarrow f(x^\infty)$, and (cf. Lemma 3.3(i)) $\tilde\epsilon_f^k := f(x^k) - \tilde f_k(x^k) \to 0$, in (4.8). Hence if $\tilde\lambda_i^\infty > 0$, then $h_i(x^k) \xrightarrow{K} h_i(x^\infty) = f(x^\infty)$ by continuity of h_i; i.e., $i \in I(x^\infty)$, and $h_i(x^k) - \tilde h_i^k(x^k) \xrightarrow{K} 0$, so $\tilde\epsilon_i^k := h_i(x^k) - \tilde h_i^k(x^k) \xrightarrow{K} 0$ with $\nabla \tilde h_i^k \in \partial_{\tilde\epsilon_i^k} h_i(x^k)$ (Lemma 4.2(iii)) yield $\nabla \tilde h_i^\infty \in \partial h_i(x^\infty)$ by the closedness of $\partial_\epsilon h_i(x)$. Otherwise, $\tilde\lambda_i^\infty = 0$. Therefore, we have $\tilde\lambda^\infty \in \Lambda(x^\infty)$ and

$$p_f^\infty := \sum\nolimits_i \tilde\lambda_i^\infty \nabla \tilde h_i^\infty \in \sum\nolimits_i \tilde\lambda_i^\infty \partial h_i(x^\infty).$$

Combining this with (4.7) yields $\tilde\lambda^\infty \in \Lambda^*(x^\infty)$ (cf. (4.2)).

(iii) Use (i), (ii), and the continuity of $d_{\Lambda^*(x^\infty)}$ (cf. the proof of Theorem 3.4(iv)). □

Remark 4.4. Under the assumptions of Lemma 4.1(iii), $\breve\Lambda^*(x^\infty)$ may replace $\Lambda^*(x^\infty)$ in Theorem 4.3; i.e., $\Lambda^*(x^\infty)$ is the set of optimal multipliers for both $\min_S f$ and $\min_{\breve S} f$. This is useful if constraints are appended to $\breve S$ in order to make S bounded.

Example 4.5. Let $S = \mathbb{R}^n_+$ and $f(\cdot) = \max_{i=1}^{|I|} \langle a^i, \cdot \rangle + b_i$, where a^i is column i of $A \in \mathbb{R}^{n \times |I|}$, $b \in \mathbb{R}^{|I|}$. Let $e := (1, \dots, 1)^T \in \mathbb{R}^{|I|}$. Then, by linear programming (LP) duality, for any $\bar x \in S_*$,

$$\Lambda^*(\bar x) = \operatorname{Arg\,max} \left\{ b^T\lambda : A\lambda \geq 0,\ e^T\lambda = 1,\ \lambda \geq 0 \right\}, \tag{4.9}$$

and if $A\breve\lambda > 0$, $e^T\breve\lambda = 1$ for some $\breve\lambda \geq 0$, then for any $x \in S$,

$$\bar x_i \leq [f(x) - b^T\breve\lambda]/(A\breve\lambda)_i, \quad i = 1 \colon n.$$

Such bounds may be used for choosing $S = \bar S = \{x : 0 \leq x \leq x^{\mathrm{up}}\}$ with $x^{\mathrm{up}} > \bar x$; also in this case every cluster point of $\{\tilde\lambda^k\}$ solves the dual problem in (4.9).

Remark 4.6. Following Remark 2.2(i), note that Theorem 4.3 holds if, for each $i \in I$, h_i is finite convex on S, and $g_{h_i}^k := g_{h_i}(y^k) \in \partial h_i(y^k)$ for all k with $g_{h_i}(\cdot)$ locally bounded on S.

4.2. Constraint multipliers.

In this subsection we assume that the feasible set S is represented as

$$S = \{x : c_i(x) \leq 0, i \in I, \langle a^i, x \rangle = b_i, i \in \bar I\}, \tag{4.10}$$

where $c_i : \mathbb{R}^n \to \mathbb{R}$ is convex, $i \in I := \{1 \colon \check m\}$, $(a^i, b_i) \in \mathbb{R}^{n+1}$, $i \in \bar I := \{\check m + 1 \colon \check m + \bar m\}$. We shall need the following additional assumption.

Assumption 4.7 (strong Slater constraint qualification). The vectors $\{a^i\}_{i \in \bar I}$ are linearly independent and there exists a point $\check x \in S$ such that $\max_{i \in I} c_i(\check x) < 0$.

For each $x \in S$, let

$$I(x) := \{i \in I : c_i(x) = 0\},$$

$$\Sigma(x) := \left\{\mu \in \mathbb{R}^{|I|}_{+} \times \mathbb{R}^{|\bar{I}|} : \mu_i = 0, i \in I \setminus I(x)\right\}, \tag{4.11}$$

so that

$$\mathcal{N}_S(x) = \left\{\sum_{i\in I} \mu_i \partial c_i(x) + \sum_{i\in \bar{I}} \mu_i a^i : \mu \in \Sigma(x)\right\}. \tag{4.12}$$

The (possibly empty) set

$$\Sigma^*(x) = \left\{\mu \in \Sigma(x) : \partial f(x) \cap -\left(\sum\nolimits_{i\in I} \mu_i \partial c_i(x) + \sum\nolimits_{i\in \bar{I}} \mu_i a^i\right) \neq \emptyset\right\} \tag{4.13}$$

of *optimal multipliers* associated with any $x \in S$ has the following properties.

FACT 4.8 (cf. [LPS98, Prop. 5.2]). *$x \in S_*$ iff $x \in S$ and $\Sigma^*(x) \neq \emptyset$. Further, for each $\bar{x} \in S_*$, $\Sigma^*(\bar{x}) = \Sigma^*(x)$ $\forall x \in S_*$, and $\Sigma^*(\bar{x})$ is compact and convex.*

In view of (4.12), we assume that each $p^k_S \in \mathcal{N}_S(y^{k+1})$ (cf. (2.8)) has the form

$$p^k_S = \sum_{i\in I} \mu^k_i g^{k+1}_{c_i} + \sum_{i\in \bar{I}} \mu^k_i a^i \quad \text{with} \quad \mu^k \in \Sigma(y^{k+1}),\ g^{k+1}_{c_i} \in \partial c_i(y^{k+1}),\ i \in I. \tag{4.14}$$

Thus μ^k is the multiplier of the current subproblem (2.4). We shall show that μ^k may serve as an estimate of optimal multipliers. At first sight, the current setting differs from the preceding one, which employed explicit aggregation for constructing the objective multiplier estimate $\tilde{\lambda}^k$ (cf. (4.5)) and related it to the aggregate objective linearization $\tilde{f}_k$. Yet there are many similarities, since μ^k may be related to the aggregate constraint linearization $\tilde{\imath}^k_S$ via the linearizations of the constraints c_i at y^{k+1} (cf. (4.14))

$$c^{k+1}_i(\cdot) := c_i(y^{k+1}) + \left\langle g^{k+1}_{c_i}, \cdot - y^{k+1}\right\rangle \leq c_i(\cdot), \quad i \in I. \tag{4.15}$$

Indeed, we have the following counterparts of Lemma 4.2(ii), (iii).

LEMMA 4.9.

(i) $\tilde{\imath}^k_S(\cdot) = \sum_{i\in I} \mu^k_i c^{k+1}_i(\cdot) + \sum_{i\in \bar{I}} \mu^k_i(\langle a^i, \cdot\rangle - b_i)$.

(ii) $g^{k+1}_{c_i} \in \partial_{\tilde{\epsilon}^k_i} c_i(x^k)$ *with* $\tilde{\epsilon}^k_i := c_i(x^k) - c^{k+1}_i(x^k) \geq 0$, $i \in I$.

Proof. (i) Since (cf. (2.4)) $y^{k+1} \in S$, (4.10)–(4.11) and (cf. (4.14)) $\mu^k \in \Sigma(y^{k+1})$ imply that $\mu^k_i c_i(y^{k+1}) = 0$, $i \in I$, $\langle a^i, y^{k+1}\rangle = b_i$, $i \in \bar{I}$. Hence (3.3) and (4.14)–(4.15) give

$$\begin{aligned}\tilde{\imath}^k_S(\cdot) := \left\langle p^k_S, \cdot - y^{k+1}\right\rangle &= \sum_{i\in I} \mu^k_i \left\langle g^{k+1}_{c_i}, \cdot - y^{k+1}\right\rangle + \sum_{i\in \bar{I}} \mu^k_i \left\langle a^i, \cdot - y^{k+1}\right\rangle \\ &= \sum_{i\in I} \mu^k_i \left[c_i(y^{k+1}) + \left\langle g^{k+1}_{c_i}, \cdot - y^{k+1}\right\rangle\right] + \sum_{i\in \bar{I}} \mu^k_i \left(\langle a^i, \cdot\rangle - b_i\right) \\ &= \sum_{i\in I} \mu^k_i c^{k+1}_i(\cdot) + \sum_{i\in \bar{I}} \mu^k_i \left(\langle a^i, \cdot\rangle - b_i\right).\end{aligned}$$

(ii) By (4.15), $c_i(\cdot) \geq c_i(x^k) - \tilde{\epsilon}^k_i + \left\langle g^{k+1}_{c_i}, \cdot - x^k\right\rangle$, $i \in I$. □

We now show that the subproblem multipliers μ^k of (4.14) converge to the set of optimal multipliers $\Sigma^*(x^\infty)$ (cf. (4.13)) as x^k approaches x^∞. Our proof is similar to that of Theorem 4.3. The main technical complication is that we must first use Assumption 4.7 to ensure boundedness of μ^k, whereas the boundedness of $\tilde{\lambda}^k$ was automatic.

Theorem 4.10.

(i) *$\{\mu^k\}$ is bounded.*

(ii) *Each cluster point of $\{\mu^k\}$ lies in $\Sigma^*(x^\infty)$.*

(iii) *$d_{\Sigma^*(x^\infty)}(\mu^k) \to 0$ as $k \to \infty$.*

Proof. (i) For each $i \in I$, by (4.14), $\{g_{c_i}^{k+1} \in \partial c_i(y^{k+1})\}$ is bounded (so is $\{y^k\}$ by Lemma 3.3(ii), and ∂c_i is locally bounded). Suppose $\{\mu^k\}$ is not bounded. Pick $K \subset \{1, 2, \dots\}$ s.t. $|\mu^k| \xrightarrow{K} \infty$, $\bar{\mu}^k := \mu^k/|\mu^k| \xrightarrow{K} \bar{\mu}^\infty$, $g_{c_i}^{k+1} \xrightarrow{K} g_{c_i}^\infty$, $i \in I$ (using the boundedness of $\{\bar{\mu}^k\}$ and $\{g_{c_i}^{k+1}\}$). Clearly, $0 \neq \bar{\mu}^\infty \in \mathbb{R}_+^{|I|} \times \mathbb{R}^{|\bar{I}|}$, since $|\bar{\mu}^k| = 1$ and $\mu^k \in \mathbb{R}_+^{|I|} \times \mathbb{R}^{|\bar{I}|}$ by (4.14), (4.11). Further, dividing the equality in (4.14) by $|\mu^k|$, we get

$$p_S^k/|\mu^k| = \sum_{i \in I} \bar{\mu}_i^k g_{c_i}^{k+1} + \sum_{i \in \bar{I}} \bar{\mu}_i^k a^i \xrightarrow{K} \sum_{i \in I} \bar{\mu}_i^\infty g_{c_i}^\infty + \sum_{i \in \bar{I}} \bar{\mu}_i^\infty a^i =: \bar{g}^\infty; \tag{4.16}$$

in fact $\bar{g}^\infty = 0$, since $|\mu^k| \xrightarrow{K} \infty$ and $\{p_S^k\}$ is bounded (Theorem 3.4(iii)). Next, using $x^k \in S$ with (cf. (4.10)) $\langle a^i, x^k \rangle = b_i$, $i \in \bar{I}$, in Lemma 4.9(i) gives

$$-\tilde{\imath}_S^k(x^k) = \sum_{i \in I} \mu_i^k \left[-c_i^{k+1}(x^k)\right] = \sum_{i \in I} \mu_i^k \left\{\left[-c_i(x^k)\right] + \left[c_i(x^k) - c_i^{k+1}(x^k)\right]\right\}, \tag{4.17}$$

where $\mu_i^k \geq 0$ and (cf. (4.15)) $c_i^{k+1}(x^k) \leq c_i(x^k) \leq 0$, $i \in I$, so that all the bracketed terms in (4.17) are nonnegative. Further, we have $-\tilde{\imath}_S^k(x^k) =: \tilde{\epsilon}_S^k \to 0$ by Lemma 3.3(i) and $x^k \to x^\infty$ by Theorem 2.1. Hence if $\bar{\mu}_i^\infty > 0$, then (since $\mu_i^k \xrightarrow{K} \infty$) (4.17) yields $c_i(x^k) \xrightarrow{K} c_i(x^\infty) = 0$ by continuity of c_i, and $c_i(x^k) - c_i^{k+1}(x^k) \xrightarrow{K} 0$, so $\tilde{\epsilon}_i^k := c_i(x^k) - c_i^{k+1}(x^k) \xrightarrow{K} 0$ with $g_{c_i}^{k+1} \in \partial_{\tilde{\epsilon}_i^k} c_i(x^k)$ (Lemma 4.9(ii)) give $g_{c_i}^\infty \in \partial c_i(x^\infty)$ by the closedness of $\partial_\epsilon c_i(x)$; otherwise, $\bar{\mu}_i^\infty = 0$. Therefore, using $\bar{g}^\infty = 0$ in (4.16), $\langle a^i, x^\infty \rangle = b_i = \langle a^i, \breve{x} \rangle$ for $i \in \bar{I}$ ($x^\infty, \breve{x} \in S$), the subgradient inequality, and $\max_{i \in I} c_i(\breve{x}) < 0$ (Assumption 4.7), we get

$$0 = \langle \bar{g}^\infty, \breve{x} - x^\infty \rangle = \sum_{i \in I} \bar{\mu}_i^\infty \left[c_i(x^\infty) + \langle g_{c_i}^\infty, \breve{x} - x^\infty \rangle\right] \leq \sum_{i \in I} \bar{\mu}_i^\infty c_i(\breve{x}) \leq 0$$

and $\bar{\mu}_i^\infty = 0$, $i \in I$. Thus (cf. (4.16)) $0 = \bar{g}^\infty = \sum_{i \in \bar{I}} \bar{\mu}_i^\infty a^i$, so by Assumption 4.7, we also have $\bar{\mu}_i^\infty = 0$, $i \in \bar{I}$, contradicting $\bar{\mu}^\infty \neq 0$. Hence $\{\mu^k\}$ must be bounded.

(ii) Let μ^∞ be a cluster point of $\{\mu^k\}$. Then, by the boundedness of $\{g_{c_i}^{k+1}\}$ (cf. the proof of (i)), there are $K \subset \{1, 2, \dots\}$ and $g_{c_i}^\infty$ such that $\mu^k \xrightarrow{K} \mu^\infty$, $g_{c_i}^{k+1} \xrightarrow{K} g_{c_i}^\infty$, $i \in I$. By (4.14), we have $\mu^\infty \in \mathbb{R}_+^{|I|} \times \mathbb{R}^{|\bar{I}|}$ (since $\mu^k \in \mathbb{R}_+^{|I|} \times \mathbb{R}^{|\bar{I}|}$ by (4.11)) and

$$p_S^k = \sum_{i \in I} \mu_i^k g_{c_i}^{k+1} + \sum_{i \in \bar{I}} \mu_i^k a^i \xrightarrow{K} \sum_{i \in I} \mu_i^\infty g_{c_i}^\infty + \sum_{i \in \bar{I}} \mu_i^\infty a^i =: p_S^\infty.$$

Then $p_S^k \xrightarrow{K} p_S^\infty$ and (cf. Lemma 3.3(i)) $p_f^k + p_S^k =: p^k \to 0$ give $p_f^k \xrightarrow{K} -p_S^\infty \in \partial f(x^\infty)$ by Theorem 3.4(i). Next, using the argument of (i) yields $g_{c_i}^\infty \in \partial c_i(x^\infty)$ and $c_i(x^\infty) = 0$ if $\mu_i^\infty > 0$, $i \in I$. Therefore, combining the preceding relations, we have

$$-\partial f(x^\infty) \ni p_S^\infty := \sum_{i \in I} \mu_i^\infty g_{c_i}^\infty + \sum_{i \in \bar{I}} \mu_i^\infty a^i \in \sum_{i \in I} \mu_i^\infty \partial c_i(x^\infty) + \sum_{i \in \bar{I}} \mu_i^\infty a^i$$

and (cf. (4.11)) $\mu^\infty \in \Sigma(x^\infty)$, i.e., $\mu^\infty \in \Sigma^*(x^\infty)$ (cf. (4.13)).

(iii) This follows from (i), (ii), and the continuity of $d_{\Sigma^*(x^\infty)}$. □

Remark 4.11.

(i) Following Remark 2.2(i), note that Theorem 4.10 holds if, for each $i \in I$, c_i is finite convex on S, and $g_{c_i}^k = g_{c_i}(y^k) \in \partial c_i(y^k)$ $\forall k$ with $g_{c_i}(\cdot)$ locally bounded on S.

(ii) The linear independence part of Assumption 4.7 is not really necessary: it suffices to assume that $\{a^i\}_{i\in\bar{I}^k}$ is linearly independent for all k, where $\bar{I}^k = \{i \in \bar{I} : \mu_i^k \neq 0\}$. Then in the proof of Theorem 4.10(i) we may pick K s.t. $\bar{I}^k$ is constant $\forall k \in K$. Note that $\{a^i\}_{i\in\bar{I}^k}$ is linearly independent if (2.11) is solved by an active-set method [Kiw89]. Similarly, we may *remove* Assumption 4.7 if, for each k and $i \in I$, c_i is polyhedral, $g_{c_i}^{k+1} = g_{c_i}(y^{k+1})$, where $g_{c_i}(\cdot) \in \partial c_i(\cdot)$ is finite-valued, and $I^k = \{i \in I : \mu_i^k > 0\}$ is s.t. $\sum_{i\in I^k} \mu_i g_{c_i}^{k+1} + \sum_{i\in\bar{I}^k} \mu_i a^i = 0$, $\mu_i \geq 0$, $i \in I^k$, implies $\mu_i = 0$, $i \in I^k \cup \bar{I}^k$.

5. Lagrangian decomposition. In this section we consider the special case where problem (1.1) (i.e., $\min_S f$) is the Lagrangian dual problem of the following *primal* convex optimization problem:

$$\psi_0^{\max} := \max\ \psi_0(z) \quad \text{s.t.} \quad \psi_j(z) \geq 0,\ j = 1\colon n,\ z \in Z, \tag{5.1}$$

where $\emptyset \neq Z \subset \mathbb{R}^{\bar{m}}$ is compact and convex, and each ψ_j is closed (upper semicontinuous) and concave with $\operatorname{dom}\psi_j \supset Z$. The Lagrangian of (5.1) has the form $\psi_0(z) + \langle x, \psi(z)\rangle$, where $\psi := (\psi_1, \ldots, \psi_n)$ and x is a multiplier. Suppose that, at each multiplier x in the *dual feasible* set $\breve{S} := \mathbb{R}_+^n$, the *dual function*

$$f(x) := \max\{\psi_0(z) + \langle x, \psi(z)\rangle : z \in Z\} \tag{5.2}$$

can be evaluated by finding a partial Lagrangian solution

$$z(x) \in Z(x) := \operatorname{Arg\,max}\{\psi_0(z) + \langle x, \psi(z)\rangle : z \in Z\}. \tag{5.3}$$

Thus f is finite convex and has a subgradient mapping $g_f(\cdot) := \psi(z(\cdot))$ on $\breve{S}$. In view of Remark 2.2(i), we suppose that $\psi(z(\cdot))$ is locally bounded on $\breve{S}$ (e.g., f is the restriction to $\breve{S}$ of a convex function finite on an open neighborhood of $\breve{S}$, or $\inf_Z \min_{j=1}^n \psi_j > -\infty$, or ψ is continuous on Z). Assuming nonemptiness of the dual optimal set $\breve{S}_* := \operatorname{Arg\,min}_{\breve{S}} f$ (e.g., Slater's condition $\psi(\check{z}) > 0$ for some $\check{z} \in Z$), we consider the following two choices:

$$S := \breve{S} := \mathbb{R}_+^n \quad \text{or} \quad S := \{x : 0 \leq x \leq x^{\text{up}}\} \text{ with } x^{\text{up}} > \bar{x} \text{ for some } \bar{x} \in \breve{S}_*. \tag{5.4}$$

For the second choice, $\min_S f$ is a restricted version of the classical dual problem $\min_{\breve{S}} f$ in the sense of Lemma 3.7.

We shall use the partial Lagrangian solutions and their constraint values

$$z^k := z(y^k) \quad \text{and} \quad g_f^k := \psi(z^k) \tag{5.5}$$

for generating and analyzing the following estimates of solutions to (5.1). Using the weights $\{\nu_j^k\}_{j\in J^k}$ (cf. (2.9)), we define the kth *aggregate primal solution* by

$$\tilde{z}^k := \sum_{j\in J^k} \nu_j^k z^j. \tag{5.6}$$

This construction is related to the aggregate linearization $\tilde{f}_k = \sum_{j\in J^k} \nu_j^k f_j$ (Lemma 3.2(ii)). By expressing each f_j in terms of $\psi_0(z^j)$ and $\psi(z^j)$, below we derive bounds on $\psi_0(\tilde{z}^k)$ and $\psi(\tilde{z}^k)$ that will be used in our subsequent asymptotic analysis.

Lemma 5.1.

(i) $f_k(\cdot) = \psi_0(z^k) + \langle \cdot, \psi(z^k) \rangle$.

(ii) $\tilde{z}^k \in Z$, $\psi_0(\tilde{z}^k) \geq f(x^k) - \tilde{\epsilon}_k - \langle p^k, x^k \rangle$, $\psi(\tilde{z}^k) \geq p_f^k$, *where* $p_f^k \geq p^k$ *if* $S = \mathbb{R}^n_+$.

Proof. (i) Use (cf. (2.2)) $f_k(\cdot) = f(y^k) + \langle g_f^k, \cdot - y^k \rangle$, (5.2), (5.3), and (5.5).

(ii) We have (cf. (2.9)) $\sum_{j \in J^k} \nu_j^k = 1$ with $\nu_j^k \geq 0$. Hence $\tilde{z}^k \in \operatorname{co}\{z^j\}_{j \in J^k} \subset Z$, $\psi_0(\tilde{z}^k) \geq \sum_j \nu_j^k \psi_0(z^j)$, $\psi(\tilde{z}^k) \geq \sum_j \nu_j^k \psi(z^j)$ by convexity of Z and concavity of ψ_0, ψ. Next, using Lemma 3.2(ii), (i) with $\psi(z^j) =: g_f^j$ (cf. (5.5)) and (2.9), we get

$$\tilde{f}_k(\cdot) = \sum\nolimits_j \nu_j^k f_j(\cdot) = \sum\nolimits_j \nu_j^k \left[\psi_0(z^j) + \langle \cdot, \psi(z^j) \rangle \right] = \sum\nolimits_j \nu_j^k \psi_0(z^j) + \langle p_f^k, \cdot \rangle$$

with $p_f^k = \sum_j \nu_j^k \psi(z^j)$. The above equality, $\tilde{f}_S^k := \tilde{f}_k + \tilde{\imath}_S^k$ (cf. (3.3)), $\tilde{\imath}_S^k(0) \leq \imath_S(0) = 0$ (cf. Lemma 3.2(i) and (5.4)), and (3.6) imply

$$\sum\nolimits_j \nu_j^k \psi_0(z^j) = \tilde{f}_k(0) = \tilde{f}_S^k(0) - \tilde{\imath}_S^k(0) \geq \tilde{f}_S^k(0) = f(x^k) - \tilde{\epsilon}_k - \langle p^k, x^k \rangle .$$

Finally, if $S = \mathbb{R}^n_+$ then (cf. (3.4)) $p_S^k \in \mathcal{N}_S(y^{k+1})$ gives $p_S^k \leq 0$, and hence (cf. (3.4)) $p_f^k = p^k - p_S^k \geq p^k$. Combining the preceding relations gives the conclusion. □

Let Z_* denote the solution set of the primal problem (5.1). We now show that the aggregate primal solution $\tilde{z}^k$ converges to Z_* as x^k approaches the dual solution x^∞. Our proof is deceptively simple thanks to the "heavy" machinery of Theorem 3.4 and the subtle content of Lemma 3.7. Note that nonemptiness of Z_* and several standard duality relations are demonstrated in a constructive way.

Theorem 5.2.

(i) $\{\tilde{z}^k\}$ *is bounded and all its cluster points lie in* Z.

(ii) $f(x^k) \downarrow f(x^\infty)$, $\tilde{\epsilon}_k + \langle p^k, x^k \rangle \to 0$ *as* $k \to \infty$, *and* $\liminf_k \min_{i=1}^n (p_f^k)_i \geq 0$.

(iii) *Let* $\tilde{z}^\infty$ *be a cluster point of* $\{\tilde{z}^k\}$. *Then* $\tilde{z}^\infty \in Z_*$. *Further,* $\psi_0^{\max} = f(x^\infty)$ *and* $\tilde{z}^\infty \in Z(x^\infty)$ (*cf.* (5.3)).

(iv) $d_{Z_*}(\tilde{z}^k) \to 0$, *and* $f(x^k) \downarrow \psi_0^{\max}$ *as* $k \to \infty$.

Proof. (i) By Lemma 5.1(ii), $\{\tilde{z}^k\}$ lies in Z, which is compact by our assumption.

(ii) By Theorem 2.1 and Lemma 3.3(i), $f(x^k) \downarrow f(x^\infty)$, $\tilde{\epsilon}_k + \langle p^k, x^k \rangle \to 0$. By Theorem 3.4(i), (ii), (5.4), and Remark 3.8, $\{p_f^k\}$ is bounded and its cluster points lie in $\mathcal{G} \subset -\mathcal{N}_{\breve{S}}(x^\infty) \subset \mathbb{R}^n_+$.

(iii) By (i), $\tilde{z}^\infty \in Z$. Using (ii) in Lemma 5.1(ii) gives $\psi_0(\tilde{z}^\infty) \geq f(x^\infty)$, $\psi(\tilde{z}^\infty) \geq 0$ by closedness of ψ_0, ψ. Since $\psi_0(\tilde{z}^\infty) \leq \psi_0^{\max} \leq f(x^\infty)$ by weak duality, $\tilde{z}^\infty$ must solve (5.1) and $\psi_0(\tilde{z}^\infty) = \psi_0^{\max} = f(x^\infty)$. Further, $\psi(\tilde{z}^\infty) \geq 0$ and $x^\infty \geq 0$ yield $\psi_0(\tilde{z}^\infty) + \langle x^\infty, \psi(\tilde{z}^\infty) \rangle \geq f(x^\infty)$, so $\tilde{z}^\infty \in Z(x^\infty)$ by (5.2)–(5.3), using $\tilde{z}^\infty \in Z$.

(iv) This follows from (i), (iii), and the continuity of d_{Z_*}. □

Remark 5.3.

(i) Given $\epsilon > 0$, the method may stop if

$$\psi_0(\tilde{z}^k) \geq f(x^k) - \epsilon \quad \text{and} \quad \psi_i(\tilde{z}^k) \geq -\epsilon, \quad i = 1\colon n.$$

Then $\psi_0(\tilde{z}^k) \geq \psi_0^{\max} - \epsilon$ from $f(x^k) \geq \psi_0^{\max}$ (weak duality), so $\tilde{z}^k \in Z$ is an ϵ-solution of (5.1). This stopping criterion will be satisfied for some k (cf. Lemma 5.1(ii) and Theorem 5.2(ii)).

(ii) If $\psi(\check{z}) > 0$ for some $\check{z} \in Z$, then for any $\bar{x} \in \breve{S}_* := \operatorname{Arg\,min}_{\mathbb{R}^n_+} f$ and $x \geq 0$,

$$\bar{x}_i \leq [f(x) - \psi_0(\check{z})]/\psi_i(\check{z}), \quad i = 1\colon n,$$

(since $\psi_0(\check{z}) + \langle \bar{x}, \psi(\check{z})\rangle \le f(\bar{x}) \le f(x)$ by (5.2)). Such bounds may be used for choosing $x^{\mathrm{up}} > \bar{x}$ in (5.4).

(iii) For Example 4.5, we may identify $\psi_0(z) = b^T z$, $\psi(z) = Az$, $Z = \{z \in \mathbb{R}^{|I|}_+ : e^T z = 1\}$, $Z(x) = \Lambda(x)$, $Z_* = \Lambda^*(\bar{x})$ in (4.9), $z^k = \lambda^k$, and hence $\tilde{z}^k = \tilde{\lambda}^k$.

Remark 5.4. Consider the equality constrained version of (5.1):

$$\psi_0^{\max} := \max\ \psi_0(z) \quad \text{s.t.} \quad \psi(z) := Az = 0,\ z \in Z, \tag{5.7}$$

where $A \in \mathbb{R}^{n\times\bar{m}}$. Then $\check{S} := \mathbb{R}^n$ and (cf. (5.4)) $S := \check{S}$ or $S := \{x : x^{\mathrm{low}} \le x \le x^{\mathrm{up}}\}$ with $x^{\mathrm{low}} < \bar{x} < x^{\mathrm{up}}$ for some $\bar{x} \in \check{S}_*$. Clearly, Lemma 5.1 holds with $\psi(\tilde{z}^k) = p_f^k$ (where $p_f^k = p^k$ if $S = \mathbb{R}^n$), and Theorem 5.2 holds with (5.1) replaced by (5.7), $p_f^k \to 0$ in (ii), and $\psi(\tilde{z}^\infty) = 0$ in (iii) (use $\mathcal{N}_{\check{S}}(x^\infty) = \{0\}$ in the proof of (ii) and $\psi(\tilde{z}^k) = p_f^k \to 0$ for (iii)). Next, we may use the stopping criterion of Remark 5.3 augmented by $\psi(\tilde{z}^k) \le \epsilon e$. For the alternative stopping criterion $\tilde{\epsilon}_k \le \epsilon$, $|p_f^k| \le \delta$, [Rob89] gives conditions on (5.7) under which, for each $\eta > 0$, there exist $\epsilon, \delta > 0$ s.t. upon termination $\max\{d_{\check{S}_*}(x^k), d_{Z_*}(\tilde{z}^k)\} \le \eta$. In fact [Rob86, Rob89] replace compactness of Z by other conditions on ψ_0 and A for the ϵ-steepest descent bundle method [LSB81], but the analysis of [Rob89] carries over to our setting; it also carries over when ψ_0 and Z have separable forms (cf. (6.2)).

6. Lagrangian relaxation of nonconvex problems. In this section we no longer assume that the primal problem (5.1) is convex, but we retain the remaining assumptions of section 5; in particular, $\{\psi_j\}_{j=0}^n$ are finite and closed (upper semicontinuous) on the compact set Z.

Since problem (5.1) may be nonconvex, consider its *relaxed convexified version*

$$\begin{aligned} &\psi_0^{\mathrm{rel}} := \max_{(\nu_j, z^j)_{j=1}^M} \sum_{j=1}^M \nu_j \psi_0(z^j) \\ &\qquad\qquad \text{s.t.} \quad \sum_{j=1}^M \nu_j \psi(z^j) \ge 0,\ \sum_{j=1}^M \nu_j = 1,\ z^j \in Z,\ \nu_j \ge 0, \end{aligned} \tag{6.1}$$

where $M := n + 1$. An interpretation of (6.1) stemming from [BLSP83] is that we choose decisions z^j and their probabilities ν_j that solve the problem $\max\{\mathrm{E}\,\psi_0(z) : \mathrm{E}\,\psi(z) \ge 0\}$, where E denotes expected value; in other words, the feasible set is expanded to include all *randomized decisions.* (In contrast to separable programming, the points z^j are *not* fixed.) Alternative interpretations are given in [Las70, section 3.4], [Sha79, section 5.3], and [MSW76, LeR96]. Both (5.1) and (6.1) have the same dual function (5.2), and hence the same dual problem and optimal Lagrange multipliers (if any). Further, $\psi_0^{\max} \le \psi_0^{\mathrm{rel}} = \inf_S f$ [MSW76].

We shall need the following specialization of Lemma 5.1, using $\hat{J}^k$ (cf. (2.10)).

LEMMA 6.1.

(i) $\sum_{j\in\hat{J}^k} \nu_j^k \psi_0(z^j) \ge f(x^k) - \tilde{\epsilon}_k - \langle p^k, x^k\rangle$, $\sum_{j\in\hat{J}^k} \nu_j^k \psi(z^j) = p_f^k$.

(ii) *If* $\nu_j^k > 0$, *then* $\psi_0(z^j) + \langle x^k, \psi(z^j)\rangle = f(x^k) + v_k - \langle g_f^j, y^{k+1} - x^k\rangle$.

Proof. (i) This follows from the proof of Lemma 5.1(ii).

(ii) By Lemma 5.1(i), (2.9) with (cf. (5.5)) $g_f^j := \psi(z^j)$, and (2.5), we have

$$\psi_0(z^j) + \langle y^{k+1}, \psi(z^j)\rangle = f_j(y^{k+1}) = \check{f}_k(y^{k+1}) = f(x^k) + v_k.$$

Subtract $\langle g_f^j, y^{k+1} - x^k\rangle$ to get the conclusion. □

Since (cf. (5.4)) S is polyhedral, we assume that $|\hat{J}^k| \le M$, where $\hat{J}^k := \{j : \nu_j^k > 0\}$ and $M := n+1$ (cf. Remark 2.2(ii)). Let $\tilde{Z}_*$ denote the solution set of the relaxed primal problem (6.1). It turns out that $(\nu_j^k, z^j)_{j\in\hat{J}^k}$ is a "natural" estimate of a relaxed solution, except that we may have $|\hat{J}^k| < M$, whereas points in $\tilde{Z}_*$ have the form $(\nu_j, z^j)_{j=1}^M$. Fortunately, this difficulty is just notational, since we may always arrange for $\hat{J}^k$ to have precisely M elements by splitting any (ν_j^k, z^j) into several elements with suitably adjusted weights. Specifically, let us relabel $(\nu_j^k, z^j)_{j\in\hat{J}^k}$ as follows: Denote $(\nu_j^k, z^j)_{j\in\hat{J}^k}$ as $(\hat{\nu}_j^k, \hat{z}^{jk})_{j=1}^{j_k}$, where $j_k = |\hat{J}^k|$; if $j_k < M$, divide $\hat{\nu}_{j_k}^k$ by $(M - j_k + 1)$ and set $(\hat{\nu}_j^k, \hat{z}^{jk}) = (\hat{\nu}_{j_k}^k, \hat{z}^{j_k k})$, $j = j_k + 1\colon M$. We now show that $(\hat{\nu}_j^k, \hat{z}^{jk})_{j=1}^M$ converges to $\tilde{Z}_*$. (Without this relabeling, the corresponding result for $(\nu_j^k, z^j)_{j\in\hat{J}^k}$ would be more cumbersome to state and prove.)

THEOREM 6.2.

(i) $\{(\hat{\nu}_j^k, \hat{z}^{jk})_{j=1}^M\}$ *lies in a compact set.*

(ii) $f(x^k) \downarrow f(x^\infty)$, $\tilde{\epsilon}_k + \langle p^k, x^k\rangle \to 0$ *as* $k \to \infty$, *and* $\liminf_k \min_{i=1}^n (p_f^k)_i \ge 0$.

(iii) *Let* $(\hat{\nu}_j, \hat{z}^j)_{j=1}^M$ *be a cluster point of* $\{(\hat{\nu}_j^k, \hat{z}^{jk})_{j=1}^M\}$. *Then* $(\hat{\nu}_j, \hat{z}^j)_{j=1}^M \in \tilde{Z}_*$. *Further,* $f(x^\infty) = \psi_0^{\mathrm{rel}}$ *and* $\hat{z}^j \in Z(x^\infty)$, $j = 1\colon M$.

(iv) $d_{\tilde{Z}_*}((\hat{\nu}_j^k, \hat{z}^{jk})_{j=1}^M) \to 0$, *and* $f(x^k) \downarrow \psi_0^{\mathrm{rel}}$ *as* $k \to \infty$.

Proof. (i) By construction (cf. (2.9)), $\sum_j \hat{\nu}_j^k = 1$, $\hat{\nu}_j^k > 0$, $\hat{z}^{jk} \in Z$, a compact set.

(ii) The proof of Theorem 5.2(ii) remains valid.

(iii) By (i), $\sum_j \hat{\nu}_j = 1$, $\hat{\nu}_j \ge 0$, $\hat{z}^j \in Z$, $j = 1\colon M$. Next, by construction

$$\sum_{j=1}^M \hat{\nu}_j^k \psi_0(\hat{z}^{jk}) = \sum_{j\in\hat{J}^k} \nu_j^k \psi_0(z^j) \quad \text{and} \quad \sum_{j=1}^M \hat{\nu}_j^k \psi(\hat{z}^{jk}) = \sum_{j\in\hat{J}^k} \nu_j^k \psi(z^j),$$

so using (ii) and the upper semicontinuity of ψ_0, ψ in Lemma 6.1(i) gives

$$\sum_{j=1}^M \hat{\nu}_j \psi_0(\hat{z}^j) \ge f(x^\infty) \quad \text{and} \quad \sum_{j=1}^M \hat{\nu}_j \psi(\hat{z}^j) \ge 0.$$

In particular, $(\hat{\nu}_j, \hat{z}^j)_{j=1}^M$ is feasible in (6.1). Since also $\sum_j \hat{\nu}_j \psi_0(\hat{z}^j) \le \psi_0^{\mathrm{rel}} \le f(x^\infty)$ by weak duality, $(\hat{\nu}_j, \hat{z}^j)_{j=1}^M$ solves (6.1) and $f(x^\infty) = \psi_0^{\mathrm{rel}}$. Finally, fix $i \in \{1\colon M\}$. If $(\hat{\nu}_i^k, \hat{z}^{ik})$ corresponds to (ν_j^k, z^j) in Lemma 6.1(ii), then (cf. Theorem 2.1 and Lemma 3.3(ii))

$$\psi_0(\hat{z}^{ik}) + \left\langle x^k, \psi(\hat{z}^{ik})\right\rangle = f(x^k) + v^k - \left\langle g_f^j, y^{k+1} - x^k \right\rangle \to f(x^\infty)$$

yields $\psi_0(\hat{z}^i) + \langle x^\infty, \psi(\hat{z}^i)\rangle \ge f(x^\infty)$, i.e., $\hat{z}^i \in Z(x^\infty)$ by (5.2)–(5.3).

(iv) This follows from (i), (iii), and the continuity of $d_{\tilde{Z}_*}$. $\square$

Remark 6.3.

(i) Given $\epsilon > 0$, the method may stop if

$$\sum_{j\in\hat{J}^k} \nu_j^k \psi_0(z^j) \ge f(x^k) - \epsilon \quad \text{and} \quad \sum_{j\in\hat{J}^k} \nu_j^k \psi_i(z^j) \ge -\epsilon, \quad i = 1\colon n.$$

Then $\sum_{j\in\hat{J}^k} \nu_j^k \psi_0(z^j) \ge \psi_0^{\mathrm{rel}} - \epsilon$ from $f(x^k) \ge \psi_0^{\mathrm{rel}}$ (weak duality), so $(\nu_j^k, z^j)_{j\in\hat{J}^k}$ is an ϵ-solution of the relaxed primal problem (6.1). This stopping criterion will be satisfied for some k (cf. Lemma 6.1(i) and Theorem 6.2(ii)).

(ii) If $\sum_j \check\nu_j \psi(\check z^j) > 0$ for some $\check\nu_j \ge 0$, $\check z^j \in Z$, $\sum_j \check\nu_j = 1$, then for any $x \ge 0$,

$$\bar x_i \le \left[f(x) - \sum\nolimits_j \check\nu_j \psi_0(\check z^j)\right] \Big/ \sum\nolimits_j \check\nu_j \psi_i(\check z^j) \quad \text{for } i = 1\colon n \text{ and each } \bar x \in \check S_*$$

(since $\sum_j \check\nu_j[\psi_0(\check z^j) + \langle \bar x, \psi(\check z^j)\rangle] \le f(\bar x) \le f(x)$). Such bounds may be used for choosing $x^{\mathrm{up}} > \bar x$ in (5.4).

(iii) The method will find a solution in finite time if f is polyhedral (e.g., Z is finite) and either $\kappa = 1$ or certain technical conditions are satisfied [Kiw91].

(iv) Extensions to cases where approximate maximizers of (5.2) are used for estimating $f(x)$ or where Z is not compact are easily developed as in [Kiw95].

If, as frequently happens in applications, (5.1) has the *separable* form

$$\text{(6.2a)} \quad \max \quad \psi_0(z) := \sum_{i=1}^m \psi_{0i}(z_i) \quad \text{s.t.} \quad \psi_j(z) := \sum_{i=1}^m \psi_{ji}(z_i) \ge 0,\ j = 1\colon n,$$

$$\text{(6.2b)} \qquad z := (z_1, \dots, z_m) \in Z := Z_1 \times \cdots \times Z_m,$$

with each $Z_i \subset \mathbb{R}^{m_i}$ compact and $\psi_{ji} : Z_i \to \mathbb{R}$ upper semicontinuous, then the preceding results may be specialized as follows. Letting

$$\text{(6.3)} \qquad f_i(x) := \max\left\{\psi_{0i}(z_i) + \langle x, \psi_{\cdot i}(z_i)\rangle : z_i \in Z_i\right\},$$

$$\text{(6.4)} \qquad z_i(x) \in Z_i(x) := \operatorname{Arg\,max}\left\{\psi_{0i}(z_i) + \langle x, \psi_{\cdot i}(z_i)\rangle : z_i \in Z_i\right\},$$

where $\psi_{\cdot i} := (\psi_{1i}, \dots, \psi_{ni})$, we have $f = \sum_{i=1}^m f_i$ in (5.2), and $z(\cdot) = (z_1(\cdot), \dots, z_m(\cdot))$ in (5.3). The relaxed problem (6.1) becomes

$$\max \sum_{i=1}^m \sum_{j=1}^M \nu_{ij} \psi_{0i}(z_i^j) \quad \text{s.t.} \quad \sum_{i=1}^m \sum_{j=1}^M \nu_{ij} \psi_{\cdot i}(z_i^j) \ge 0,\ \sum_{j=1}^M \nu_{ij} = 1,\ z_i^j \in Z_i,\ \nu_{ij} \ge 0. \tag{6.5}$$

To exploit the additive structure of $f = \sum_{i=1}^m f_i$, we may use the models (cf. (2.3))

$$\check f_k := \sum_{i=1}^m \check f_i^k \quad \text{with} \quad \check f_i^k := \max_{j \in J_i^k} f_i^j$$

constructed from the linearizations $f_i^j(\cdot) := \psi_{0i}(z_i^j) + \langle \cdot, \psi_{\cdot i}(z_i^j)\rangle$ of f_i, where $z^j := z(y^j)$. The sets J_i^k are selected [Kiw90, Kiw95] by finding Lagrange multipliers $\nu_{ij}^k \ge 0$, $j \in J_i^k$, $i = 1\colon m$, of the corresponding extension of (2.11)

$$\text{(6.6a)} \qquad \min \quad \tfrac{1}{2} u^k |x - x^k|^2 + \sum_{i=1}^m \xi_i \qquad \text{over all } (x, \xi) \in S \times \mathbb{R}^m$$

$$\text{(6.6b)} \qquad \text{s.t.} \quad \psi_{0i}(z_i^j) + \left\langle x, \psi_{\cdot i}(z_i^j)\right\rangle \le \xi_i \quad \forall j \in J_i^k,\ i = 1\colon m,$$

such that

$$\text{(6.7a)} \qquad \sum_{i=1}^m |\hat J_i^k| \le n + m, \qquad \sum_{j \in \hat J_i^k} \nu_{ij}^k = 1, \quad i = 1\colon m,$$

where

$$\text{(6.7b)} \qquad \hat J_i^k := \{j \in J_i^k : \nu_{ij}^k > 0\}, \quad i = 1\colon m.$$

Choosing $J_i^{k+1} \supset \hat{J}_i^k \cup \{k+1\}$ suffices for Theorem 2.1 (see [Kiw90, Kiw95] for details).

Then $\sum_i \sum_j \nu_{ij}^k \psi_{0i}(z_i^j)$ and $\sum_i \sum_j \nu_{ij}^k \psi_{\cdot i}(z_i^j)$ replace the left sides of the estimates of Lemma 6.1(i). The relabeling that preceded Theorem 6.2 may be extended as follows. By (6.7), $j_i^k := |\hat{J}_i^k| \le n+1$, $i = 1\colon m$. Denote $(\nu_{ij}^k, z_i^j)_{j \in \hat{J}_i^k}$ as $(\hat{\nu}_{ij}^k, \hat{z}_i^{jk})_{j=1}^{j_i^k}$; if $j_i^k < M$, divide $\hat{\nu}_{ij_i^k}^k$ by $(M - j_i^k + 1)$ and set $(\hat{\nu}_{ij}^k, \hat{z}_i^{jk}) = (\hat{\nu}_{ij_i^k}^k, \hat{z}_i^{j_i^k k})$, $j = j_i^k + 1\colon M$, $i = 1\colon m$. The following extension of Theorem 6.2 has an analogous proof, which is omitted.

THEOREM 6.4. *The sequence* $\{(\hat{\nu}_{ij}^k, \hat{z}_i^{jk})_{i=1\colon m}^{j=1\colon M}\}$ *is bounded and converges to the solution set of* (6.5). *Further, each of its cluster points* $\{(\hat{\nu}_{ij}, \hat{z}_i^j)_{i=1\colon m}^{j=1\colon M}\}$ *solves* (6.5), *and* $\hat{z}_i^j \in Z_i(x^\infty)$, $i = 1\colon m$, $j = 1\colon M$. *Finally, the optimal value of* (6.5) *equals* $f(x^\infty)$.

Remark 6.5.

(i) Remark 6.3 extends naturally to the separable case.

(ii) The QP subproblem (6.6) has at most $n + 2m$ constraints if, for each k, one chooses $J_i^{k+1} = \hat{J}_i^k \cup \{k+1\}$, $i = 1\colon m$ (cf. (6.7a)).

(iii) As in [Ber82, p. 370], we note that (6.7) yields

$$|\{i : |\hat{J}_i^k| > 1\}| \le n \quad \text{and} \quad |\{i : |\hat{J}_i^k| = 1\}| \ge m - n.$$

In particular, if $m > n$ and $Z_i \subset \{0,1\}^{m_i}$ $\forall i$, then $\sum_{j \in \hat{J}_i^k} \nu_{ij}^k z_i^j \notin Z_i$ for at most n indices i. This suggests that for $m \gg n$ it should be possible to devise heuristic rules for modifying the relaxed solution of (6.5) to obtain a feasible solution of (6.2) with value relatively close to the optimal value of (6.5). Some supporting evidence will be given in the next section.

7. Application to the unit commitment problem.

7.1. Unit commitment model. Our mathematical model of the UC problem is given by

$$\text{(7.1a)} \qquad \min_{u,p} \quad F(u,p) := \sum_{i=1}^{I} \left\{ \sum_{t=1}^{T} u_{it} C_i(p_{it}) + S_i(u_i) \right\}$$

$$\text{(7.1b)} \qquad \text{s.t.} \quad \sum_{i=1}^{I} u_{it} p_{it} \ge D_t, \quad \sum_{i=1}^{I} u_{it} r_i(p_{it}) \ge R_t, \ t = 1\colon T,$$

$$\text{(7.1c)} \qquad u_{it} \underline{p}_i \le p_{it} \le u_{it} \overline{p}_i, \ t = 1\colon T, \ i = 1\colon I, \quad u_i \in U_i, \ i = 1\colon I,$$

where I is the number of units, T is the number of time periods, D_t and R_t are the demand and reserve in period t, and for each unit i, C_i is a convex cost-power generation function, S_i is the startup/shutdown cost, $u_{it} = 1$ (0) if unit i is operating (shutdown, resp.) at time t, p_{it} is the output power in period t, $\underline{p}_i$ and $\overline{p}_i$ are the minimum and maximum output powers,

$$\text{(7.2)} \qquad r_i(p_{it}) := \min\{\overline{p}_i - p_{it}, p_i^{\Delta}\}$$

is the reserve function, where p_i^{Δ} is the maximum increase in power, $u_i = (u_{i1}, \ldots, u_{iT})$ is the schedule, and U_i represents minimum up/down times and required on/off constraints.

Our UC problem is an instance of (6.2) with $n = 2T$, $m = I + 1$, $z_i = (u_i, p_i)$, $\psi_{0i}(z_i) = -\sum_{t=1}^{T} u_{it} C_i(p_{it}) - S_i(u_i)$, $\psi_{ti}(z_i) = u_{it} p_{it}$, $\psi_{T+t,i}(z_i) = u_{it} r_i(p_{it})$, $t = 1\colon T$,

$Z_i = \{z_i : u_{it}\underline{p}_i \le p_{it} \le u_{it}\overline{p}_i, t = 1\colon T, u_i \in U_i\}$, $i = 1\colon I$, $\psi_{0m}(z_m) = 0$, $\psi_{tm}(z_m) = -D_t z_m$, $\psi_{T+t,m}(z_m) = -R_t z_m$, $t = 1\colon T$, $Z_m = \{1\}$. Then (cf. (6.3)–(6.4))

$$f_i(x) = -\min_{u_i \in U_i} \left\{ \sum_{t=1}^{T} u_{it} \min_{\underline{p}_i \le p_{it} \le \overline{p}_i} [C_i(p_{it}) - x_t p_{it} - x_{T+t} r_i(p_{it})] + S_i(u_i) \right\} \tag{7.3}$$

may be evaluated by finding a minimizer $z_i(x) = (u_i(x), p_i(x))$ of (7.3) via dynamic programming, for $i = 1\colon I$, whereas $f_m(x) = -\langle x, (D, R)\rangle$.

7.2. Obtaining a primal feasible solution. In view of Theorem 6.4, we may suppose that, for k large enough, ν_{ij}^k and $z_i^j = (u_i^j, p_i^j)$, $j \in \hat{J}_i^k$, $i = 1\colon I$, form a relaxed solution to (7.1) treated as an instance of (6.2). Thus we may use the relaxed schedules $\tilde{u}_i^k = \sum_{j \in \hat{J}_i^k} \nu_{ij}^k u_i^j$, and the interpretation of $\tilde{u}_{it}^k \in [0, 1]$ as the probability of unit i to be on-line at time t, in various ways to generate feasible solutions to the original problem (7.1). An important observation is that, in view of Remark 6.5(iii), for problems with many more units than time periods, only relatively few $\tilde{u}_{it}^k$ can be fractional.

The next subsection gives computational results for four simple heuristics, which are only sketched below; their detailed descriptions can be found in [Fel97, FeK97].

First, we note that, due to (7.2), if the schedules $u_i \in U_i$, $i = 1\colon I$, satisfy

$$\sum_{i=1}^{I} u_{it}\overline{p}_i \ge D_t + R_t \quad \text{and} \quad \sum_{i=1}^{I} u_{it} p_i^{\Delta} \ge R_t \quad \text{for } t = 1\colon T, \tag{7.4}$$

then we may solve T continuous optimization problems in p to obtain a feasible solution to (7.1). Conversely, any feasible solution to (7.1) must satisfy (7.4).

Our first heuristic PFS1 works as follows. For successive $t = 1\colon T$, it attempts to satisfy inequalities (7.4) by turning on units i available for startup at time t in order of decreasing probabilities $\tilde{u}_{it}^k$, while respecting the requirement $u_i \in U_i$. In our second *randomized* heuristic PFS2, unit i is turned on with probability $\tilde{u}_{it}^k$ by "tossing a coin," whereas in the third heuristic PFS3, unit i is turned on (off) with probability $\tilde{u}_{it}^k$ if there is a schedule u_i^j, $j \in \hat{J}_i^k$, where the unit is turned on (off, resp.) at time t. The fourth heuristic PFS4 is a randomized extension of PFS1, in which free units (not turned on at time t) are sampled for startup/shutdown with probability $\tilde{u}_{it}^k$.

It may be interesting to relate our heuristics to the heuristic of [ZhG88], which works. Starting from an approximate minimizer $\breve{x} = x^k$ of f_S, until $\psi(z(\breve{x})) \ge 0$ do: pick $\breve{\jmath} \in \operatorname{Arg\,min}_{j=1}^{n} \psi_j(z(\breve{x}))$ and increase $\breve{x}_{\breve{\jmath}}$ until $\psi_{\breve{\jmath}}(z(\breve{x})) \ge 0$. Thus exact coordinate descent on f is made until the partial Lagrangian solution becomes feasible. However, since $\psi(z(\cdot))$ may be discontinuous, no guarantee of success is available, and in practice quite complicated inexact line-searches must be made "intelligently" [ZhG88, p. 768].

7.3. Computational results. In this subsection we report on our preliminary numerical experience.

Table 7.1 gives some details of our test problems. The final two problems are fairly large. We used $x^1 = 0$, $x^{\mathrm{up}} = 100e$, $\kappa = 0.1$. The maximum number of stored subgradients was $2T + 3$ in the *aggregate* case with subproblems (2.11), and $2T{+}2I{+}3$ in the *disaggregate* case with subproblems (6.6). Tables 7.2–7.3 compare the quality of primal feasible solutions generated via the various heuristics by presenting percentages of approximation to the best known values of Table 7.1, with stars

Table 7.1
Test problems and their best known dual and primal values.

Case	I	T	Best dual	Best primal	Gap (%)	Origin
Bard	10	24	5.409528e+05	5.433704e+05	0.44	[Bar88]
Greece	10	24	5.608615e+05	5.658277e+05	0.88	[KBP96]
Irina	10	24	4.719926e+04	4.729448e+04	0.20	[Ris96]
Durham	12	24	2.797602e+04	2.798166e+04	0.02	[ChS86]
Shaw	16	24	1.098853e+06	1.098978e+06	0.01	[Sha95]
Pacific	19	24	1.887889e+06	1.891327e+06	0.18	[LJS97]
Ohio	20	24	1.859801e+05	1.860323e+05	0.04	[FaV86]
EPRI	48	48	2.843720e+06	2.853591e+06	0.35	[ZWC$^+$77]
Emod	48	168	9.909559e+06	9.973104e+06	0.64	scaled EPRI
Bard168	100	168	3.760644e+07	3.767200e+07	0.17	scaled Bard

Table 7.2
Relative primal and dual errors (in %) of the disaggregate bundle.

Case	Iter	Dual error	PFS1	PFS2	PFS3	PFS4
Bard	29	2.4e−01	5.7e+00	*	*	3.9e−01
	47	1.4e−01	5.3e+00	*	*	7.3e−01
	108	1.4e−03	3.2e+00	*	*	1.4e+00
	140	1.5e−04	7.4e+00	*	*	8.1e−01
Greece	26	5.8e−02	*	*	*	1.3e−02
	38	8.1e−03	*	*	*	1.3e−02
	63	9.8e−04	*	*	*	1.9e−01
	78	4.5e−04	*	*	*	1.9e−01
Irina	12	3.6e−02	1.8e+00	2.6e−01	2.6e−01	3.2e−01
	20	5.4e−03	8.7e−01	2.6e−01	2.6e−01	4.4e−01
	29	5.5e−04	1.8e+00	2.6e−01	2.6e−01	4.8e−01
	37	1.3e−04	4.5e−01	2.6e−01	2.6e−01	2.6e−01
Durham	33	3.4e−02	2.2e+01	2.6e−02	1.2e−03	8.7e−02
	39	3.3e−03	1.8e+01	3.3e−02	1.2e−03	8.3e−02
	48	4.3e−04	1.8e+01	2.6e−02	1.2e−03	9.1e−02
	55	7.2e−05	1.8e+01	2.6e−02	1.2e−03	1.1e−01
Shaw	19	3.8e−02	*	6.6e−02	2.0e−03	2.0e−02
	27	3.0e−03	5.9e+00	4.8e−02	2.0e−03	1.1e−02
	34	2.7e−04	6.1e+00	4.8e−02	2.0e−03	1.1e−02
	43	2.7e−04	6.1e+00	4.8e−02	2.0e−03	2.9e−02
Pacific	16	8.0e−02	3.0e−02	*	*	3.0e−02
	37	9.0e−03	1.1e−01	*	*	3.0e−02
	60	5.8e−03	1.1e−01	*	*	3.0e−02
	69	5.3e−03	1.1e−01	*	*	1.1e−01
Ohio	15	2.2e−02	2.3e−01	*	*	9.0e−02
	20	6.5e−03	2.1e−01	*	*	3.6e−02
	28	2.2e−03	2.1e−01	*	*	1.5e−02
	44	5.4e−05	2.3e−01	1.5e−02	9.5e−03	6.3e−02
EPRI	14	5.5e−02	5.1e+00	*	*	2.5e−01
	27	4.6e−03	3.8e+00	*	*	1.1e−02
	50	3.5e−04	4.0e+00	*	*	2.5e−01
	75	0.0e+00	4.6e+00	*	*	2.5e−01
Emod	35	5.1e−02	5.2e+00	*	*	1.6e+00
	70	4.1e−03	4.5e+00	*	*	1.2e+00
	95	6.0e−04	5.9e+00	*	*	1.5e+00
	113	9.1e−05	6.4e+00	*	*	1.5e+00
Bard168	53	2.1e−01	1.5e+00	*	*	2.2e−01
	90	1.4e−01	1.3e+00	*	*	2.9e−01
	272	6.0e−03	2.0e+00	*	*	1.4e−01
	652	1.2e−03	1.6e+00	*	*	8.0e−02

Table 7.3

Relative primal and dual errors (in %) of the aggregate bundle.

Case	Iter	Dual value	PFS1	PFS2	PFS3	PFS4
Bard	38	2.6e−01	2.6e+00	*	*	2.0e+00
	70	1.7e−01	2.6e+00	*	*	7.2e−01
	218	3.7e−03	3.5e+00	*	*	8.0e−01
	258	2.0e−03	2.8e+00	*	*	6.4e−01
Greece	43	7.8e−02	*	*	*	4.6e−02
	78	5.1e−03	*	*	*	9.3e−01
	116	8.0e−04	*	*	*	1.3e+00
	169	8.9e−05	*	*	*	9.3e−01
Irina	19	5.6e−02	8.1e−01	4.1e−01	2.7e−02	7.5e−01
	37	6.1e−03	4.6e−01	2.6e−01	2.6e−01	5.0e−01
	56	5.5e−04	9.2e−01	2.6e−01	2.6e−01	2.6e−01
	83	1.3e−04	1.8e+00	2.6e−01	2.6e−01	3.4e−01
Durham	53	2.4e−02	8.6e+00	3.7e−02	1.2e−03	9.4e−02
	70	5.1e−03	1.4e+01	2.6e−02	1.2e−03	1.6e−01
	84	4.3e−04	8.6e+00	2.6e−02	1.2e−03	1.8e−01
	105	7.2e−05	1.4e+01	2.6e−02	1.2e−03	1.8e−01
Shaw	27	3.7e−02	*	5.7e−02	2.0e−03	7.5e−02
	39	2.1e−03	4.8e+00	8.4e−02	2.0e−03	3.6e−01
	48	2.7e−04	*	4.8e−02	2.0e−03	1.1e−02
	61	2.7e−04	*	4.8e−02	2.0e−03	1.1e−02
Pacific	37	5.6e−02	1.0e−01	*	*	2.3e−01
	57	5.8e−03	3.0e−02	*	*	1.4e−01
	80	2.1e−03	3.0e−02	*	*	3.0e−02
	108	1.6e−03	1.1e−01	*	*	3.0e−02
Ohio	19	4.3e−02	1.2e−01	*	*	1.0e−01
	29	6.6e−03	1.2e−01	6.3e−02	1.5e−02	8.5e−02
	54	2.2e−03	1.2e−01	6.3e−02	9.5e−03	7.9e−02
	103	5.4e−05	1.2e−01	2.6e−02	9.5e−03	7.9e−02
EPRI	27	4.1e−02	5.8e+00	*	*	2.1e−02
	52	6.0e−03	4.9e+00	*	*	1.1e−02
	75	7.0e−04	3.7e+00	*	*	3.5e−01
	105	0.0e+00	5.5e+00	*	*	7.3e−03
Emod	50	5.0e−02	5.9e+00	*	*	1.7e+00
	76	6.9e−03	3.8e+00	*	*	1.1e+00
	115	1.8e−03	4.9e+00	*	*	1.8e+00
	159	1.1e−03	8.2e+00	*	*	1.4e+00
Bard168	82	2.5e−01	1.8e+00	*	*	8.2e−01
	152	1.7e−01	2.0e+00	*	*	6.9e−01
	740	4.6e−03	2.4e+00	*	*	4.3e−01
	1037	3.0e−03	2.0e+00	*	*	4.6e−01

denoting failures. Since the heuristics are relatively cheap, in practice one might run all of them to pick the best solution. For each problem, we give results obtained with the stopping criterion $-v_k \le \epsilon_{\rm opt}(1+|f(x^k)|)$ for successive $\epsilon_{\rm opt} = 10^{-3}$, 10^{-4}, 10^{-5}, 10^{-6}. Usually, when $\epsilon_{\rm opt} = 10^{-l}$ is used, upon termination the dual objective value has l correct digits. We observed the rather surprising phenomenon that higher dual objective accuracy need not imply better quality of the heuristic primal solutions. The best primal results are obtained for an intermediate dual accuracy of $\epsilon_{\rm opt} = 10^{-4}$: using a looser precision does not reveal the right schedules, whereas a too tight precision ($\epsilon_{\rm opt} = 10^{-6}$) discards good schedules. Further, somewhat contrary to our expectations, the disaggregate version need not deliver better solutions. We also note the following: PFS3 always produces the best solution when it delivers any feasible solution; PFS1 is inferior on all problems except Pacific; PFS4 is robust and gives good solutions in comparison with PFS1 and, when possible, with PFS2–PFS3.

Table 7.4
Iteration count and CPU timing (in seconds).

	Disaggregate			Aggregate		
Case	Iter	Mas_%	Time	Iter	Mas_%	Time
Bard	29	71	3.13e−01	38	47	1.64e−01
	47	73	5.39e−01	70	49	3.17e−01
	108	74	1.26e+00	218	55	1.12e+00
	140	74	1.68e+00	258	55	1.33e+00
Greece	26	69	2.76e−01	43	47	1.94e−01
	38	69	4.03e−01	78	49	3.64e−01
	63	69	6.53e−01	116	50	5.53e−01
	78	69	7.96e−01	169	52	8.31e−01
Irina	12	70	1.37e−01	19	51	9.38e−02
	20	71	2.27e−01	37	50	1.80e−01
	29	71	3.28e−01	56	51	2.78e−01
	37	71	4.17e−01	83	52	4.21e−01
Durham	33	69	3.69e−01	53	43	2.47e−01
	39	69	4.44e−01	70	44	3.32e−01
	48	69	5.46e−01	84	46	4.07e−01
	55	69	6.22e−01	105	46	5.17e−01
Shaw	19	69	2.89e−01	27	38	1.60e−01
	27	69	4.03e−01	39	39	2.33e−01
	34	70	5.14e−01	48	40	2.89e−01
	43	69	6.41e−01	61	40	3.69e−01
Pacific	16	57	2.78e−01	37	35	2.79e−01
	36	56	5.84e−01	57	34	4.29e−01
	61	56	9.89e−01	80	35	6.05e−01
	73	56	1.18e+00	108	35	8.28e−01
Ohio	15	70	3.54e−01	19	35	1.78e−01
	20	71	4.81e−01	29	34	2.68e−01
	28	71	6.59e−01	54	33	4.90e−01
	44	71	1.04e+00	103	33	9.33e−01
EPRI	14	68	1.32e+00	27	26	7.97e−01
	27	71	2.67e+00	52	24	1.49e+00
	50	72	4.93e+00	75	24	2.12e+00
	75	72	7.41e+00	105	24	2.97e+00
Emod	35	81	1.92e+01	50	29	5.34e+00
	70	82	3.89e+01	76	26	7.80e+00
	95	82	5.30e+01	115	24	1.15e+01
	113	83	6.33e+01	159	24	1.58e+01
Bard168	53	90	1.18e+02	82	20	1.72e+01
	90	90	1.96e+02	152	20	3.21e+01
	272	90	5.89e+02	740	30	1.78e+02
	652	90	1.43e+03	1037	32	2.57e+02

In Table 7.4 we give the iteration counts and timings obtained on a SUN Enterprise 4000 machine for the successive stopping criteria, with Mas_% being the percentage of time spent on the master QP subproblems. We see that in terms of the CPU time, the decrease in the number of iterations required to reach a certain stopping criterion by using the disaggregate version is offset by the computationally heavier master problems.

Since the heuristics PFS2–PFS4 are based on sampling, there is a question of sample size (and hence solution time) versus solution quality. We found that a sample size of 200 was sufficient in most cases, i.e., after 200 samples usually little improvement was made. The CPU time requirements of the primal heuristics are quite modest compared with the time spent on solving the dual problem. For instance, the CPU times (in seconds) for each heuristic after the termination of the disaggregate version for $\epsilon_{\mathrm{opt}} = 10^{-3}$ had the following ranges: 0.0003–0.0148 for PFS1, 0.0302–4.0858 for PFS2, 0.0109–0.9231 for PFS3, 0.0820–8.6980 for PFS4. Note that these times depend heavily on whether or not a heuristic is successful: each time a feasible solution is found, one has to solve T economic dispatch problems. In general we observe that the total solution times are dominated by the solution of the dual problem.

Acknowledgments. We thank the Associate Editor and two anonymous referees for their constructive comments.

REFERENCES

[AnW93] K. M. Anstreicher and L. Wolsey, *On Dual Solutions in Subgradient Optimization*, Tech. report, Dept. of Management Sciences, University of Iowa, Iowa City, IA, September 1993.

[Bar88] J. F. Bard, *Short-term scheduling of thermal-electric generators using Lagrangian relaxation*, Oper. Res., 36 (1988), pp. 756–766.

[Ber82] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[Ber95] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[BLSP83] D. Bertsekas, G. S. Lauer, N. R. Sandell, and T. A. Posberg, *Optimal short-term scheduling of large-scale power systems*, IEEE Trans. Automat. Control, AC-28 (1983), pp. 1–11.

[BuF91] J. V. Burke and M. C. Ferris, *Characterization of solutions sets of convex programs*, Oper. Res. Lett., 10 (1991), pp. 57–60.

[ChS86] C. H. Cheung and M. J. H. Sterling, *Large-scale unit commitment using a composite thermal generator operator cost function*, in Proceedings of the Second International Conference on Power Systems Monitoring and Control, IEEE Press, Piscataway, 1986, pp. 332–337.

[CoL93] R. Correa and C. Lemaréchal, *Convergence of some algorithms for convex minimization*, Math. Programming, 62 (1993), pp. 261–275.

[Dan63] G. B. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, NJ, 1963.

[DGM$^+$97] D. Dentcheva, R. Gollmer, A. Möller, W. Römisch, and R. Schultz, *Solving the unit commitment problem in power generation by primal and dual methods*, in Progress in Industrial Mathematics at ECMI 96, M. Brøns, M. P. Bendsøe, and M. P. Sørensen, eds., Teubner, Stuttgart, 1997, pp. 332–339.

[FaV86] B. Fardanesh and F. E. Villaseca, *Two-step optimal thermal generation scheduling*, Automatica, 22 (1986), pp. 361–366.

[FeK97] S. Feltenmark and K. C. Kiwiel, *Generalized Linear Programming Solves the Relaxed Primal*, Tech. report TRITA/MAT-97-OS11, Dept. of Mathematics, Royal Institute of Technology, Stockholm, Sweden, August 1997.

[Fel97] S. Feltenmark, *On Optimization of Power Production*, Ph.D. thesis, Dept. of Mathematics, Royal Institute of Technology, Stockholm, Sweden, 1997.

[Fis81] M. L. Fisher, *The Lagrangian relaxation method for solving integer programming problems*, Management Sci., 27 (1981), pp. 1–18.

[HUL93] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Springer-Verlag, Berlin, 1993.

[KBP96] S. A. Kazarlis, A. G. Bakirtzis, and V. Petridis, *A genetic algorithm solution to the unit commitment problem*, IEEE Trans. Power Systems, 11 (1996), pp. 83–91.

[Kiw89] K. C. Kiwiel, *A dual method for certain positive semidefinite quadratic programming problems*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 175–186.

[Kiw90] K. C. Kiwiel, *Proximity control in bundle methods for convex nondifferentiable minimization*, Math. Programming, 46 (1990), pp. 105–122.

[Kiw91] K. C. Kiwiel, *Exact penalty functions in proximal bundle methods for constrained convex nondifferentiable minimization*, Math. Programming, 52 (1991), pp. 285–302.

[Kiw94] K. C. Kiwiel, *A Cholesky dual method for proximal piecewise linear programming*, Numer. Math., 68 (1994), pp. 325–340.

[Kiw95] K. C. Kiwiel, *Approximations in proximal bundle methods and decomposition of convex programs*, J. Optim. Theory Appl., 84 (1995), pp. 529–548.

[Kiw96] K. C. Kiwiel, *Restricted step and Levenberg–Marquardt techniques in proximal bundle methods for nonconvex nondifferentiable optimization*, SIAM J. Optim., 6 (1996), pp. 227–249.

[LaL89] T. Larsson and Z. Liu, *A Primal Convergence Result for Dual Subgradient Optimization with Application to Multicommodity Network Flows*, Tech. report, Dept. of Mathematics, Linköping University, Linköping, Sweden, 1989.

[Las70] S. Lasdon, *Optimization Theory for Large Systems*, MacMillan, New York, 1970.

[Lem75] C. Lemaréchal, *An extension of Davidon methods to nondifferentiable problems. Nondifferentiable optimization*, Math. Programming Stud., 3 (1975), pp. 95–109.

[Lem77] C. Lemaréchal, *Nonsmooth Optimization and Descent Methods*, Research report RR–78–4, International Institute of Applied Systems Analysis, Laxenburg, Austria, 1977.

[LeR96] C. Lemaréchal and A. Renaud, *Dual-equivalent Convex and Nonconvex Problems*, Research report, INRIA, Rocquencourt, France, 1996.

[LeS97] C. Lemaréchal and C. Sagastizábal, *Variable metric bundle methods*: *From conceptual to implementable forms*, Math. Programming, 76 (1997), pp. 393–410.

[LJS97] C. A. Li, R. A. Johnson, and A. J. Svoboda, *A new unit commitment method*, IEEE Trans. Power Systems, 12 (1997), pp. 113–119.

[LPRS96] C. Lemaréchal, F. Pellegrino, A. Renaud, and C. Sagastizábal, *Bundle methods applied to the unit-commitment problem*, in System Modelling and Optimization, Proceedings of the Seventeenth IFIP TC7 Conference on System Modelling and Optimization, 1995, J. Doležal and J. Fidler, eds., Chapman & Hall, London, 1996, pp. 395–402.

[LPS98] T. Larsson, M. Patriksson, and A.-B. Strömberg, *Ergodic convergence in subgradient optimization*, Optim. Methods Softw., 9 (1998), pp. 93–120.

[LPS99] T. Larsson, M. Patriksson, and A.-B. Strömberg, *Ergodic, primal convergence in dual subgradient schemes for convex programming*, Math. Programming, 86 (1999), pp. 283–312.

[LSB81] C. Lemaréchal, J.-J. Strodiot, and A. Bihain, *On a bundle algorithm for nonsmooth optimization*, in Nonlinear Programming 4, O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, eds., Academic Press, New York, 1981, pp. 245–282.

[Mif82] R. Mifflin, *A modification and an extension of Lemaréchal's algorithm for nonsmooth minimization*, Math. Programming Stud., 17 (1982), pp. 77–90.

[MSW76] T. L. Magnanti, J. F. Shapiro, and M. H. Wagner, *Generalized linear programming solves the dual*, Management Sci., 23 (1976), pp. 1195–1203.

[MuK77] J. A. Muckstadt and S. A. Koenig, *An application of Lagrangian relaxation to scheduling in thermal power-generation systems*, Oper. Res., 25 (1977), pp. 387–403.

[Ris96] I. Rish, *private communication*, Department of Information and Computer Science, University of California, Irvine, CA, September, 1996.

[Rob86] S. M. Robinson, *Bundle-based decomposition*: *Description and preliminary results*, in System Modelling and Optimization, Lecture Notes in Control and Inform. Sci. 84, A. Prékopa, J. Szelezsán, and B. Strazicky, eds., Springer-Verlag, Berlin, 1986, pp. 751–756.

[Rob89] S. M. Robinson, *Bundle-based decomposition*: *Conditions for convergence*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 6 (1989), pp. 435–447.

[Rzh89] S. V. Rzhevskii, *A conditional ϵ-subgradient method for the simultaneous solution of the dual and primal problems of convex programming*, Kibernetika, (1989), pp. 54–64 (in Russian). English translation in Cybernetics, 25 (1989), pp. 203–218.

[RzK85] S. V. Rzhevskii and A. V. Kuncevich, *An application of the ϵ-subgradient method to the solution of the dual and primal problems of convex programming*, Kibernetika, (1985), pp. 51–54 (in Russian).

[ScZ88] H. Schramm and J. Zowe, *A combination of the bundle approach and the trust region concept*, in Advances in Mathematical Optimization, Math. Res. 45, J. Guddat et al., eds., Akademie-Verlag, Berlin, 1988, pp. 196–209.

[Sha79] J. F. Shapiro, *Mathematical Programming*: *Structures and Algorithms*, Wiley, New York, 1979.

[Sha95] J. J. Shaw, *A direct method for security-constrained unit commitment*, IEEE Trans. Power Systems, 10 (1995), pp. 1329–1342.

[ShC96] H. D. Sherali and G. Choi, *Recovery of primal solutions when using subgradient optimization methods to solve Lagrangian duals of linear programs*, Oper. Res. Lett., 19 (1996), pp. 105–113.

[ShF94] G. B. Sheblé and G. N. Fahd, *Unit commitment literature synopsis*, IEEE Trans. Power Systems, 9 (1994), pp. 128–135.

[Sho79] N. Z. Shor, *Minimization Methods for Non-Differentiable Functions*, Naukova Dumka, Kiev, 1979 (in Russian). English translation published by Springer-Verlag, Berlin, 1985.

[Wol75] P. Wolfe, *A method of conjugate subgradients for minimizing nondifferentiable functions*, Math. Programming Stud., 3 (1975), pp. 145–173.

[ZhG88] G. Zhuang and F. D. Galiana, *Towards a more rigorous and practical unit commitment by Lagrangian relaxation*, IEEE Trans. Power Systems, 3 (1988), pp. 763–773.

[Zhu77] N. G. Zhurbenko, *Investigation of a Class of Algorithms for Minimizing Nonsmooth Functions and Their Application to the Solution of Large-Scale Problems*, Dissertation, Cybernetics Institute, Academy of Sciences of the Ukrainian SSR, Kiev, 1977 (in Russian).

[ZWC$^+$77] H. W. Zeminger, A. J. Wood, H. K. Clark, T. F. Laskowski, and J. D. Burns, *Synthetic Electric Utility Systems for Evaluating Advanced Technologies*, Final report EM-285, Electrical Power Research Institute (EPRI), Palo Alto, CA, 1977.

SIAM J. Optim.
Vol. 10, No. 3, pp. 722–731

CONVERGENCE OF A GENERAL CLASS OF ALGORITHMS FOR SEPARATED CONTINUOUS LINEAR PROGRAMS[*]

MALCOLM C. PULLAN[†]

Abstract. Separated continuous linear programs (SCLP) are a type of infinite-dimensional linear program which can serve as a useful model for a variety of dynamic network problems where storage is permitted at the nodes. This paper proves the convergence of a general class of algorithms for solving SCLP under certain restrictions on the problem data. This is the first such proof for any nondiscretization algorithm for solving any form of continuous linear programs.

Key words. continuous linear programming, linear optimal control

AMS subject classifications. 49M99, 49N05, 65K05, 90C45

PII. S1052623494278827

1. Introduction.

This paper is concerned with a particular form of infinite-dimensional linear programs called *separated continuous linear programs* (SCLP), first introduced by Anderson [1] in an attempt to model job-shop scheduling problems. This problem is a special case of a more general class of problems known as *continuous linear programs* (CLP) first introduced by Bellman [8] in 1953. The problem SCLP can also be viewed as a useful model for various forms of dynamic network flow problems where storage is permitted at the nodes. Such problems occur in many real-life situations, for instance, in the dynamic routing of traffic in a network (see, for example, Segall [15]) or the closely related problem of routing fluid flows in networks (see, for example, Weiss [16]).

Separated continuous linear programs may be defined as follows:

$$\text{SCLP:}\quad \text{minimize}\quad \int_0^T c(t)^T x(t)\,dt$$

$$\text{subject to}\quad \int_0^t Gx(s)\,ds + y(t) = a(t), \tag{1.1}$$

$$Hx(t) + z(t) = b(t), \tag{1.2}$$

$$x(t), y(t), z(t) \geq 0, \qquad t \in [0,T].$$

Here $x(t)$, $z(t)$, $b(t)$, and $c(t)$ are bounded measurable functions and $y(t)$ and $a(t)$ are absolutely continuous functions. The dimensions of $x(t)$, $y(t)$, and $z(t)$ are n_1, n_2, and n_3, respectively. Thus G is an $n_2 \times n_1$ matrix and H is an $n_3 \times n_1$ matrix. We let $\omega(t)$ denote a complete set of variables for SCLP, i.e., $\omega(t)^T = (x(t)^T, y(t)^T, z(t)^T)$.

Ever since the introduction of CLP, the development of an efficient and convergent algorithm to solve any form of the problem has eluded many people, for instance, Perold [10, 11] and Anstreicher [7]. The difficulty in the development of algorithms has been at the very fundamental level of trying to find an improvement step, that is, a step to construct an improved feasible solution to the problem starting from a

[*] Received by the editors December 19, 1994; accepted for publication (in revised form) July 23, 1999; published electronically March 21, 2000.
http://www.siam.org/journals/siopt/10-3/27882.html

[†] St. John's College, Cambridge CB2 1TP, UK.

nonoptimal one. While authors such as Perold [10, 11] and Anstreicher [7] certainly did describe improvement steps, they worked only under certain "nondegeneracy" assumptions on the current solution. Therefore, previous researchers studying CLP did not even begin to answer the question of convergence of an algorithm because there were no such algorithms.

In 1989 Anderson and Philpott [4] broke the trend and developed an algorithm aimed at solving a dynamic single-commodity network program (called CNP) under certain practical restrictions on the problem data. This network problem is a very specialized case of SCLP and hence of CLP. It was the first algorithm for any class of CLP problems to give a general improvement step (i.e., one which did not require assumptions about the current solution). Unfortunately, though, it was later observed that in many instances the algorithm did not converge to an optimal solution.

Recently, Pullan [12] developed an algorithm aimed at solving SCLP under similar restrictions on the problem data as for the network problem in [4]. Strictly speaking, this algorithm is a whole class of algorithms based on a single idea. It is also a complete departure from previous work in that it was not based on a simplex-like approach, although such ideas did motivate its development. Moreover, unlike the algorithm for CNP in [4], this algorithm did appear to converge in every case tried, although no proof of convergence was given.

Shortly after the development of the algorithm in Pullan [12], Philpott and Craddock [9] utilized the ideas of [12] to produce an algorithm for solving CNP (but with a direct extension to include SCLP) for which they did prove convergence. The algorithm is called the adaptive discretization algorithm, a name which accurately summarizes its properties. It is essentially a discretization algorithm which proceeds by adding and removing points in the partition used in the current discretization. However, the precise points entered into the partition are somewhat arbitrary in that they are always equally spaced between two existing ones. This arbitrary nature is somewhat against the general philosophy of most previous work on CLP. The aim of this previous work has been to develop an algorithm that would not discretize arbitrarily, with the hope that if such an algorithm were to be found, it would prove to be more efficient than discretization methods and reveal more information about the problem.

The algorithm in Pullan [12] for SCLP is such an algorithm that does not discretize arbitrarily. As predicted, it has also revealed a lot more information about the problem. This is exemplified by the extensive duality theory developed in Pullan [14] as a result. The purpose of this paper is to prove that the algorithm in [12], in its full generality, always converges. This is the first such proof for any kind of algorithm for solving any class of CLP. Not only that but, as far as this author is aware, it is only the second convergence proof for an algorithm for solving any type of infinite-dimensional linear programs, the other being for a continuous transportation problem in Anderson and Nash [2, Chapter 5]. (Here we make the distinction between infinite-dimensional linear programs, where there are both an infinite number of variables and an infinite number of constraints, and semi-infinite linear programs, where either the number of variables or the number of constraints is finite.)

The plan of this paper is as follows. In the next section we summarize the necessary results and algorithm from Pullan [12]. We assume that the reader is already familiar with this, and we just collect the results together for ease of reference. In section 3 we formally state the algorithm for which we prove convergence. The algorithm given includes one completely general step for which we list several possible alternatives. In section 4 we then prove the convergence of the algorithm. Finally in section 5, we comment on the implications of this result for future work and discuss its relationship to the work of Philpott and Craddock [9].

Before we begin, we formally state the assumptions on SCLP under which we will work in this paper.

Assumption 1.1. *The costs, $c(t)$, are piecewise linear; $a(t)$ is piecewise linear and continuous; and $b(t)$ is piecewise constant on $[0,T]$. Also the feasible region for* SCLP *is nonempty and bounded.*

Here *bounded* means there exists $M < \infty$ such that for any $\omega(\cdot)$ feasible for SCLP, $\|\omega_i(\cdot)\|_\infty \leq M$ for each i.

This assumption is often satisfied in practical problems (e.g., in both Segall [15] and Weiss [16]). In addition, Anderson, Nash, and Perold [3] have shown that this assumption ensures that SCLP has an optimal solution in which $x(t)$ is piecewise constant on $[0,T]$ (as does Pullan [13], which, along with Anderson and Philpott [5], also gives several, more general results of a similar nature).

Finally in this introduction we give the following definition.

Definition 1.1.

1. *The* breakpoints *of a piecewise linear or piecewise constant function are the discontinuities in either the function or its derivative.*
2. *We define the* initial breakpoint partition *to be the smallest partition of* $[0,T]$ *consisting of all the breakpoints of* $a(\cdot)$, $b(\cdot)$, *and* $c(\cdot)$.
3. *Let* $\omega(t)$ *be a feasible solution for* SCLP *such that* $x(t)$ *is piecewise constant on* $[0,T]$. *We define the* breakpoint partition for $\omega(t)$ *to be the partition of* $[0,T]$ *consisting of all the breakpoints of* $\omega(t)$ *and the points in the initial breakpoint partition.*
4. *Let* f *be any real valued function. We use the notation* $f(t-)$ *to denote* $\lim_{s\uparrow t} f(s)$ *and* $f(t+)$ *to denote* $\lim_{s\downarrow t} f(s)$ *when these limits exist.*

2. Summary of results from Pullan [12]. In this section we summarize the results and concepts from Pullan [12] needed for this paper. The key to all the main results in [12] lies in the study of a special discretization called AP(P). We will see that the proof of convergence of the algorithm to follow will also rely on the properties of this special discretization. We therefore state this discretization and summarize its important properties.

Let $P = \{t_0, t_1, \ldots, t_m\}$ be any refinement of the initial breakpoint partition. Given P, define

$$u_i = \frac{t_{i-1} + t_i}{2},$$

$$\tau_i = \frac{t_i - t_{i-1}}{2}.$$

The variables in the discretization AP(P) are $\hat{x}(t_{i-1}+)$, $\hat{x}(t_i-)$, $\hat{y}(t_i)$, $\hat{y}(u_i)$, $\hat{z}(t_{i-1}+)$, and $\hat{z}(t_i-)$ which, as the notation suggests, correspond in some way to the function values of x, y, and z at $t_{i-1}+$, t_i-, and u_i of a feasible solution for SCLP. (This correspondence is made precise in Definition 2.2 below.) We define the discretization AP(P) of SCLP as follows (written in a slightly different form from Pullan [12]):

$$
\begin{array}{lll}
\mathrm{AP}(P): & \text{minimize} & \displaystyle\sum_{i=1}^{m} \big(c(t_{i-1}+)^T\hat{x}(t_{i-1}+) + c(t_i-)^T\hat{x}(t_i-)\big) \\
& \text{subject to} & G\hat{x}(t_0+) + \hat{y}(u_1) = a(u_1), \\
& & G\hat{x}(t_i-) + \hat{y}(t_i) - \hat{y}(u_i) = a(t_i) - a(u_i), \quad i = 1, \dots, m, \\
& & G\hat{x}(t_{i-1}+) + \hat{y}(u_i) - \hat{y}(t_{i-1}) = a(u_i) - a(t_{i-1}), \quad i = 2, \dots, m, \\
& & H\hat{x}(t_{i-1}+) + \hat{z}(t_{i-1}+) = \tau_i b(t_{i-1}+), \quad i = 1, \dots, m, \\
& & H\hat{x}(t_i-) + \hat{z}(t_i-) = \tau_i b(t_i-), \quad i = 1, \dots, m, \\
& & \hat{x}(t_{i-1}+), \hat{x}(t_i-), \hat{y}(t_i), \hat{y}(u_i), \hat{z}(t_{i-1}+), \hat{z}(t_i-) \geq 0, \\
& & \qquad\qquad\qquad\qquad\qquad\qquad i = 1, \dots, m,
\end{array}
$$

or, in matrix form,

$$
\begin{array}{lll}
\mathrm{AP}(P): & \text{minimize} & \hat{c}^T\hat{\omega} \\
& \text{subject to} & \hat{A}\hat{\omega} = \hat{b}, \\
& & \hat{\omega} \geq 0,
\end{array}
$$

for appropriately defined $\hat{c}$, $\hat{A}$, and $\hat{b}$. The most important result about AP(P) is the following (see Theorem 3.5 in [12]).

Lemma 2.1. *Let P be any refinement of the initial breakpoint partition. Then $V[\mathrm{AP}(P)] \leq V[\mathrm{SCLP}]$.*

Here and throughout the paper we use the notation $V[\mathrm{LP}]$ to denote the optimal value of a linear program LP.

The correspondence between feasible solutions of AP(P) and SCLP is given in the next definition and following lemma (the latter being an amalgamation of Theorems 3.4 and 3.7 and Corollary 3.6 in [12]). It is important to note the different properties of SCLP solutions constructed from solutions to AP(P), and AP(P) solutions constructed from solutions to SCLP.

Definition 2.2. *Let $P = \{t_0, t_1, \dots, t_m\}$ be any refinement of the initial breakpoint partition. Suppose that $\omega(t)$ is feasible for* SCLP *with $x(t)$ piecewise constant with breakpoints in P. We say that $\hat{\omega}$ defined by*

$$
\begin{aligned}
\hat{x}(t_{i-1}+) &= \tau_i x(t_{i-1}+), \\
\hat{x}(t_i-) &= \tau_i x(t_i-), \\
\hat{y}(t_i) &= y(t_i), \\
\hat{y}(u_i) &= y(u_i), \\
\hat{z}(t_{i-1}+) &= \tau_i z(t_{i-1}+), \\
\hat{z}(t_i-) &= \tau_i z(t_i-), \qquad i = 1, \dots, m,
\end{aligned}
$$

is the natural solution *for* AP(P) *(constructed from $\omega(t)$). Similarly, suppose now that $\hat{\omega}$ is any feasible solution for* AP(P)*; then we say that $\omega(t)^T = (x(t)^T, y(t)^T, z(t)^T)$ defined by*

$$
x(t) = \begin{cases}
\dfrac{1}{\tau_i}\hat{x}(t_{i-1}+), & t \in [t_{i-1}, u_i), \quad i = 1, \dots, m, \\
\dfrac{1}{\tau_i}\hat{x}(t_i-), & t \in [u_i, t_i), \quad i = 1, \dots, m, \\
\dfrac{1}{\tau_m}\hat{x}(t_m-), & t = T,
\end{cases}
$$

and with $y(t)$ and $z(t)$ from the constraints of SCLP (*i.e., satisfying* (1.1) *and* (1.2)) *is the* natural solution *for* SCLP *(constructed from $\hat{\omega}$).*

Lemma 2.3. *Suppose that $\omega(t)$ is feasible for* SCLP *with $x(t)$ piecewise constant. Let P be any refinement of the breakpoint partition for $\omega(t)$. Then the natural solution $\hat{\omega}$ for* AP(P) *is feasible for* AP(P) *and the objective function values of the two solutions are the same in their respective linear programs. Furthermore, if $\hat{\omega}$ is optimal for* AP(P)*, then $\omega(t)$ is optimal for* SCLP.

Conversely, let $P = \{t_0, t_1, \dots, t_m\}$ be any refinement of the initial breakpoint partition and suppose that $\hat{\omega}$ is feasible for AP(P)*. Then the natural solution $\omega(t)$ for* SCLP *is feasible for* SCLP *and the difference in the values of the objective function is given by*

$$\begin{aligned}\alpha(\omega) &\equiv \hat{c}^T\hat{\omega} - \int_0^T c(t)x(t)\,dt \\ &= \sum_{i=1}^{m} \left(\frac{(t_i - t_{i-1})^2}{8}\right)(x(t_i-) - x(t_{i-1}+))^T \dot{c}(t_i-).\end{aligned}$$

Having discussed the discretization AP(P) we now summarize the improvement step given in Pullan [12], that is, the method whereby a nonoptimal solution for SCLP can be improved. This step will form the basis of the algorithm to be studied in the next sections.

Let $\omega(t)$ be a feasible solution for SCLP such that $x(t)$ is piecewise constant. Let P be any refinement of the breakpoint partition for $\omega(t)$, and $\hat{\omega}$ be the natural solution for AP(P). If $\hat{\omega}$ is optimal for AP(P), then by Lemma 2.3, $\omega(t)$ is optimal for SCLP. Otherwise we may construct an improved feasible solution $\hat{\tilde{\omega}}$ for AP(P). Let $\delta \equiv \hat{c}^T\hat{\tilde{\omega}} - \hat{c}^T\hat{\omega} < 0$ and $\tilde{\omega}(t)$ be the natural solution for SCLP constructed from $\hat{\tilde{\omega}}$. Choose $\varepsilon \in [0,1]$ and set $\varepsilon_i = \tau_i\varepsilon$. We may then define a new feasible solution $\bar{\omega}_\varepsilon(t)$ by

$$\bar{x}_\varepsilon(t) = \begin{cases} \tilde{x}(t), & t \in [t_{i-1}, t_{i-1}+\varepsilon_i) \cup [t_i - \varepsilon_i, t_i), \quad i = 1, \dots, m, \\ x(t) & \text{otherwise,} \end{cases}$$

with again $\bar{y}_\varepsilon(t)$ and $\bar{z}_\varepsilon(t)$ derived from the constraints of SCLP. We refer to this as *patching $\omega(t)$ and $\tilde{\omega}(t)$ together.* Not only do we get a new feasible solution (Corollary 4.2 in [12]) but this solution also gives an improvement over $\omega(t)$ in objective function value for appropriately chosen ε (see Corollary 4.4 in [12]).

Theorem 2.4. *For ε sufficiently small, $\int_0^T c(t)^T\bar{x}_\varepsilon(t)\,dt < \int_0^T c(t)^T x(t)\,dt$ and*

$$\min_\varepsilon \int_0^T c(t)^T\bar{x}_\varepsilon(t)\,dt - \int_0^T c(t)^T x(t)\,dt = \begin{cases} \dfrac{\delta^2}{4\alpha}, & \alpha < 0 \textit{ and } \dfrac{\delta}{2\alpha} < 1, \\ \delta - \alpha & \textit{otherwise} \end{cases}$$

and occurs at

$$\varepsilon^* = \begin{cases} \dfrac{\delta}{2\alpha}, & \alpha < 0 \textit{ and } \dfrac{\delta}{2\alpha} < 1, \\ 1 & \textit{otherwise,} \end{cases}$$

where $\alpha = \alpha(\tilde{\omega})$ given in Lemma 2.3.

We refer to patching $\omega(t)$ and $\tilde{\omega}(t)$ together with $\varepsilon = \varepsilon^*$ above as *patching* $\omega(t)$ *and* $\tilde{\omega}(t)$ *together optimally.*

With these preliminaries we now proceed to state a general algorithm based on these ideas and prove its convergence.

3. The algorithm. We now formally state the algorithm that we will study in this paper.

0. Let P_1 be the initial breakpoint partition and $\omega^{(0)}(t)$ be any feasible solution for SCLP with breakpoints in P_1. Let $\hat{\omega}^{(0)}$ be the natural solution for AP(P_1). Set $n = 1$.
1. If $\hat{\omega}^{(n-1)}$ is optimal for AP(P_n) then stop as $\omega^{(n-1)}(t)$ is optimal for SCLP (Lemma 2.1).
2. Optimize AP(P_n) to produce $\hat{\tilde{\omega}}^{(n)}$. Let $\tilde{\omega}^{(n)}(t)$ be the natural solution for SCLP.
3. Patch $\omega^{(n-1)}(t)$ and $\tilde{\omega}^{(n)}(t)$ together optimally to produce $\bar{\omega}^{(n)}(t)$.
4. Perform any other step to produce a feasible solution $\omega^{(n)}(t)$ for SCLP whose objective function value is at least as good as that of $\bar{\omega}^{(n)}(t)$.
5. Let P_{n+1} be the breakpoint partition for $\omega^{(n)}(t)$ (or some refinement of it) and $\hat{\omega}^{(n)}$ the natural solution for AP(P_{n+1}). Set $n = n + 1$ and return to Step 1.

The generality of this algorithm, of course, lies in step 4. Some of the possible choices for this step are as follows:

- Do nothing.
- Purify $\bar{\omega}^{(n)}(t)$; i.e., produce an extreme-point solution without increasing the value of the objective function. Such a scheme has been given in Anderson and Pullan [6].
- Some steps of the above algorithm where AP is not optimized but just merely improved at each stage.
- Optimize DP(Q), where Q is the breakpoint partition for $\bar{\omega}^{(n)}(t)$, and set $\omega^{(n)}(t)$ to be the corresponding SCLP solution obtained from this optimal solution to DP(Q). (See Pullan [12]. DP(P) is another discretization for SCLP which is simpler and more obvious than AP(P). Unlike AP(P), any solution to DP(P) has a natural solution for SCLP with the same objective function value and vice versa, that is, if P is a refinement of the breakpoint partition for the SCLP solution.)
- Any combination of the above.

4. Convergence of the algorithm. We now proceed to prove the convergence of the general algorithm stated in the previous section. Let the partition P_n be given by

$$P_n = \{t_0^{(n)}, t_1^{(n)}, \dots, t_{m_n}^{(n)}\},$$

and let $\hat{c}^{(n)}$ denote the cost vector for AP(P_n). We define the following quantities:

$$\delta_n = \hat{c}^{(n)^T} \hat{\tilde{\omega}}^{(n)} - \hat{c}^{(n)^T} \hat{\omega}^{(n-1)},$$

$$\alpha_n = \hat{c}^{(n)^T} \hat{\tilde{\omega}}^{(n)} - \int_0^T c(t)^T \tilde{x}^{(n)}(t)\, dt,$$

$$f_n = \int_0^T c(t)^T \bar{x}^{(n)}(t)\, dt - \int_0^T c(t)^T x^{(n-1)}(t)\, dt.$$

Using the results from Pullan [12] introduced in section 2 we obtain

$$\alpha_n = \sum_{i=1}^{m_n} \left(\frac{(t_i^{(n)} - t_{i-1}^{(n)})^2}{8} \right) (\tilde{x}^{(n)}(t_i^{(n)}-) - \tilde{x}^{(n)}(t_{i-1}^{(n)}+))^T \dot{c}(t_i^{(n)}-), \tag{4.1}$$

$$f_n = \begin{cases} \dfrac{\delta_n^2}{4\alpha_n}, & \alpha_n < 0 \text{ and } \dfrac{\delta_n}{2\alpha_n} < 1, \\ \delta_n - \alpha_n & \text{otherwise.} \end{cases} \tag{4.2}$$

We now establish some simple results concerning these quantities. First, it is clear that by definition, $\delta_n \leq 0$ for each n and $\delta_n = 0$ if and only if the algorithm stops at the nth iteration. The next lemma gives the properties of α_n that we will require for the convergence proof.

LEMMA 4.1. *We have $\alpha_n \leq 0$ for each n and $\alpha_n = 0$ if and only if the algorithm terminates at the $(n+1)$th iteration. Also there exists N such that $|\alpha_n| \leq N$ for all n.*

Proof. We have by step 2 of the algorithm and Lemma 2.1,

$$\begin{aligned} \hat{c}^{(n)^T} \hat{\tilde{\omega}}^{(n)} &= V[\mathrm{AP}(P_n)] \\ &\leq V[\mathrm{SCLP}] \\ &\leq \int_0^T c(t)^T \tilde{x}^{(n)}(t)\, dt. \end{aligned}$$

Hence, by the definition of α_n, $\alpha_n \leq 0$ with equality if and only if $\tilde{\omega}^{(n)}(t)$ is optimal for SCLP, in which case the algorithm will terminate at the next iteration.

To show that α_n is uniformly bounded, let M be a uniform bound on $\|x(t)\|$ for any feasible solution for SCLP and C be a bound on $\|\dot{c}(t)\|$. Then by (4.1),

$$\begin{aligned} |\alpha_n| &= \left| \sum_{i=1}^{m_n} \left(\frac{(t_i^{(n)} - t_{i-1}^{(n)})^2}{8} \right) (\tilde{x}^{(n)}(t_i^{(n)}-) - \tilde{x}^{(n)}(t_{i-1}^{(n)}+))^T \dot{c}(t_i^{(n)}-) \right| \\ &\leq \frac{MC}{4} \sum_{i=1}^{m_n} (t_i^{(n)} - t_{i-1}^{(n)})^2 \\ &\leq \frac{MC}{4} \left(\sum_{i=1}^{m_n} (t_i^{(n)} - t_{i-1}^{(n)}) \right)^2 \\ &= \frac{MCT^2}{4} \end{aligned}$$

and so the result follows. □

We now establish the required properties of f_n for the convergence proof.

LEMMA 4.2. *We have $f_n < 0$ for each n and* $\lim_{n\to\infty} f_n = 0$.

Proof. The fact that $f_n < 0$ follows from Theorem 2.4. Now by the general nature of step 4,

$$\begin{aligned} V[\mathrm{SCLP}] - \int_0^T c(t)^T x^{(0)}(t)\,dt &\le \sum_{n=1}^{\infty} \int_0^T c(t)^T (x^{(n)}(t) - x^{(n-1)}(t))\,dt \\ &\le \sum_{n=1}^{\infty} \int_0^T c(t)^T (\bar{x}^{(n)}(t) - x^{(n-1)}(t))\,dt \\ &= \sum_{n=1}^{\infty} f_n \end{aligned}$$

by the definition of f_n. Hence $f_n \to 0$ as $n \to \infty$. □

We may now prove the convergence result for the algorithm of the previous section.

THEOREM 4.3. *The algorithm in section* 3 *converges for any implementation of step* 4*; i.e., either the algorithm terminates in a finite number of steps with an optimal solution or*

$$\lim_{n\to\infty} \int_0^T c(t)^T x^{(n)}(t)\,dt = V[\mathrm{SCLP}].$$

Proof. Assume that the algorithm does not terminate after a finite number of steps. We now have two cases to consider: either $\delta_n/(2\alpha_n) \ge 1$ for only finitely many n or not. Consider first the case where $\delta_n/(2\alpha_n) \ge 1$ for only finitely many n. Then there exists m such that for all $n \ge m$, $\delta_n/(2\alpha_n) < 1$. Hence, from (4.2) and the fact that $\alpha_n < 0$ by Lemma 4.1, we have

$$f_n = \frac{\delta_n^2}{4\alpha_n}, \qquad n \ge m.$$

But $\lim_{n\to\infty} f_n = 0$ by Lemma 4.2, and α_n is uniformly bounded with respect to n by Lemma 4.1. Hence $\lim_{n\to\infty} \delta_n = 0$. Now by the definition of δ_n we have

$$\begin{aligned} \delta_n &= \hat{c}^{(n)^T} \hat{\hat{\omega}}^{(n)} - \hat{c}^{(n)^T} \hat{\omega}^{(n-1)} \\ &= V[\mathrm{AP}(P_n)] - \int_0^T c(t)^T x^{(n-1)}(t)\,dt \\ &\le V[\mathrm{SCLP}] - \int_0^T c(t)^T x^{(n-1)}(t)\,dt \\ &\le 0 \end{aligned}$$

by Lemmas 2.1 and 2.3. Hence $\delta_n \to 0$ implies

$$\lim_{n\to\infty} \int_0^T c(t)^T x^{(n)}(t)\,dt = V[\mathrm{SCLP}].$$

Now consider the case where $\delta_n/(2\alpha_n) \ge 1$ for infinitely many n. Let $\{n_k\}_{k=1}^{\infty}$ be such that $\delta_{n_k}/(2\alpha_{n_k}) \ge 1$. Then from (4.2) we have

$$f_{n_k} = \delta_{n_k} - \alpha_{n_k}, \qquad k = 1, 2, \ldots .$$

Thus by Lemma 4.2, $\lim_{k\to\infty}(\delta_{n_k} - \alpha_{n_k}) = 0$. However, $\delta_{n_k}/(2\alpha_{n_k}) \ge 1$ for each k and so we have $\delta_{n_k} - \alpha_{n_k} \le \alpha_{n_k} < 0$. Hence $\lim_{k\to\infty} \alpha_{n_k} = 0$. Now by the definition of α_{n_k} we have

$$\begin{aligned}\alpha_{n_k} &= \hat{c}^{(n_k)^T}\hat{\tilde{\omega}}^{(n_k)} - \int_0^T c(t)^T \tilde{x}^{(n_k)}(t)\,dt\\ &\leq V[\mathrm{AP}(P_n)] - \int_0^T c(t)^T \bar{x}^{(n_k)}(t)\,dt,\end{aligned}$$

since patching together is done optimally (recall that $\tilde{\omega}(t)$ is a possible outcome of the patching together process with $\varepsilon = 1$). Hence, in a way similar to the above, we now obtain

$$\begin{aligned}\alpha_{n_k} &\leq V[\mathrm{SCLP}] - \int_0^T c(t)^T x^{(n_k)}(t)\,dt\\ &\leq 0\end{aligned}$$

by Lemma 2.1 and the general nature of step 4 in the algorithm. Thus we now have

$$\lim_{k\to\infty}\int_0^T c(t)^T x^{(n_k)}(t)\,dt = V[\mathrm{SCLP}].$$

Finally, the objective function values of $\omega^{(n)}(t)$ are strictly monotonic decreasing and so we have convergence of the whole sequence of objective function values. $\square$

5. Remarks. The algorithm discussed in this paper is quite general and certainly includes all the possibilities mentioned in the final section of Pullan [12]. The convergence proof in this paper has opened the way for a detailed numerical study of the various possible implementations of the algorithm. The numerical results in Anderson and Pullan [6] suggest that the above algorithm performs very well if step 4 includes a purification step.

It is also worth commenting on the algorithm given in Philpott and Craddock [9] for which convergence was also proved. While this algorithm is not quite a special case of the algorithm in this paper, it is essentially the algorithm obtained by replacing step 3 in the algorithm of section 3 by patching together using $\varepsilon = 1$ and by doing nothing in the general step 4. Thus the two algorithms would coincide when it is optimal to patch together using $\varepsilon = 1$ at every stage. As the algorithm in Philpott and Craddock [9] does not always patch together optimally, we would expect the algorithm in this paper to perform better in the sense of needing fewer iterations. However, as observed in the numerical results in [9], one implementation of the algorithm in this paper leads to very large discretizations which, therefore, take a very long time to solve. Consequently this implementation did not compare favorably with the algorithm in [9]. It is thus desirable to find an operation to include in step 4 that will tend to reduce the size of the partitions. Given such an operation, it would then be plausible that we would obtain better numerical results than those in [9]. The preliminary results obtained by including a purification step in step 4, as mentioned above, do appear to indicate that such an operation is possible.

REFERENCES

[1] E. J. Anderson, *A Continuous Model for Job-Shop Scheduling*, Ph.D. thesis, University of Cambridge, Cambridge, UK, 1978.

[2] E. J. Anderson and P. Nash, *Linear Programming in Infinite-Dimensional Spaces*, Wiley-Interscience, Chichester, UK, 1987.

[3] E. J. Anderson, P. Nash, and A. F. Perold, *Some properties of a class of continuous linear programs*, SIAM J. Control Optim., 21 (1983), pp. 758–765.

[4] E. J. Anderson and A. B. Philpott, *A continuous-time network simplex algorithm*, Networks, 19 (1989), pp. 395–425.

[5] E. J. Anderson and A. B. Philpott, *On the solutions of a class of continuous linear programs*, SIAM J. Control Optim., 32 (1994), pp. 1289–1296.

[6] E. J. Anderson and M. C. Pullan, *Purification for separated continuous linear programs*, Z. Oper. Res., 43 (1996), pp. 3–33.

[7] K. M. Anstreicher, *Generation of Feasible Descent Directions in Continuous Time Linear Programming*, Tech. Report SOL 83-18, Department of Operations Research, Stanford University, Stanford, CA, 1983.

[8] R. E. Bellman, *Bottleneck problems and dynamic programming*, Proc. Nat. Acad. Sci. U.S.A., 39 (1953), pp. 947–951.

[9] A. B. Philpott and M. Craddock, *An adaptive discretization algorithm for a class of continuous network programs*, Networks, 26 (1995), pp. 1–11.

[10] A. F. Perold, *Fundamentals of a Continuous Time Simplex Method*, Tech. Report SOL 78-26, Department of Operations Research, Stanford University, Stanford, CA, 1978.

[11] A. F. Perold, *On a Continuous Time Simplex Method* I: *Local Basis Change*, Tech. Report, Graduate School of Business Administration, Harvard University, Boston, MA, 1982.

[12] M. C. Pullan, *An algorithm for a class of continuous linear programs*, SIAM J. Control Optim., 31 (1993), pp. 1558–1577.

[13] M. C. Pullan, *Forms of optimal solutions for separated continuous linear programs*, SIAM J. Control Optim., 33 (1995), pp. 1952–1977.

[14] M. C. Pullan, *A duality theory for separated continuous linear programs*, SIAM J. Control Optim., 34 (1996), pp. 931–965.

[15] A. Segall, *The modeling of adaptive routing in data-communications networks*, IEEE Trans. Comm., 25 (1977), pp. 85–95.

[16] G. Weiss, *On optimal draining of re-entrant fluid lines*, in Stochastic Networks, IMA Vol. Math. Appl., F. Kelly and R. Williams, eds., Springer-Verlag, New York, 1996, pp. 93–105.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 732–749

REGULAR CASTAING REPRESENTATIONS OF MULTIFUNCTIONS WITH APPLICATIONS TO STOCHASTIC PROGRAMMING[*]

DARINKA DENTCHEVA[†]

Abstract. We consider set-valued mappings defined on a topological space with closed convex images in $\mathbb{R}^n$. The measurability of a multifunction is characterized by the existence of a Castaing representation for it: a countable set of measurable selections that pointwise fills up the graph of the multifunction. Our aim is to construct a Castaing representation which inherits the regularity properties of the multifunction. The construction uses Steiner points. A notion of a generalized Steiner point is introduced. A Castaing representation called regular is defined by using generalized Steiner selections. All selections are measurable, continuous, resp., Hölder-continuous, or directionally differentiable, if the multifunction has the corresponding properties. The results are applied to various multifunctions arising in stochastic programming. In particular, statements about the asymptotic behavior of measurable selections of solution sets via the delta-method are obtained.

Key words. Steiner center, selections, Castaing representation, stochastic programs, random sets, delta-theorems

AMS subject classifications. 54C65, 28B20, 90C15

PII. S1052623498341454

1. Introduction.

Analysis of the behavior of multifunctions includes questions on existence of selections with some regularity properties. When measurability plays a role, one of the most celebrated results is the Castaing representation theorem [7]. It is known (see [22]) that a closed-valued measurable multifunction in a Polish target space admits a measurable selection. Furthermore, for a multifunction F with nonempty closed values in a Polish target space (in our case this will be $\mathbb{R}^n$), we can choose a countable family of measurable selections $\{f_n\}$ that pointwise fills up the values of the multifunction:

$$\text{For each } x \in X,\ F(x) = \operatorname{cl}\Big(\cup_{n=1}^{\infty} f_n(x)\Big).$$

Such a countable family is called a *Castaing representation* for the multifunction. The existence of such a representation characterizes measurability (see [7]). The terminology "Castaing representation" seems to have been introduced in [29]. Besides this survey, there are several publications dealing with regularity properties of multifunctions [3, 8, 21, 30].

Our aim is to construct a Castaing representation of a multifunction $F : X \rightrightarrows \mathbb{R}^n$, defined on a linear metric space X, which inherits its regularity properties. An overview of the basic facts on how selections inherit measurability, Lipschitz-continuity, etc., is given in [3]. The reader also can find there a presentation of some special selections and their properties which are widely studied in the literature. Various results on continuous selections are presented in the recent monograph [27].

[*] Received by the editors July 6, 1998; accepted for publication (in revised form) June 14, 1999; published electronically March 21, 2000.

http://www.siam.org/journals/siopt/10-3/34145.html

[†] Rutgers Center for Operations Research, Rutgers University, 640 Bartholomew Road, Piscataway, NJ 08854-8003 (darina@rutcor.rutgers.edu). Current address: Department of Industrial and Manufacturing Systems, Lehigh University, Bethlehem, PA 18015 (dad8@lehigh.edu).

Although the well-known Steiner selection preserves measurability and continuity properties of a multifunction, its definition does not provide tools for constructing a Castaing representation. We shall generalize the definition of a Steiner center by using an arbitrary probability measure with smooth density on the unit ball. We will obtain different Steiner points with respect to different measures which will be the basis of our construction. All generalized Steiner selections will preserve measurability, continuity, Hölder- or Lipschitz-continuity, and some kind of differentiability.

Several concepts of differentiability of set-valued mappings have been developed in the literature (see, e.g., [3, 4, 28]). We shall work with the notion of semidifferentiability, which was introduced by Penot [25] and corresponds to the concept of tangential approximation due to Shapiro [37, 38]. Semidifferentiability plays an important role in the delta-method, which provides information about the asymptotic behavior of stochastic processes. In particular, mappings containing feasible and optimal solutions of stochastic programs are of this kind. The existence of a differentiable selection has been treated in [14, 9, 11]. In [9] another construction of a Castaing representation is developed that is suitable for applications to delta-theorems. The construction is based on metric projections and it is sufficient for working with the delta-method, but the selections of that Castaing representation do not preserve the regularity properties of the multifunction.

Our results have a specific application to stochastic programming. We shall demonstrate the existence of a regular Castaing representation for various multifunctions arising in stochastic programming.

2. Generalized Steiner points.

In this section, the notion of a generalized Steiner point for a convex compact set is introduced. The notion of Steiner center can be generalized also for some unbounded sets, as shown in [9]. We restrict our investigations to the case of compact sets in order to simplify the presentation; moreover, this situation corresponds to all applications we have in mind.

Let $C \subseteq \mathbb{R}^n$ be a compact convex set. Furthermore, let the Lebesgue measure of the unit ball $\mathbb{B}$ in $\mathbb{R}^n$ be denoted by $\mathcal{V}$ and its surface area by $\mathcal{S}$, i.e.,

$$\mathcal{V} = \frac{\pi^{n/2}}{\Gamma(1+\frac{n}{2})}, \qquad \mathcal{S} = n\mathcal{V} = \frac{2\pi^{n/2}}{\Gamma(\frac{n}{2})}.$$

The surface area $\mathcal{S}$ is computed with the n-dimensional spherical Lebesgue measure (used, for example, as in [5, p. 187, Exercise 13.12]).

Definition 2.1. *The Steiner center* $\mathrm{s}(C)$ *is defined in the following way:*

$$\mathrm{s}(C) = \frac{1}{\mathcal{V}} \int\limits_{\Sigma} p\sigma(p, C)\ \omega(dp), \tag{2.1}$$

where Σ *denotes the unit sphere in* $\mathbb{R}^n$, ω *is the Lebesgue measure on* Σ, *and* $\sigma(\cdot, C)$ *is the support function of* C.

Recall that the support function $\sigma(\cdot, C) : \mathbb{R}^n \to \mathbb{R}$ of a closed convex set $C \subseteq \mathbb{R}^n$ is defined by $\sigma(p, C) = \sup_{y\in C}\langle p, y\rangle$.

This point was first introduced by Steiner [41] in 1840 for a C^2-convex plane curve as the barycenter of the curvature measure. A definition using normalized isometry-invariant measure was introduced by Shephard [40]. The properties of the Steiner center have been widely investigated in the literature. We refer to the monograph [36], where the interested reader can find several facts and references on this topic.

It is easy to see that changing the measure in the formula above could easily lead to obtaining points that do not belong to the set C. However, there is another representation of the Steiner point, which we shall use. Following [3], we use the notation $\partial\sigma(p,C) = \{y \in C : \langle p, y\rangle = \sigma(p,C)\}$ for the subdifferential of the support function and $m(\partial\sigma(p,C))$ for the norm-minimal element in it. The Steiner center can be expressed equivalently as follows:

$$\mathrm{s}(C) = \frac{1}{\mathcal{V}} \int\limits_{\mathbb{B}} m(\partial\sigma(p,C))dp. \tag{2.2}$$

Let μ denote the normalized Lebesgue measure on $\mathbb{B}$, i.e., $d\mu = \frac{dp}{\mathcal{V}}$. We define the set

$$\mathcal{M} = \{\alpha : \text{ probability measure on } \mathbb{B} \text{ having } C^1 - \text{density with respect to } \mu\}.$$

Definition 2.2. *A generalized Steiner center* $\mathrm{St}_\alpha(C)$ *of a compact convex set* $C \subseteq \mathbb{R}^n$ *with respect to the measure* $\alpha \in \mathcal{M}$ *is defined as follows:*

$$\mathrm{St}_\alpha(C) = \int\limits_{\mathbb{B}} m(\partial\sigma(p,C))\alpha(dp). \tag{2.3}$$

It is well-known that

(1) $\mathrm{s}(C) \in C$ for all compact convex sets $C \subseteq \mathbb{R}^n$;

(2) $\mathrm{s}(aA + bB) = a\mathrm{s}(A) + b\mathrm{s}(B)$ for any real numbers a and b and any compact convex sets A and B.

We shall show that this is true for the generalized Steiner points, too. Let $\nabla f(x)$ denote the gradient of f calculated at x. In order to show some regularity of the generalized Steiner points a representation using only the values of the support function instead of its subdifferential is of interest. The equivalence of the two representations (2.1) and (2.2) known for the Steiner center holds only for isometry-invariant measures and, therefore, we cannot simply change the measure in (2.1). The following equivalent representation can be obtained by using a divergence theorem applied to the Moreau–Yosida approximation of the support function $\sigma(p,C)$ of C.

Theorem 2.3 (see [9]). *It holds that for any convex compact set* C *and probability measure* $\alpha \in \mathcal{M}$ *with a density* $\theta(\cdot)$

$$\mathrm{St}_\alpha(C) = \frac{1}{\mathcal{V}}\left[\int\limits_{\Sigma} p\sigma(p,C)\theta(p)\omega(dp) - \int\limits_{\mathbb{B}} \sigma(p,C)\nabla\theta(p)dp\right]. \tag{2.4}$$

The point $\mathrm{St}_\alpha(C)$ *belongs to* C *and* $\mathrm{St}_\alpha(aA+bB) = a\mathrm{St}_\alpha(A) + b\mathrm{St}_\alpha(B)$ *for any real numbers* a *and* b *and any compact convex sets* A *and* B.

Throughout the paper we denote the Hausdorff distance between two sets $A, B \subseteq \mathbb{R}^n$ by

$$d_{\mathrm{H}}(A,B) = \max\{e(A,B), e(B,A)\}, \text{ and } e(A,B) = \sup_{y\in A} d(y,B),$$

where $d(\cdot, A)$ denotes the distance function associated with a closed set $A \subseteq \mathbb{R}^n$.

Proposition 2.4. *The mapping* $\mathrm{St}_\alpha(\cdot)$ *is Lipschitzian with respect to the Hausdorff distance with a Lipschitz constant*

$$\hat{L} = \left(n \max_{p\in\Sigma} \theta(p) + \max_{p\in\mathbb{B}} \|\nabla\theta(p)\|\right),$$

where $\theta(\cdot)$ stands for the density of the probability measure $\alpha \in \mathcal{M}$.

Proof. Let us recall that for every $p \in \Sigma$ and for all nonempty, convex, compact sets A and B it holds that $|\sigma(p, A) - \sigma(p, B)| \leq d_{\mathrm{H}}(A, B)$. We deduce the following chain of inequalities:

$$\begin{aligned}
&\|\mathrm{St}_\alpha(A) - \mathrm{St}_\alpha(B)\| \\
&= \Bigg\| \tfrac{1}{\mathcal{V}} \left[\int_\Sigma p\sigma(p, A)\theta(p)\omega(dp) - \int_{\mathbb{B}} \sigma(p, A)\nabla\theta(p)dp \right] \\
&- \tfrac{1}{\mathcal{V}} \left[\int_\Sigma p\sigma(p, B)\theta(p)\omega(dp) - \int_{\mathbb{B}} \sigma(p, B)\nabla\theta(p)dp \right] \Bigg\| \\
&\leq \tfrac{1}{\mathcal{V}} \left[\int_\Sigma p|\sigma(p, A) - \sigma(p, B)|\theta(p)\omega(dp) + \int_{\mathbb{B}} |\sigma(p, A) - \sigma(p, B)|\nabla\theta(p)dp \right] \\
&\leq \tfrac{1}{\mathcal{V}} \left[\int_\Sigma p d_{\mathrm{H}}(A, B)\theta(p)\omega(dp) + \int_{\mathbb{B}} d_{\mathrm{H}}(A, B)\nabla\theta(p)dp \right] \\
&\leq d_{\mathrm{H}}(A, B)(n \max\nolimits_{p\in\Sigma} \theta(p) + \max\nolimits_{p\in\mathbb{B}} \|\nabla\theta(p)\|) = \hat{L} d_{\mathrm{H}}(A, B). \qquad \square
\end{aligned}$$

3. Measurability and Castaing representations.

Let the space X be equipped with a σ-algebra $\mathcal{A}$. We use the following definition of measurability (see also [3, 8]).

DEFINITION 3.1. *A mapping $f : (X, \mathcal{A}) \to \mathbb{R}^n$ is* measurable *if for any open set $C \subseteq \mathbb{R}^n$ the preimage $f^{-1}(C) = \{x \in X \ : \ f(x) \in C\}$ belongs to $\mathcal{A}$. A multifunction $F : (X, \mathcal{A}) \rightrightarrows \mathbb{R}^n$ is measurable if for any open set $C \subseteq \mathbb{R}^n$ the preimages $F^{-1}(C) = \{x \in X \ : \ F(x) \cap C \neq \emptyset\} \in \mathcal{A}$.*

Recall that $f : (X, \mathcal{A}) \to \mathbb{R}^n$ is called *a measurable selection* of F if f is measurable and $f(x) \in F(x)$ almost surely. It is known (see [22]) that a closed-valued measurable multifunction in a Polish target space admits a measurable selection. Furthermore, for a multifunction F with nonempty closed values in a Polish target space, we can choose a Castaing representation of it—a countable family of measurable selections $\{f_n\}$ such that

$$\text{for each } x \in X,\ F(x) = \mathrm{cl}\left(\cup_{n=1}^\infty f_n(x)\right).$$

The existence of such a representation characterizes measurability (cf., e.g., [8]). In this section we shall construct Castaing representations of a multifunction $F : X \rightrightarrows \mathbb{R}^n$ with convex compact images, which preserves regularity properties of F using generalized Steiner selections.

LEMMA 3.2 (see [9]). *Let C be a convex compact set. The set of generalized Steiner points $D = \{\mathrm{St}_\alpha(C) : \alpha \in \mathcal{M}\}$ is dense in C.*

DEFINITION 3.3. *The function $f_\alpha : X \to \mathbb{R}^n$, defined by $f(x) = \mathrm{St}_\alpha\big(F(x)\big)$, is said to be a generalized Steiner selection of F with respect to the measure α.*

THEOREM 3.4. *Let $F : X \rightrightarrows \mathbb{R}^n$ be a measurable multifunction with nonempty compact convex images. Then F admits a representation by countably many generalized Steiner selections $\{f_n\}$ such that*

$$\textit{for each } x \in X,\ F(x) = \mathrm{cl}\left(\cup_{n=1}^\infty f_n(x)\right).$$

Proof. We consider the set of functions $C_d^1 = \{f \in C^1(\mathbb{B}, \mathbb{R}_+) : \int_{\mathbb{B}} f\mu(dp) = 1\}$. By modification of standard arguments in functional analysis, it can be shown that there is a countable set $\{\theta_i\}_{i=1}^\infty$, which is dense in C_d^1 with respect to the supremum-norm.

Consider the probability measures $\{\alpha_i\}_{i=1}^{\infty}$ with densities $\{\theta_i\}_{i=1}^{\infty}$ on $\mathbb{B}$. We denote the Steiner selection with respect to the measure α_i by f_i. We shall show that the union of selections $\{f_i\}_{i=1}^{\infty}$ is the Castaing representation we are looking for.

Let a point $(x, y) \in \operatorname{graph} F$ and $\delta > 0$ be given. By virtue of Lemma 3.2, there is a measure $\alpha \in \mathcal{M}$ such that $\|\mathrm{St}_\alpha\big(F(x)\big) - y\| \le \frac{1}{2}\delta$. Let θ be the density of this measure. Further, we set $\kappa := \max_{y\in F(x)} \|y\|$. There exists a density θ_δ such that $\sup_{y\in\mathbb{B}} |\theta(y) - \theta_\delta(y)| \le \frac{\delta}{2\kappa}$. Taking the Steiner point with respect to the measure α_δ with this density, we obtain

$$\begin{aligned}\|\mathrm{St}_\alpha\big(F(x)\big) - \mathrm{St}_{\alpha_\delta}\big(F(x)\big)\| &\le \|\int_{\mathbb{B}} m(\partial\sigma(p, C))\big(\theta(p) - \theta_\delta(p)\big)\mu(dp)\| \\ &\le \tfrac{\delta}{2\kappa}\kappa \int_{\mathbb{B}} \mu(dp) = \tfrac{1}{2}\delta.\end{aligned}$$

Consequently, $\|\mathrm{St}_{\alpha_\delta}\big(F(x)\big) - y\| \le \delta$ and this proves the assertion since δ is arbitrary. □

4. Regularity properties of multifunctions and their generalized Steiner selections.

The goal of this section is to show that the representation constructed in Theorem 3.4 preserves regularity properties of the multifunction. We shall show that all selections are measurable, continuous, Hölder- or Lipschitz-continuous, or directionally differentiable whenever the multifunction is so.

Suppose that X is a metric space with a metric ρ. We shall use the following notions of continuity for multifunctions.

A multifunction $F : X \rightrightarrows \mathbb{R}^n$ is called *continuous* at a point $\bar{x}$ if

$$\text{for all } \varepsilon > 0 \text{ there is a } \delta > 0 \text{ such that } d_{\mathrm{H}}(F(x), F(\bar{x})) \le \varepsilon \text{ for all } x : \rho(\bar{x}, x) < \delta.$$

Furthermore, a multifunction is called *Hölder-continuous* of order k around $\bar{x} \in X$ if there exist a constant L and a neighborhood U of $\bar{x}$ such that

$$d_{\mathrm{H}}(F(x_1), F(x_2)) \le L\rho(x_1, x_2)^k \text{ for all } x_1, x_2 \in U,$$

If $k = 1$, then the multifunction is called *Lipschitz-continuous* at this point.

A multifunction will be called *Hölder-stable* of order k at $\bar{x} \in X$ if there exist a constant L and a neighborhood U of $\bar{x}$ such that

$$d_{\mathrm{H}}(F(x), F(\bar{x})) \le L\rho(x, \bar{x})^k \text{ for all } x \in U.$$

If $k = 1$, then such a multifunction is called *Lipschitz-stable* at that point.

From now on we assume that the multifunction F under consideration has nonempty compact convex images.

THEOREM 4.1. *Let a multifunction F be continuous, resp., Hölder-continuous, or Hölder-stable of order k at a point $\bar{x}$ with a constant L. Then each generalized Steiner selection f_α is continuous, resp., Hölder-continuous, or Hölder-stable of order k at this point with a constant:*

$$\hat{L} = \left(n \max_{p\in\Sigma} \theta(p) + \max_{p\in\mathbb{B}} \|\nabla\theta(p)\|\right) L,$$

where θ is the density of the measure α. Moreover, all generalized Steiner selections are measurable whenever F is measurable.

Proof. Let us observe that a generalized Steiner selection f_α is a composition of two mappings: $\mathrm{St}_\alpha \circ F$. The assertion follows by virtue of Proposition 2.4. □

Results about existence of Lipschitz-continuous selections are given in [2, 3, 9, 11], including the case of $F(x)$ being unbounded sets. An interesting result on existence of a Lipschitz-continuous selection through any given point of the graph of the multifunction is contained in [11].

The Hölder-continuity of the generalized Steiner selections can be extended to multifunctions with unbounded images in the same way as [3] or [9]. We do not provide those considerations in order to concentrate on the main goal of this paper: the existence of a regular Castaing representation.

Let us now discuss the relation between differentiability of a multifunction and its generalized Steiner selections. For the purpose of this investigation we need to assume that X is a linear metric space. We denote the graph of F by $\operatorname{graph} F$.

The following notions of differentiability of set-valued mappings will be used.

DEFINITION 4.2. *A mapping $F : X \rightrightarrows \mathbb{R}^n$ is called radially differentiable at a point $(\bar{x}, \bar{y}) \in \operatorname{graph} F$ in direction $h \in X$ if the limit*

$$F'(\bar{x}, \bar{y}; h) = \lim_{t_n \downarrow 0} t_n^{-1}[F(\bar{x} + t_n h) - \bar{y}]$$

exists in the sense of Kuratowski–Painlevé convergence.

Recall that

$$\liminf_{n\to\infty} A_n = \left\{ z \ : \ \limsup_{n\to\infty} d(z, A_n) = 0 \right\}, \quad \limsup_{n\to\infty} A_n = \left\{ z \ : \ \liminf_{n\to\infty} d(z, A_n) = 0 \right\}.$$

A sequence of closed sets $\{A_n\}$, $A_n \subseteq \mathbb{R}^n$, converges to some $A \subseteq \mathbb{R}^n$ in the sense of Kuratowski–Painlevé if and only if the sequence of distance functions converges pointwise (cf. [3]), i.e.,

$$A = \lim_{n\to\infty} A_n \text{ if and only if } d(y, A) = \lim_{n\to\infty} d(y, A_n)$$

or, equivalently,

$$\liminf_{n\to\infty} A_n = A = \limsup_{n\to\infty} A_n.$$

DEFINITION 4.3 (see [25]). *A mapping $F : X \rightrightarrows \mathbb{R}^n$ is called semidifferentiable at a point $(\bar{x}, \bar{y}) \in \operatorname{graph} F$ in direction $h \in X$ if for any sequence $h_n \to h$ the limit*

$$DF(\bar{x}, \bar{y}; h) = \lim_{t_n \downarrow 0, h_n \to h} t_n^{-1}[F(\bar{x} + t_n h_n) - \bar{y}]$$

exists in the sense of Kuratowski–Painlevé.

Various differentiability concepts are compared in [4, 28]. Semidifferentiability generates a derivative that forms a continuous multifunction with respect to the direction (see [4]), i.e., $\lim_{h_n \to h} DF(x, y; h_n) = DF(x, y; h)$, where the limit is taken with respect to the Kuratowski–Painlevé convergence. The derivatives above build some cone-approximation of the graph of the multifunction. Continuous tangential approximations of set-valued mappings are considered also in [37, 38]. It has been shown in [4] that such tangential approximations, if they exist, coincide with the semiderivatives.

THEOREM 4.4 (see [9]). *Suppose that a multifunction $F : X \rightrightarrows \mathbb{R}^n$ is Lipschitz-stable at all $x \in X$ and semidifferentiable at all points (x, y) such that $y \in \operatorname{bd} F(x)$. Here* bd *stands for the boundary of $F(x)$. Let $F(x)$ be polyhedra for all $x \in X$. Then* *the generalized Steiner selection f of F is Hadamard directionally differentiable at all points $x \in X$. Moreover, the directional derivative of f is given by the following formula:*

$$(4.1)\ f'(x;h) = \frac{1}{\mathcal{V}}\left[\int_{\Sigma} p\sigma(p, DF(x, y_p; h))\theta(p)\omega(dp) - \int_{\mathbb{B}} \sigma(p, DF(x, y_p; h))\nabla\theta(p)dp\right],$$

where $y_p \in \partial\sigma(p, F(x))$.

Differentiability properties of the classical Steiner selection are investigated in [9, 11, 14].

Corollary 4.5. *Let $F : X \rightrightarrows \mathbb{R}^n$ be Lipschitz-stable, semidifferentiable at any point (x, y) with $y \in \operatorname{bd} F(x)$, and let $F(x)$ be polyhedra for all $x \in X$. Then F admits a Castaing representation by Hadamard directionally differentiable Steiner selections $\{f_n\}$. Moreover, if F is semidifferentiable at $(x, f_n(x))$, then $f'_n(x;h) \in DF(x, f_n(x); h)$ for all $h \in X$.*

Proof. The statement follows from Theorem 3.4 and Theorem 4.4, having in mind that all generalized Steiner selections are measurable by their continuity. In case F is semidifferentiable at $(x, f_n(x))$, the inclusion $f'_n(x;h) \in DF(x, f_n(x); h)$ follows from the definition of the semiderivative. □

Now, we would like to formulate a statement relating the radial differentiability of a set-valued mapping with the existence of a Castaing representation with directionally differentiable selections.

Corollary 4.6. *Suppose that a multifunction $F : X \rightrightarrows \mathbb{R}^n$ is radially differentiable into a direction h at all points $(\bar{x}, y) \in \operatorname{graph} F :\ y \in \operatorname{bd} F(\bar{x})$, $F(x)$ are polyhedra for all $x \in X$, and it satisfies the following condition on Lipschitz behavior: There exist constants $L > 0$ and $\delta > 0$ such that*

$$\text{(LB)} \qquad d_{\mathrm{H}}(F(\bar{x}), F(\bar{x} + th)) \le Lt \quad \textit{whenever } t \in (0, \delta).$$

Then F admits a Castaing representation by generalized Steiner selections $\{f_n\}$ which are directionally differentiable in the direction h at $\bar{x}$. Moreover, if F is directionally differentiable at $(\bar{x}, f_n(\bar{x}))$, then $f'_n(\bar{x};h) \in F'(\bar{x}, f_n(\bar{x}); h)$, and the directional derivative satisfies formula (4.1)*. If F is Lipschitzian at $\bar{x}$ and directionally differentiable into all directions, then f_n are Hadamard directionally differentiable at $\bar{x}$.*

Proof. Under the assumption (LB), we follow the same line of argument as in the proof of Theorem 4.4, considering all limits for the fixed direction h. In this way, we obtain directional differentiability of all generalized Steiner selections into the direction h at the point $\bar{x}$. Under the stronger assumption that F is Lipschitzian, the proof is the same as the previous corollary. We have to take into account that directional differentiability, together with Lipschitz-continuity, implies semidifferentiability [28]. The formula and the inclusion of the directional derivative follow analogously. □

These statements are of interest when dealing with the delta-method as we shall see in the last section.

5. Feasible and optimal solutions of stochastic programs.

5. Feasible and optimal solutions of stochastic programs. In this section we shall discuss some nontrivial applications for the existence of a regular Castaing representation. We apply the results of the previous section to mappings expressing optimal solutions of stochastic programs subjected to perturbations.

While working with stochastic optimization models, one assumes that the underlying probability measure is given. In practical situations this is rarely the case; one usually works with some approximations or statistical estimates. These circumstances motivate the stability investigations of stochastic programs with respect to perturbations of the probability distributions. We shall consider two basic types of stochastic models: stochastic programs with recourse and stochastic programs with probabilistic constraints.

In order to discuss stability with respect to the probability measure, we need to work with a suitable metric space. Let (X, d) be a separable linear normed space and $\mathcal{P}(X)$ be the set of all Borel probability measures on X. We denote

$$\begin{array}{rcl} \mathcal{M}(X) & := & \Big\{\mu \in \mathcal{P}(X) : \int_X d(x,y)\mu(dx) < \infty\Big\}, \\ \mathcal{D}(\mu,\nu) & := & \Big\{\eta \in \mathcal{P}(X \times X) : \eta \circ \pi_1^{-1} = \mu, \eta \circ \pi_2^{-1} = \nu\Big\}, \end{array}$$

using π_1 and π_2 as the canonical first and second projections, respectively. The L_1-Wasserstein metric W_1 is defined as follows:

$$W_1(\mu,\nu) := \inf\Big\{ \int_{X\times X} d(x,y)\eta(dx,dy) : \eta \in \mathcal{D}(\mu,\nu)\Big\} \quad \text{for all } \mu,\nu \in \mathcal{M}(X).$$

Furthermore, let $\|f\|_L$ be the usual Lipschitz-norm:

$$\|f\|_L = \|f\|_\infty + \sup_{x,y\in Y} \frac{|f(x) - f(y)|}{\|x-y\|}.$$

It is known (cf. [15]) that $(\mathcal{M}(X), W_1)$ is a metric space. Quantitative stability of stochastic programs with respect to perturbations of probability measures is investigated in [16, 17, 31, 32, 33, 34]. We shall utilize some of the results presented in those papers.

5.1. Stochastic recourse programs. Let us consider a two-stage stochastic program with linear recourse and random right-hand side:

$$\min\{g(x) + Q_\mu(Ax) : x \in C\}, \tag{5.1}$$

$$Q_\mu(\chi) = \int_{\mathbb{R}^m} \tilde{Q}(\theta - \chi)\mu(d\theta), \tag{5.2}$$

$$\tilde{Q}(z) = \min\{q^\top y : Wy = z, y \geq 0\}, \tag{5.3}$$

where $g : \mathbb{R}^n \to \mathbb{R}$ is a convex function, $C \subseteq \mathbb{R}^n$ is a nonempty closed convex set, and μ is a Borel probability measure on $\mathbb{R}^m$. Furthermore, $q \in \mathbb{R}^s$, A is an $n \times m$ matrix, and W is an $s \times m$ matrix. We make use of the general assumptions (A1)–(A3), which are common in the literature, in order to make the problem well defined.

$$W(\mathbb{R}^s_+) = \mathbb{R}^m \qquad \text{(complete recourse)}, \tag{A1}$$

$$M_D := \{u \in \mathbb{R}^m : W^\top u \leq q\} \neq \emptyset \qquad \text{(dual feasibility)}, \tag{A2}$$

$$\int_{\mathbb{R}^m} \|z\|\mu(\mathrm{d}z) < +\infty \qquad \text{(finite first moment)}. \tag{A3}$$

Having in mind linear programming theory, observe that (A1) and (A2) imply $\tilde{Q}(z)$ to be finite for all $z \in \mathbb{R}^m$. Due to (A3) the integral of $\tilde{Q}(z)$ is also finite [18, 42].

The model is derived from an optimization problem with uncertain data, where some statistical information about the random data is available. The decision x of the first stage has to be made *here and now* before observing some realization of θ. It is supposed to solve the problem

$$\inf\{g(x) : x \in C, Ax = \theta\}.$$

After observing a realization of θ we fix a second-stage decision y (called recourse action) in order to overcome the deviation $\theta - Ax$. The matrix W determines the rule to react and q the costs of our reaction. Assumption (A1) means that we are able to overcome any deviation. To choose y properly, we minimize its costs. To choose x properly, we minimize the sum of the first-stage costs and the expected second-stage costs, caused by the corrective action y. Further details and fundamental properties of two-stage stochastic programs can be found in [18, 26, 42].

We consider the multifunction assigning to each probability measure μ the set of optimal solutions of the problem (5.1), i.e.,

$$\psi(\mu) = \operatorname{argmin}\{g(x) + Q_\mu(Ax) :\ x \in C\}.$$

Two-stage stochastic programs hardly have a unique solution. This fact has motivated the attempt to avoid the assumption on the multifunction to be a singleton at certain points in our investigations. The next example gives an impression on how restrictive this assumption is.

Example (see [33]). $g(x) = 0,\ A = (1,0),\ C = [0,1] \times [0,1],\ q = (1,1),\ W = (1,-1)$. Let μ be the uniform distribution on [–1/2,1/2]. Then

$$\begin{aligned} \psi(\mu) &= \operatorname{argmin}\{Q_\mu(Ax) :\ x \in [0,1] \times [0,1]\} \\ &= \operatorname{argmin}\left\{\int_{\mathbb{R}} |\omega - x_1|\mu(d\omega) :\ x \in [0,1] \times [0,1]\right\} \\ &= \{(0, x_2) : x_2 \in [0,1]\} = kerA \cap C. \end{aligned}$$

One can see that even for very simple examples the solution set is not a singleton. Under an assumption that Q_μ is a strictly, resp., strongly convex function we have the uniqueness of $A\psi(Q_\mu)$, but we cannot expect that $kerA = \{0\}$.

Proposition 5.1. *Let g be a convex quadratic function and C a polyhedron. Given $\mu \in \mathcal{M}(\mathbb{R})$, let $\psi(\mu)$ be nonempty and let the function Q_μ be strongly convex on an open neighborhood of the set $A(\psi(\mu))$. Then the mapping ψ admits a Castaing representation of generalized Steiner selections which are Hölder-stable of order* 1/2 *at the point $\mu \in (\mathcal{M}(X), W_1)$.*

Proof. According to Theorem 2.7 in [31], under the assumption of the theorem, there are constants $L > 0$ and $\delta > 0$ such that

$$d_{\mathrm{H}}(\psi(\mu), \psi(\nu)) \leq L\ W_1(\mu, \nu)^{1/2}$$

whenever $\nu \in \mathcal{M}(\mathbb{R}), W_1(\mu, \nu) < \delta$. Hence, we can apply Theorem 4.1 and conclude that each generalized Steiner selection is Hölder-stable of order 1/2 at the point μ. Consequently, our construction of Theorem 3.4 yields a Castaing representation of ψ with the stated property. □

We consider also general perturbations of the recourse function without referring to metrics for probability measures. The following setting of a perturbed problem is relevant:

$$\inf\{g(x) + Q(Ax) : x \in C\},$$

where $Q : \mathbb{R}^m \to \mathbb{R}$ is a convex function, considered to be a perturbation (resp., approximation) of the expected recourse function Q_μ. Resorting to convex perturbations is motivated by the fact that, given (A1) and (A2), Q_μ is convex for any probability measure with finite first moment (cf. [18, 42]). Then the definition space (X) of the mapping ψ changes to a functional space:

$$\psi(Q) = \operatorname{argmin}\{g(x) + Q(Ax) : \ x \in C\}.$$

Setting $Y = A(C)$, we consider two functional spaces as definition spaces: the space $C^{1,1}(Y, R)$ of all real-valued continuously differentiable functions with locally Lipschitz derivative, defined on Y, and the space $C^{0,1}(Y, \mathbb{R})$ of all real-valued locally Lipschitz functions, defined on Y. Both spaces are metrizable (cf. [10]). We suppose here that the set C is bounded and endow the space $C^{0,1}(Y, \mathbb{R})$ with the usual Lipschitz-norm. We work with the corresponding norm-convergence in $C^{1,1}(Y, \mathbb{R})$.

In the following, we always consider the restriction of the solution set mapping ψ to the cone of convex functions in one of the spaces above. One more piece of notation is that

$$\phi(y) = \operatorname{argmin}\{g(x) : x \in C, Ax = y\} \qquad (y \in Y).$$

PROPOSITION 5.2. *Let $\psi(Q_\mu)$ be nonempty and Q_μ be strongly convex on some open neighborhood of $A\psi(Q_\mu)$. Assume, in addition, that there is a constant $L > 0$ and a neighborhood U of $\bar{y}$ with $\bar{y} = A\psi(Q_\mu)$ such that*

(i) $$d(\phi(\bar{y}), \phi(y)) \leq L\|\bar{y} - y\| \text{ for all } y \in Y \cap U.$$

Then ψ admits a Castaing representation by generalized Steiner selections which are Lipschitz-stable at the point $Q_\mu \in C^{0,1}(Y, \mathbb{R})$. Moreover, if g is linear or convex quadratic and C is a polyhedron, then the assumption (i) *is satisfied.*

Proof. We refer here to Theorem 2.3 and Remark 2.4 in [10]. Under the assumption of the proposition, there are constants $\hat{L} > 0$ and $\delta > 0$ such that

$$d_H(\psi(Q_\mu), \psi(Q)) \leq \hat{L}\|Q_\mu - Q\|_L$$

for any convex function $Q \in C^{0,1}(Y, \mathbb{R})$ such that $\|Q - Q_\mu\|_L < \delta$, which means that the mapping ψ is locally Lipschitz-stable at Q_μ. Consequently, according to Theorem 4.1 each generalized Steiner selection is Lipschitz-stable at that point. Applying the construction of a Castaing representation by Steiner selections according to Theorem 3.4 we accomplish the goal of the proposition. □

A result similar to Theorem 2.3 in [10] is shown in [33]. We can use it and obtain a similar statement to the above proposition. Here we have chosen to present only one of them to illustrate existence of a Castaing representation for the solution set mapping, which has Lipschitz behavior.

Restricting the solution set mapping ψ to the cone K of convex functions in one of the spaces above has an impact on the notions of differentiability. Considering the semiderivative at a point (Q_μ, x) in a certain direction $\bar{v}$, we assume that the arguments of ψ lie in K. Hence, we consider only sequences $v \to \bar{v}$ such that $Q_\mu + tv \in K$ for all v. Consequently, the directions v are elements of the closure of the radial tangent cone to K at the point Q_μ. We denote the radial tangent cone to K at the point Q_μ by

$$T_K^r(Q_\mu) = \{\lambda(Q - Q_\mu) : \ \lambda \geq 0, \ Q \in K\}.$$

Proposition 5.3. *Assume that $\psi : K \subset C^{0,1}(Y,R) \rightrightarrows \mathbb{R}^n$ and that $\psi(Q_\mu)$ is nonempty. Let Q_μ be strictly convex on some open neighborhood of $A(\psi(Q_\mu))$ and twice continuously differentiable at $\chi_* : \ A(\psi(Q_\mu)) = \{\chi_*\}$. Let g be convex quadratic, C be a polyhedron, and $v \in T^r_K(Q_\mu)$. Then ψ is radially differentiable at $(Q_\mu, x) \in$* graph F *in direction v and*

$$\begin{aligned}\psi'(Q_\mu; x)(v) &= \lim_{t\to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - x)\\ &= \operatorname{argmin}\left\{\frac{1}{2}\langle \nabla^2 g(x)y, y\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(Ax)Ay, Ay\rangle + v'(Ax; Ay) : y \in S(x)\right\}.\end{aligned}$$

Moreover, ψ admits a Castaing representation by Steiner selections f_i which are directionally differentiable at Q_μ in the direction v and it holds that

$$f_i'(Q_\mu; v) \in \psi'(Q_\mu; f_i(Q_\mu))(v).$$

Proof. The first statement of the proposition, i.e., the directional differentiability of ψ and the formula of the derivative, is proved by Theorem 4.1 in [10]. The second statement follows from the first by virtue of our Corollary 4.6. □

Now we come to the semidifferentiability of the solution set mapping and its consequences. We consider the restriction of ψ to the space $C^{1,1}(Y,R)$.

Proposition 5.4. *Assume $\psi(Q_\mu)$ to be nonempty, g a quadratic function, and C a polyhedron. Let Q_μ be strongly convex on some open neighborhood of $A(\psi(Q_\mu))$ and twice continuously differentiable at $\chi_* : \ A(\psi(Q_\mu)) = \{\chi_*\}$. Let $x \in \psi(Q_\mu)$. Then ψ admits a Castaing representation by Steiner selections. All selections are Hadamard directionally differentiable at (Q_μ, x), and the directional derivatives of the selections belong to the semiderivative of ψ, which is given by the formula of the previous proposition.*

Proof. The semidifferentiability of ψ and the formula for the semiderivative are proved by Theorem 4.7 in [10]. As in the proof of Proposition 5.2 we obtain that ψ is also Lipschitz-stable at Q_μ. Thus, we can apply Corollary 4.5, which states the existence of the Castaing representation with the desired differentiability property. □

5.2. Stochastic programs with probabilistic constraints. We shall be concerned with the following stochastic problem:

$$\min\{g(x) : x \in \mathbb{R}^n,\ \mu(\{z \in \mathbb{R}^s : x \in H(z)\}) \geq p\}, \tag{5.4}$$

where $g : \mathbb{R}^n \to \mathbb{R}$ is a convex function, $p \in (0,1)$ is a probability (or reliability) level, $\mu \in \mathcal{P}(\mathbb{R}^s)$, and $H : \mathbb{R}^s \to \mathbb{R}^n$ is a measurable mapping. It is assumed that the constraint $x \in H(z)$ is satisfied with a probability p.

Let $\mathcal{B}$ be a subset of the Borel σ-algebra on $\mathbb{R}^s$. The $\mathcal{B}$-discrepancy of two measures is defined by

$$\alpha_{\mathcal{B}}(\mu,\nu) = \sup_{B\in\mathcal{B}} |\mu(B) - \nu(B)|, \qquad \mu,\nu \in \mathcal{P}(\mathbb{R}^s).$$

The preimages $H^{-1}(x) = \{z \in \mathbb{R}^s : x \in H(z)\}$ are Borel sets because H is measurable. Consequently, we can use the subset $\mathcal{B}_H = \{H^{-1}(x), x \in \mathbb{R}^n\}$ as a subset of discrepancy and denote $\bar\alpha := \alpha_{\mathcal{B}_H}$.

A special case of the $\mathcal{B}_H$-discrepancy is the Kolmogorov distance on $\mathcal{P}(\mathbb{R}^s)$ defined by

$$\alpha(\mu,\nu) = \sup_{y\in\mathbb{R}^s} |F_\mu(y) - F_\nu(y)|, \qquad \mu,\nu \in \mathcal{P}(\mathbb{R}^s),$$

where F_μ is the distribution function of μ.

In the setting of the previous section, recourse problems preserve the same set of feasible points when the measure is subjected to perturbations. In the models with probabilistic constraints the solution changes because the feasible set changes when the measure is perturbed. Stability investigations of probabilistically constrained models are mainly concerned with changes that affect the feasible set. The feasible set can be expressed in the following way:

$$\{x \in \mathbb{R}^n : \mu(H^{-1}(x)) \geq p\}. \tag{5.5}$$

Mostly investigated is the case of a mapping H given by linear inequalities, i.e.,

$$H(z) = \{x \in C \,:\, Ax \geq z\}, \qquad z \in \mathbb{R}^s,$$

where A is an $s \times n$-matrix and $C \subseteq \mathbb{R}^n$ is a closed set, often supposed to be a polyhedron. Then we deal with the problem

$$\min\{g(x) : \, x \in C, \, F_\mu(Ax) \geq p\}, \tag{5.6}$$

where F_μ is the distribution function of the probability measure $\mu \in \mathcal{P}(\mathbb{R}^s)$.

We assume μ to be r-concave for some $r \in (-\infty, 0)$. Recall that r-concavity is introduced in the following way. Let the generalized mean function m_r be defined on $\mathbb{R}_+ \times \mathbb{R}_+ \times [0,1]$ as

$$m_r(a,b,\lambda) = \begin{cases} (\lambda a^r + (1-\lambda)b^r)^{1/r} & \text{if } r \neq 0, ab > 0, \\ 0 & \text{if } ab = 0, \\ a^\lambda b^{1-\lambda} & \text{if } r = 0, \\ \max\{a,b\} & \text{if } r = \infty, \\ \min\{a,b\} & \text{if } r = -\infty. \end{cases}$$

The measure $\mu \in \mathcal{P}(\mathbb{R}^s)$ is called r-concave if the inequality $\mu(\lambda B_1 + (1-\lambda)B_2) \geq m_r(\mu(B_1), \mu(B_2), \lambda)$ holds for all $\lambda \in [0,1]$ and all Borel subsets B_1, B_2 of $\mathbb{R}^s$ such that $\lambda B_1 + (1-\lambda)B_2$ is a Borel set.

Due to r-concavity of μ, the problem (5.6) represents a convex program.

We shall consider the following mapping $\Phi : \mathcal{P}(\mathbb{R}^s) \times (0,1) \to \mathbb{R}^n$ defined by setting

$$\Phi(\mu, p) := \{x \in C : p - F_\mu(Ax) \leq 0\}.$$

PROPOSITION 5.5. *Assume that μ is r-concave and C is a convex compact set. Suppose that the mapping $\Phi(\mu,\cdot)$ is Lipschitzian at a certain point p_0. Then Φ has a Castaing representation by generalized Steiner selections $\{f_i\}$ such that there exist constants $\delta > 0$ and $L_i > 0$, and it holds that*

$$|f_i(\nu, p_0) - f_i(\mu, p_0)| \leq L_i \bar{\alpha}(\nu, \mu) \tag{5.7}$$

whenever $\bar{\alpha}(\nu,\mu) \leq \delta$.

Proof. The set of feasible points is convex and compact under the assumptions of the proposition. Hence, the Steiner points are well defined. In Proposition 5.3 of [32] a kind of pseudo-Lipschitzian behavior is shown for Φ under local assumptions on $\Phi(\mu, \cdot)$. Applying this result we obtain that for all $x \in \Phi(\mu, p_0)$ there is a neigborhood V_x and $\delta_x > 0$, $L_x > O$ such that

$$d_{\mathrm{H}}(\Phi(\nu, p_0) \cap V_x, \Phi(\mu, p_0) \cap V_x) \leq L_x \bar{\alpha}(\nu, \mu) \text{ for all } \mu \text{ and } \nu \text{ such that } \bar{\alpha}(\nu, \mu) \leq \delta_x.$$

The set $\Phi(\mu, p_0)$ is compact; therefore, we can choose a finite number of those neighborhoods that cover the whole feasible set $\Phi(\mu, p_0)$. Let us denote these neighborhoods by $V_1, V_2, \ldots, V_k$ and the corresponding constants by $\delta_1, \delta_2, \ldots, \delta_k$, resp., $\bar{L}_1, \bar{L}_2, \ldots, \bar{L}_k$. We set $L = \max_i \bar{L}_i$ and $\delta = \min_i \delta_i$ for $i = 1, \ldots, k$. Then for each $x \in \Phi(\mu, p_0)$ let $x \in V_j$ for some $j \in \{1, 2, \ldots, k\}$. We have

$$d(x, \Phi(\nu, p_0)) \leq d(x, \Phi(\nu, p_0) \cap V_j) \leq d_{\mathrm{H}}(\Phi(\nu, p_0) \cap V_j, \Phi(\mu, p_0) \cap V_j) \leq L \bar{\alpha}(\nu, \mu)$$

whenever $\bar{\alpha}(\nu, \mu) \leq \delta$. In the same way we obtain that for all $x \in \Phi(\nu, p_0)$, it holds that

$$d(x, \Phi(\mu, p_0)) \leq L \bar{\alpha}(\nu, \mu)$$

whenever $\bar{\alpha}(\nu, \mu) \leq \delta$. The latter two inequalities imply that

$$d_{\mathrm{H}}(\Phi(\nu, p_0), \Phi(\mu, p_0)) \leq L \bar{\alpha}(\nu, \mu)$$

whenever $\bar{\alpha}(\nu, \mu) \leq \delta$. Then, following the proof of Theorem 4.1, we can show that the relation (5.7) is satisfied for each generalized Steiner selection. Applying our usual technique of Theorem 3.4 we obtain the assertion. ∎

Determining the probability level p is a significant modeling decision. Therefore, it is natural to investigate changes of the feasible set when this level changes.

PROPOSITION 5.6. *Let μ be r-concave and its distribution function F_μ be locally Lipschitzian. Furthermore, let p_0 be a given probability level and C be a convex compact set. Assume that for all $x \in \Phi(\mu, p_0)$ it holds that if $F_\mu(Ax) = p_0$, then the Clarke subdifferential of $F_\mu(A\cdot)$ at x and the normal cone to C at x have an empty intersection. Then $\Phi(\mu, \cdot)$ has a Castaing representation by generalized Steiner selections which are Lipschitzian at p_0.*

Proof. The set of feasible points is convex and compact under the assumptions of the proposition. Therefore, the Steiner points are well defined. Furthermore, we can apply Proposition 2.1 in [34] and obtain that $\Phi(\mu, \cdot)$ is pseudo-Lipschitzian at (x, p_0) for any $x \in \Phi(\mu, p_0)$. Since the images $\Phi(\mu, p)$ are compact, it follows as in the proof of the previous proposition that $\Phi(\mu, \cdot)$ is Lipschitzian at those points. Consequently, according to Theorem 4.1 each generalized Steiner selection is Lipschitz-continuous at p_0. Applying the construction of a Castaing representation by Steiner selections according to Theorem 3.4, we accomplish our goal. ∎

Now, we focus our attention on sets of optimal solutions. Following the notation of the previous section, we understand that $\psi(\mu)$ designates the set of global solutions to (5.6), and $\psi_U(\nu)$ refers to the localized solution set of this problem, where $\nu \in \mathcal{P}(\mathbb{R}^s)$ is a perturbation of μ and $U \subseteq \mathbb{R}^n$ is a neighborhood of $\psi(\mu)$.

PROPOSITION 5.7. *Assume that*

(i) *$\psi(\mu)$ is nonempty and bounded;*

(ii) *$\psi(\mu) \cap \operatorname{argmin}\{g(x) : x \in C\} = \emptyset$;*

(iii) *there is $\bar{x} \in C : F_\mu(A\bar{x}) > p$ (Slater condition);*

(iv) *F_μ^r is strongly convex on some open convex neighborhood V of $A\psi(\mu)$, where $r \in (-\infty, 0)$ is chosen such that μ is r-concave.*

Then there exist a neighborhood U of $\psi(\mu)$ and $\delta > 0$ such that setting $\hat{\psi} : \mathcal{U} \to \mathbb{R}^n$ as $\hat{\psi}(\nu) = \psi_U(\nu)$, where $\mathcal{U} = \{\nu \in \mathcal{P}(\mathbb{R}^s) : \alpha(\mu, \nu) < \delta\}$; it holds that the mapping $\hat{\psi}$ admits a Castaing representation by Steiner selections which are Hölder-stable of order $1/2$ at μ.

Proof. We apply Theorem 4.3 of [17]. Under the assumption of the proposition, there are constants $L > 0$, $\delta > 0$ and some neighborhood U of $\psi(\mu)$ such that

$$d_{\mathrm{H}}(\psi(\mu), \psi_U(\nu)) \leq L\alpha(\mu, \nu)^{1/2} \tag{5.8}$$

for any probability measure $\nu \in \mathcal{U}$. Using the notation $\hat{\psi}$ for the restriction of the solution set mapping to the mapping of local minimizers the above inequality means that $\hat{\psi}$ is locally Hölder-stable of order $1/2$ at μ. Consequently, according to Theorem 4.1 each generalized Steiner selection is locally Hölder-stable of order $1/2$ at that point. Applying the construction of a Castaing representation by Steiner selections we obtain the result. □

The assumptions of the above proposition are commented on in [16] and illustrated by examples. Condition (i) is satisfied, for example, if C is a polytope. The conditions (ii) and (iii) mean that the probability level p is not chosen too low and too high, respectively. From the modeling point of view both conditions show the significance of the choice of the reliability level p. Assumption (iv) is decisive for obtaining a growth condition of the objective function around the original solution set.

As a conclusion, following [16], we formulate a large deviation result for the selections of the constructed Castaing representation when estimating μ by empirical measures. Let $\xi_1, \xi_2, \ldots, \xi_n, \ldots$ be independent identically distributed $\mathbb{R}^s$-valued random variables having common distribution μ, and let $\mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{\xi_i}$ denote the empirical measure of $\xi_1, \xi_2, \ldots, \xi_n$.

Corollary 5.8. *Under the conditions of the previous proposition, let L and δ be the constants involved in the inequality (5.8) and the statement. Let $\hat{L_j} = (n \max_{p\in\mathbb{B}} \theta_j(p) + \max_{p\in\mathbb{B}} \|\nabla\theta_j(p)\|)L$, where θ_j designates the density of the jth measure applied to calculate the generalized Steiner points. Then for any selection f_j of the Castaing representation of $\hat{\psi}$ and all $\varepsilon > 0$ it holds that*

$$\limsup_{n\to\infty} \frac{1}{n} \log P(\|f_j(\mu) - f_j(\mu_n)\| \geq \varepsilon) \leq -2\min\left\{\delta^2, \varepsilon^4 \hat{L_j}^{-4}\right\}.$$

Proof. According to Theorem 4.1, the generalized Steiner selections f_j will have a constant $\hat{L_j} = (n \max_{p\in\mathbb{B}} \theta_j(p) + \max_{p\in\mathbb{B}} \|\nabla\theta_j(p)\|)L$ of Hölder-stability. The assertion follows from Corollary 4.29 in [16], the construction of the Castaing representation, and Theorem 4.1. □

6. Asymptotic behavior of random sets.

One way to obtain information about the asymptotic behavior of random elements is the so-called delta-method. Delta-theorems are concerned with the asymptotic distribution of functions of random elements, when those elements satisfy a central limit formula.

Theorem 6.1 (see [39]). *Let $f : (X, \mathcal{B}(X)) \to \mathbb{R}^n$ be measurable and Hadamard directionally differentiable at some point $\bar{x} \in X$. Suppose that X is a Banach space and $t_n(x_n - \bar{x})$ are some random elements of X converging in distribution to some* *element h, written*

$$t_n^{-1}(x_n - \bar{x}) \xrightarrow{D} h,$$

while $t_n \downarrow 0$ and h is a random element in some separable subspace of X. Then

$$t_n^{-1}(f(x_n) - f(\bar{x})) \xrightarrow{D} f'(\bar{x}; h).$$

Here $\xrightarrow{D}$ denotes convergence in distribution. Recall that convergence in distribution of a sequence of random elements $x_n : (\Omega, \mathcal{A}, P) \to X$ means the weak* convergence of the measures $\mu_n = P \circ x_n^{-1}$ that these elements induce on the space X. A sequence of probability measures μ_n on a metric space X weakly* converges to μ (cf. [6]) if

$$\lim_{n\to\infty} \int g(x)\ \mu_n(dx) = \int g(x)\ \mu(dx)$$

for all bounded uniformly continuous functions $g : X \to \mathbb{R}$.

Convergence in distribution of set-valued mappings is considered in [35]. The first generalized delta-theorem for set-valued mappings was formulated by King [19]. It is the following statement.

THEOREM 6.2 (see [19]). *Let $F : (X, \mathcal{B}(X)) \rightrightarrows \mathbb{R}^n$ be a closed-valued measurable multifunction defined on a separable complete metric space X. Suppose that x_n satisfy a generalized central limit formula with limit $\bar{x}$, i.e., there is a sequence $\{t_n\}, t_n \geq 0$, monotonically decreasing to 0 and a limit element h such that*

$$t_n^{-1}(x_n - \bar{x}) \xrightarrow{D} h$$

as random variables in X. Assume, additionally, that F is almost surely semidifferentiable at $(\bar{x}, \bar{y})$ for some $\bar{y} \in F(\bar{x})$ with respect to the measure μ induced by h. Then $F(x_n)$ satisfy the generalized central limit formula

$$t_n^{-1}(F(x_n) - \bar{y}) \xrightarrow{D} DF(\bar{x}, \bar{y}; h)$$

as random closed sets in $\mathbb{R}^n$ or, equivalently,

$$d(\cdot, t_n^{-1}[F(x_n) - \bar{y}]) \xrightarrow{D} d(\cdot, DF(\bar{x}, \bar{y}; h))$$

as stochastic processes on $\mathbb{R}^n$.

Here semidifferentiability almost surely means that the convergence of the differential quotients holds for all directions, except for a set of μ-measure 0.

In general, the distribution of a random set does not determine the distributions of its measurable selections (cf., e.g., [1]). The results of this section will contribute to the investigations of this matter.

COROLLARY 6.3. *Assume that the random elements $x_n \in X$ satisfy a generalized central limit formula with limit $\bar{x}$, i.e.,*

$$t_n^{-1}(x_n - \bar{x}) \xrightarrow{D} h$$

as random variables in X, where $t_n \downarrow 0$. In addition to the assumptions of Theorem 4.4, suppose that F is semidifferentiable at all points $(\bar{x}, y) \in$ graph F. Then *for any point $\bar{y} \in F(\bar{x})$, the random sets $F(x_n)$ satisfy the generalized central limit formula*

$$t_n^{-1}(F(x_n) - \bar{y}) \xrightarrow{D} DF(\bar{x}, \bar{y}; h)$$

and F admits a Castaing representation $\{f_k\}$ by generalized Steiner selections such that all f_k satisfy the generalized central limit formula

$$t_n^{-1}[f_k(x_n) - f_k(\bar{x})] \xrightarrow{D} f_k'(\bar{x}; h) \in DF(\bar{x}, f_k(\bar{x}); h).$$

Proof. The proof follows from Theorem 6.1, Theorem 6.2, and Corollary 4.5. □

Let us return again to the solution set mapping of the recourse problem, which assigns to each approximation $Q \in C^{1,1}(Y, R)$ of the recourse function the set of optimal solutions of the approximate problem.

Supposed we have some approximations (resp., estimates) $Q_n,\ n = 1, 2, \ldots,$ of Q_μ that satisfy a generalized central limit formula in the above functional space. The application of our investigations leads to the following consequences for the delta-method.

COROLLARY 6.4. *Assume the conditions of Proposition* 5.4. *Suppose that $Q_n,\ n = 1, 2, \ldots,$ satisfy the functional central limit formula*

$$t_n^{-1}[Q_n - Q_\mu] \xrightarrow{D} \zeta \quad in \quad C^{1,1}(D, R)$$

for some monotonically decreasing sequence $t_n \downarrow 0$. Given a point $\bar{x} \in \psi(Q_\mu)$, then ψ satisfies the generalized central limit formula

$$t_n^{-1}[\psi(Q_n) - \bar{x}] \xrightarrow{D} D\psi(Q_\mu, \bar{x}; \zeta)$$

as random sets in $\mathcal{F}(\mathbb{R}^n)$. Moreover, ψ admits a Castaing representation $\{f_i\}_{i=1}^{\infty}$ of Steiner selections such that all f_i satisfy the central limit formula

$$t_n^{-1}[f_i(Q_n) - f_i(Q_\mu)] \xrightarrow{D} f_i'(Q_\mu; \zeta) \in D\psi(Q_\mu, f_i(Q_\mu); \zeta)$$

as random variables on $\mathbb{R}^n$.

Proof. The assertion follows by Corollary 6.4 and Proposition 5.4. □

Investigating the asymptotic behavior of solution sets of stochastic programs is beyond the scope of this paper. The last statements have been included for the sake of giving an application of the results of this paper and yielding nontrivial statements. For investigations on the asymptotic behavior of stochastic programs the interested reader is referred to [13, 24, 20, 39] and the references therein.

Let us mention some of the results published on convergence in distribution of measurable selections of multifunctions. Interesting results are given in [1] by Artstein in a different setting. The primary object there is a given probability distribution on some compact subset of a complete separable metric space. The problem of which distributions on the space are induced by selections of random sets with the given probability distribution is investigated. Relevant results are given by King [19] and Lachout [23]. In Theorem 4.3 in [19] a generalized central limit formula for all measurable selections is established under the assumption that the multifunction is upper Lipschitzian, $F(\bar{x}) = \{\bar{y}\}$, and $DF(\bar{x}, \bar{y}; h)$ is single-valued almost everywhere. In [23], the values $F(x)$ are supposed to be compact and $F(\bar{x}) = \{\bar{y}\}$ to be a singleton. The statement is that the measurable selections f of F do not satisfy the central limit formula themselves, but there are subsequences for which the formula holds. Those assumptions, in particular the assumption about $F(x)$ being singleton, are too strong for the applications we were aiming at. As mentioned, stochastic programs very seldom have unique solutions and, therefore, we are interested in statements that are applicable to solution sets.

Acknowledgments. The author wishes to express her gratitude to Werner Römisch for many helpful suggestions and encouragement while this work was in progress. Thanks are due to the referees for their constructive remarks.

REFERENCES

[1] Z. Artstein, *Distributions of random sets and random selections*, Israel J. Math., 46 (1984), pp. 313–324.

[2] J.-P. Aubin and A.Cellina, *Differential Inclusions. Set-Valued Maps and Viability Theory*, Grundlehren Math. Wiss. 264, Springer, Berlin, 1984.

[3] J.-P. Aubin and H. Frankowska, *Set-Valued Analysis*, Birkhäuser, Boston, 1990.

[4] A. Auslender and R. Cominetti, *A comparative study of multifunction differentiability with applications in mathematical programming*, Math. Oper. Res., 16 (1991), pp. 240–258.

[5] P. Billingsley, *Probability and Measure*, John Wiley, New York, 1995.

[6] P. Billingsley, *Convergence of Probability Measures*, John Wiley, New York, 1968.

[7] Ch. Castaing, *Sur les multi-applications measurables*, Rev. Française Informat. Recherche Opérationnell, 1 (1967), pp. 91–126.

[8] Ch. Castaing and M. Valadier, *Convex Analysis and Measurable Multifunctions*, Lecture Notes in Math. 580, Springer, New York, 1977.

[9] D. Dentcheva, *Differentiable selections and Castaing representations of multifunctions*, J. Math. Anal. Appl., 223 (1998), pp. 371–396.

[10] D. Dentcheva and W. Römisch, *Differential stability of two-stage stochastic programs*, SIAM J. Optim, to appear.

[11] G. Dommisch, *On the existence of Lipschitz-continuous and differentiable selections for multifunctions*, in Parametric Optimization and Related Topics, J. Guddat, H. Th. Jongen, B. Kummer, and F. Nožička, eds., Math. Res. 35, Akademie-Verlag, Berlin, 1987, pp. 60–73.

[12] R. M. Dudley, *Probabilities and Metrics*, Lecture Notes Ser. 45, Aarhus Universitet, Aarhus, 1976.

[13] J. Dupačová and R. J.-B. Wets, *Asymptotic behaviour of statistical estimators and of optimal solutions of stochastic optimization*, Ann. Statist., 16 (1988), pp. 1517–1549.

[14] S. Gautier and R. Morchadi, *A selection of convex-compact-valued multifunctions with remarkable properties: The Steiner selection*, Numer. Funct. Anal. Optim., 13 (1992), pp. 513–522.

[15] C. R. Givens and R. M. Shortt, *A class of Wasserstein metrics for probability distributions*, Michigan Math. J., 31 (1984), pp. 231–240.

[16] R. Henrion, *The Approximate Subdifferential and Parametric Optimization*, Habilitationschrift, Humboldt-Universität zu Berlin, Berlin, 1997.

[17] R. Henrion and W. Römisch, *Metric regularity and quantitative stability in stochastic programs with probabilistic constraints*, Math. Programming, 84 (1999), pp. 55–88.

[18] P. Kall, *Stochastic Linear Programming*, Springer-Verlag, Berlin, 1976.

[19] A. J. King, *Generalized delta theorems for multivalued mappings and measurable selections*, Math. Oper. Res., 14 (1989), pp. 720–736.

[20] A. J. King and R. T. Rockafellar, *Asymptotic theory for solutions in statistical estimation and stochastic programming*, Math. Oper. Res., 18 (1993), pp. 148–162.

[21] E. Klein and A. C. Thompson, *Theory of Correspondences, Including Applications to Mathematical Economics*, John Wiley, New York, 1984.

[22] K. Kuratowski and C. Ryll-Nardzewski, *A general theorem on selectors*, Bull. Acad. Pol. Sci., 13 (1965), pp. 397–403.

[23] P. Lachout, *On multifunction transforms of probability measures*, Ann. Oper. Res., 56 (1995), pp. 241–250.

[24] G. Pflug, *Asymptotic stochastic programs*, Math. Oper. Res., 20 (1996), pp. 769–789.

[25] J.-P. Penot, *Differentiability of relations and differential stability of perturbed optimization problems*, SIAM J. Control Optim., 22 (1984), pp. 529–551.

[26] A. Prekopa, *Stochastic Programming*, Math. Appl. 324, Kluwer, Dordrecht, the Netherlands, 1995.

[27] D. Repovs and P. V. Semenov, *Continuous Selections of Multivalued Mappings*, Math. Appl. 455, Kluwer, Dordrecht, the Netherlands, 1998.

[28] R. T. Rockafellar, *Proto-differentiability of set-valued mappings and its application in optimization*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 6 (1989), pp. 449–482.

[29] R. T. Rockafellar, *Integral functionals, normal integrands and measurable selections*, in Nonlinear Operators and the Calculus of Variations, L. Waelbroeck, ed., Lecture Notes in Math. 543, Springer-Verlag, New York, 1976, pp. 157–207.

[30] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Grundlehren Math. Wiss. 317, Springer-Verlag, Berlin, 1998.

[31] W. Römisch and R. Schultz, *Stability analysis for stochastic programs*, Ann. Oper. Res., 30 (1991), pp. 241–266.

[32] W. Römisch and R. Schultz, *Distribution sensitivity in stochastic programming*, Math. Programming, 50 (1991), pp. 197–226.

[33] W. Römisch and R. Schultz, *Lipschitz stability for stochastic programs with complete recourse*, SIAM J. Optim., 6 (1996), pp. 531–547.

[34] W. Römisch and R. Schultz, *Distribution sensitivity for certain classes of chance constrained models with application to power dispatch*, J. Optim. Theory Appl., 71 (1991), pp. 569–588.

[35] G. Salinetti and R. J.-B.Wets, *On the convergence in distribution of measurable multifunctions (random sets), normal integrands, stochastic processes and stochastic infima*, Math. Oper. Res., 11 (1986), pp. 385–419.

[36] R. Schneider, *Convex Bodies: The Brunn-Minkowski Theory*, Encyclopedia of Mathematics and Its Applications 44, Cambridge University Press, Cambridge, 1993.

[37] A. Shapiro, *Existence and differentiability of metric projections in Hilbert spaces*, SIAM J. Optim., 4 (1994), pp. 130–141.

[38] A. Shapiro, *On differentiability of metric projections in $\mathbb{R}^n$: Boundary case*, Proc. Amer. Math. Soc., 99 (1987), pp. 123–128.

[39] A. Shapiro, *Asymptotic analysis of stochastic programs*, Ann. Oper. Res., 30 (1991), pp. 169–186.

[40] G. C. Shephard, *A uniqueness theorem for the Steiner point of a convex region*, J. London Math. Soc., 43 (1968), pp. 439–444.

[41] J. Steiner, *Von dem Krümmungsschwerpunkte ebener Kurven*, Z. Reine Angew. Math., 21 (1840), pp. 33–63; 101–122.

[42] R. J.-B. Wets, *Stochastic programs with fixed recourse: The equivalent deterministic program*, SIAM Rev., 16 (1974), pp. 309–339.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 750–778

CONES OF MATRICES AND SUCCESSIVE CONVEX RELAXATIONS OF NONCONVEX SETS*

MASAKAZU KOJIMA† AND LEVENT TUNÇEL‡

Abstract. Let F be a compact subset of the n-dimensional Euclidean space R^n represented by (finitely or infinitely many) quadratic inequalities. We propose two methods, one based on successive semidefinite programming (SDP) relaxations and the other on successive linear programming (LP) relaxations. Each of our methods generates a sequence of compact convex subsets C_k $(k = 1, 2, \dots)$ of R^n such that
(a) the convex hull of $F \subseteq C_{k+1} \subseteq C_k$ (monotonicity),
(b) $\cap_{k=1}^{\infty} C_k =$ the convex hull of F (asymptotic convergence).
Our methods are extensions of the corresponding Lovász–Schrijver lift-and-project procedures with the use of SDP or LP relaxation applied to general quadratic optimization problems (QOPs) with infinitely many quadratic inequality constraints. Utilizing descriptions of sets based on cones of matrices and their duals, we establish the exact equivalence of the SDP relaxation and the semi-infinite convex QOP relaxation proposed originally by Fujie and Kojima. Using this equivalence, we investigate some fundamental features of the two methods including (a) and (b) above.

Key words. semidefinite programming, nonconvex quadratic optimization problem, linear matrix inequality, bilinear matrix inequality, semi-infinite programming, global optimization

AMS subject classifications. 15A48, 52A47, 49M39, 90C05, 90C25, 90C26, 90C30, 90C34

PII. S1052623498336450

1. Introduction.

Consider a maximization problem with a linear objective function $\boldsymbol{c}^T\boldsymbol{x}$:

$$\text{maximize } \boldsymbol{c}^T\boldsymbol{x} \quad \text{subject to } \boldsymbol{x} \in F, \tag{1.1}$$

where $\boldsymbol{c}$ denotes a constant vector in the n-dimensional Euclidean space R^n and F a subset of R^n. We can reduce a more general maximization problem with a nonlinear objective function $f(\boldsymbol{x})$ to a maximization problem having a linear objective function represented by a new variable, x_{n+1}, if we replace $f(\boldsymbol{x})$ by x_{n+1} and then add the inequality $f(\boldsymbol{x}) \geq x_{n+1}$ to the constraint. Thus (1.1) covers such a general optimization problem. Throughout the paper we assume that F is compact. Then the problem (1.1) has a global maximizer whenever the feasible region F is nonempty.

For any compact convex set C containing F, the maximization problem

$$\text{maximize } \boldsymbol{c}^T\boldsymbol{x} \quad \text{subject to } \boldsymbol{x} \in C \tag{1.2}$$

serves as a convex relaxation problem, which satisfies the properties that

(i) the maximum objective value ζ of the problem (1.2) gives an upper bound for the maximum objective value ζ^* of the problem (1.1), i.e., $\zeta \geq \zeta^*$, and

*Received by the editors March 31, 1998; accepted for publication (in revised form) July 19, 1999; published electronically March 21, 2000.

http://www.siam.org/journals/siopt/10-3/33645.html

†Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552, Japan (kojima@is.titech.ac.jp).

‡Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (ltuncel@math.uwaterloo.ca). This work was completed while this author was visiting Tokyo Institute of Technology, Department of Mathematical and Computing Sciences, on a sabbatical leave from University of Waterloo. The research of this author was supported in part by Tokyo Institute of Technology and by a research grant from NSERC of Canada.

(ii) if a maximizer $\hat{\boldsymbol{x}} \in C$ of (1.2) lies in F, it is a maximizer of (1.1).

Since the objective function of (1.1) is linear, we know that if we take the convex hull c.hull(F) (defined as the intersection of all the convex sets containing F) for C in (1.2), then

(i)′ $\zeta = \zeta^*$, and

(ii)′ the set of the maximizers of (1.2) forms a compact convex set whose extreme points are maximizers of (1.1).

Therefore, if we solve the relaxation problem (1.2) with a convex feasible region C which closely approximates c.hull(F), we can expect to get not only a good upper bound ζ for the maximum objective value ζ^* but also an approximate maximizer of the problem (1.1). We can further prove that for almost every $\boldsymbol{c} \in R^n$ (in the sense of measure), any maximizer $\boldsymbol{x}' \in C =$ c.hull(F) of (1.2) is an extreme point of c.hull(F), which also lies in F; hence $\boldsymbol{x}'$ is a maximizer of (1.1). This follows from a result due to Ewald, Larman, and Rogers [5] for consequences of related results; see also [17]. Furthermore, for many representations of various convex sets C, given $\hat{\boldsymbol{x}} \in C$, we can very efficiently find $\boldsymbol{x}^*$, an extreme point of C, such that $\boldsymbol{c}^T\boldsymbol{x}^* \geq \boldsymbol{c}^T\hat{\boldsymbol{x}}$.

Indeed, the relaxation technique mentioned above has been playing an essential role in practical computational methods for solving various problems in the fields of combinatorial optimization and global optimization. It is often used in hybrid schemes with the branch-and-bound and branch-and-cut techniques in those fields. See, for instance, [2].

The aim of this paper is to present a basic idea on how we can approximate the convex hull of F. This is a quite difficult problem, and also too general. Before making further discussions, we at least need to provide an appropriate (algebraic) representation for the compact feasible region F of the problem (1.1) and the compact convex feasible region C of the relaxation problem (1.2). We employ quadratic inequalities for this purpose.

Let $\mathcal{S}^n$ and $\mathcal{S}^n_+ \subset \mathcal{S}^n$ denote the set of $n \times n$ symmetric matrices and the set of $n \times n$ symmetric positive semidefinite matrices, respectively. Given $\boldsymbol{Q} \in \mathcal{S}^n$, $\boldsymbol{q} \in R^n$, and $\gamma \in R$, we write a quadratic function on R^n with the quadratic term $\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}$, the linear term $2\boldsymbol{q}^T\boldsymbol{x}$, and the constant term γ as $p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q})$:

$$p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \equiv \gamma + 2\boldsymbol{q}^T\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} \quad \forall \boldsymbol{x} \in R^n.$$

Then the set $\mathcal{Q}$ of quadratic functions on R^n and the set $\mathcal{Q}_+$ of convex quadratic functions are defined as

$$\mathcal{Q} \equiv \{p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) : \boldsymbol{Q} \in \mathcal{S}^n,\ \boldsymbol{q} \in R^n \text{ and } \gamma \in R\}$$

and

$$\mathcal{Q}_+ \equiv \{p(\cdot;\gamma,\,\boldsymbol{q},\boldsymbol{Q}) : \boldsymbol{Q} \in \mathcal{S}^n_+,\ \boldsymbol{q} \in R^n \text{ and } \gamma \in R\},$$

respectively. We also write $p(\cdot) \in \mathcal{Q}$ (or $\mathcal{Q}_+$) instead of $p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{Q}$ (or $\mathcal{Q}_+$) if $\boldsymbol{Q} \in \mathcal{S}^n$, $\boldsymbol{q} \in R^n$, and $\gamma \in R$ are irrelevant. Throughout the paper, we assume that the feasible region F of the problem (1.1) is represented by a set of quadratic inequalities such that

$$F = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0\ \forall p(\cdot) \in \mathcal{P}_F\},$$

where $\mathcal{P}_F$ denotes a set of quadratic functions, i.e., $\mathcal{P}_F \subseteq \mathcal{Q}$, and we will derive convex relaxations, C, represented by convex quadratic inequalities such that

$$C = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0\ \forall p(\cdot) \in \mathcal{P}_C\},$$

where $\mathcal{P}_C$ denotes a set of convex quadratic functions, i.e., $\mathcal{P}_C \subseteq \mathcal{Q}_+$. We allow cases where $\mathcal{P}_F$ and/or $\mathcal{P}_C$ involve infinitely many quadratic functions. Thus (1.1) or (1.2) (or both) can be a semi-infinite quadratic optimization problem (QOP). Here we use the word "semi-infinite" for optimization problems having a finite number of scalar variables and possibly an infinite number of inequality constraints.

There are some reasons why we have chosen quadratic inequalities for the representation of both problems, the maximization problem (1.1) that we want to solve and its convex relaxation problem (1.2). First, quadratic inequalities form a class of relatively easily manageable nonlinear inequalities, yet they have enough power to describe any compact feasible region F in R^n. Indeed, if F is closed, then its complement $R^n \backslash F$ is open so that it can be represented as the union of the open balls

$$\{\boldsymbol{x} \in R^n : (\boldsymbol{x} - \boldsymbol{x}')^T(\boldsymbol{x} - \boldsymbol{x}') < \epsilon(\boldsymbol{x}')\} \quad \text{with } \exists \epsilon(\boldsymbol{x}') > 0$$

over all $\boldsymbol{x}' \in \mathcal{G}$ for some $\mathcal{G} \subseteq R^n \backslash F$; hence

$$F = \{\boldsymbol{x} \in R^n : (\boldsymbol{x} - \boldsymbol{x}')^T(\boldsymbol{x} - \boldsymbol{x}') \geq \epsilon(\boldsymbol{x}') \ \forall \boldsymbol{x}' \in \mathcal{G}\}.$$

We also know that any single polynomial inequality can be converted into a system of quadratic inequalities; for example,

$$x_1^2 x_2 + 2x_1 x_2^2 - 5 \leq 0$$

can be converted into

$$x_3 - x_1 x_2 \leq 0, \ -x_3 + x_1 x_2 \leq 0 \quad \text{and} \quad x_1 x_3 + 2x_2 x_3 - 5 \leq 0.$$

See [23, 24].

Second, we know that we can solve some classes of maximization problems having linear objective functions and a convex-quadratic-inequality constrained feasible region C efficiently. Among others, we can apply interior-point methods [1, 16] to the problem (1.2) when either $\mathcal{P}_C$ is finite or $\mathcal{P}_C$ is infinite, but its feasible region C is described as the projection of a set characterized by linear matrix inequalities in the space $\mathcal{S}^n$ of $n \times n$ symmetric matrices onto the n-dimensional Euclidean space R^n.

Third, and also most importantly, we can apply the semidefinite programming (SDP) relaxation, which was originally developed for 0-1 integer programming problems by Lovász and Schrijver [12] and later extended to nonconvex quadratic optimization problems [6, 18, 19], to the entire class of maximization problems having a linear objective function and finitely or infinitely many quadratic inequality constraints. See also [1, 8, 9, 13, 15, 23, 24, 29].

In addition to the reasons above, we should mention that the maximization problem with a linear objective function and quadratic inequality constraints involves various optimization problems such as 0-1 integer linear (or quadratic) programming problems which, in principle, include all combinatorial optimization problems [1, 9, 18]. Linear complementarity problems [4], bimatrix games, and bilinear matrix inequalities [14, 20] are also included as special cases.

For some optimization problems, some of the semidefinite programming (SDP) relaxations we provide may be solved in polynomially many iterations (of an interior-point method or an ellipsoid algorithm) approximately. Such conclusion requires, in the case of the ellipsoid method, the existence of a certain polynomial-time separation oracle for the underlying convex cone constraint (see [9]). In the case of interior-point algorithms (whose efficiency in the theory and practice of SDP has been well established), we need to have an efficiently computable self-concordant barrier for the feasible solutions set or at least for the underlying cone constraints (see [16]).

Some of the most exciting activities in combinatorial optimization are currently centered around the applications of SDP to combinatorial optimization problems (see [7]). Such activity in theory and practice is fueled by theoretical results establishing that certain simple SDP relaxations of a combinatorial optimization problem can be effectively utilized in developing polynomial-time approximation algorithms with worst-case approximation-ratio guarantees much better than those previously proven using linear programming or other techniques. (See Goemans [7], Goemans and Williamson [8], Nesterov [15], and Ye [29].) Also outstanding are the results on the stable set problem establishing the fact that SDP techniques can be used in optimizing over a relaxation of the stable set polytope which is contained in the polytope defined by the clique inequalities. (Note that it is NP-hard to optimize over the latter-mentioned polytope, whereas Grötschel, Lovász, and Schrijver [9] and Lovász, and Schrijver [12] were able to utilize polynomial-time methods to achieve a better goal, as far as the proof of approximate optimality of some feasible solutions of the stable set problem is concerned.)

Given an initial approximation C_0 of F, i.e., a compact convex set C_0 containing F, both of the methods, proposed in this paper, generate a sequence of compact convex subsets C_k $(k = 1, 2, \dots)$ of R^n such that

(a) c.hull$(F) \subseteq C_{k+1} \subseteq C_k$ (monotonicity),

(b) $\cap_{k=1}^{\infty} C_k =$ c.hull(F) (asymptotic convergence).

It should be noted that the compactness of each C_k and property (b) imply that

(c) if $F = \emptyset$, then $\cap_{k=1}^{k^*} C_k = \emptyset$ for some finite number k^* (detecting infeasibility).

To generate C_{k+1} at each iteration, the SDP relaxation and the linear programming (LP) relaxation play an essential role, and the entire method may be regarded as an extension of the Lovász–Schrijver lift-and-project procedure for 0-1 integer programming problems to semi-infinite nonconvex quadratic optimization problems, with the use of the SDP relaxation in the first method and the LP relaxation in the second method. The LP relaxation, referred to above, is essentially the same as the reformulation-linearization technique developed for nonconvex quadratic optimization problems by Sherali and Alameddine [21]; see also [2, 22]. However, we should caution the reader that the methods presented here are mostly conceptual in the general settings, because we need to solve a semi-infinite SDP (or a semi-infinite LP) at each iteration. For such a task, an efficient practical algorithm may not be currently available.

In their paper [6], Fujie and Kojima proposed the semi-infinite convex QOP relaxation for nonconvex quadratic optimization problems and showed that the semi-infinite convex QOP relaxation is not stronger than the SDP relaxation in general, but the two relaxations are essentially equivalent under Slater's constraint qualification. We establish the exact equivalence between the two relaxations for semi-infinite nonconvex quadratic optimization problems without any constraint qualification. Using this equivalence, we derive some fundamental features of our methods including (a) and (b) above. One of the common themes in this paper is the usage of cones of matrices (and duality) in our constructions. This was also one of the themes of [12]. The other themes of this paper are the successive applications of SDP relaxations and LP relaxations. We call the related procedures the successive SDP relaxation method and the successive semi-infinite LP relaxation method, respectively.

Section 2 is devoted to preliminaries, where we provide some basic definitions and properties on quadratic inequality representations for closed subsets of R^n, the homogeneous form of quadratic functions, the SDP relaxation, etc. In section 3, we present our first method in detail as well as the main results, including the features (a) and (b). After we present some fundamental characterizations of the SDP relaxation in section 4, we give proofs of the main results in section 5. In section 6, we apply our method to 0-1 semi-infinite nonconvex quadratic optimization problems. Incorporating the basic results on the lift-and-project procedure given by Lovász and Schrijver [12] for 0-1 integer convex optimization problems, we show that our method terminates in at most $(n+1)$ iterations either to generate the convex hull of the feasible region or to detect the emptiness of the feasible region, where n denotes the number of 0-1 variables of the problem. Section 7 contains our second method, which is based on semi-infinite LP relaxations. We establish the same theoretical properties as we do for the successive SDP relaxation method. In section 8, we present two numerical examples showing the worst-case behavior of some of our procedures. In particular, we know from the second example that the best of our procedures requires infinitely many iterations to generate the convex hull of F in the worst case.

2. Preliminaries.

2.1. Semi-infinite quadratic inequality representation. In this subsection, we discuss some representations of a closed subset F of R^n in terms of (possibly infinitely many) quadratic inequalities. If $p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{Q}$, and $p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0$ holds for all $\boldsymbol{x} \in F$, we say that $p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0$ is *a quadratic valid inequality* for F and that $p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q})$ induces a quadratic valid inequality for F. A quadratic valid inequality $p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0$ for F is

linear if $\boldsymbol{Q} = \boldsymbol{O}$,

rank-1 *quadratic* if $p(\boldsymbol{x}) = (\boldsymbol{a}^T\boldsymbol{x} - \alpha)(\boldsymbol{a}^T\boldsymbol{x} - \beta)$ for $\exists \boldsymbol{a} \in R^n$, $\exists \alpha \in R$ and $\exists \beta \in R$ such that $\alpha \leq \boldsymbol{a}^T\boldsymbol{x} \leq \beta \ \forall \boldsymbol{x} \in F$,

rank-2 *quadratic* if $p(\boldsymbol{x}) = -(\boldsymbol{a}^T\boldsymbol{x} - \alpha)(\boldsymbol{b}^T\boldsymbol{x} - \beta)$ for $\exists \boldsymbol{a} \in R^n$, $\exists \boldsymbol{b} \in R^n$, $\exists \alpha \in R$ and $\exists \beta \in R$ such that $\boldsymbol{a}^T\boldsymbol{x} \leq \alpha$ and $\boldsymbol{b}^T\boldsymbol{x} \leq \beta \ \forall \boldsymbol{x} \in F$,

spherical if $p(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{d})^T(\boldsymbol{x} - \boldsymbol{d}) - \rho$ for $\exists \boldsymbol{d} \in R^n$ and $\exists \rho > 0$,

ellipsoidal if $p(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{d})^T\boldsymbol{Q}(\boldsymbol{x} - \boldsymbol{d}) - \rho$ for $\exists \boldsymbol{Q} \in \mathcal{S}^n_{++}$, $\boldsymbol{d} \in R^n$ and $\exists \rho > 0$,

convex quadratic if $\boldsymbol{Q} \in \mathcal{S}^n_{+}$,

respectively. It should be noted that if a quadratic valid inequality $p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0$ for F is rank-2, then the rank of the matrix $\boldsymbol{Q}$ is at most 2 but that the converse is not necessarily true.

We say that F has *a (semi-infinite) quadratic inequality representation* $\mathcal{P} \subseteq \mathcal{Q}$ if

$$F = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0 \ \forall p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{P}\}$$

holds. To designate the underlying representation $\mathcal{P}$ of F, we often write $F(\mathcal{P})$ instead of F. Whenever F is a closed proper subset of R^n, F has infinitely many representations. We allow the cases where $\mathcal{P}$ consists of infinitely many quadratic functions. Hence $p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}$ can be a semi-infinite system of quadratic inequalities. If $\mathcal{P} \subseteq \mathcal{Q}$ is a quadratic inequality representation of F and if $p(\cdot) \in \text{c.cone}(\mathcal{P})$, then $p(\boldsymbol{x}) \leq 0$ is a quadratic valid inequality, where $\text{c.cone}(\mathcal{P})$ denotes the closed convex cone generated by $\mathcal{P}$. Hence if $\mathcal{P} \subseteq \mathcal{P}' \subseteq \text{c.cone}(\mathcal{P})$, then $\mathcal{P}'$ is a quadratic inequality representation of F; $F(\mathcal{P}) = F(\mathcal{P}') = F(\text{c.cone}(\mathcal{P}))$. A quadratic inequality representation $\mathcal{P}$ of F is *finite* if it consists of a finite number of quadratic functions, and *infinite* otherwise. If F is a compact convex subset of R^n, it has a quadratic inequality representation; in fact, the set of all the linear (rank-2 quadratic or spherical) valid inequalities for F forms an inequality representation of F. If, in addition, F is polyhedral, we can take a finite linear inequality representation.

Let C be a compact subset of R^n. We use the following symbols:

$\mathcal{P}^L(C) =$ the set of $p(\cdot)$'s that induce linear valid inequalities for C,

$\mathcal{P}^1(C) =$ the set of $p(\cdot)$'s that induce rank-1 quadratic valid inequalities for C,

$\mathcal{P}^2(C) =$ the set of $p(\cdot)$'s that induce rank-2 quadratic valid inequalities for C,

$\mathcal{P}^S(C) =$ the set of $p(\cdot)$'s that induce spherical valid inequalities for C,

$\mathcal{P}^E(C) =$ the set of $p(\cdot)$'s that induce ellipsoidal valid inequalities for C,

$\mathcal{P}^C(C) =$ the set of $p(\cdot)$'s that induce convex quadratic valid inequalities for C,

$\mathcal{P}^\sharp(C) =$ the set of $p(\cdot)$'s that induce all quadratic valid inequalities for C.

By definition, we see that

$$\begin{array}{l}\left(\mathcal{P}^L(C) \cup \mathcal{P}^1(C) \cup \mathcal{P}^S(C) \cup \mathcal{P}^E(C)\right) \subset \mathcal{P}^C(C) \subset \mathcal{P}^\sharp(C),\\ \mathcal{P}^S(C) \subset \mathcal{P}^E(C) \ \text{ and } \ \left(\mathcal{P}^L(C) \cup \mathcal{P}^1(C)\right) \subset \mathcal{P}^2(C) \subset \mathcal{P}^\sharp(C).\end{array}$$

Note that if C is convex, then the equality

$$C = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}\}$$

holds with each $\mathcal{P} = \mathcal{P}^L(C), \mathcal{P}^1(C), \mathcal{P}^2(C), \mathcal{P}^S(C), \mathcal{P}^E(C), \mathcal{P}^C(C), \mathcal{P}^\sharp(C)$. Among these, $\mathcal{P}^\sharp(C)$ is *the strongest quadratic inequality representation* of C.

2.2. Homogeneous form of quadratic functions—lifting to the space of symmetric matrices. We introduce a different description of quadratic functions, which we call the homogeneous form. This form leads us to a lifting of a quadratic function defined on the Euclidean space to the space of symmetric matrices and to the SDP relaxation (or to the semi-infinite LP relaxation in section 4.2). For every quadratic function $p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \mathcal{Q}$, we connect the variable vector $\boldsymbol{x} \in R^n$ to the $(1+n) \times (1+n)$ rank-1 positive semidefinite matrix

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} = \begin{pmatrix} 1 \\ \boldsymbol{x} \end{pmatrix} (1, \boldsymbol{x}^T) \in \mathcal{S}^{1+n}_+$$

and the triplet of the constant $\gamma \in R$, $\boldsymbol{q} \in R^n$, and $\boldsymbol{Q} \in \mathcal{S}^n$ to the $(1+n) \times (1+n)$ symmetric matrix $\left(\begin{smallmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{smallmatrix}\right) \in \mathcal{S}^{1+n}$. Then we have the identity

$$p(\boldsymbol{x}; \gamma, \boldsymbol{q}, \boldsymbol{Q}) = (1, \boldsymbol{x}^T) \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} \begin{pmatrix} 1 \\ \boldsymbol{x} \end{pmatrix} = \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \ \forall \boldsymbol{x} \in R^n.$$

Thus, if $\mathcal{P} \subseteq \mathcal{Q}$ is a quadratic inequality representation of F, then

$$\underline{\mathcal{P}} \equiv \left\{ \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} : p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \mathcal{P} \right\}$$

provides an equivalent representation of F;

$$F(\mathcal{P}) = \left\{ \boldsymbol{x} \in R^n : \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \leq 0 \ \forall \boldsymbol{P} \in \underline{\mathcal{P}} \right\}.$$

Now we have two kinds of description for a quadratic function on R^n: the usual form $p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) = \gamma + 2\boldsymbol{q}^T\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}$ and the homogeneous form introduced above.

The former is used in section 5, where we prove our main results, while the latter is suitable for the compact description of the SDP relaxation in section 2.3 and the proof of its equivalence to the semi-infinite convex QOP relaxation in section 4. We will use both forms in parallel, choosing whichever is convenient to us in a given situation. It should be noted that the correspondence

$$p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{Q} \iff \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} \in \mathcal{S}^{1+n}$$

is not only one-to-one but also linear. To save notation, we identify the set $\mathcal{Q}$ of quadratic functions with the set $\mathcal{S}^{1+n}$ of $(1+n)\times(1+n)$ symmetric matrices and any subset of $\mathcal{Q}$ with the corresponding subset of $\mathcal{S}^{1+n}$. Specifically, we write $P = \left(\begin{smallmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{smallmatrix}\right) \in \mathcal{P}$ whenever $p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{P}$ and identify the set of $(1+n)\times(1+n)$ symmetric matrices

$$\left\{ \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} : \gamma \in R,\ \boldsymbol{q} \in R^n,\ \boldsymbol{Q} \in \mathcal{S}^n \right\}$$

with the set $\mathcal{Q}$ of quadratic functions from R^n to R.

2.3. SDP relaxation. Let $\mathcal{P}$ be a semi-infinite quadratic inequality representation of F:

$$\begin{aligned} F(\mathcal{P}) &= \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \le 0 \ \forall p(\cdot) \in \mathcal{P}\} \\ &= \left\{ \boldsymbol{x} \in R^n : \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \le 0 \ \forall \boldsymbol{P} \in \mathcal{P} \right\}. \end{aligned}$$

The SDP relaxation $\hat{F}(\mathcal{P})$ of $F(\mathcal{P})$ with the quadratic inequality representation $\mathcal{P}$ is given by

$$\begin{aligned} \hat{F}(\mathcal{P}) &\equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \text{ and} \\ \gamma + 2\boldsymbol{q}^T\boldsymbol{x} + \boldsymbol{Q} \bullet \boldsymbol{X} \le 0 \quad \forall p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{P} \end{array} \right\} \\ &= \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \text{ and} \\ \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \le 0 \quad \forall \boldsymbol{P} \in \mathcal{P} \end{array} \right\}. \end{aligned}$$

If $\boldsymbol{x} \in F(\mathcal{P})$ and $\boldsymbol{P} \in \mathcal{P}$, then $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T$ satisfies that

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} = \begin{pmatrix} 1 \\ \boldsymbol{x} \end{pmatrix} (1, \boldsymbol{x}^T) \in \mathcal{S}^{1+n}_+ \text{ and } \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \le 0.$$

This implies that $\boldsymbol{x} \in \hat{F}(\mathcal{P})$ and $F(\mathcal{P}) \subseteq \hat{F}(\mathcal{P})$. We also see that $\hat{F}(\mathcal{P})$ is convex. Hence c.hull$(F(\mathcal{P})) \subseteq \hat{F}(\mathcal{P})$. The SDP relaxation was originally proposed for combinatorial optimization problems and 0-1 integer programming problems [12], and later extended to quadratic optimization problems. See [1, 6, 8, 9, 15, 19, 18, 23, 24, 29].

3. Main results. Now we are ready to describe our method for approximating a quadratic-inequality-constrained compact feasible region F of the minimization problem (1.1). Before running the method, we need to fix a semi-infinite quadratic inequality representation $\mathcal{P}_F$ of F, and choose an initial approximation C_0 of the convex hull of F, i.e., a compact convex set which contains c.hull(F). Starting from C_0, the method generates a sequence of compact convex sets C_k $(k = 0, 1, 2, \dots)$, which we expect to converge to c.hull(F). At each iteration, we choose a semi-infinite quadratic inequality representation $\mathcal{P}_k$ of the kth approximation C_k of c.hull(F). Since c.hull$(F) \subseteq C_k$, the union $(\mathcal{P}_F \cup \mathcal{P}_k)$ forms a semi-infinite quadratic inequality representation of F. We then apply the SDP relaxation to $(\mathcal{P}_F \cup \mathcal{P}_k)$ to generate the next iterate $C_{k+1} = \hat{F}(\mathcal{P}_F \cup \mathcal{P}_k)$. It should be emphasized that during none of the iterations do we modify or strengthen the representation $\mathcal{P}_F$ directly. We only utilize the semi-infinite quadratic inequality representation of the compact convex set C_k that has been computed in the previous iteration.

Successive SDP Relaxation Method.

Step 0: Let $k = 0$.

Step 1: If $C_k = \emptyset$ or $C_k = \text{c.hull}(F)$, then stop.

Step 2: Choose a semi-infinite quadratic inequality representation $\mathcal{P}_k$ for C_k.

Step 3: Let

$$\begin{aligned} C_{k+1} &= \hat{F}(\mathcal{P}_F \cup \mathcal{P}_k) \\ &= \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \\ \text{and} \\ \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \leq 0 \quad \forall \boldsymbol{P} \in \mathcal{P}_F \cup \mathcal{P}_k \end{array} \right\}. \end{aligned} \tag{3.1}$$

Step 4: Let $k = k + 1$, and go to Step 1.

We state two convergence theorems below. We choose the spherical inequality representation $\mathcal{P}^S(C_k)$ for C_k at Step 2 of each iteration in the first theorem, while we choose the rank-2 quadratic inequality representation $\mathcal{P}^2(C_k)$ for C_k at Step 2 of each iteration in the second theorem. Their proofs will be given in section 5.

Theorem 3.1. *Assume that $\mathcal{P}_F$ is a semi-infinite quadratic inequality representation of a compact subset F of R^n, and that $C_0 \supseteq F$ is a compact convex subset of R^n. If we choose $\mathcal{P}_k = \mathcal{P}^S(C_k)$ at Step* 2 *of each iteration in the successive SDP relaxation method, then the monotonicity property* (a) *and the asymptotic convergence property* (b) *stated in the introduction hold.*

Theorem 3.2. *Under the same assumptions as in Theorem* 3.1*, if we choose $\mathcal{P}_k = \mathcal{P}^2(C_k)$ at Step* 2 *of each iteration in the successive SDP relaxation method, then* (a) *and* (b) *remain valid.*

We know that if $\mathcal{P} \subset \mathcal{Q}$ and $\mathcal{P}' \subset \mathcal{Q}$ are semi-infinite quadratic inequality representations of C_k and if $\mathcal{P} \subset \mathcal{P}'$, then $\hat{F}(\mathcal{P}') \subseteq \hat{F}(\mathcal{P})$. Hence, even if we replace "$\mathcal{P}_k = \mathcal{P}^S(C_k)$" in Theorem 3.1 by "$\mathcal{P}_k \supseteq \mathcal{P}^S(C_k)$" (or "$\mathcal{P}_k = \mathcal{P}^2(C_k)$" in Theorem 3.2 by "$\mathcal{P}_k \supseteq \mathcal{P}^2(C_k)$"), the properties (a) and (b) remain valid. In particular, (a) and (b) remain valid when we choose any of $\mathcal{P}^E(C_k)$, $\mathcal{P}^C(C_k)$, and $\mathcal{P}^\sharp(C_k)$ for $\mathcal{P}_k$.

If we take the linear representation $\mathcal{P}^L(C_k)$ of C_k at every iteration, then we can prove that

$$C_1 = \tilde{F}(\mathcal{P}_F \cup \mathcal{P}_0) = \tilde{F}(\mathcal{P}_F) \cap C_0 \quad \text{and} \quad C_{k+1} = \tilde{F}(\mathcal{P}_F) \cap C_k = C_1 \quad (k = 1, 2, \dots).$$

(See Lemma 4.1.) Hence (b) does not follow in general.

In section 8, we will give two numerical examples. The first example shows that the rank-1 quadratic inequality representation $\mathcal{P}_k = \mathcal{P}^1(C_k)$ is not strong enough to ensure (b). The second example shows that even when we choose the strongest quadratic inequality representation $\mathcal{P}^{\sharp}(C_k)$ of C_k for $\mathcal{P}_k$ at every iteration, not only does the convergence "$C_k \to$ c.hull(F)" require infinitely many iterations, but its speed also becomes extremely slow in the worst case.

4. Fundamental characterization of successive convex relaxation.

4.1. Semi-infinite convex QOP relaxation and its equivalence to SDP relaxation.

The semi-infinite convex QOP relaxation of $F(\mathcal{P})$ with the semi-infinite quadratic inequality representation $\mathcal{P}$ is defined as

$$\begin{aligned}\tilde{F}(\mathcal{P}) &\equiv \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \text{c.cone}(\mathcal{P}) \cap \mathcal{Q}_+\} \\ &= \left\{\boldsymbol{x} \in R^n : \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \leq 0 \ \forall \boldsymbol{P} \in \text{c.cone}(\mathcal{P}) \cap \mathcal{Q}_+ \right\}.\end{aligned}$$

We observe that

$$F(\mathcal{P}) = \left\{\boldsymbol{x} \in R^n : \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \leq 0 \ \forall \boldsymbol{P} \in \text{c.cone}(\mathcal{P}) \right\} \subseteq \tilde{F}(\mathcal{P})$$

and that the set $\tilde{F}(\mathcal{P})$ is a closed convex set. Hence $F(\mathcal{P}) \subseteq$ c.hull$(F(\mathcal{P})) \subseteq \tilde{F}(\mathcal{P})$.

The semi-infinite convex QOP relaxation was introduced by Fujie and Kojima [6]. It was called the relaxation using convex-quadratic valid inequalities for $F(\mathcal{P})$ in their paper [6]. The following basic properties of the relaxation are essentially due to them.

LEMMA 4.1. *Let $\mathcal{P}_F$ be a semi-infinite quadratic inequality representation of a closed set $F \subset R^n$.*

(i) *Let $\mathcal{P}$ be a set of convex quadratic valid inequalities for F, i.e., $\mathcal{P} \subseteq \mathcal{P}^C(F)$. Then*

$$\tilde{F}(\mathcal{P}_F \cup \mathcal{P}) \subseteq \tilde{F}(\mathcal{P}) = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}\}.$$

(ii) *Let $\mathcal{P}$ be a set of linear valid inequalities for F, i.e., $\mathcal{P} \subseteq \mathcal{P}^L(F)$. Then*

$$\tilde{F}(\mathcal{P}_F \cup \mathcal{P}) = \tilde{F}(\mathcal{P}_F) \cap \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}\}.$$

(iii) *Let $\boldsymbol{x}' \notin$ c.hull(F). Suppose that $p(\boldsymbol{x}'; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \geq 0$ for some $p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \mathcal{P}_F$ with a positive definite $\boldsymbol{Q}$. Then $\boldsymbol{x}' \notin \tilde{F}(\mathcal{P}_F)$.*

Proof. Part (i) follows directly from the definition of the semi-infinite convex QOP relaxation. Now we show (ii). Let $C = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}\}$. Then we see that

$$\tilde{F}(\mathcal{P}_F \cup \mathcal{P}) \subseteq \tilde{F}(\mathcal{P}_F) \cap \tilde{F}(\mathcal{P}) = \tilde{F}(\mathcal{P}_F) \cap C.$$

Hence it suffices to show that $\tilde{F}(\mathcal{P}_F) \cap C \subseteq \tilde{F}(\mathcal{P}_F \cup \mathcal{P})$. Let $p(\cdot) \in$ c.cone$(\mathcal{P}_F \cup \mathcal{P}) \cap \mathcal{Q}_+$. Then there exist $p(\cdot)_i \in \mathcal{P}_F$ $(i = 1, 2, \ldots, \ell)$, $p(\cdot)_j \in \mathcal{P}$ $(j = \ell + 1, \ldots, m)$, and positive numbers λ_i $(i = 1, 2, \ldots, m)$ such that

$$p(\cdot) = \sum_{i=1}^{\ell} \lambda_i p(\cdot)_i + \sum_{j=\ell+1}^{m} \lambda_i p(\cdot)_i \in \mathcal{Q}_+.$$

Since $p(\cdot)_j \in \mathcal{P}$ $(j = \ell + 1, \ldots, m)$ are linear functions, we see that

$$\sum_{i=1}^{\ell} \lambda_i p(\cdot)_i \in \text{c.cone}(\mathcal{P}_F) \cap \mathcal{Q}_+; \quad \text{hence,} \ \sum_{i=1}^{\ell} \lambda_i p(\boldsymbol{x})_i \leq 0 \quad \forall \boldsymbol{x} \in \tilde{F}(\mathcal{P}_F).$$

Moreover,

$$\sum_{j=\ell+1}^{m} \lambda_i p(\cdot)_i \in \text{c.cone}(\mathcal{P}) \cap \mathcal{Q}_+; \quad \text{hence,} \quad \sum_{j=\ell+1}^{m} \lambda_i p(\boldsymbol{x})_i \leq 0 \quad \forall \boldsymbol{x} \in C.$$

Therefore,

$$p(\boldsymbol{x}) = \sum_{i=1}^{\ell} \lambda_i p_i(\boldsymbol{x}) + \sum_{j=\ell+1}^{m} \lambda_i p_i(\boldsymbol{x}) \leq 0 \quad \forall \boldsymbol{x} \in \tilde{F}(\mathcal{P}_F) \cap C.$$

This proves (ii). Finally we will show (iii). Since $\boldsymbol{x}' \not\in F$, there is a $p'(\cdot) \in \mathcal{P}_F$ such that $p'(\boldsymbol{x}') > 0$. Hence, if $\epsilon > 0$ is sufficiently small, we obtain that

$$\epsilon p(\cdot)' + p(\cdot) \in \text{c.cone}(\mathcal{P}_F) \cap \mathcal{Q}_+ \text{ and } \epsilon p'(\boldsymbol{x}') + p(\boldsymbol{x}') > 0.$$

This implies $\boldsymbol{x}' \not\in \tilde{F}(\mathcal{P}_F)$, and proves (iii). □

When $\mathcal{P}$ is finite and $F(\mathcal{P})$ satisfies Slater's constraint qualification, Fujie and Kojima [6] showed that the semi-infinite convex QOP relaxation is essentially equivalent to the SDP relaxation in the sense that $\tilde{F}(\mathcal{P})$ coincides with the closure of $\hat{F}(\mathcal{P})$. The theorem below shows the exact equivalence between them, without any constraint qualification, for more general semi-infinite quadratic inequality representation cases. Since $\tilde{F}(\mathcal{P})$ is closed, one of the consequences of the next theorem is that $\hat{F}(\mathcal{P})$ is always closed. Note that we can assume without loss of generality that $\mathcal{P}$ is a closed convex cone, since every closed set F admits such a representation.

THEOREM 4.2. *Let $\mathcal{P}$ be a closed convex cone, giving a semi-infinite quadratic inequality representation of a closed subset F of R^n; $F(\mathcal{P}) = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}\}$. Then its SDP relaxation and its semi-infinite convex QOP relaxation coincide with each other; $\hat{F}(\mathcal{P}) = \tilde{F}(\mathcal{P})$.*

Proof. Using the dual cone

$$\mathcal{P}^* = \{\boldsymbol{V} \in \mathcal{S} : \boldsymbol{V} \bullet \boldsymbol{U} \geq 0 \ \forall \boldsymbol{U} \in \mathcal{P}\}$$

of $\mathcal{P}$, we can express the sets $\hat{F}(\mathcal{P})$ and $\tilde{F}(\mathcal{P})$ as follows:

$$\hat{F}(\mathcal{P}) = \left\{ \boldsymbol{x} \in R^n : \exists \boldsymbol{X} \in \mathcal{S}^n \ \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in (-\mathcal{P}^*) \cap \mathcal{S}^{1+n}_+ \right\}$$

and

$$\begin{aligned} \tilde{F}(\mathcal{P}) &= \left\{ \boldsymbol{x} \in R^n : \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \in -\left(\mathcal{P} \cap \mathcal{Q}_+\right)^* \right\} \\ &= \left\{ \boldsymbol{x} \in R^n : \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \in -\left[\mathcal{P}^* + \begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \mathcal{S}^n_+ \end{pmatrix} \right] \right\}. \end{aligned}$$

For the last identity above, we have used the fact that for any pair of closed convex cones $\mathcal{K}_1$ and $\mathcal{K}_2$ in R^m, we have $(\mathcal{K}_1 \cap \mathcal{K}_2)^* = \mathcal{K}_1^* + \mathcal{K}_2^*$.

First let $\boldsymbol{x} \in \hat{F}(\mathcal{P})$. Then there exists an $\boldsymbol{X} \in \mathcal{S}^n$ such that

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in (-\mathcal{P}^*) \cap \mathcal{S}^{1+n}_+.$$

Consider the identity

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} = -\left[\begin{pmatrix} -1 & -\boldsymbol{x}^T \\ -\boldsymbol{x} & -\boldsymbol{X} \end{pmatrix} + \begin{pmatrix} 0 & -\boldsymbol{0}^T \\ -\boldsymbol{0} & \boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix}\right].$$

The first matrix on the right-hand side is in $\mathcal{P}^*$ and in the second matrix of the right-hand side, we have $\boldsymbol{X} - \boldsymbol{x}\boldsymbol{x}^T \in \mathcal{S}^n_+$ since it is the Schur complement of 1 in the symmetric, positive semidefinite matrix $\left(\begin{smallmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{smallmatrix}\right)$. We have proved $\boldsymbol{x} \in \tilde{F}(\mathcal{P})$ and hence $\hat{F}(\mathcal{P}) \subseteq \tilde{F}(\mathcal{P})$.

For the converse, let $\boldsymbol{x} \in \tilde{F}(\mathcal{P})$; that is, there exists some $\boldsymbol{H} \in \mathcal{S}^n_+$ such that

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T + \boldsymbol{H} \end{pmatrix} \in -\mathcal{P}^*.$$

The matrix

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T + \boldsymbol{H} \end{pmatrix}$$

is positive semidefinite if and only if $(\boldsymbol{H} + \boldsymbol{x}\boldsymbol{x}^T - \boldsymbol{x}\boldsymbol{x}^T) = \boldsymbol{H}$ is. But the latter was already established. So,

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T + \boldsymbol{H} \end{pmatrix} \in (-\mathcal{P}^*) \cap \mathcal{S}^{1+n}_+.$$

Therefore $\boldsymbol{x} \in \hat{F}(\mathcal{P})$, and $\tilde{F}(\mathcal{P}) \subseteq \hat{F}(\mathcal{P})$ is proved. □

4.2. Semi-infinite LP relaxation. In section 7, we will also need an analog of the above theorem for our successive semi-infinite LP relaxation method. For every semi-infinite quadratic inequality representation $\mathcal{P}$ of a compact subset F of R^n, let us define

$$\hat{F}^L(\mathcal{P}) \equiv \left\{\boldsymbol{x} \in R^n : \exists X \in \mathcal{S}^n \text{ such that } \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \le 0 \ \ \forall \boldsymbol{P} \in \mathcal{P}\right\}$$

and

$$\tilde{F}^L(\mathcal{P}) \equiv \left\{\boldsymbol{x} \in R^n : \gamma + 2\boldsymbol{q}^T\boldsymbol{x} \le 0 \ \forall p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \text{c.cone}(\mathcal{P}) \cap \mathcal{L}\right\}$$

of Sherali and Alameddine [21]. Here, $\mathcal{L}$ denotes the set of linear functions on R^n:

$$\mathcal{L} \equiv \{p(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \mathcal{Q} : \boldsymbol{Q} = \boldsymbol{O}\}.$$

The next result can be obtained by following the steps of the proof of Theorem 4.2.

COROLLARY 4.3. *Let $\mathcal{P}$ be a closed convex cone, giving a semi-infinite quadratic inequality representation of a closed subset F of R^n; $F(\mathcal{P}) = \{\boldsymbol{x} \in R^n : p(\boldsymbol{x}) \le 0 \ \forall p(\cdot) \in \mathcal{P}\}$. Then $\hat{F}^L(\mathcal{P}) = \tilde{F}^L(\mathcal{P})$.*

Proof. We observe that

$$\hat{F}^L(\mathcal{P}) = \left\{\boldsymbol{x} \in R^n : \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in -\mathcal{P}^*\right\}$$

and

$$\begin{aligned}\tilde{F}^L(\mathcal{P}) &= \left\{ \boldsymbol{x} \in R^n : \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \in -\left(\mathcal{P} \cap \mathcal{L}\right)^* \right\} \\ &= \left\{ \boldsymbol{x} \in R^n : \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \in -\left[\mathcal{P}^* + \begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \mathcal{S}^n \end{pmatrix} \right] \right\}.\end{aligned}$$

Since it is easy to see that $\exists \boldsymbol{X} \in \mathcal{S}^n$ such that $\left(\begin{smallmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{smallmatrix}\right) \in -\mathcal{P}^*$ if and only if

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \in -\left[\mathcal{P}^* + \begin{pmatrix} 0 & \boldsymbol{0}^T \\ \boldsymbol{0} & \mathcal{S}^n \end{pmatrix} \right],$$

the proof is complete. □

4.3. Invariance under one-to-one affine transformation. Let $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$ be an arbitrary one-to-one affine transformation on R^n, where $\boldsymbol{A}$ is an $n \times n$ nonsingular matrix and $\boldsymbol{b} \in R^n$.

Then

$$\begin{aligned}\boldsymbol{f}(\hat{F}(\mathcal{P})) = \boldsymbol{f}(\tilde{F}(\mathcal{P})) &= \{\boldsymbol{y} \in R^n : p'(\boldsymbol{y}) \leq 0 \ \forall p'(\cdot) \in \text{c.cone}(\mathcal{P}') \cap \mathcal{Q}_+\}, \\ \boldsymbol{f}(\hat{F}^L(\mathcal{P})) = \boldsymbol{f}(\tilde{F}^L(\mathcal{P})) &= \{\boldsymbol{y} \in R^n : p'(\boldsymbol{y}) \leq 0 \ \forall p'(\cdot) \in \text{c.cone}(\mathcal{P}') \cap \mathcal{L}\},\end{aligned}$$

where $\mathcal{P}' \equiv \{p(\boldsymbol{f}^{-1}(\cdot)) : p(\cdot) \in \mathcal{P}\}$ forms a semi-infinite quadratic inequality representation of $\boldsymbol{f}(F(\mathcal{P}))$. This means that the semi-infinite SDP and LP relaxations are invariant under the one-to-one affine transformation $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{b}$.

We also see that

$$\mathcal{P}^U(\boldsymbol{f}(C)) = \{p(\boldsymbol{f}^{-1}(\cdot)) : p(\cdot) \in \mathcal{P}^U(C)\}$$

holds, where $U \in \{L,\ 1,\ 2,\ E,\ C,\ \sharp\}$. Therefore, $\mathcal{P}^L(C)$, $\mathcal{P}^1(C)$, $\mathcal{P}^2(C)$, $\mathcal{P}^E(C)$, $\mathcal{P}^C(C)$, and $\mathcal{P}^\sharp(C)$ are invariant under one-to-one affine transformations on R^n. If in addition $\boldsymbol{A}$ is a scalar multiple of an orthogonal matrix, then the above identity also holds for $U = S$; hence $\mathcal{P}^S(C)$ is invariant under such a one-to-one affine transformation on R^n.

At each iteration of the successive SDP relaxation method, we observe that

$$\boldsymbol{f}(C_{k+1}) = \{\boldsymbol{y} \in R^n : p'(\boldsymbol{y}) \leq 0 \ \forall p'(\cdot) \in \text{c.cone}(\mathcal{P}'_F \cup \mathcal{P}'_k) \cap \mathcal{Q}_+\},$$

where $\mathcal{P}'_F \equiv \{p(\boldsymbol{f}^{-1}(\cdot)) : p(\cdot) \in \mathcal{P}_F\}$ forms a semi-infinite quadratic inequality representation of $\boldsymbol{f}(F)$ and $\mathcal{P}'_k \equiv \{p(\boldsymbol{f}^{-1}(\cdot)) : p(\cdot) \in \mathcal{P}_k\}$ forms a semi-infinite quadratic inequality representation of $\boldsymbol{f}(C_k)$. Furthermore, if we choose one of the invariant semi-infinite quadratic inequality representations $\mathcal{P}^L(C_k)$, $\mathcal{P}^1(C_k)$, $\mathcal{P}^2(C_k)$, $\mathcal{P}^E(C_k)$, $\mathcal{P}^C(C_k)$, and $\mathcal{P}^\sharp(C_k)$ of C_k under any one-to-one affine transformation for $\mathcal{P}_k$, we see that $\mathcal{P}^U(\boldsymbol{f}(C)) = \{p(\boldsymbol{f}^{-1}(\cdot)) : p(\cdot) \in \mathcal{P}^U(C)\}$; hence the identity above turns out to be

$$\boldsymbol{f}(C_{k+1}) = \left\{\boldsymbol{y} \in R^n : p'(\boldsymbol{y}) \leq 0 \ \forall p'(\cdot) \in \text{c.cone}(\mathcal{P}'_F \cup \mathcal{P}^U(\boldsymbol{f}(C_k))) \cap \mathcal{Q}_+\right\}.$$

Here $U \in \{L,\ 1,\ 2,\ E,\ C,\ \sharp\}$. Therefore the successive SDP relaxation method is invariant under any one-to-one affine transformation. The same comment applies to the successive semi-infinite LP relaxation method, which we will present in section 7.

5. Proofs of Theorems 3.1 and 3.2.

We present three lemmas, Lemma 5.1 in section 5.1, Lemma 5.2 in section 5.2, and Lemma 5.3 in section 5.4. Lemma 5.1 proves the monotonicity property (a) in Theorems 3.1 and 3.2 simultaneously. Lemma 5.2 is used to prove Theorem 3.1 in section 5.3, and Lemma 5.3 to prove Theorem 3.2 in section 5.5.

5.1. Monotonicity.

We first establish the monotonicity in general.

LEMMA 5.1. *Let C_0 be a compact convex set containing F. Fix a closed convex cone $\mathcal{S}^{1+n}_+ \subseteq \mathcal{K} \subseteq \mathcal{S}^{1+n}$ and $U \in \{L, 1, 2, S, E, C, \sharp\}$. Define*

$$C_{k+1} \equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \textit{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{K} \textit{ and} \\ \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \leq 0 \quad \forall \boldsymbol{P} \in \mathcal{P}_F \cup \mathcal{P}^U(C_k) \end{array} \right\}$$

for $k = 1, 2, \dots$. Assume that

$$C_0 = \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \textit{ such that } \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{K} \textit{ and} \\ \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \leq 0 \quad \forall \boldsymbol{P} \in \mathcal{P}^U(C_0) \end{array} \right\}.$$

Then c.hull$(F) \subseteq C_{k+1} \subseteq C_k$ for all $k = 0, 1, 2, \dots$.

Proof. Since $\mathcal{K} \supseteq \mathcal{S}^{1+n}_+$, it contains all symmetric rank-1 matrices of the form

$$\begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix}.$$

Now, as in the arguments in section 2.3, it follows that c.hull$(F) \subseteq C_k$ for all $k = 0, 1, 2, \dots$. We will show by induction that $C_{k+1} \subseteq C_k$ for all $k = 0, 1, \dots$. By the construction of C_1 and the assumption imposed on C_0, we first observe that $C_1 \subseteq C_0$. Now assume that $C_k \subseteq C_{k-1}$ for some $k \geq 1$. Then $\mathcal{P}^U(C_{k-1}) \subseteq \mathcal{P}^U(C_k)$, which implies that $\mathcal{P}_F \cup \mathcal{P}^U(C_{k-1}) \subseteq \mathcal{P}_F \cup \mathcal{P}^U(C_k)$. Therefore, $C_{k+1} \subseteq C_k$, as desired. ☐

5.2. Separating hypersphere.

The following lemma easily follows from the separating hyperplane theorem, and the proof is omitted here.

LEMMA 5.2. *Let C be a compact convex subset of R^n and $\boldsymbol{x}' \notin C$. Then there exists a hypersphere $S \equiv \{\boldsymbol{x} \in R^n : \|\boldsymbol{x} - \boldsymbol{d}\| = \eta\}$ which strictly separates the point $\boldsymbol{x}'$ and C such that*

$$\|\boldsymbol{x}' - \boldsymbol{d}\| > \eta > \|\boldsymbol{x} - \boldsymbol{d}\| \quad \forall \boldsymbol{x} \in C, \tag{5.1}$$

where $\boldsymbol{d} \in R^n$ and $\eta > 0$.

5.3. Proof of Theorem 3.1.

The monotonicity property (a) follows from Lemma 5.1 by letting $\mathcal{K} \equiv \mathcal{S}^{1+n}_+$ and $U \equiv S$. Let $C \equiv \cap_{k=0}^{\infty} C_k$. We know by (a) that c.hull$(F) \subseteq C \subseteq C_{k+1} \subseteq C_k$ $(k = 0, 1, \dots)$, and that all the sets c.hull(F), C, and C_k are compact sets. To prove (b), we have the following left to show: $C \subseteq$ c.hull(F). Assume on the contrary that there exists some $\boldsymbol{x}' \in C$ such that $\boldsymbol{x}' \notin$ c.hull(F). Then, by Lemma 5.2, there exists a hypersphere $S \equiv \{\boldsymbol{x} \in R^n : \|\boldsymbol{x} - \boldsymbol{d}\| = \eta\}$ that strictly separates the point $\boldsymbol{x}' \in C$ from c.hull(F) such that

$$\|\boldsymbol{x}' - \boldsymbol{d}\| > \eta > \|\boldsymbol{x} - \boldsymbol{d}\| \quad \forall \boldsymbol{x} \in \text{c.hull}(F),$$

where $\boldsymbol{d} \in R^n$ and $\eta > 0$. Let $\eta^* \equiv \sup\{\|\boldsymbol{x} - \boldsymbol{d}\| : \boldsymbol{x} \in C\}$. Obviously, $\eta < \eta^* = \|\boldsymbol{x}^* - \boldsymbol{d}\|$ for some $\boldsymbol{x}^* \in C$. Since $\boldsymbol{x}^* \notin \text{c.hull}(F)$, there is a quadratic function, $p_1(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{P}_F$ that cuts off $\boldsymbol{x}^*$; $0 < p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q})$. Note that if $p_1(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q})$ is such a quadratic function, then so is $\alpha p_1(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q})$ for any $\alpha > 0$. Hence we may assume that the minimum eigenvalue of the matrix $\boldsymbol{Q} \in \mathcal{S}^n$ is at least (-1). Now consider a quadratic function $p_2(\cdot)$ defined by

$$p_2(\boldsymbol{x}) = (\boldsymbol{x} - \boldsymbol{d})^T(\boldsymbol{x} - \boldsymbol{d}) - (\eta^*)^2 - p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q})/2 \quad \forall \boldsymbol{x} \in R^n.$$

By the definition of η^*, we see that

$$p_2(\boldsymbol{x}) \leq -p_1(\boldsymbol{x}^*)/2 < 0 \quad \forall \boldsymbol{x} \in C.$$

This means that the open ball $B_+ \equiv \{\boldsymbol{x} \in R^n : p_2(\boldsymbol{x}) < 0\}$ with the center $\boldsymbol{d}$ and the radius $\sqrt{(\eta^*)^2 + p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q})/2}$ forms a neighborhood of the compact set C. On the other hand, the sequence $\{C_k\}$ of compact subsets of R^n satisfies

$$C_{k+1} \subseteq C_k \ (k = 0,1,2,\dots) \quad \text{and} \quad C = \cap_{k=0}^{\infty} C_k.$$

So, we can find a finite positive number ℓ such that the open ball B_+ contains C_ℓ. Hence, $p_2(\boldsymbol{x}) \leq 0$ is a convex quadratic valid inequality for C_ℓ; $p_2(\cdot) \in \mathcal{P}_\ell$. We also see that

$$p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\boldsymbol{x}^*) = p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q})/2 > 0 \ \text{ and } p_1(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\cdot) \in \mathcal{Q}_+.$$

Thus we have shown that

$$p_1(\boldsymbol{x}^*;\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\boldsymbol{x}^*) > 0 \ \text{ and } p_1(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\cdot) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_\ell) \cap \mathcal{Q}_+.$$

Therefore, $\boldsymbol{x}^* \notin C_{\ell+1} = \tilde{F}(\mathcal{P}_F \cup \mathcal{P}_\ell)$, so that $\boldsymbol{x}^* \notin C = \cap_{k=0}^{\infty} C_k$. This is a contradiction. The theorem is proved.

5.4. A family of inequalities of the convex cone of rank-2 quadratic valid inequalities for the unit ball. Let B denote the unit ball $\{\boldsymbol{x} \in R^n : \|\boldsymbol{x}\| \leq 1\}$. Let $\boldsymbol{Q}$ be an arbitrary $n \times n$ symmetric matrix, and let $\boldsymbol{u} \in R^n$ be an arbitrary vector on the boundary of B; $\|\boldsymbol{u}\| = 1$. We will construct a family of quadratic valid inequalities, which lie in the convex cone of rank-2 quadratic valid inequalities, $p_\theta(\boldsymbol{x}) \leq 0$, with a parameter $\theta \in (0, \pi/8)$ for the unit ball B satisfying the properties (i), (ii), and (iii) listed in Lemma 5.3.

We first apply the eigenvalue decomposition to the matrix $\boldsymbol{Q} \in \mathcal{S}^n$. We may assume that the first m eigenvalues are nonnegative and the last $n - m$ eigenvalues are nonpositive for some nonnegative integer $m \leq n$. Then we can write the matrix $\boldsymbol{Q} \in \mathcal{S}^n$ as

$$\boldsymbol{Q} = \sum_{j=1}^{m} \mu_j \boldsymbol{r}_j \boldsymbol{r}_j^T - \sum_{j=m+1}^{n} \mu_j \boldsymbol{r}_j \boldsymbol{r}_j^T,$$

where $\|\boldsymbol{r}_j\| = 1 \ (j = 1,2,\dots,n)$ and $\mu_j \geq 0 \ (j = 1,2,\dots,n)$, $\boldsymbol{r}_j \ (j = 1,2,\dots,n)$ denote eigenvectors of $\boldsymbol{Q}$, which are orthogonal to each other, and $\mu_j \ (j = 1,2,\dots,m)$ and $-\mu_j \ (j = m+1,\dots,n)$ denote the eigenvalues corresponding to them.

For each $\theta \in (0, \pi/8)$, we define

$$(5.2)\qquad \begin{cases} \boldsymbol{a}_j(\theta) & \equiv & \boldsymbol{u}\cos\theta + \boldsymbol{r}_j \sin\theta \quad (j = 1, 2, \ldots, n), \\ \bar{\boldsymbol{a}}_j(\theta) & \equiv & \boldsymbol{u}\cos\theta - \boldsymbol{r}_j \sin\theta \quad (j = 1, 2, \ldots, n), \\ \boldsymbol{b}_j & \equiv & +\boldsymbol{r}_j, \ \bar{\boldsymbol{b}}_j \equiv -\boldsymbol{r}_j \quad (j = 1, 2, \ldots, m), \\ \boldsymbol{b}_j & \equiv & -\boldsymbol{r}_j, \ \bar{\boldsymbol{b}}_j \equiv +\boldsymbol{r}_j \quad (j = m+1, \ldots, n), \\ \alpha_j(\theta) & \equiv & \max\{\boldsymbol{a}_j(\theta)^T\boldsymbol{x} : \boldsymbol{x} \in B\} = \|\boldsymbol{a}_j(\theta)\| \quad (j = 1, 2, \ldots, n), \\ \bar{\alpha}_j(\theta) & \equiv & \max\{\bar{\boldsymbol{a}}_j(\theta)^T\boldsymbol{x} : \boldsymbol{x} \in B\} = \|\bar{\boldsymbol{a}}_j(\theta)\| \quad (j = 1, 2, \ldots, n), \\ \beta_j & \equiv & \max\{\boldsymbol{b}_j^T\boldsymbol{x} : \boldsymbol{x} \in B\} = \|\boldsymbol{b}_j\| = 1 \quad (j = 1, 2, \ldots, n), \\ \bar{\beta}_j & \equiv & \max\{\bar{\boldsymbol{b}}_j^T\boldsymbol{x} : \boldsymbol{x} \in B\} = \|\bar{\boldsymbol{b}}_j\| = 1 \quad (j = 1, 2, \ldots, n), \\ \lambda_j(\theta) & \equiv & \dfrac{\mu_j}{2\sin\theta} \geq 0 \quad (j = 1, 2, \ldots, n). \end{cases}$$

Then, $\forall\theta \in (0, \pi/8)$ and $j = 1, 2, \ldots, n$, $\boldsymbol{a}_j(\theta)$, $\bar{\boldsymbol{a}}_j(\theta)$, $\boldsymbol{b}_j(\theta)$, and $\bar{\boldsymbol{b}}_j(\theta)$ are nonzero vectors, and

$$(5.3)\qquad \begin{cases} \boldsymbol{a}_j(\theta)^T\boldsymbol{x} - \alpha_j(\theta) & \leq 0, \quad \boldsymbol{b}_j^T\boldsymbol{x} - \beta_j \leq 0, \\ \bar{\boldsymbol{a}}_j(\theta)^T\boldsymbol{x} - \bar{\alpha}_j(\theta) & \leq 0, \quad \bar{\boldsymbol{b}}_j^T\boldsymbol{x} - \bar{\beta}_j \leq 0 \end{cases}$$

are linear valid inequalities for the unit ball B. For all $\theta \in (0, \pi/8)$, define

$$(5.4)\qquad p_\theta(\boldsymbol{x}) \equiv -\sum_{j=1}^{n} \lambda_j(\theta)\Big((\boldsymbol{a}_j(\theta)^T\boldsymbol{x} - \alpha_j(\theta))(\boldsymbol{b}_j^T\boldsymbol{x} - \beta_j) \\ + (\bar{\boldsymbol{a}}_j(\theta)^T\boldsymbol{x} - \bar{\alpha}_j(\theta))(\bar{\boldsymbol{b}}_j^T\boldsymbol{x} - \bar{\beta}_j)\Big).$$

Then $p_\theta(\cdot) \in \text{c.cone}(\mathcal{P}^2(B))$ for all $\theta \in (0, \pi/8)$. In particular, $p_\theta(\boldsymbol{u}) \leq 0 \; \forall\theta \in (0, \pi/8)$.

Lemma 5.3.

(i) *$p_\theta(\cdot) \in c.cone(\mathcal{P}^2(B))$.*

(ii) *$p_\theta(\boldsymbol{u}) \to 0$ as $\theta \in (0, \pi/8)$ tends to 0.*

(iii) *The Hessian matrix of $p_\theta(\cdot)$ coincides with $-\boldsymbol{Q}$.*

Proof. Part (i) was already shown.

(ii) Let j be fixed. It suffices to show that

$$\epsilon_j(\theta) \equiv \lambda_j(\theta)(\boldsymbol{a}_j(\theta)^T\boldsymbol{u} - \alpha_j(\theta))(\boldsymbol{b}_j^T\boldsymbol{u} - \beta_j) \quad \text{and} \\ \bar{\epsilon}_j(\theta) \equiv \lambda_j(\theta)(\bar{\boldsymbol{a}}_j(\theta)^T\boldsymbol{u} - \bar{\alpha}_j(\theta))(\bar{\boldsymbol{b}}_j^T\boldsymbol{u} - \bar{\beta}_j)$$

converge to zero as $\theta \in (0, \pi/8)$ tends to 0. First, we derive that $\epsilon_j(\theta)$ converges to zero as $\theta \in (0, \pi/8)$ tends to 0. We see from (5.2) that

$$(5.5)\qquad \epsilon_j(\theta) = \frac{\mu_j(\cos\theta + \boldsymbol{u}^T\boldsymbol{r}_j\sin\theta - \|\boldsymbol{u}\cos\theta + \boldsymbol{r}_j\sin\theta\|)}{2\sin\theta}(\boldsymbol{b}_j^T\boldsymbol{u} - 1) \\ = \frac{\mu_j\left(\cos\theta + \boldsymbol{u}^T\boldsymbol{r}_j\sin\theta - (\cos^2\theta + 2\boldsymbol{u}^T\boldsymbol{r}_j\sin\theta\cos\theta + \sin^2\theta)^{\frac{1}{2}}\right)}{2\sin\theta} \\ \times (\boldsymbol{b}_j^T\boldsymbol{u} - 1).$$

Since both the numerator and the denominator above converge to zero as $\theta \in (0, \pi/8)$ tends to 0, we calculate their derivatives at $\theta = 0$. The derivative of the numerator turns out to be

$$\mu_j \left(-\sin\theta + \boldsymbol{u}^T \boldsymbol{r}_j \cos\theta + \frac{\boldsymbol{u}^T \boldsymbol{r}_j (\sin^2\theta - \cos^2\theta)}{(2\boldsymbol{u}^T \boldsymbol{r}_j \sin\theta\cos\theta + 1)^{1/2}} \right) (\boldsymbol{b}_j^T \boldsymbol{u} - 1),$$

which vanishes at $\theta = 0$. On the other hand, the derivative "$2\cos\theta$" of the denominator "$2\sin\theta$" in (5.5) does not vanish at $\theta = 0$. Thus, $\epsilon_j(\theta)$ converges to 0 as $\theta \in (0, \pi/8)$ tends to 0. Similarly, we can prove that $\bar{\epsilon}_j(\theta)$ converges to 0 as $\theta \in (0, \pi/8)$ tends to 0.

(iii) It follows from the definitions (5.2) and (5.4) that the Hessian matrix of the quadratic function $p_\theta(\cdot)$

$$\begin{aligned} &= -\sum_{j=1}^{n} \lambda_j(\theta) \frac{\boldsymbol{a}_j(\theta)\boldsymbol{b}_j^T + \boldsymbol{b}_j \boldsymbol{a}_j^T(\theta) + \bar{\boldsymbol{a}}_j(\theta)\bar{\boldsymbol{b}}_j^T + \bar{\boldsymbol{b}}_j \bar{\boldsymbol{a}}_j(\theta)^T}{2} \\ &= -\sum_{j=1}^{m} \mu_j \boldsymbol{r}_j \boldsymbol{r}_j^T + \sum_{j=m+1}^{n} \mu_j \boldsymbol{r}_j \boldsymbol{r}_j^T \\ &= -\boldsymbol{Q}. \qquad \square \end{aligned}$$

From the lemma above, we see that the cone $\mathcal{P}^2(B)$ is rich enough to contain rank-2 quadratic functions with any prescribed Hessian, leading to valid inequalities that are tight at any given point on the boundary of B.

5.5. Proof of Theorem 3.2. The monotonicity property (a) follows from Lemma 5.1 by letting $\mathcal{K} \equiv \mathcal{S}_+^{1+n}$ and $U \equiv 2$. To derive (b), it suffices to show that $C \equiv \cap_{k=0}^{\infty} C_k \subseteq \text{c.hull}(F)$ as in the proof of Theorem 3.1. Assume on the contrary that $\boldsymbol{x}' \notin \text{c.hull}(F)$ for some $\boldsymbol{x}' \in C$. By Lemma 5.2, there exists a hypersphere $S \equiv \{\boldsymbol{x} \in R^n : \|\boldsymbol{x} - \boldsymbol{d}\| = \eta\}$ which strictly separates the point $\boldsymbol{x}' \in C$ and c.hull(F) such that

$$\|\boldsymbol{x}' - \boldsymbol{d}\| > \delta > \|\boldsymbol{x} - \boldsymbol{d}\| \quad \forall \boldsymbol{x} \in \text{c.hull}(F),$$

where $\boldsymbol{d} \in R^n$ and $\delta > 0$. Let $\delta^* \equiv \sup\{\|\boldsymbol{x} - \boldsymbol{d}\| : \boldsymbol{x} \in C\}$. Obviously, $\delta^* = \|\boldsymbol{u} - \boldsymbol{d}\| > \delta$ for some $\boldsymbol{u} \in C$. Since the successive SDP relaxation method using the rank-2 quadratic representation for C_k at each iteration is invariant under the affine transformation $(\boldsymbol{x} - \boldsymbol{d})/\delta^* \to \boldsymbol{x}'$, which maps $\boldsymbol{d}$ to the origin and the hypersphere $S \equiv \{\boldsymbol{x} \in R^n : \|\boldsymbol{x} - \boldsymbol{d}\| = \delta^*\}$ onto the unit hypersphere $\{\boldsymbol{x}' \in R^n : \|\boldsymbol{x}'\| = 1\}$, we may assume that $\boldsymbol{d} = \boldsymbol{0}$ and $\delta^* = 1$. Thus, we have obtained that

$$C \subseteq B \equiv \{\boldsymbol{x} \in R^n : \|\boldsymbol{x}\| \leq 1\} \quad \text{and} \quad \boldsymbol{u} \in C,\ \boldsymbol{u} \notin \text{c.hull}(F),\ \|\boldsymbol{u}\| = 1.$$

Since $\boldsymbol{u} \notin F$, there is a quadratic function $p_1(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) \in \mathcal{P}_F$ that cuts off $\boldsymbol{u}$; $p_1(\boldsymbol{u}; \gamma, \boldsymbol{q}, \boldsymbol{Q}) > 0$. Now, let $p_\theta(\cdot) \in \mathcal{P}^2(B) \cap \mathcal{Q}_+$ be the quadratic function introduced in section 5.4. See (5.2) and (5.4). By Lemma 5.3, we can choose a $\theta \in (0, \pi/8)$ for which $p_\theta(\boldsymbol{u}) \geq -p_1(\boldsymbol{u}; \gamma, \boldsymbol{q}, \boldsymbol{Q})/3$ holds. Now we define

$$\begin{aligned} \alpha_j^k &= \max\{\boldsymbol{a}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C_k\}, \quad \beta_j^k = \max\{\boldsymbol{b}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C_k\} \quad (1 \leq j \leq n), \\ \bar{\alpha}_j^k &= \max\{\bar{\boldsymbol{a}}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C_k\}, \quad \bar{\beta}_j^k = \max\{\bar{\boldsymbol{b}}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C_k\} \quad (1 \leq j \leq n), \end{aligned}$$

$$\begin{aligned} p_k'(\boldsymbol{x}) = -\sum_{j=1}^{n} \lambda_j(\theta) \big(&(\boldsymbol{a}_j(\theta)^T \boldsymbol{x} - \alpha_j^k)(\boldsymbol{b}_j(\theta)^T \boldsymbol{x} - \beta_j^k) \\ &+ (\bar{\boldsymbol{a}}_j(\theta)^T \boldsymbol{x} - \bar{\alpha}_j^k)(\bar{\boldsymbol{b}}_j(\theta)^T \boldsymbol{x} - \bar{\beta}_j^k) \big) \end{aligned}$$

for $k = 0, 1, 2, \dots$. By construction, we know that $p'_k(\cdot) \in \text{c.cone}(\mathcal{P}^2(C_k))$. Since both quadratic functions $p_\theta(\cdot)$ and $p'_k(\cdot)$ have the common Hessian matrix $-\boldsymbol{Q}$,

$$p_1(\cdot; \gamma, \boldsymbol{q}, \boldsymbol{Q}) + p'_k(\cdot) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{L} \subset \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+ \quad \forall k = 0, 1, 2, \dots.$$

We will show that

$$p_1(\boldsymbol{u}) + p'_k(\boldsymbol{u}) \geq p_1(\boldsymbol{u})/3 > 0 \tag{5.6}$$

for every sufficiently large k. Then the above two relations imply $\boldsymbol{u} \notin C_{k+1}$ for such a large k. This contradicts the fact $\boldsymbol{u} \in C = \cap_{k=0}^{\infty} C_k$.

Since the sequence of compact convex subsets C_k $(k = 0, 1, 2, \dots)$ satisfies

$$C_{k+1} \subseteq C_k \ (k = 0, 1, 2, \dots) \ \text{ and } \ \cap_{k=0}^{\infty} C_k = C \subseteq B = \{\boldsymbol{x} : \|\boldsymbol{x}\| \leq 1\},$$

we see that

$$\begin{aligned}
&\alpha_j^k \to \alpha_j^* \equiv \max\{\boldsymbol{a}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C\} \leq \alpha_j(\theta), \ \beta_j^k \to \beta_j^* \equiv \max\{\boldsymbol{b}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C\} \leq \beta_j, \\
&\bar{\alpha}_j^k \to \bar{\alpha}_j^* \equiv \max\{\bar{\boldsymbol{a}}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C\} \leq \bar{\alpha}_j(\theta), \ \bar{\beta}_j^k \to \bar{\beta}_j^* \equiv \max\{\bar{\boldsymbol{b}}_j(\theta)^T \boldsymbol{x} : \boldsymbol{x} \in C\} \leq \bar{\beta}_j
\end{aligned}$$

as $k \to \infty$ $(j = 2, 3, \dots, n)$. By continuity, we see then that for every sufficiently large k

$$\begin{aligned}
p'_k(\boldsymbol{u}) &\geq -p_1(\boldsymbol{u})/3 - \sum_{j=1}^{n} \lambda_j(\theta) \big((\boldsymbol{a}_j(\theta)^T \boldsymbol{u} - \alpha_j^*)(\boldsymbol{b}_j(\theta)^T \boldsymbol{u} - \beta_j^*) \\
&\qquad\qquad + (\bar{\boldsymbol{a}}_j(\theta)^T \boldsymbol{u} - \bar{\alpha}_j^*)(\bar{\boldsymbol{b}}_j(\theta)^T \boldsymbol{u} - \bar{\beta}_j^*) \big) \\
&\geq -p_1(\boldsymbol{u})/3 + p_\theta(\boldsymbol{u}) \\
&\geq -2p_1(\boldsymbol{u})/3 \ (\text{since } p_\theta(\boldsymbol{u}) \geq -p_1(\boldsymbol{u})/3).
\end{aligned}$$

Thus we have shown that (5.6) holds for every sufficiently large k. This completes the proof of Theorem 3.2.

6. Application to 0-1 semi-infinite, nonconvex quadratic optimization problems. We briefly recall two of the Lovász–Schrijver procedures for 0-1 integer programming problems, and relate them to our successive SDP relaxation method. Let F be a subset of $\{0, 1\}^n$ whose convex hull is to be approximated. In the Lovász–Schrijver procedures, we assume that a compact convex subset C_0 of R^n satisfying $F = C_0 \cap \{0, 1\}^n$ is given in advance. We define

$$\mathcal{K}_0 \equiv \{(\lambda, \lambda \boldsymbol{x}^T) \in R^{1+n} : \lambda \geq 0, \ \text{and } \boldsymbol{x} \in C_0\}.$$

Let $\mathcal{K}_I$ denote the convex cone spanned by the 0-1 vectors in $\mathcal{K}_0$:

$$\mathcal{K}_I = \{(\lambda, \lambda \boldsymbol{x}^T) \in R^{1+n} : \lambda \geq 0, \ \text{and } \boldsymbol{x} \in \text{c.hull}(F)\}.$$

Here the 0th coordinate is special. It is used in homogenizing the sets of interest in R^n. Clearly

$$C_0 = \{\boldsymbol{x} \in R^n : (1, \boldsymbol{x}^T) \in \mathcal{K}_0\} \quad \text{and} \quad \text{c.hull}(F) = \{\boldsymbol{x} \in R^n : (1, \boldsymbol{x}^T) \in \mathcal{K}_I\}.$$

The closed convex cone $\mathcal{K}_0$ serves as an initial relaxation of $\mathcal{K}_I$. Given the current relaxation $\mathcal{K}_k$ of $\mathcal{K}_I$, first a convex cone $M_+(\mathcal{K}_k, \mathcal{K}_k)$ in the space of $(1+n) \times (1+n)$

symmetric matrices is defined (the lifting operation). Then a projection of this cone gives the next relaxation $N_+(\mathcal{K}_k)$ of $\mathcal{K}_I$.

Now, we define the lifting operation in general. Let $\mathcal{K}$ and $\mathcal{T}$ be closed convex cones in R^{1+n}. A $(1+n) \times (1+n)$ symmetric matrix, $\boldsymbol{Y}$, with real entries is in $M_+(\mathcal{K}, \mathcal{T})$ if

(i) $\boldsymbol{Y} \in \mathcal{S}_+^{1+n}$,

(ii) $\boldsymbol{Y}\boldsymbol{e}_0 = \mathrm{Diag}(\boldsymbol{Y})$,

(iii) $\boldsymbol{u}^T\boldsymbol{Y}\boldsymbol{v} \geq 0 \ \forall u \in \mathcal{K}^*,\ v \in \mathcal{T}^*$. (This condition is equivalent to $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{T}$.)

Here, $\boldsymbol{e}_0$ denotes the unit vector with 0th coordinate 1. Item (ii) above serves an important role in Lovász–Schrijver procedures as well as in some of the SDP relaxations used by Goemans and Williamson [8], Nesterov [15], and Ye [29]. This equation is valid simply because for each j for which $x_j \in \{0, 1\}$, the equation $x_j^2 = x_j$ is valid. Indeed, our general framework applies to any compact set in R^n, and the equation $\boldsymbol{Y}\boldsymbol{e}_0 = \mathrm{Diag}(\boldsymbol{Y})$ was not utilized in earlier sections (as it is not valid). In this section, however, the equation is valid and we utilize it. As will be noted in the proof of Theorem 6.3, the inclusion of this equation will be guaranteed by our choice of the initial formulation.

The third condition of Lovász–Schrijver procedures is very interesting. They present a couple of possibilities for the choice of cone $\mathcal{T}$ in 0-1 integer programming. Among them is the cone spanned by all 0-1 vectors with the first component $x_0 = 1$. This choice, since the cone $\mathcal{T}^*$ has a very simple set of generators, allows for the development of polynomial-time algorithms for approximately solving the successive SDP relaxations as long as the number of iterations of the successive procedure is $O(1)$. Their result only assumes that a polynomial-time *weak separation oracle* is available for $\mathcal{K}$. The key is that since $\mathcal{T}^*$ has only $O(n)$ extreme rays, it becomes trivial to check condition (iii) in polynomial time. On the other hand, Lovász and Schrijver [12] note that the choice $\mathcal{T} \equiv \mathcal{K}$ is also possible and leads to at least as good relaxations as the former choice for $\mathcal{T}$. (In many cases the successive relaxations for $\mathcal{T} \equiv \mathcal{K}$ are significantly tighter than the successive relaxations with the simpler choice of $\mathcal{T}$.) In the case of the latter choice, the possibility of polynomial-time solvability of the first few successive relaxations depends on the availability of polynomial-time algorithms to check $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{K}$. Our procedure uses $\mathcal{T} \equiv \mathcal{K}$.

Now, we describe the projection step.

$$N_+(\mathcal{K}) \equiv \{\boldsymbol{Y}\boldsymbol{e}_0 : \boldsymbol{Y} \in M_+(\mathcal{K}, \mathcal{K})\}.$$

We also define the iterated operators $N_+^k(\mathcal{K})$ as follows: $N_+^0(\mathcal{K}) := \mathcal{K}$ and $N_+^k(\mathcal{K}) := N_+(N_+^{k-1}(\mathcal{K}))$ for all integers $k \geq 1$. (We use the notation $N_+(\mathcal{K})$, whereas $N_+(\mathcal{K}, \mathcal{K})$ is used in [12].)

Another procedure studied in [12] uses a weaker relaxation by removing the condition (i) in the lifting procedure. Let $M(\mathcal{K}, \mathcal{K})$ and $N(\mathcal{K})$ denote the related sets for this procedure. We will refer to the first procedure using the lifting $M_+(\mathcal{K}, \mathcal{K})$ (and the projection N_+) as the N_+ procedure. We will call the other (using $M(\mathcal{K}, \mathcal{K})$, and N) the N procedure. Lovász and Schrijver prove the following.

Theorem 6.1.

$$\mathcal{K} \supseteq N_+(\mathcal{K}) \supseteq N_+^2(\mathcal{K}) \supseteq \cdots \supseteq N_+^n(\mathcal{K}) = \mathcal{K}_I$$

and

$$\mathcal{K} \supseteq N(\mathcal{K}) \supseteq N^2(\mathcal{K}) \supseteq \cdots \supseteq N^n(\mathcal{K}) = \mathcal{K}_I.$$

Let us see how our successive SDP relaxation method applies to 0-1 nonconvex quadratic optimization problems. Consider a 0-1 nonconvex quadratic program:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in F \equiv \{\boldsymbol{x} \in \{0,1\}^n : p(\boldsymbol{x}) \leq 0 \ \forall p(\cdot) \in \mathcal{P}'\}. \end{array} \tag{6.1}$$

We may assume that the set $\mathcal{P}'$ contains the quadratic functions $x_i(x_i-1),\ i = 1,2,\ldots,n$. Then we can replace the 0-1 constraint imposed on the variable x_i by the inequality $-x_i(x_i-1) \leq 0$. Thus by adding the quadratic functions $-x_i(x_i-1),\ i = 1,2,\ldots,n$, to $\mathcal{P}'$, we obtain a quadratic inequality representation $\mathcal{P}_F$ of the feasible region F. Let $C_0 \equiv [0,1]^n$. Note that $F \neq C_0 \cap \{0,1\}^n = \{0,1\}^n$ in our general setting here. However, $F = C_0 \cap \{0,1\}^n$ has been assumed for some compact convex subset C_0 of R^n in the Lovász–Schrijver procedures discussed above.

LEMMA 6.2. *Suppose that we take $C_0 = [0,1]^n$ and $\mathcal{P}_0 \equiv \{x_i(x_i-1) : \ i = 1,2,\ldots,n\} \subset \mathcal{P}^2(C_0)$. Then $F = C_1 \cap \{0,1\}^n$, where*

$$C_1 = \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \textit{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \ \textit{ and} \\ \boldsymbol{P} \bullet \boldsymbol{Y} \leq 0 \ \ \forall \boldsymbol{P} \in \mathcal{P}_F \cup \mathcal{P}_0 \end{array} \right\}.$$

Proof. Let C_1' be the semi-infinite convex QOP relaxation of the set F with the quadratic inequality representation $\mathcal{P}_F \cup \mathcal{P}_0$:

$$C_1' \equiv \left\{\boldsymbol{x} \in R^n : p(\boldsymbol{x};\gamma,\boldsymbol{q},\boldsymbol{Q}) \leq 0 \ \forall p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \text{c.cone}\,(\mathcal{P}_F \cup \mathcal{P}_0) \cap \mathcal{Q}_+\right\}.$$

In view of Theorem 4.2 and Lemma 5.1, we know that

$$F \subseteq \text{c.hull}(F) \subseteq C_1 = C_1' \subseteq C_0.$$

Hence it suffices to show that

$$\{\boldsymbol{x} \in C_1 : x_i = 0 \ \text{ or } 1,\ i = 1,2,\ldots,n\} \subseteq F.$$

If F contains all the 0-1 vectors, the inclusion relation above obviously holds. Now assume that $\boldsymbol{x}' \notin F$ is a 0-1 vector. Then there is a quadratic function $p_1(\cdot,\gamma,\boldsymbol{q},\boldsymbol{Q}) \in \mathcal{P}_F$ such that

$$p_1(\boldsymbol{x}',\gamma,\boldsymbol{q},\boldsymbol{Q}) > 0.$$

On the other hand, we know that the quadratic function

$$p_2(\boldsymbol{x}) \equiv \sum_{i=1}^n x_i(x_i-1),$$

with the identity matrix as its Hessian matrix, is a member of c.cone$(\mathcal{P}_0)$, and that $p_2(\boldsymbol{x}') = 0$. Hence if $\epsilon > 0$ is sufficiently small, then

$$\begin{array}{l} \epsilon p_1(\cdot,\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\cdot) \in \text{c.cone}\,(\mathcal{P}_F \cup \mathcal{P}_0) \cap \mathcal{Q}_+, \\ \epsilon p_1(\boldsymbol{x}',\gamma,\boldsymbol{q},\boldsymbol{Q}) + p_2(\boldsymbol{x}') > 0. \end{array}$$

This implies that $\boldsymbol{x}' \notin C_1'$. □

As a consequence of the lemma above, we see that the 0-1 nonconvex quadratic optimization problem (6.1) is equivalent to the 0-1 convex quadratic optimization problem

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^T\boldsymbol{x} \\ \text{subject to} & \boldsymbol{x} \in F = C_1 \cap \{0,1\}^n. \end{array} \tag{6.2}$$

Using this observation, we can prove that in the case of 0-1 nonconvex quadratic optimization problem (6.1), our successive SDP relaxation method converges in $(1+n)$ iterations.

THEOREM 6.3. *The successive SDP relaxation method, applied to the* 0-1 *nonconvex quadratic optimization problem* (6.1)*, using* $C_0 = [0,1]^n$ *as the initial approximation of c.hull*(F) *and* $\mathcal{P}_k = \mathcal{P}^2(C_k)$ *in each iteration, terminates in at most* $(1+n)$ *iterations with* $C_{1+n} = c.hull(F)$.

Proof. We note that by Lemma 6.2, after one iteration of the successive SDP relaxation method, we obtain the 0-1 convex quadratic optimization problem (6.2) that can be used with the original Lovász–Schrijver procedure. We only have to note that the successive SDP relaxation method becomes the Lovász–Schrijver procedure after the first iteration. For this purpose, we compare conditions (i), (ii), and (iii) of the Lovász–Schrijver procedure for $\mathcal{K} = \mathcal{T} = \mathcal{K}_k$ to the conditions used to construct $C_{k+1} = \hat{F}(\mathcal{P}_F \cup \mathcal{P}_k)$ in the successive SDP relaxation method. Here

$$\mathcal{K}_k \equiv \{(\lambda, \lambda\boldsymbol{x}^T) \in R^{1+n} : \lambda \geq 0,\ \boldsymbol{x} \in C_k\}.$$

First, we observe that $\exists \boldsymbol{X'} \in \mathcal{S}^n$ such that $\boldsymbol{Y'} = \left(\begin{smallmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X'} \end{smallmatrix}\right) \in \mathcal{S}^{1+n}_+$ if and only if $\forall \lambda \geq 0$, $\exists \boldsymbol{X} \in \mathcal{S}^n$ such that $\boldsymbol{Y} = \left(\begin{smallmatrix} \lambda & \lambda\boldsymbol{x}^T \\ \lambda\boldsymbol{x} & \boldsymbol{X} \end{smallmatrix}\right) \in \mathcal{S}^{1+n}_+$. Hence (i) is satisfied. For (ii), note that $x_i(x_i-1) \in \mathcal{P}_F\ \forall\ i$ implies the constraint $\boldsymbol{Y}\boldsymbol{e}_0 \geq \mathrm{Diag}(\boldsymbol{Y})$ and $-x_i(x_i-1) \in \mathcal{P}_F\ \forall\ i$ implies $\boldsymbol{Y}\boldsymbol{e}_0 \leq \mathrm{Diag}(\boldsymbol{Y})$. Finally, for (iii), note that a linear inequality $\boldsymbol{a}^T\boldsymbol{x} \leq \alpha$ is valid for C_k if and only if $(\alpha, -\boldsymbol{a}^T) \in \mathcal{K}^*_k$ (recall $C_k = \{\boldsymbol{x} \in R^n : (1, \boldsymbol{x}^T) \in \mathcal{K}_k\}$). Therefore, we see that

$$\mathcal{P}^2(C_k) = \text{c.cone}\{-\boldsymbol{u}\boldsymbol{v}^T :\ \boldsymbol{u}, \boldsymbol{v} \in \mathcal{K}^*_k\}.$$

Step (3.1) of the successive SDP relaxation method implies that $\boldsymbol{Y} = \left(\begin{smallmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{smallmatrix}\right) \in -(\mathcal{P}^2(C_k))^*$. Thus, we conclude by noting that

$$\boldsymbol{Y} \in -(\mathcal{P}^2(C_k))^* \quad \text{if and only if } \boldsymbol{Y} \bullet \boldsymbol{u}\boldsymbol{v}^T = \boldsymbol{u}^T\boldsymbol{Y}\boldsymbol{v} \geq 0 \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{K}^*.$$

Now, Theorem 6.1 implies that n more steps of the procedure is sufficient. □

The above discussion and the results show that our successive SDP relaxation method generalizes the Lovász–Schrijver N_+ procedure by ignoring condition (ii), which is no longer valid. Our results in the previous sections already showed that in this full generality, we still have the asymptotic convergence of the method. It is therefore interesting to investigate the same questions about the weaker procedure N:

- What is the generalization of procedure N?
- Does the generalization of procedure N satisfy the same theoretical properties as the successive SDP relaxation method?

We answer both of these questions in the next section. As is shown in [12], in some cases the procedure N_+ is significantly better than N. Procedure N is weaker, but the relaxations given by it are always polyhedral sets (so LP techniques can be employed) and N_+ requires more general techniques. Hence, sometimes procedure N might be more manageable even if the procedure N_+ is not.

We should expect that the generalization of procedure N should be only using condition (iii), $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{K}$, in the definition of the lifting. We would also expect that the generalization should lead to semi-infinite LP (rather than SDP) relaxations. We show in the next section that the above-mentioned generalization of procedure N leads to successive semi-infinite LP relaxations and all the analogs of the theoretical properties established for our successive SDP relaxations can also be established for the successive semi-infinite LP relaxations.

7. Successive semi-infinite LP relaxation.

Successive Semi-Infinite LP Relaxation Method.
Step 0: Let $k = 0$.
Step 1: If $C_k = \emptyset$ or $C_k = \text{c.hull}(F)$, then stop.
Step 2: Choose a quadratic inequality representation $\mathcal{P}_k$ for C_k.
Step 3: Let

$$\begin{aligned} C_{k+1} &= \hat{F}^L(\mathcal{P}_F \cup \mathcal{P}_k) \\ &\equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that} \\ \boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \leq 0 \;\; \forall \boldsymbol{P} \in \mathcal{P}_F \cup \mathcal{P}_k \end{array} \right\} \\ &= \tilde{F}^L(\mathcal{P}_F \cup \mathcal{P}_k) \\ &\equiv \left\{ \boldsymbol{x} \in R^n : \gamma + 2\boldsymbol{q}^T\boldsymbol{x} \leq 0 \; \forall p(\cdot;\gamma,\boldsymbol{q},\boldsymbol{Q}) \in (\text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k)) \cap \mathcal{L} \right\}. \end{aligned}$$

(The equalities above follow from Corollary 4.3.)
Step 4: Let $k = k + 1$, and go to Step 1.

Theorem 7.1. *Assume that $\mathcal{P}_F$ is a semi-infinite quadratic inequality representation of a compact subset F of R^n, and that $C_0 \supseteq F$ is a compact convex subset of R^n. If we choose $\mathcal{P}_k = \mathcal{P}^2(C_k)$ at Step 2 of each iteration in the successive semi-infinite LP relaxation method, then the monotonicity property* (a) *and the asymptotic convergence property* (b) *stated in the introduction hold.*

Proof. We can apply the same proof as the one given for Theorem 3.2 in section 5.5 to the theorem. □

Note that we can define another semi-infinite LP relaxation based on the semi-infinite convex QOP relaxation. Clearly, if $\boldsymbol{Q} \in \mathcal{S}^n_+$, then

$$\gamma + 2\boldsymbol{q}^T\boldsymbol{x} + \boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} \leq 0 \;\text{ implies }\; \gamma + 2\boldsymbol{q}^T\boldsymbol{x} \leq 0 \quad \forall \boldsymbol{x} \in R^n.$$

So, we can define a semi-infinite LP relaxation based on the above observation:

$$\hat{F}^L_+ \equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n, \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{O} \end{pmatrix} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \leq 0, \\ \forall \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} \in \text{c.cone}(\mathcal{P}) \cap \mathcal{Q}_+ \end{array} \right\}$$

and

$$\tilde{F}^L_+ \equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{O} \end{pmatrix} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \leq 0, \\ \forall \begin{pmatrix} \gamma & \boldsymbol{q}^T \\ \boldsymbol{q} & \boldsymbol{Q} \end{pmatrix} \in \text{c.cone}(\mathcal{P}) \cap \mathcal{Q}_+ \end{array} \right\}.$$

In this case, the equivalence $\hat{F}^L_+ = \tilde{F}^L_+$ is evident. The convergence of the successive semi-infinite LP relaxation method using $\hat{F}^L_+$ can be established by following the proofs of Theorems 3.1 and 3.2. Instead, we note $\tilde{F}^L_+ \subseteq \tilde{F}^L$. Therefore, Theorem 7.1 also implies that this particular semi-infinite LP relaxation method has the properties (a) and (b) mentioned in the theorem.

8. Further discussions on successive convex relaxations.

8.1. Conic quadratic inequality representation. The conic quadratic inequality presented below is a generalization of the linear matrix inequality [3, 28] and the bilinear matrix inequality [14, 20]. It will be shown that any conic quadratic inequality can be reduced to a semi-infinite system of standard quadratic inequalities and vice versa.

Let $\mathcal{K}$ and $\mathcal{K}^* = \{\boldsymbol{v} \in R^m : \boldsymbol{v} \cdot \boldsymbol{u} \geq 0 \ \forall \boldsymbol{u} \in \mathcal{K}\}$ be a closed convex cone in R^m and its dual. Here $\boldsymbol{u} \cdot \boldsymbol{v}$ denotes an inner product of $\boldsymbol{u} \in R^m$ and $\boldsymbol{v} \in R^m$. For all $\boldsymbol{u} \in R^m$, we write $\boldsymbol{u} \preceq_K \boldsymbol{0}$ when $-\boldsymbol{u}$ lies in $\mathcal{K}$. Now we introduce a *conic quadratic inequality*:

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T, \quad \sum_{i=0}^{n}\sum_{j=0}^{n} \boldsymbol{g}_{ij} x_i x_j \preceq_K \boldsymbol{0} \quad \text{and} \quad x_0 = 1. \tag{8.1}$$

Here $\boldsymbol{g}_{ij}$, $i = 0, 1, \ldots, n,\ j = 0, 1, \ldots, n$, are constant vectors in R^m. We may assume without loss of generality that $\boldsymbol{g}_{ij} = \boldsymbol{g}_{ji}$. The inequality (8.1) turns out to be a system of m usual quadratic inequalities on R^n if we take the nonnegative orthant R^m_+ of R^m for the cone $\mathcal{K}$. The inequality (8.1) turns out to be a *quadratic matrix inequality*, which is a generalization of linear and bilinear matrix inequalities [3, 28] if we identify the space of $\ell \times \ell$ symmetric matrices with R^m and we take the positive semidefinite cone $\mathcal{S}^\ell_+$ of matrices for the cone $\mathcal{K}$, where $m = \ell \times (\ell + 1)/2$ for some $\ell \geq 1$.

We can rewrite the conic quadratic inequality (8.1) as a semi-infinite system of standard quadratic inequalities in the homogeneous form.

$$\boldsymbol{P} \bullet \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{x}\boldsymbol{x}^T \end{pmatrix} \leq 0 \quad \forall \boldsymbol{P} \in \mathcal{P} \tag{8.2}$$

for some $\mathcal{P} \subseteq \mathcal{Q} = \mathcal{S}^{1+n}$. This means that we can easily include any conic quadratic inequality in the semi-infinite quadratic inequality representation of the feasible region F of the maximization problem (1.1). To see the equivalence between (8.1) and (8.2) for some $\mathcal{P} \subseteq \mathcal{Q} = \mathcal{S}^{1+n}$, we observe that (8.1) can be rewritten as

$$\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T, \quad \left(\sum_{i=0}^{n}\sum_{j=0}^{n} \boldsymbol{g}_{ij} x_i x_j \right) \cdot \boldsymbol{v} \leq 0 \quad \forall \boldsymbol{v} \in \mathcal{K}^* \quad \text{and} \quad x_0 = 1.$$

Therefore, if we define

$$\boldsymbol{P}(\boldsymbol{v}) \equiv \begin{pmatrix} \boldsymbol{g}_{00} \cdot \boldsymbol{v} & \boldsymbol{g}_{01} \cdot \boldsymbol{v} & \cdots & \boldsymbol{g}_{0n} \cdot \boldsymbol{v} \\ \boldsymbol{g}_{10} \cdot \boldsymbol{v} & \boldsymbol{g}_{11} \cdot \boldsymbol{v} & \cdots & \boldsymbol{g}_{1n} \cdot \boldsymbol{v} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{g}_{n0} \cdot \boldsymbol{v} & \boldsymbol{g}_{n2} \cdot \boldsymbol{v} & \cdots & \boldsymbol{g}_{nn} \cdot \boldsymbol{v} \end{pmatrix} \in \mathcal{Q} = \mathcal{S}^{1+n} \quad \forall \boldsymbol{v} \in \mathcal{K}^*,$$

$$\mathcal{P} \equiv \{\boldsymbol{P}(\boldsymbol{v}) : \boldsymbol{v} \in \mathcal{K}^*\},$$

we obtain the desired semi-infinite system (8.2) of standard quadratic inequalities, which is equivalent to (8.1).

Let $F(\mathcal{P})$ denote the solution set of (8.2) with its quadratic inequality representation $\mathcal{P} \equiv \{\boldsymbol{P}(\boldsymbol{v}) : \boldsymbol{v} \in \mathcal{K}^*\}$. Applying the SDP relaxation to $F(\mathcal{P})$, we obtain that

$$\begin{aligned}
\hat{F}(\mathcal{P}) &\equiv \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \text{ and} \\ \boldsymbol{P}(\boldsymbol{v}) \bullet \boldsymbol{Y} \leq 0 \ \forall \boldsymbol{v} \in \mathcal{K}^* \end{array} \right\} \\
&= \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \text{ and} \\ \left(\sum_{i=0}^n \sum_{j=0}^n \boldsymbol{g}_{ij} Y_{ij} \right) \cdot \boldsymbol{v} \leq 0 \ \forall \boldsymbol{v} \in \mathcal{K}^* \end{array} \right\} \\
&= \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \text{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \text{ and} \\ \sum_{i=0}^n \sum_{j=0}^n \boldsymbol{g}_{ij} Y_{ij} \preceq_K \boldsymbol{0} \end{array} \right\}.
\end{aligned}$$

The set in the last line corresponds to the SDP relaxation to the solution set of (8.1). This implies that we can apply the SDP relaxation directly to the conic quadratic inequality (8.1) without converting it into the semi-infinite system (8.2) of standard quadratic inequalities.

Conversely, we can reduce any semi-infinite system of standard quadratic inequalities to a conic quadratic inequality. To show this, consider a semi-infinite system (8.2) of standard quadratic inequalities in the homogeneous form. We may assume without loss of generality that $\mathcal{P} \subseteq \mathcal{S}^{1+n}$ is a closed convex cone. We can rewrite (8.2) as

$$\left(\begin{pmatrix} 1 \\ \boldsymbol{x} \end{pmatrix} (1, \boldsymbol{x}^T) \right) \preceq_{\mathcal{P}^*} \boldsymbol{O}, \tag{8.3}$$

which is a conic quadratic inequality.

Let F denote the solution set of the conic quadratic inequality (8.3) that we have derived from (8.2) above. Applying the SDP relaxation to F, we obtain that

$$\begin{aligned}
\hat{F} &\equiv \left\{ \boldsymbol{x} \in R^n : \ \exists \boldsymbol{X} \in \mathcal{S}^n \ \text{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \ \text{ and } \boldsymbol{Y} \preceq_{\mathcal{P}^*} \boldsymbol{O} \right\} \\
&= \left\{ \boldsymbol{x} \in R^n : \begin{array}{l} \exists \boldsymbol{X} \in \mathcal{S}^n \ \text{ such that } \boldsymbol{Y} = \begin{pmatrix} 1 & \boldsymbol{x}^T \\ \boldsymbol{x} & \boldsymbol{X} \end{pmatrix} \in \mathcal{S}^{1+n}_+ \ \text{ and} \\ \boldsymbol{P} \bullet \boldsymbol{Y} \leq 0 \ \forall \boldsymbol{P} \in \mathcal{P} \end{array} \right\}.
\end{aligned}$$

Note that the set in the last line corresponds to the SDP relaxation of the solution set of the semi-infinite system (8.2) of standard quadratic inequalities.

In view of the discussions above, we know that the conic quadratic inequality representation is as general as the semi-infinite quadratic inequality representation and that the SDP relaxations to both representations are equivalent. When we deal with the semi-infinite convex QOP relaxation, however, the semi-infinite quadratic inequality representation seems more convenient than the conic quadratic inequality representation.

8.2. A counterexample to the convergence for the rank-1 quadratic inequality representation case. The example below shows that the rank-1 quadratic inequality representation is not strong enough to ensure the convergence of the successive SDP relaxation method. Let

$$\begin{aligned}
F &\equiv \{\boldsymbol{x} = (x_1, x_2)^T : p_0(\boldsymbol{x}) \le 0,\ \|\boldsymbol{x}\|^2 \le 1\},\\
S &\equiv \{\boldsymbol{a} \in R^2 : a_1^2 + a_2^2 = 1\},\\
B &\equiv \{\boldsymbol{x} = (x_1, x_2)^T \in R^2 : x_1^2 + x_2^2 \le 1\},\\
C_0 &\equiv B,\\
p_0(\boldsymbol{x}) &\equiv -(x_1 - 1)^2 - (x_2 - 1)^2 + 1,\\
\mathcal{P}_F &\equiv \{p_0(\boldsymbol{x})\} \cup \mathcal{P}^1(B),
\end{aligned}$$

where $\mathcal{P}^1(B)$ denotes the rank-1 quadratic inequality representation of the unit ball, which consists of all quadratic functions such that $(\boldsymbol{a}^T\boldsymbol{x} - 1)(\boldsymbol{a}^T\boldsymbol{x} + 1)$ $(\boldsymbol{a} \in \boldsymbol{S})$. We see that

$$\text{c.hull}(F) = \{\boldsymbol{x} = (x_1, x_2)^T \in B : x_1 + x_2 \le 1\}.$$

THEOREM 8.1. *Suppose that we take $\mathcal{P}_k = \mathcal{P}^1(C_k)$ (the rank-1 quadratic inequality representation of C_k) in the successive SDP relaxation method applied to the example above. Then $C_k = B$ $(k = 0, 1, 2, \dots)$.*

Proof. By definition, $C_0 = B$. We will prove $C_1 = B$, which suffices to establish the theorem. First observe that $C_1 \subseteq B$. Hence it suffices to show $B \subseteq C_1$ or equivalently for all $p(\cdot) \in \text{c.cone}(\mathcal{P}_F) \cap \mathcal{Q}_+$,

$$p(\bar{\boldsymbol{x}}) \le 0 \quad \forall \bar{\boldsymbol{x}} \in B.$$

Let $p(\cdot) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+$ and $\bar{\boldsymbol{x}} \in B$ be fixed. Then we can choose $\lambda_i \ge 0$ $(i = 0, 1, \dots, \ell)$ and $\boldsymbol{a}_i \in S$ $(i = 1, 2, \dots, \ell)$ such that

$$p(\boldsymbol{x}) = \lambda_0 p_0(\boldsymbol{x}) + \sum_{i=1}^{\ell} \lambda_i (\boldsymbol{a}_i^T\boldsymbol{x} - 1)(\boldsymbol{a}_i^T\boldsymbol{x} + 1) \quad \forall \boldsymbol{x} \in R^n.$$

If $\lambda_0 = 0$, then $p(\bar{\boldsymbol{x}}) \le 0$. Now assume that $\lambda_0 > 0$. In this case, we may further assume without loss of generality that $\lambda_0 = 1$; hence, for all $\boldsymbol{x} \in R^n$,

$$\begin{aligned}
p(\boldsymbol{x}) &= p_0(\boldsymbol{x}) + \sum_{i=1}^{\ell} \lambda_i (\boldsymbol{a}_i^T\boldsymbol{x} - 1)(\boldsymbol{a}_i^T\boldsymbol{x} + 1)\\
&= \boldsymbol{x}^T \left(\sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i \boldsymbol{a}_i^T - \boldsymbol{I} \right) \boldsymbol{x} - \sum_{i=1}^{\ell} \lambda_i + 2\boldsymbol{e}^T\boldsymbol{x} - 1.
\end{aligned}$$

It follows from $p(\cdot) \in \mathcal{Q}_+$ that the Hessian matrix $\left(\sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i \boldsymbol{a}_i^T - \boldsymbol{I}\right)$ is positive semidefinite. Hence if we denote the largest and the smallest eigenvalues of the matrix $\sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i \boldsymbol{a}_i^T$ by $\mu_{\max}$ and $\mu_{\min}$, then $1 \le \mu_{\min} \le \mu_{\max}$. We also see that

$$\mu_{\max} + \mu_{\min} = \text{trace}\left(\sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i \boldsymbol{a}_i^T \right) = \sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i^T \boldsymbol{a}_i = \sum_{i=1}^{\ell} \lambda_i.$$

Hence

$$\begin{aligned} p(\bar{\boldsymbol{x}}) &= \bar{\boldsymbol{x}}^T \left(\sum_{i=1}^{\ell} \lambda_i \boldsymbol{a}_i \boldsymbol{a}_i^T - \boldsymbol{I} \right) \bar{\boldsymbol{x}} - \sum_{i=1}^{\ell} \lambda_i + 2\boldsymbol{e}^T \bar{\boldsymbol{x}} - 1 \\ &\leq \mu_{\max} - 1 - \sum_{i=1}^{\ell} \lambda_i + 2\boldsymbol{e}^T \bar{\boldsymbol{x}} - 1 \\ &= 2\boldsymbol{e}^T \bar{\boldsymbol{x}} - \mu_{\min} - 2 \\ &\leq 2\sqrt{2} - 3 \\ &< 0. \qquad \square \end{aligned}$$

8.3. A counterexample to the finite termination for the strongest quadratic inequality representation case. The example below shows that in the worst case, even when we take the strongest quadratic inequality representation $\mathcal{P}^{\sharp}(C_k)$ for C_k at every iteration,

- the successive SDP relaxation method requires infinitely many iterations, and
- the convergence is extremely slow.

For every $\boldsymbol{x} = (x_1, x_2)^T \in R^2$, let

$$\begin{aligned} p_1(\boldsymbol{x}) &\equiv x_1^2 + x_2^2 - 4, \\ p_2(\boldsymbol{x}) &\equiv -(x_1 - 1)^2 - (x_2 - 2)^2 + 5, \\ p_3(\boldsymbol{x}) &\equiv p_2(-x_1, x_2) = -(x_1 + 1)^2 - (x_2 - 2)^2 + 5. \end{aligned}$$

Define

$$\begin{aligned} F &\equiv \{\boldsymbol{x} = (x_1, x_2)^T \in R^2 : p_i(\boldsymbol{x}) \leq 0 \ (i = 1, 2, 3)\}, \\ \mathcal{P}_F &\equiv \{p_1(\cdot), \ p_2(\cdot), \ p_3(\cdot)\}, \\ C_0 &= \{\boldsymbol{x} = (x_1, x_2)^T \in R^2 : p_1(\boldsymbol{x}) \leq 0\}. \end{aligned}$$

Then

$$\begin{aligned} \text{c.hull}(F) &= \{\boldsymbol{x} = (x_1, x_2)^T \in R^2 : p_1(\boldsymbol{x}) \leq 0, \ x_2 \leq 0\} \\ &= \{\boldsymbol{x} = (x_1, x_2)^T \in R^2 : x_1^2 + x_2^2 \leq 4, \ x_2 \leq 0\}. \end{aligned}$$

THEOREM 8.2. *Suppose that we take $\mathcal{P}_k = \mathcal{P}^{\sharp}(C_k)$ (the strongest quadratic inequality representation of C_k) in the successive SDP relaxation method applied to the example above.*

(i) *C_k is symmetric with respect to the x_2 axis:*

$$(x_1, x_2)^T \in C_k \quad \textit{if and only if } (-x_1, x_2)^T \in C_k.$$

(ii) *Let*

$$\xi_k \equiv \max\{x_2 \ : \ (0, x_2)^T \in C_k\}.$$

Then

$$0 < \xi_k \leq 2, \tag{8.4}$$

$$0 < \bar{\xi}_{k+1} \equiv \frac{\xi_k}{1 + \xi_k(1 - \xi_k/4)} \leq \xi_{k+1}. \tag{8.5}$$

Proof. We will prove (i) and (ii) by induction.

(i) Obviously the assertion is true for $k = 0$. Assume that C_k is symmetric with respect to the x_2 axis. Then we know that

$$p(x_1, x_2) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+ \text{ if and only if } p(-x_1, x_2) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+.$$

This ensures that C_{k+1} is symmetric with respect to the x_2 axis.

(ii) By definition, we know that $\xi_0 = 2$. Hence (8.4) holds for $k = 0$. Assuming that (8.4) holds, we prove that (8.5) holds. We first observe that

$$(8.6) \qquad (2, 0)^T \in \text{c.hull}(F) \subseteq C_k, \quad (0, \xi_k)^T \in C_k \quad \text{and} \quad (0, \bar{\xi}_{k+1})^T \in C_k.$$

It suffices to show that $(0, \bar{\xi}_{k+1})^T \in C_{k+1}$ or equivalently

$$p(0, \bar{\xi}_{k+1}) \leq 0 \quad \forall p(x_1, x_2) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+.$$

Assume on the contrary that

$$p(0, \bar{\xi}_{k+1}) > 0 \text{ for } \exists p(\cdot) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+.$$

Since $p(\cdot) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+$, we can choose $\lambda_i \geq 0$ $(i = 2, 3)$ and

$$p'(\boldsymbol{x}) \equiv Q_{11}x_1^2 + 2Q_{12}x_1x_2 + Q_{22}x_2^2 + 2q_1x_1 + 2q_2x_2 + \gamma \in \mathcal{P}_k$$

such that

$$p(\boldsymbol{x}) = \sum_{i=2}^{3} \lambda_i p_i(\boldsymbol{x}) + p'(\boldsymbol{x}) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+.$$

Here we remark that $p_1(\cdot)$ can be incorporated into $p'(\cdot)$ since $p_1(\cdot) \in \mathcal{P}_k$. By the symmetry with respect to the x_2 axis, we see that

$$p(-x_1, x_2) = \sum_{i=2}^{3} \lambda_i p_i(-x_1, x_2) + p'(-x_1, x_2) \in \text{c.cone}(\mathcal{P}_F \cup \mathcal{P}_k) \cap \mathcal{Q}_+.$$

Thus, defining $\tilde{p}(\boldsymbol{x}) = (p(x_1, x_2) + p(-x_1, x_2))/2$, $\mu = \lambda_2 + \lambda_3$, and $p''(x_1, x_2) = (p'(x_1, x_2) + p'(-x_1, x_2))/2$, we obtain that

$$(8.7) \qquad \begin{aligned} \tilde{p}(0, \bar{\xi}_{k+1}) &= p(0, \bar{\xi}_{k+1}) > 0, \\ p''(x_1, x_2) &= Q_{11}x_1^2 + Q_{22}x_2^2 + q_2x_2 + \gamma \in \mathcal{P}_k, \\ \tilde{p}(x_1, x_2) &= \mu(-x_1^2 - (x_2 - 2)^2 + 4) + p''(x_1, x_2) \in \mathcal{Q}_+. \end{aligned}$$

It follows from $p''(x_1, x_2) \in \mathcal{P}_k$ and the third inclusion relation of (8.6) that $p''(0, \bar{\xi}_{k+1}) \leq 0$. Hence $\mu > 0$. We may further assume without loss of generality that $\mu = 1$; redefine $p(\boldsymbol{x}) = p(\boldsymbol{x})/\mu$, $p''(\boldsymbol{x}) = p''(\boldsymbol{x})/\mu, \ldots$, etc.; then all the relations above remain valid. Since $\tilde{p}(x_1, x_2) \in \mathcal{Q}_+$, we see that $Q_{11} \geq 1$ and $Q_{22} \geq 1$. By (8.6) and $p''(x_1, x_2) \in \mathcal{P}_k$,

$$0 \geq p''(2, 0) = 4Q_{11} + \gamma \geq 4 + \gamma \quad \text{and} \quad 0 \geq p''(0, \xi_k);$$

hence

$$\begin{aligned} \tilde{p}(0, 0) &= (-0^2 - 2^2 + 4) + p''(0, 0) = \gamma \leq -4 \text{ and} \\ \tilde{p}(0, \xi_k) &= (-0^2 - (\xi_k - 2)^2 + 4) + p''(0, \xi_k) \leq (4 - \xi_k)\xi_k. \end{aligned}$$

Therefore, by the convexity of the quadratic function $\tilde{p}(\boldsymbol{x})$, we obtain that

$$\begin{aligned}\tilde{p}(0,\bar{\xi}_{k+1}) &= \tilde{p}\left(\frac{\xi_k-\bar{\xi}_{k+1}}{\xi_k}(0,0)^T+\frac{\bar{\xi}_{k+1}}{\xi_k}(0,\xi_k)^T\right)\\ &\leq \frac{\xi_k-\bar{\xi}_{k+1}}{\xi_k}\tilde{p}(0,0)+\frac{\bar{\xi}_{k+1}}{\xi_k}\tilde{p}(0,\xi_k)\\ &\leq \frac{\xi_k-\bar{\xi}_{k+1}}{\xi_k}(-4)+\frac{\bar{\xi}_{k+1}}{\xi_k}(4-\xi_k)\xi_k\\ &= \frac{4\bar{\xi}_{k+1}}{\xi_k}\left(1+(1-\xi_k/4)\xi_k\right)-4\\ &= 0.\end{aligned}$$

This contradicts (8.7). □

The above example is simple, yet it illustrates great difficulties for the successive SDP relaxation method. For example, $\xi_{k+1}/\xi_k \to 1$. Therefore, the convergence is slower than linear.

Note that, in any dimension, if we take a pair of ball constraints, one convex (inclusion), the other nonconvex (exclusion), then both of the successive SDP and semi-infinite LP relaxation methods stop in one iteration, returning the convex hull of the intersection. Also, in the above example, if we knew that $p_2(\cdot)$ affects only the definition of F in the region $x_1 \geq 0$ and that $p_3(\cdot)$ is only effective in the region $x_1 \leq 0$, we could do elementary modifications to the method to speed up convergence tremendously. This is a good elementary example to illustrate the fact that for such methods to become more efficient in practice, hybrid approaches including branch-and-bound and branch-and-cut seem necessary. We make further remarks in the next section.

9. Concluding remarks.

We propose extensions of two fundamental lift-and-project procedures N and N_+ of Lovász and Schrijver [12]. The original procedures were proposed for 0-1 integer programming problems to compute the convex hull of feasible (integer) solutions. Our procedure applies to any nonconvex region and as a result we do not use the key equations, $\boldsymbol{Y}\boldsymbol{e}_0 = \mathrm{Diag}(\boldsymbol{Y})$, used in N and N_+ procedures. Therefore, our relaxations are based either on two conditions: $\boldsymbol{Y}$ is positive semidefinite and $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{K}$ (successive SDP relaxation method), or on only one condition: $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{K}$ (successive semi-infinite LP relaxation method). In both cases we established the properties (a) monotonicity and (b) asymptotic convergence. The weakest version of our procedures satisfying the properties (a) and (b) uses only rank-2 quadratic valid inequalities. We showed in section 6 that such inequalities ensure the condition $\boldsymbol{Y}\mathcal{K}^* \subseteq \mathcal{K}$. Finally, in section 8 we showed that even the strongest of such relaxation procedures (using all quadratic valid inequalities) uses infinitely many iterations to converge. In the above sense, the strongest positive result is given in section 7 by the successive semi-infinite LP relaxation method based on rank-2 valid inequalities.

On the one hand, theoretically speaking, the best results are given in section 7: the weakest algorithm achieving the strongest results. Moreover, the successive semi-infinite LP relaxation method is more likely to be practical for a given general problem. On the other hand, the relative value of SDP relaxations has been quite impressive so far on some very special problems (e.g., the stable set problem [12]) and less impressive on others (e.g., the matching problem [25]). Therefore, one interesting research direction is to search for interesting classes of nonconvex sets for which the successive SDP relaxation method is significantly better than the successive semi-infinite LP relaxation method. For the same reason, (partial) characterizations of nonconvex sets on which both methods perform comparably are also important.

Our convergence proofs are by contradiction, but the main argument is about cutting off a point using valid inequalities induced by the underlying construction. The strongest convergence result (for the weakest algorithm) uses separating hyperspheres. In the other proofs, for the *bad* points, the separating hyperspheres may have huge radii and converge to hyperplanes. However, for certain points and shapes, the advantage of using more general convex quadratic inequalities is clear. This discussion motivates us to suggest another avenue for research. It would be interesting to find certain invariants and measures of the input of our procedures that lead to nontrivial, *descriptive* convergence rates for our methods, perhaps only for some interesting subclass of problems.

Recently, Kojima and Takeda [11] discussed the computational complexity of the successive SDP and semi-infinite LP relaxation methods. They gave an upper bound on the number of iterations which the methods require to attain a convex relaxation of a quadratically constrained compact set F with a given accuracy $\epsilon > 0$, in terms of ϵ, the diameter of the initial relaxation C_0, the diameter of F, and some other quantities characterizing the Lipschitz continuity and the nonconvexity and nonlinearity of the quadratic inequality representation $\mathcal{P}_F$ of F.

The major difficulty in implementing the idea of the successive SDP (or semi-infinite LP) relaxation method in practice is the solution of a continuum of semi-infinite SDPs (or semi-infinite LPs) to generate a new approximation C_{k+1} of the convex hull of the feasible region F of a nonconvex quadratic program at each iteration. In their succeeding paper [10], the authors propose implementable variants by introducing two new techniques, a discretization technique for approximating continuum of semi-infinite SDPs (or semi-infinite LPs) by a finite number of standard SDPs (or LPs) with a finite number of linear inequality constraints, and a localization technique for generating a convex relaxation of F that is accurate only in certain directions in a neighborhood of the objective direction $\boldsymbol{c}$. They established that, *Given any positive number* ϵ*, there is an implementable discretized-localized variant of the successive SDP (or semi-infinite LP) relaxation method which generates an upper bound of the objective values within* ϵ *of their maximum in a finite number of iterations.* See also [27] for a practical implementation of this variant and some numerical results.

REFERENCES

[1] F. Alizadeh, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.

[2] E. Balas, S. Ceria, and G. Cornuéjols, *A lift-and-project cutting plane algorithm for mixed* 0-1 *programs*, Math. Programming, 58 (1993), pp. 295–323.

[3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, Stud. Appl. Math. 15, SIAM, Philadelphia, 1994.

[4] R. W. Cottle, J. S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, New York, 1992.

[5] G. Ewald, D. G. Larman, and C. A. Rogers, *The directions of the line segments and of the r-dimensional balls on the boundary of a convex body in Euclidean space*, Mathematika, 17 (1970), pp. 1–20.

[6] T. Fujie and M. Kojima, *Semidefinite relaxation for nonconvex programs*, J. Global Optim., 10 (1997), pp. 367–380.

[7] M. X. Goemans, *Semidefinite programming in combinatorial optimization*, Math. Programming, 79 (1997), pp. 143–161.

[8] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, J. Assoc. Comput. Mach., 42 (1995), pp. 1115–1145.

[9] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer, New York, 1988.

[10] M. Kojima and L. Tunçel, *Discretization and Localization in Successive Convex Relaxation Methods for Nonconvex Quadratic Optimization Problems*, Technical Report B-341, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo, Japan, 1998. Also issued as CORR 98-34, Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Waterloo, ON N2L 3G1, Canada.

[11] M. Kojima and A. Takeda, *Complexity Analysis of Conceptual Successive Convex Relaxation of Nonconvex Sets*, Technical Report B-350, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Meguro, Tokyo, Japan, 1999.

[12] L. Lovász and A. Schrijver, *Cones of matrices and set-functions and* 0-1 *optimization*, SIAM J. Optim., 1 (1991), pp. 166–190.

[13] R. Horst and H. Tuy, *Global Optimization*, 2nd rev. ed., Springer-Verlag, Berlin, 1992.

[14] M. Mesbahi and G. P. Papavassilopoulos, *A cone programming approach to the bilinear matrix inequality problem and its geometry*, Math. Programming, 77 (1997), pp. 247–272.

[15] Yu. E. Nesterov, *Semidefinite relaxation and nonconvex quadratic optimization*, Optim. Methods Softw., 9 (1998), pp. 141–160.

[16] Yu. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[17] G. Pataki and L. Tunçel, *On the Generic Properties of Convex Optimization Problems in Conic Form*, Research Report 97–16, Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada, 1997, revised 1999.

[18] S. Poljak, F. Rendl, and H. Wolkowicz, *A recipe for semidefinite relaxation for* (0,1)-*quadratic programming*, J. Global Optim., 7 (1995), pp. 51–73.

[19] M. V. Ramana, *An Algorithmic Analysis of Multiquadratic and Semidefinite Programming Problems*, Ph.D. Thesis, Johns Hopkins University, Baltimore, MD, 1993.

[20] M. G. Safonov, K. G. Goh, and J. H. Ly, *Control system synthesis via bilinear matrix inequalities*, in Proceedings of the 1994 American Control Conference, Baltimore, MD, IEEE, 1994, pp. 45–49.

[21] H. D. Sherali and A. Alameddine, *A new reformulation-linearization technique for bilinear programming problems*, J. Global Optim., 2 (1992), pp. 379–410.

[22] H. D. Sherali and C. H. Tuncbilek, *A reformulation-convexification approach for solving nonconvex quadratic programming problems*, J. Global Optim., 7 (1995), pp. 1–31.

[23] N. Z. Shor, *Quadratic optimization problems*, Soviet J. Comput. Systems Sci., 25 (1987), pp. 1–11.

[24] N. Z. Shor, *Dual quadratic estimates in polynomial and boolean programming*, Ann. Oper. Res., 25 (1990), pp. 163-168.

[25] T. Stephen and L. Tunçel, *On a representation of the matching polytope via semidefinite liftings*, Math. Oper. Res., 24 (1999), pp. 1–7.

[26] J. Stoer and C. Witzgall, *Convexity and Optimization in Finite Dimensions* I, Springer-Verlag, Berlin, 1970.

[27] A. Takeda, Y. Dai, M. Fukuda, and M. Kojima, *Towards the implementation of successive convex relaxation method for nonconvex quadratic optimization problem*, in Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems, P. M. Pardalos, ed., Kluwer Academic Press, Dordrecht, to appear.

[28] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[29] Y. Ye, *Approximating quadratic programming with bound and quadratic constraints*, Math. Programming, 84 (1999), pp. 219–226.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 779–808

SMOOTH SQP METHODS FOR MATHEMATICAL PROGRAMS WITH NONLINEAR COMPLEMENTARITY CONSTRAINTS[*]

HOUYUAN JIANG[†] [‡] AND DANIEL RALPH[†]

Abstract. Mathematical programs with nonlinear complementarity constraints are reformulated using better posed but nonsmooth constraints. We introduce a class of functions, parameterized by a real scalar, to approximate these nonsmooth problems by smooth nonlinear programs. This smoothing procedure has the extra benefits that it often improves the prospect of feasibility and stability of the constraints of the associated nonlinear programs and their quadratic approximations. We present two globally convergent algorithms based on sequential quadratic programming (SQP) as applied in exact penalty methods for nonlinear programs. Global convergence of the *implicit smooth SQP* method depends on existence of a lower-level nondegenerate (strictly complementary) limit point of the iteration sequence. Global convergence of the *explicit smooth SQP* method depends on a weaker property, i.e., existence of a limit point at which a generalized constraint qualification holds. We also discuss some practical matters relating to computer implementations.

Key words. mathematical programs with equilibrium constraints, bilevel optimization, complementarity problems, sequential quadratic programming, exact penalty, generalized constraint qualification, global convergence, smoothing method

AMS subject classifications. 90C, 90D65

PII. S1052623497332329

1. Introduction.

Mathematical programs with equilibrium constraints (MPECs) form a relatively new and interesting class of optimization problems. The roots of MPECs lie in game theory and especially in bilevel optimization. MPECs include a number of significant applications in economics and engineering. See the monograph [28] for comprehensive theoretical treatment, applications, and references.

The MPEC considered in this paper is a mathematical program with nonlinear complementarity problem (NCP) constraints:

$$
\begin{array}{rl}
\min\limits_{x,y} & f(x,y) \\
\text{subject to} & g(x,y) \geq 0, \\
 & 0 \leq F(x,y) \ \perp y \ \geq 0,
\end{array}
\tag{1}
$$

where $f : \Re^{n+m} \to \Re$, $g : \Re^{n+m} \to \Re^l$, $F : \Re^{n+m} \to \Re^m$ are continuously differentiable and $w \perp y$ indicates orthogonality of any vectors $w, y \in \Re^m$. The constraints $g(x,y) \geq 0$ are called the upper-level constraints. By lower-level or equilibrium constraints we mean the system $0 \leq F(x,y) \perp y \geq 0$, which constitutes a nonlinear complementarity problem in y for each fixed x.

We omit equality constraints in the upper level for simplicity, but these can easily be handled and would be useful for the following case. Lower-level mixed complementarity constraints [7] can be dealt with quite easily by moving equations and their

[*] Received by the editors January 8, 1998; accepted for publication (in revised form) July 21, 1999; published electronically March 30, 2000. This work was supported by the Australian Research Council.

http://www.siam.org/journals/siopt/10-3/33232.html

[†] Department of Mathematics and Statistics, University of Melbourne, Parkville, Victoria 3052, Australia (danny@ms.unimelb.edu.au).

[‡] Current address: CSIRO Mathematical and Information Science, Private Bag 2, Glen Osmond, SA 5064, Australia (Houyuan.Jiang@cmis.csiro.au).

associated variables to the upper level. For example, consider the following lower-level mixed complementarity constraints

$$\begin{aligned}&F_1(x,y,z)=0,\\&0\le F_2(x,y,z)\ \perp\ z\ge 0,\end{aligned}$$

where $F_1:\Re^{n+m_1+m_2}\to\Re^{m_1}$, $F_2:\Re^{n+m_1+m_2}\to\Re^{m_2}$. By renaming the tuple (x,y) as the upper-level vector and z as the lower-level vector, and moving the equations $F_1(x,y,z)=0$ to the upper level, we obtain an MPEC with upper-level constraints that are specified by nonlinear equalities and inequalities and by lower-level nonlinear complementarity constraints.

Clearly, the MPEC (1) is equivalent to the smooth nonlinear program (NLP) obtained by writing the complementarity condition $F(x,y)\perp y$ as an inner product $F(x,y)^Ty=0$. Unfortunately, it has been proved [4] that the Mangasarian–Fromovitz constraint qualification does not hold at any feasible point of this smooth NLP even if the usual inequality constraints $g(x,y)\ge 0$ are omitted and the lower-level NCP problem has very fine properties such as strong monotonicity with respect to y. Since this constraint qualification is almost synonymous with numerical stability of the feasible set, its failure to hold suggests that well-developed nonlinear programming theory and numerical methods are not readily applicable to solving this form of MPEC: the feasible set of the smooth NLP is numerically ill posed. See [19, 28] for more discussions and numerical examples.

Instead we let $w=F(x,y)$ and substitute a nonsmooth equation $\Phi(y,w)=0\in\Re^m$, constructed using the Fischer–Burmeister functional [9], for example, for the complementarity problem $y,w\ge 0$, $y^Tw=0$:

$$(2)\qquad\begin{aligned}\min_{x,y,w}\quad & f(x,y)\\ \text{subject to}\quad & g(x,y)\ge 0,\\ & F(x,y)-w=0,\\ & \Phi(y,w)=0.\end{aligned}$$

The mapping Φ is then "smoothed" by introducing a parameterization $\Psi(y,w,\mu)$ that is differentiable if the scalar μ is nonzero but coincides with $\Phi(y,w)$ when $\mu=0$. By a smoothing method we mean an algorithm that solves (1) either by solving an augmented problem like

$$(3)\qquad\begin{aligned}\min_{x,y,w,\mu}\quad & f(x,y)\\ \text{subject to}\quad & g(x,y)\ge 0,\\ & F(x,y)-w=0,\\ & \Psi(y,w,\mu)=0,\\ & e^\mu-1=0,\end{aligned}$$

where e is Euler's constant, so that the last constraint requires $\mu=0$ (cf. [18] for complementarity problems), or by approximately solving the following problem for a sequence of values $\mu=\mu_k\to 0$:

$$(4)\qquad\begin{aligned}\min_{x,y,w}\quad & f(x,y)\\ \text{subject to}\quad & g(x,y)\ge 0,\\ & F(x,y)-w=0,\\ & \Psi(y,w,\mu)=0.\end{aligned}$$

The introduction of the smoothing parameter μ has three consequences: Nonsmooth problems are transformed into smooth problems, except when $\mu = 0$; well-posedness can be improved in the sense that feasibility and constraint qualifications, hence stability, are often more likely to be satisfied for all values of μ; and solvability of quadratic approximation problems is improved. This opens the way to use sequential quadratic programming (SQP) methods from classical nonlinear programming.

The methods presented in this paper follow some ideas from [8, 12] which try to use well-developed numerical methods for the solution of smooth nonlinear programs. In [8], smooth nonlinear programs of the type (4) are formed and assumed to be solvable by an unspecified (black box) method. Under further conditions, which will be relaxed in the explicit smoothing method to be presented in section 6, it is shown that limit points of the sequence of approximate solutions of the parametric nonlinear programs satisfy generalized Karush–Kuhn–Tucker (KKT) conditions [16] given in terms of the Clarke generalized derivatives [5]. We call this an *explicit* smoothing method because the smoothing parameter is updated separately from the direction-finding process. In [12] another explicit smoothing method is proposed, which is an SQP-based method for MPECs with linear complementarity constraints and upper-level constraints only on x, and limit points satisfying a lower-level nondegeneracy (strict complementarity) condition are shown to be piecewise stationary points for (1).

This paper details methods for solving the problems (2) and (3) using SQP in an ℓ_1-exact penalty framework. The first method, *implicit smooth SQP*, applies to (3); Theorem 5.10 assumes lower-level nondegeneracy at limit points amongst other conditions to ensure that limit points of the iteration sequence are piecewise stationary points of (1). The term *implicit* means that the smoothing parameter is included as one of the variables of the problem formulation and updated at each iteration using the QP solution, like the other variables. This convergence result is not surprising given that lower-level nondegeneracy at a feasible point of (1) implies locally that the problem is a smooth nonlinear program.

To move beyond the realm of standard nonlinear programming, we present the *explicit smooth SQP* method that is aimed at solving the problem (2) by solving a sequence of problems (4), where we expect $\mu = \mu_k \to 0$ and limit points of the iteration sequence to satisfy generalized KKT conditions of (2). Perhaps the most novel result is Theorem 6.4, which extends global convergence theory for exact penalty methods in nonlinear programming to MPECs by using a generalized constraint qualification at limit points of the iteration sequence. Explicit smooth SQP can be viewed as an implementation of the smoothing method of [8] though the convergence analysis of the new method is more demanding.

Our main goal is to explore convergence conditions and analysis for smooth SQP methods. Given this and the length of the paper, a numerical investigation will be pursued in a future publication.

We mention that the development of numerical methods for the solution of MPECs is at a less advanced stage than optimality theory [4, 26, 27, 28, 29, 30, 32, 33, 34, 40, 43]. When the upper-level constraints exclude y, i.e., take the form $g(x) \geq 0$, the implicit function approach may be possible. In this approach it is assumed that y can be found as a function of x by solving the NCP appearing in the constraints, and the MPEC is collapsed to the problem of minimizing the nondifferentiable objective function $f(x, y(x))$ subject to $g(x) \geq 0$. This nonsmooth problem can be tackled by bundle methods as proposed in [23, 24, 33, 34] or using another nonsmooth optimization method such as Shor's R-algorithm as implemented in SolvOpt [25]; see [7] for some computational comparisons. However, with mixed upper-level constraints, i.e., involving y and possibly x, the implicit programming approach transforms an MPEC into a problem with nondifferentiable constraints in addition to a nonsmooth objective, a format which has not been seriously studied with regard to computational methods.

Some methods which can be extended to handle mixed upper-level constraints include the penalty interior-point algorithm (PIPA) [28]; smoothing methods [8, 12], which are related to the interior-point approach; and piecewise sequential quadratic programming (PSQP) [28, 29, 37]. Apart from this paper, the only implementations of these algorithms we know of that handle joint upper-level constraints are discussed in [19]. PIPAs converge globally under suitable conditions, at least in the implicit case [28], while the PSQP method exhibits local superlinear convergence under the uniqueness of multipliers and some second-order sufficient conditions, but surprisingly without requiring a strict complementarity condition. Some preliminary numerical experiments have been carried out for the PIPA and PSQP [28, 29, 19], and smoothing methods [8, 12]. See also [7] for a comparison of PIPA with implicit programming methods. The theoretical results and numerical experience show some promise for these methods. We also refer the reader to [24, 33, 34, 41] for other numerical methods and applications of MPECs.

The rest of the paper is organized as follows. In the next section, we review first-order optimality theory for nonsmooth programs using the Clarke calculus. In section 3, we reformulate the MPEC (1) into equivalent (in the sense of global optima, local optima, generalized stationarity, or piecewise stationarity as the case may be) but generally better-posed nonsmooth programs by means of functions introduced there. Constraint qualifications for the reformulated nonsmooth programs are studied in section 4. In section 5, we present implicit smooth SQP for solving the reformulation (3) and give details of global convergence under lower-level nondegeneracy at limit points. In section 6, we propose explicit smooth SQP and establish its global convergence to generalized KKT points under generalized constraint qualifications; the analogs of the various results developed in section 5 are given here. Section 7 briefly gives concrete examples of smoothing functions from the literature.

A word about notation: For a locally Lipschitz real-valued function f and a vector-valued locally Lipschitz function H, ∂f and ∂H denote the Clarke generalized subgradient and the Clarke generalized Jacobian, respectively; see [5]. For a continuously differentiable real-valued function f and a vector-valued continuously differentiable function H, we use ∇f and F' to indicate the gradient of f and the Jacobian of H. If x_1 and x_2 are two vectors with the same dimension, then $x_1^T x_2$ denotes the inner products of these two vectors. By $\|\cdot\|$, we mean the Euclidean norm. $\Re^n$ denotes the real Euclidean space of column vectors of length n; for $u \in \Re^n$ and $v \in \Re^m$, (u, v) denotes the column vector $[u^T \; v^T]^T$ in $\Re^{n+m}$.

2. Preliminaries on nonsmooth programming.

Consider the nonsmooth program (NSP):

$$\begin{array}{rl} \min\limits_{u} & f(u) \\ \text{subject to} & g(u) \geq 0, \\ & h(u) = 0, \end{array} \tag{5}$$

where $f : \Re^n \to \Re$, $g : \Re^n \to \Re^l$, and $h : \Re^n \to \Re^m$ are locally Lipschitz.

Definition 2.1. *The point u^* is said to be a generalized stationary point of* (5) *if there exists a KKT multiplier vector $(\lambda_g, \lambda_h) \in \Re^{l+m}$ such that the following generalized Karush–Kuhn–Tucker (GKKT) conditions hold:*

$$\begin{array}{l} \partial f(u^*) - \partial g(u^*)^T \lambda_g + \partial h(u^*)^T \lambda_h \ni 0, \\ 0 \le g(u^*) \ \perp \ \lambda_g \ge 0, \\ h(u^*) = 0, \end{array}$$

where ∂ denotes the Clarke generalized gradient for a scalar function and the Clarke generalized Jacobian for a vector-valued function [5].

If f, g, and h happen to be smooth at u^*, then the GKKT conditions reduce to the usual KKT condition:

$$\begin{array}{l} \nabla f(u^*) - g'(u^*)^T \lambda_g + h'(u^*)^T \lambda_h = 0, \\ 0 \le g(u^*) \ \perp \ \lambda_g \ge 0, \\ h(u^*) = 0. \end{array}$$

In this case, u^* is called a stationary point or a KKT point of (5).

For convenience, we may assume that in the above NSP, the first l_1 ($l_1 \le l$) inequality constraints are active and the rest are inactive at u^*, i.e.,

$$\begin{array}{ll} g_i(u^*) = 0, & 1 \le i \le l_1, \\ g_i(u^*) > 0, & i > l_1. \end{array}$$

Let

$$G(u) \ = \ \begin{pmatrix} g_1(u) \\ \vdots \\ g_{l_1}(u) \\ h_1(u) \\ \vdots \\ h_m(u) \end{pmatrix}.$$

Associated with the above NSP, we recall some well-known regularity conditions under which a local solution is a generalized stationary point [16].

Generalized Linear Independence Constraint Qualification (GLICQ). Each element of the generalized Jacobian $\partial G(u^*)$ [5] has full row rank.

Generalized Mangasarian–Fromovitz Constraint Qualification (GMFCQ). (i) there exists $d \in \Re^n$ such that for all elements $(A_1, \ldots, A_{l_1}, B_1, \ldots, B_m) \in \partial G(u^*)$,

$$\begin{array}{ll} A_i^T d > 0 & \text{for } i = 1, \ldots, l_1, \\ B_j^T d = 0 & \text{for } j = 1, \ldots, m; \end{array}$$

(ii) for any element of $(A_1, \ldots, A_{l_1}, B_1, \ldots, B_m) \in \partial G(u^*)$, $(B_1, \ldots, B_m)$ has full row rank.

Generalized Constant Rank Constraint Qualification (GCRCQ). There is a neighborhood of u^* such that for any u in this neighborhood, the rank of each element of the generalized Jacobian $\partial G(u)$ is invariant.

We mention that the above three constraint qualifications are slightly stronger than those given in [16] to keep notation simple. When (5) is defined by smooth (continuously differentiable) functions, the GLICQ and GMFCQ reduce to the classical LICQ and MFCQ (see [28, 31]); and, as in the smooth case, the GLICQ implies the GMFCQ. However, the CRCQ usually used in the smooth case [17] is stronger than the smooth version of the GCRCQ in that the former requires constant rank of submatrices of rows of the Jacobian $G'(u)$ for u near u^*. (We mention an example to distinguish these two CRCQs: let $g(u_1, u_2) = (u_1 + u_2, (u_1 + u_2)^2)$ and observe that the rank of $g'(u_1, u_2)$ is always 1, whereas the rank of $g'_2(u_1, u_2)$ is either 0 if $(u_1, u_2) = (0, 0)$, or 1 otherwise.)

Under these generalized CQs, Hiriart-Urruty [16] proved the optimality conditions in the following proposition. These optimality conditions also hold under the next assumption.

Piecewise Affine Constraint Condition (PACC). Both g and h are piecewise affine. See [40, section 2.1] for discussion of generalized stationarity which includes the PACC.

Proposition 2.2. *Suppose u^* is a local minimizer of the nonsmooth program* (5) *and one of the GCRCQ, GLICQ, GMFCQ, or PACC holds at u^*. Then u^* is a generalized stationary point of* (5)*. Furthermore, if f, g, and h are smooth at u^*, then u^* is a stationary point or a KKT point of* (5).

3. Equivalent reformulations of MPECs. As explained in section 1, the smooth nonlinear programming reformulation of the MPEC (1) is numerically ill posed. The strategy we use in this article is to approximate the MPEC by well-behaved nonlinear programming problems (NLP). To this end, we introduce a class of smoothing functions on which some properties are imposed as we proceed. Suppose the function $\psi : \Re^3 \to \Re$ satisfies the following assumptions:

(A1) ψ is locally Lipschitz and directionally differentiable on $\Re^3$, and ψ is continuously differentiable at every point (a, b, c) with $c \neq 0$.

(A2) $\psi(a, b, 0) = 0$ if and only if $a \geq 0,\ b \geq 0,\ ab = 0$.

Section 7 contains standard examples, all of which satisfy the assumptions (A1)–(A2) and the assumptions (A3)–(A5) to be introduced in what follows.

Let $\phi : \Re^2 \to \Re$ and the parametric function $\phi_c : \Re^2 \to \Re$ be defined for any $(a, b) \in \Re^2$ and $c \in \Re$ by

$$\phi(a, b) \ = \ \psi(a, b, 0)$$

and

$$\phi_c(a, b) \ = \ \psi(a, b, c).$$

Clearly, $\phi_0 \equiv \phi$. By means of the functions ϕ and ψ, we define two nonsmooth programs:

$$\begin{array}{rl} \min\limits_{x,y,w} & f(x, y) \\ \text{subject to} & g(x, y) \geq 0, \\ & F(x, y) - w = 0, \\ & \phi(y_i, w_i) = 0, \quad i = 1, \ldots, m, \end{array} \tag{6}$$

and

$$\begin{array}{rl} \min\limits_{x,y,w,\mu} & f(x, y) \\ \text{subject to} & g(x, y) \geq 0, \\ & F(x, y) - w = 0, \\ & \psi(y_i, w_i, \mu) = 0, \quad i = 1, \ldots, m, \\ & e^\mu - 1 = 0. \end{array} \tag{7}$$

It is easy to see that (6) and (7) are (2) and (3), respectively, with

$$\Psi(y,w,\mu) \quad = \quad \begin{pmatrix} \psi(y_1,w_1,\mu) \\ \vdots \\ \psi(y_m,w_m,\mu) \end{pmatrix}$$

and

$$\Phi(y,w) \quad = \quad \begin{pmatrix} \phi(y_1,w_1) \\ \vdots \\ \phi(y_m,w_m) \end{pmatrix} \quad = \quad \Psi(y,w,0).$$

Since differentiability of ϕ and ψ is not assumed at (a,b) and (a,b,c), respectively, (6) and (7) are nonsmooth programs in general. On the other hand, by the assumption (A1), when $\mu \neq 0$, the functions involved in (7) are smooth at (x,y,w,μ), which is a nice property to be used in the subsequent analysis. Next we give some relationships between the MPEC (1) and the nonsmooth programs (6) and (7).

Proposition 3.1. *Under the assumptions* (A1) *and* (A2)*, the following statements are equivalent.*

(i) (x,y) *is a feasible point (local solution, global solution) of* (1).

(ii) (x,y,w) *with* $w = F(x,y)$ *is a feasible point (local solution, global solution) of* (6).

(iii) (x,y,w,μ) *with* $w = F(x,y)$ *and* $\mu = 0$ *is a feasible point (local solution, global solution) of* (7).

Proof. Given that $e^{\mu} - 1 = 0$ has a unique solution $\mu = 0$ and the assumption (A2) is satisfied, it is clear that all three statements are equivalent regarding feasible points. The equivalence with respect to local solutions or global solutions is an obvious consequence. □

Since f, g, and F are smooth, it can be shown, by Proposition 2.3.3 of [5] and its Corollary 1, that (x^*,y^*,w^*) is a generalized stationary point of (6) if and only if there exists a KKT multiplier vector $(\lambda_g,\lambda_F,\lambda_\Phi) \in \Re^{l+2m}$ such that the following GKKT conditions hold:

$$\begin{pmatrix} \nabla f(x^*,y^*) \\ 0 \end{pmatrix} - \begin{pmatrix} g'(x^*,y^*)^T \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} F'(x^*,y^*)^T \\ -I \end{pmatrix} \lambda_F$$

$$+ \begin{pmatrix} 0 \\ \partial\Phi(y^*,w^*) \end{pmatrix} \lambda_\Phi \ni 0, \tag{8}$$

$$\begin{aligned} &0 \le g(x^*,y^*) \ \perp \ \lambda_g \ge 0, \\ &F(x^*,y^*) - w^* = 0, \\ &\Phi(y^*,w^*) = 0, \end{aligned}$$

where 0 denotes appropriate zero vectors or matrices and $I \in \Re^{m\times m}$ is the identity matrix. Similarly, (x^*,y^*,w^*,μ^*) is a generalized stationary point of (7) if and only if there exists a KKT multiplier vector $(\lambda_g,\lambda_F,\lambda_\Psi,\lambda_\mu) \in \Re^{l+2m+1}$ such that the following GKKT conditions hold:

$$\begin{aligned}(9)\qquad & \begin{pmatrix} \nabla f(x^*,y^*) \\ 0 \end{pmatrix} - \begin{pmatrix} g'(x^*,y^*)^T \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} F'(x^*,y^*)^T \\ -I \\ 0 \end{pmatrix} \lambda_F \\ & + \begin{pmatrix} 0 \\ \partial\Psi(y^*,w^*,\mu^*) \end{pmatrix} \lambda_\Psi + \begin{pmatrix} 0 \\ e^{\mu^*} \end{pmatrix} \lambda_\mu \ni 0, \\ & 0 \le g(x^*,y^*) \ \perp\ \lambda_g \ge 0, \\ & F(x^*,y^*) - w^* = 0, \\ & \Psi(y^*,w^*,\mu^*) = 0, \\ & e^{\mu^*} - 1 = 0. \end{aligned}$$

Note in (9) that $\mu^* = 0$.

The assumption (A1) ensures the inclusion

$$\Pi_{ab}\partial\psi(a,b,0) \ \supseteq\ \partial\phi(a,b)$$

for any $(a,b) \in \Re^2$, where Π_{ab} denotes the projection operator on $\Re^3$: $\Pi_{ab}(\alpha,\beta,0) = (\alpha,\beta)$; see Proposition 2.3.16 in [5]. We introduce another assumption to ensure that these sets are in fact identical.

(A3) $\Pi_{ab}\partial\psi(a,b,0) = \partial\phi(a,b)$ for any $(a,b) \in \Re^2$, where $\phi(a,b)$ is defined as $\psi(a,b,0)$.

A direct consequence of the assumption (A3) is that

$$\Pi_{ab}\partial\Psi(y,w,0) \ =\ \partial\Phi(y,w) \quad \forall (y,w) \in \Re^{2m}.$$

PROPOSITION 3.2. *Under the assumptions* (A1)–(A2)*, if* (x^*,y^*,w^*) *is a generalized stationary point of* (6)*, then* $(x^*,y^*,w^*,0)$ *is a generalized stationary point of* (7)*. Conversely, if* (A3) *holds as well as* (A1)–(A2)*, and if* $(x^*,y^*,w^*,0)$ *is a generalized stationary point of* (7)*, then* (x^*,y^*,w^*) *is a generalized stationary point of* (6)*.*

Proof. Suppose (x^*,y^*,w^*) is a generalized stationary point of (6); then there exists a KKT multiplier vector $(\lambda_g,\lambda_F,\lambda_\Phi)$ such that (8) holds. Let λ_μ be an element of $-\partial_\mu\Psi(y^*,w^*,\mu^*)\lambda_\Phi$ with $\mu^* = 0$. It follows from the remark before the assumption (A3) that $(\lambda_g,\lambda_F,\lambda_\Phi,\lambda_\mu)$ is a KKT multiplier satisfying (9); i.e., $(x^*,y^*,w^*,0)$ is a generalized stationary point of (7). Conversely, if $(x^*,y^*,w^*,0)$ is a generalized stationary point of (7), it is easy to see from the assumption (A3) and the GKKT conditions (8) and (9) that (x^*,y^*,w^*) is a generalized stationary point of (6). □

By Propositions 3.1 and 3.2, (6) and (7) are completely equivalent in the sense that global solutions, local solutions, generalized stationary points, and feasible points correspond to one another. However, it is not yet clear what relationships the optimality condition of the MPEC (1) and the nonlinear programming problems (6) and (7) have.

Let $z^* = (x^*,y^*)$ be a feasible point of the MPEC (1). Let $\mathcal{F}$ be the feasible set of (1), i.e.,

$$\mathcal{F} \ =\ \{z = (x,y)\ :\ g(z) \ge 0,\ 0 \le F(z) \ \perp\ y \ge 0\}.$$

Denote by $\mathcal{T}(z^*,\mathcal{F})$ the tangent cone to $\mathcal{F}$ at z^*: $\mathcal{T}(z^*,\mathcal{F})$ is the set of limit points of sequences $\{\frac{z^k - z^*}{\tau_k}\}$, where $\{z^k\} \subseteq \mathcal{F}$ converges to z^* and $\tau_k \downarrow 0$.

DEFINITION 3.3. *A point $z^* \in \mathcal{F}$ is said to be a primal stationary* [28] *or B-stationary* [40] *point of the MPEC* (1) *if the following condition holds:*

$$\nabla f(z^*)^T d \geq 0 \;\forall d \in \mathcal{T}(z^*, \mathcal{F}).$$

A decomposition or disjunction technique was very useful in establishing optimality conditions for MPECs in [28]. For any feasible point $z^* \in \mathcal{F}$, define

$$\begin{array}{rcl} \alpha(z^*) & = & \{1 \leq i \leq m : F_i(z^*) = 0 < y_i^*\}, \\ \beta(z^*) & = & \{1 \leq i \leq m : F_i(z^*) = 0 = y_i^*\}, \\ \gamma(z^*) & = & \{1 \leq i \leq m : F_i(z^*) > 0 = y_i^*\}, \end{array} \tag{10}$$

and the family of index sets

$$\mathcal{A}(z^*) \;=\; \{(\mathcal{J}, \mathcal{K}) \,:\, \mathcal{J} \supseteq \alpha(z^*),\, \mathcal{K} \supseteq \gamma(z^*),\, \mathcal{J} \cap \mathcal{K} = \emptyset,\, \mathcal{J} \cup \mathcal{K} = \{1, 2, \ldots, m\}\}.$$

For each partition $\mathcal{J} \cup \mathcal{K}$ of $\{i \,:\, 1 \leq i \leq m\}$, let

$$\begin{array}{rcll} \mathcal{F}_{(\mathcal{J},\mathcal{K})} & = & \{z : & g(z) \geq 0, \\ & & & F_i(z) = 0 \leq y_i \quad \forall i \in \mathcal{J}, \\ & & & F_i(z) \geq 0 = y_i \quad \forall i \in \mathcal{K} \;\}. \end{array}$$

Using the family of sets $\{\mathcal{F}_{(\mathcal{J},\mathcal{K})} : (\mathcal{J}, \mathcal{K}) \in \mathcal{A}(z^*)\}$, the feasible set $\mathcal{F}$ of (1) can be locally decomposed at any feasible point $z^* \in \mathcal{F}$, and hence stationarity conditions for (1) defined in [28] can be characterized in terms of traditional nonlinear programs associated with each $\mathcal{F}_{(\mathcal{J},\mathcal{K})}$, which has the form of a standard nonlinear programming feasible region. The disjunctive approach can be carried over to constraint stability.

Piecewise Constraint Qualification at a Point $z^* \in \mathcal{F}$. For each $(\mathcal{J}, \mathcal{K}) \in \mathcal{A}(z^*)$, the above representation of $\mathcal{F}_{(\mathcal{J},\mathcal{K})}$ satisfies a standard smooth constraint qualification at z^* (for example, the MFCQ, LICQ, or CRCQ).

We now state a disjunctive first-order optimality condition studied in [28], where it was called "primal-dual stationarity"; we call it "piecewise stationarity" to distinguish it from generalized stationarity, which also has a primal-dual flavor.

DEFINITION 3.4. *A point $z^* = (x^*, y^*)$ is a piecewise stationary point of the MPEC* (1) *if it is feasible and, for each $(\mathcal{J}, \mathcal{K}) \in \mathcal{A}(z^*)$, there exist KKT multipliers $\xi \in \Re^l$, $\eta \in \Re^m$, and $\pi \in \Re^m$ such that*

$$\begin{array}{l} \nabla_x f(x^*, y^*) - g_x'(x^*, y^*)^T \xi - F_x'(x^*, y^*)^T \eta = 0, \\ \nabla_y f(x^*, y^*) - g_y'(x^*, y^*)^T \xi - F_y'(x^*, y^*)^T \eta - \pi = 0, \\ 0 \leq g(x^*, y^*) \;\perp\; \xi \geq 0, \\ F_i(x^*, y^*) = 0, \; 0 \leq y_i^* \;\perp\; \pi_i \geq 0 \quad \forall i \in \mathcal{J}, \\ 0 \leq F_i(x^*, y^*) \;\perp\; \eta_i \geq 0, \; y_i^* = 0 \quad \forall i \in \mathcal{K}. \end{array} \tag{11}$$

The next result is essentially due to [28].

PROPOSITION 3.5. *Let $z^* = (x^*, y^*)$. If z^* is a piecewise stationary point of the MPEC* (1), *then it is primal stationary for* (1). *Conversely, if z^* is primal stationary for* (1) *and a piecewise constraint qualification holds at z^*, then z^* is piecewise stationary for* (1).

The idea of strict complementarity of a solution of a complementarity problem is adapted to nonfeasible points of the MPEC (1).

Definition 3.6. *A point $(x, y) \in \Re^{n+m}$ is said to be lower-level nondegenerate if $y_i \neq F_i(x, y)$ for $i = 1, \ldots, m$. A point $(x, y, w) \in \Re^{n+2m}$ is said to be lower-level nondegenerate if $y_i \neq w_i$ for $i = 1, \ldots, m$.*

Suppose $z^* = (x^*, y^*)$ is feasible for the MPEC (1). Then lower-level nondegeneracy of (x^*, y^*) is equivalent to the strict complementarity condition: for any i $(1 \leq i \leq m)$, either $y_i^* > 0 = F_i(x^*, y^*)$ or $y_i^* = 0 < F_i(x^*, y^*)$. Lower-level nondegeneracy of (x^*, y^*) is also equivalent to lower-level nondegeneracy of (x^*, y^*, w^*) with $w^* = F(x^*, y^*)$ and to the family of index sets $\mathcal{A}(z^*)$ reducing to a singleton, i.e., $\mathcal{A}(z^*) = \{(\alpha(z^*), \gamma(z^*))\}$. If the function Φ is continuously differentiable at such a feasible lower-level nondegenerate point (x^*, y^*, w^*), then piecewise stationarity of (6) at (x^*, y^*, w^*) coincides with the classical KKT conditions.

The next result shows that stationarity conditions on the MPECs (1), (6), and (7) coincide at lower-level nondegenerate points. To this end, we impose another condition on the function ψ.

(A4) If $\psi(a, b, 0) = 0$ and $(p, q, r) \in \partial\psi(a, b, 0)$, then

$$p^2 + q^2 > 0, \qquad pq \geq 0,$$

and

$$\begin{array}{ll} p = 0,\ q \neq 0 & \text{if } a > 0, \\ p \neq 0,\ q = 0 & \text{if } b > 0. \end{array}$$

Proposition 3.7. *Suppose (x^*, y^*) is a lower-level nondegenerate feasible point of the MPEC* (1). *Assume that the assumptions* (A1)–(A4) *are satisfied. Then the following statements are equivalent.*

(i) *(x^*, y^*) is a piecewise stationary point of the MPEC* (1).

(ii) *(x^*, y^*, w^*) is a (generalized) stationary point of* (6)*, where $w^* = F(x^*, y^*)$.*

(iii) *(x^*, y^*, w^*, μ^*) is a (generalized) stationary point of* (7)*, where $w^* = F(x^*, y^*)$, $\mu^* = 0$.*

Proof. (i) $\Longrightarrow$ (ii). Let $\mathcal{J} = \alpha(z^*)$ and $\mathcal{K} = \gamma(z^*)$. $(\mathcal{J}, \mathcal{K})$ is the unique element of $\mathcal{A}(z^*)$ since $\beta(z^*) = \emptyset$. It follows that there exist multipliers $\xi \in \Re^l$, $\eta \in \Re^m$, and $\pi \in \Re^p$ such that (11) holds.

Let $\lambda_g = \xi$, $\lambda_F = -\eta$. We now define a vector λ_Φ. For $i \in \mathcal{J} = \alpha(z^*)$ and $(A_i, B_i) \in \partial\phi(y_i^*, w_i^*)$, the assumption (A4) implies that $A_i = 0, B_i \neq 0$. Therefore, $(\lambda_\Phi)_i = \frac{(\lambda_F)_i}{B_i}$ is well defined for any $i \in \alpha(z^*)$. Similarly, $(\lambda_\Phi)_i = \frac{(-\pi)_i}{A_i}$ is well defined with $(A_i, B_i) \in \partial\phi(y_i^*, w_i^*)$ for any $i \in \gamma(z^*)$ by the assumption (A4).

By the assumption (A4), it is easy to verify that $(\lambda_g, \lambda_F, \lambda_\Phi)$ is a KKT multiplier such that the GKKT conditions (8) hold, i.e., such that (x^*, y^*, w^*) is a generalized stationary point of (6).

(ii) $\Longrightarrow$ (i). Suppose there exists a KKT multiplier $(\lambda_g, \lambda_F, \lambda_\Phi) \in \Re^{l+2m}$ such that (8) holds at (x^*, y^*, w^*). Let

$$\begin{pmatrix} A \\ B \end{pmatrix} \in \partial\Phi(y^*, w^*).$$

Clearly $A = \operatorname{diag}(A_1, \ldots, A_n)$ and $B = \operatorname{diag}(B_1, \ldots, B_n)$ are diagonal matrices. Since $z^* = (x^*, y^*)$ is lower-level nondegenerate, $y_i^* \neq w_i^*$ for $i = 1, \ldots, m$ and $\beta(z^*) = \emptyset$. By the assumptions (A3) and (A4), $A_i = 0$ for $i \in \alpha(z^*)$ and $B_i = 0$ for $i \in \gamma(z^*)$. Moreover, it can be shown from (8) that $-\lambda_F + B\lambda_\Phi = 0$, i.e., $\lambda_F = B\lambda_\Phi$. Let $\xi = \lambda_g$, $\eta = -\lambda_F = -B\lambda_\Phi$, and $\pi = -A\lambda_\Phi$. We immediately obtain that (ξ, η, π) is a KKT multiplier such that (11) holds for the given $(\mathcal{J},\mathcal{K})$ such that $\mathcal{J}=\alpha(z^*)$ and $\mathcal{K}=\gamma(z^*)$. Since z^* is a lower-level nondegenerate feasible point, $(\mathcal{J},\mathcal{K})$ is the only element in $\mathcal{A}(z^*)$. This proves that (i) holds.

The desired results follow from Proposition 3.2. □

We next study the relationship between the piecewise stationary point and the generalized stationary point of (6) or (7) under the assumptions (A1)–(A3).

Suppose (x^*,y^*) is a piecewise stationary point of the MPEC (1). It turns out that $(dx,dy)=0\in\Re^{n+m}$ is a local solution of the MPEC

$$\begin{array}{rl}\displaystyle\min_{dx,dy} & \nabla f(x^*,y^*)^T(dx,dy)\\ \text{subject to} & g'(x^*,y^*)(dx,dy)+g(x^*,y^*)\geq 0,\\ & 0\leq y^*+dy\ \perp\ F(x^*,y^*)+F'(x^*,y^*)(dx,dy)\geq 0,\end{array}$$

or that $(dx,dy,dw)=0\in\Re^{n+2m}$ is a local solution of the MPEC

$$\begin{array}{rl}\displaystyle\min_{dx,dy,dw} & \nabla f(x^*,y^*)^T(dx,dy)\\ \text{subject to} & g'(x^*,y^*)(dx,dy)+g(x^*,y^*)\geq 0,\\ & F(x^*,y^*)+F'(x^*,y^*)(dx,dy)-(w^*+dw)=0,\\ & 0\leq y^*+dy\ \perp\ w^*+dw\geq 0,\end{array}$$

or, by Proposition 3.2, that $(dx,dy,dw)=0\in\Re^{n+2m}$ is a local solution of the nonsmooth program

$$\begin{array}{rl}\displaystyle\min_{dx,dy,dw} & \nabla f(x^*,y^*)^T(dx,dy)\\ \text{subject to} & g'(x^*,y^*)(dx,dy)+g(x^*,y^*)\geq 0,\\ & F(x^*,y^*)+F'(x^*,y^*)(dx,dy)-(w^*+dw)=0,\\ & \Phi(y^*+dy,w^*+dw)=0.\end{array}\tag{12}$$

Then, under the GLICQ or GMFCQ at 0 on the last nonsmooth problem (12), which is equivalent to the GLICQ or GMFCQ at (x^*,y^*,w^*) on the problem (6), we have that $0\in\Re^{n+2m}$ is a generalized stationary point of (12), which is equivalent to saying that (x^*,y^*,w^*) is a generalized stationary point of (6). Similarly, if the PACC holds for (6), so that g, F, and ϕ are affine functions, then piecewise stationarity of (1) implies generalized stationarity.

The following result summarizes the above discussion.

PROPOSITION 3.8. *Suppose the assumptions* (A1)–(A3) *hold. Suppose* (x^*,y^*) *is a piecewise stationary point of the MPEC* (1). *Assume that the GLICQ, GMFCQ, or PACC is satisfied at* (x^*,y^*,w^*) *with* $w^*=F(x^*,y^*)$ *for the problem* (6). *Then* (x^*,y^*,w^*) *is a generalized stationary point of the problem* (6) *and* (x^*,y^*,w^*,μ^*) *with* $\mu^*=0$ *is a generalized stationary point of the problem* (7).

Remark. The PACC applies in particular when g and F are affine and $\phi(a,b)=\min\{a,b\}$; see also section 7.

We point out that the converse of the above proposition does not hold in general. This can be demonstrated by the following example, which also shows that the definition of generalized stationary points is much weaker than that of piecewise stationary points.

Example 3.1. Consider the following MPEC:

$$\begin{array}{rl}\displaystyle\min_{x,y} & 0.5x^2+0.5y^2+x-y\\ \text{subject to} & 0\leq(y-x)\ \perp\ y\geq 0.\end{array}$$

This MPEC has a unique piecewise stationary point $(-1,0)$. Let

$$\psi(a,b,c) \;=\; \sqrt{a^2+b^2+c^2}-(a+b)$$

and $\phi(a,b)=\psi(a,b,0)$, which is the Fischer–Burmeister functional [9]; see also section 7. Clearly, ψ and ϕ satisfy the assumptions (A1)–(A4). However, $(0,0,0)$ is a generalized stationary point of the problem (6). Note that the feasible point $(0,0)$ of this MPEC is lower-level degenerate (strict complementarity fails).

4. Constraint qualifications for MPECs. For a given feasible point $z^*=(x^*,y^*)$ of the MPEC (1), let α, β, and γ be the respective index sets $\alpha(z^*)$, $\beta(x^*)$, and $\gamma(z^*)$ defined in (10). The MPEC is said to be R-regular in y at the feasible point (x^*,y^*) if the submatrix $F'_y(x^*,y^*)_{\alpha\alpha}$ of $F'_y(x^*,y^*)$ is nonsingular and the Schur complement

$$F'_y(x^*,y^*)_{\beta\beta}-F'_y(x^*,y^*)_{\beta\alpha}F'_y(x^*,y^*)^{-1}_{\alpha\alpha}F'_y(x^*,y^*)_{\alpha\beta}$$

is a P-matrix.

Consider the constraint mapping

$$H(x,y,w,\mu)=\begin{pmatrix} g(x,y)\\ F(x,y)-w\\ \Psi(y,w,\mu)\\ e^{\mu}-1\end{pmatrix} \tag{13}$$

associated with (7). Let V be an element of the generalized Jacobian of this mapping at (x^*,y^*,w^*,μ^*) with $\mu^*=0$:

$$V=\begin{pmatrix} g'_x(x^*,y^*) & g'_y(x^*,y^*) & 0 & 0\\ F'_x(x^*,y^*) & F'_y(x^*,y^*) & -I & 0\\ 0 & A & B & C\\ 0 & 0 & 0 & 1\end{pmatrix},$$

where $(A,B,C)\in\partial\Psi(y^*,w^*,\mu^*)$ and A and B are appropriate diagonal matrices. Then the submatrix $\bar{V}$ of V corresponding to the equilibrium constraints (i.e., the equality constraint functions $F(x,y)-w$, $\Psi(y,w,\mu)$, and $e^{\mu}-1$) is of the following form:

$$\bar{V}=\begin{pmatrix} F'_x(x^*,y^*) & F'_y(x^*,y^*) & -I & 0\\ 0 & A & B & C\\ 0 & 0 & 0 & 1\end{pmatrix}. \tag{14}$$

Clearly, the matrix V has full row rank if the following matrix is nonsingular:

$$U \;=\; \begin{pmatrix} F'_y(x^*,y^*) & -I\\ A & B\end{pmatrix}. \tag{15}$$

The following lemma and proposition can be proved in a standard way in the literature of nonlinear complementarity problems. See, for example, [42, Proposition 2.1] and also [10, Theorem 9].

LEMMA 4.1. *Suppose for $M,N,E\in\Re^{m\times m}$, M and N are diagonal matrices such that MN is positive semidefinite, M^2+N^2 is positive definite, and E is a P-matrix. Then $M+NE$ is nonsingular.*

Proposition 4.2. *Suppose the MPEC* (1) *is R-regular in y at a feasible point* (x^*, y^*) *of the MPEC* (1). *Then under the assumptions* (A1)–(A4), *the matrix U defined in* (15) *is nonsingular.*

We now study generalized constraint qualifications for the problems (6) and (7) (or (2) and (3), respectively) at the feasible point (x^*, y^*, w^*) and (x^*, y^*, w^*, μ^*) under R-regularity, respectively. Recall that $V \in \partial H(x^*, y^*, w^*, \mu^*)$. By Proposition 4.2, an equivalent reduction of the matrix V is the following matrix (reduction of a matrix under nonsingular transformation):

$$\begin{pmatrix} g'_x(x^*,y^*) - g'_y(x^*,y^*)(U^{-1})_{yy}F'_x(x^*,y^*) & 0 & 0 & 0 \\ F'_x(x^*,y^*) & F'_y(x^*,y^*) & -I & 0 \\ 0 & A & B & C \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

where A, B, C, and U are defined in (14) and (15) and $(U^{-1})_{yy}$ is a submatrix of the matrix U^{-1},

$$U^{-1} = \begin{pmatrix} (U^{-1})_{yy} & (U^{-1})_{yw} \\ (U^{-1})_{wy} & (U^{-1})_{ww} \end{pmatrix}.$$

By this observation, the following results can easily be verified.

Proposition 4.3. *Suppose* (x^*, y^*) *is a feasible point of the MPEC* (1), *the MPEC is R-regular at this point, and the assumptions* (A1)–(A4) *are satisfied. Let* $w^* = F(x^*, y^*)$ *and* $\mu^* = 0$ *and define* $\mathcal{I}_g = \{i : \; g_i(x^*, y^*) = 0\}$ *and*

$$\Gamma = g'_x(x^*,y^*) - g'_y(x^*,y^*)(U^{-1})_{yy}F'_x(x^*,y^*).$$

Then the following conclusions hold.

(i) *The GCRCQ holds for* (6) *at* (x^*, y^*, w^*) *and for* (7) *at* (x^*, y^*, w^*, μ^*) *if the row submatrix* $\Gamma_{\mathcal{I}_g}$ *corresponding to the active indexes of g at* (x^*, y^*) *for any U defined in* (15) *has constant rank around* (x^*, y^*). *In particular, the GCRCQ holds for* (6) *at* (x^*, y^*, w^*) *and for* (7) *at* (x^*, y^*, w^*, μ^*) *if* $g(x, y) = g(x)$ *and the matrix* $g'(x^*)_{\mathcal{I}_g}$ *has constant rank around* x^*.

(ii) *The GLICQ holds for* (6) *at* (x^*, y^*, w^*) *and for* (7) *at* (x^*, y^*, w^*, μ^*) *if the row submatrix* $\Gamma_{\mathcal{I}_g}$ *for any U defined in* (15) *has full row rank. In particular, the GLICQ holds for* (6) *at* (x^*, y^*, w^*) *and for* (7) *at* (x^*, y^*, w^*, μ^*) *if* $g(x, y) = g(x)$ *and the matrix* $g'(x^*)_{\mathcal{I}_g}$ *has full row rank.*

(iii) *The GMFCQ holds for* (6) *at* (x^*, y^*, w^*) *and for* (7) *at* (x^*, y^*, w^*, μ^*) *if there exists a vector* $d \in \Re^n$ *such that for any U defined in* (15)

$$\Gamma_{\mathcal{I}_g} d > 0.$$

In particular, the GMFCQ holds for (6) *at* (x^*, y^*, w^*) *and it holds for* (7) *at* (x^*, y^*, w^*, μ^*) *if* $g(x, y) = g(x)$, *and there exists a vector* $d \in \Re^n$ *such that*

$$(g_i)'(x^*)d > 0 \quad \textit{for } i \in \mathcal{I}_g.$$

5. Implicit smooth SQP.

5.1. Background and the algorithm. By the assumption (A1), the nonsmooth programming problem (7) (or (3)) is smooth for any $\mu \neq 0$. This important feature allows us to use traditional nonlinear programming approaches such as SQP methods for solving MPECs. To this end, we introduce a quadratic program (QP) that approximates (7). For any given (x, y, w, μ) with $\mu \neq 0$ and $d = (dx, dy, dw, d\mu)$,

$$
\begin{aligned}
&\min_{d \in \Re^{n+2m+1}} && \nabla f(x,y)^T \begin{pmatrix} dx \\ dy \end{pmatrix} + \tfrac{1}{2} d^T W d \\
&\text{subject to} && g'(x,y) \begin{pmatrix} dx \\ dy \end{pmatrix} + g(x,y) \geq 0, \\
& && F'(x,y) \begin{pmatrix} dx \\ dy \end{pmatrix} - dw + (F(x,y) - w) = 0, \\
& && A dy + B dw + C d\mu + \Psi(y, w, \mu) = 0, \\
& && e^{\mu} d\mu + e^{\mu} - 1 = 0,
\end{aligned}
\tag{16}
$$

where $\{(A, B, C)\} = \partial\Psi(x, y, \mu)$, which is singleton by the assumption (A1) since $\mu \neq 0$. An exact penalty merit function of (7) is defined by

$$
\begin{aligned}
\theta_\rho(x, y, w, \mu) \quad = \quad f(x,y) \quad + \quad \rho \Bigg[&\sum_{i=1}^{l} \max\{-g_i(x,y), 0\} \\
&+ \sum_{j=1}^{m} (|F_j(x,y) - w_j| + |\psi(y_j, w_j, \mu)|) + |e^{\mu} - 1| \Bigg],
\end{aligned}
$$

where ρ is a positive number. If two penalty parameters are used, then we may define another penalty function

$$
\begin{aligned}
\Theta_{(\rho^g, \rho^{\text{NCP}})}(x, y, w, \mu) = f(x,y) &+ \rho^g \sum_{i=1}^{l} \max\{-g_i(x,y), 0\} \\
&+ \rho^{\text{NCP}} \left[\sum_{j=1}^{m} (|F_j(x,y) - w_j| + |\psi(y_j, w_j, \mu)|) + |e^{\mu} - 1| \right],
\end{aligned}
$$

where ρ^g and ρ^{NCP} are two positive numbers. When $\rho^g = \rho^{\text{NCP}} = \rho$, $\Theta_{(\rho^g, \rho^{\text{NCP}})}$ reduces to the penalty function θ_ρ. It is easy to see that $\Theta_{(\rho^g, \rho^{\text{NCP}})}$ is not differentiable in general but directionally differentiable if ψ is directionally differentiable.

If the QP (16) has a solution d, then its KKT condition can be written as follows:

$$
\begin{aligned}
&\begin{pmatrix} \nabla_x f \\ \nabla_y f \\ 0 \\ 0 \end{pmatrix} + Wd - \begin{pmatrix} g'_x(x,y)^T \\ g'_y(x,y)^T \\ 0 \\ 0 \end{pmatrix} \lambda_g + \begin{pmatrix} F'_x(x,y)^T \\ F'_y(x,y)^T \\ -I \\ 0 \end{pmatrix} \lambda_F \\
&\qquad + \begin{pmatrix} 0 \\ A^T \\ B^T \\ C^T \end{pmatrix} \lambda_\Psi + \begin{pmatrix} 0 \\ 0 \\ 0 \\ e^{\mu} \end{pmatrix} \lambda_\mu = 0, \\
&0 \leq g'(x,y)(dx, dy) + g(x,y) \ \perp \ \lambda_g \geq 0, \\
&F'(x,y)(dx, dy) - dw + (F(x,y) - w) = 0, \\
&A dy + B dw + C d\mu + \Psi(y, w, \mu) = 0, \\
&e^{\mu} d\mu + e^{\mu} - 1 = 0,
\end{aligned}
\tag{17}
$$

where $(\lambda_g, \lambda_F, \lambda_\Psi, \lambda_\xi)$ is the corresponding KKT multiplier.

The existence of solutions to quadratic programs generated in traditional SQP methods plays a critical role. In particular, SQP fails if one of the associated quadratic programs is infeasible. In order to overcome QP infeasibility, some modifications have been introduced; see [1, 2]. Our strategy below is similar to that proposed in [1, 2] but with several notable differences. A modified quadratic program of (16) is defined as follows:

$$\tag{18}\begin{array}{lll} \displaystyle\min_{d\in\Re^{n+2m+1},\xi\in\Re^l} & & \nabla f(x,y)^T(dx,dy)+\frac{1}{2}d^TWd+\rho\sum_{i=1}^{l}\xi_i \\ & \text{subject to} & g'(x,y)(dx,dy)+g(x,y)\geq -\xi, \\ & & F'(x,y)(dx,dy)-dw+(F(x,y)-w)=0, \\ & & Ady+Bdw+Cd\mu+\Psi(y,w,\mu)=0, \\ & & e^{\mu}d\mu+e^{\mu}-1=0, \\ & & \xi\geq 0, \end{array}$$

where ρ is a positive penalty parameter. Note that if the constraint submatrix U given in (15) is invertible, then the second, third, and fourth block-rows of constraints can be solved for $(dy, dw, d\mu)$ in terms of dx. This means that by choice of ξ with sufficiently large components, the QP (18) is a feasible problem, an observation which is put to use in the next subsection to show that the modified SQP method is well defined.

If the modified QP (18) has a solution (d, ξ), then its KKT condition is a modification of the KKT condition (17):

$$\tag{19}\begin{array}{l} \begin{pmatrix} \nabla_x f \\ \nabla_y f \\ 0 \\ 0 \end{pmatrix} + Wd - \begin{pmatrix} g'_x(x,y)^T \\ g'_y(x,y)^T \\ 0 \\ 0 \end{pmatrix}\lambda_g + \begin{pmatrix} F'_x(x,y)^T \\ F'_y(x,y)^T \\ -I \\ 0 \end{pmatrix}\lambda_F \\ \qquad + \begin{pmatrix} 0 \\ A^T \\ B^T \\ C^T \end{pmatrix}\lambda_\Psi + \begin{pmatrix} 0 \\ 0 \\ 0 \\ e^{\mu} \end{pmatrix}\lambda_\mu = 0, \\ \rho\,\tilde{e} = \lambda_g + \lambda_\xi, \\ 0\leq g'(x,y)(dx,dy)+g(x,y)+\xi \ \perp\ \lambda_g\geq 0, \\ F'(x,y)(dx,dy)-dw+(F(x,y)-w)=0, \\ Ady+Bdw+Cd\mu+\Psi(y,w,\mu)=0, \\ e^{\mu}d\mu+e^{\mu}-1=0, \\ 0\leq \xi \ \perp\ \lambda_\xi\geq 0, \end{array}$$

where $\tilde{e}$ is the vector of all ones in $\Re^l$ and $(\lambda_g, \lambda_F, \lambda_\Psi, \lambda_\mu, \lambda_\xi)$ is the corresponding KKT multiplier.

The inequality constraints are perturbed by introducing a vector of artificial variables $\xi \in \Re^l$. This modification improves the prospect of the feasibility of the modified QP (18). One may also relax the equality constraints in the QP (16) by introducing further artificial variables. However, because of the special structure of the MPEC, we do not change the equality constraints. As shall be seen later, the modified QP (18) is always feasible under assumptions that are considered mild in the context of nonlinear complementarity problems.

Let $u = (x, y, w, \mu)$. We propose our first modified SQP method.

ALGORITHM: IMPLICIT SMOOTH SQP.

Step 0. (**Initialization**) Let $\rho_{-1} > 0$, $\delta_1 > 0$, $\delta_2 > 0$, $\sigma \in (0, 1)$, $\tau \in (0, 1)$. Choose $(x^0, y^0, w^0, \mu^0) \in \Re^{n+2m+1}$ such that $\mu^0 > 0$, and a symmetric positive definite matrix W_0 in $\Re^{(n+2m+1)\times(n+2m+1)}$. Set $k := 0$.

Step 1. (**Search direction**) Solve the modified QP (18) with $(x, y, w, \mu) = (x^k, y^k, w^k, \mu^k)$, $W = W_k$, and $\rho = \rho_{k-1}$. Let (d^k, ξ^k) be a solution of this modified QP and $\lambda^k = (\lambda_g^k, \lambda_F^k, \lambda_\Psi^k, \lambda_\mu^k, \lambda_\xi^k)$ be its corresponding multiplier.

Step 2. (**Termination check**) If a stopping rule is satisfied, terminate. Otherwise, go to Step 3.

Step 3. (**Penalty update**) Let

$$\tilde{\rho}_k = \begin{cases} \rho_{k-1} & \text{if } \rho_{k-1} \geq \max\limits_{1\leq i\leq l+2m+1} |\lambda_i^k|, \\ \delta_1 + \max\limits_{1\leq i\leq l+2m+1} |\lambda_i^k| & \text{otherwise.} \end{cases}$$

Define $\rho_k^g = \rho_{k-1}$, $\rho_k^{\mathrm{NCP}} = \tilde{\rho}_k$, and

$$\rho_k \;=\; \begin{cases} \tilde{\rho}_k & \text{if } \sum\limits_{1\leq i\leq l} \xi_i^k = 0, \\ \tilde{\rho}_k + \delta_2 & \text{otherwise.} \end{cases}$$

Step 4. (**Line search**) Let $t_k = (\tau)^{i_k}$, where i_k is the smallest nonnegative integer such that $i = i_k$ satisfies

$$\Theta_{(\rho_k^g, \rho_k^{\mathrm{NCP}})}(u^k + (\tau)^i d^k) - \Theta_{(\rho_k^g, \rho_k^{\mathrm{NCP}})}(u^k) \leq -\sigma(\tau)^i (d^k)^T W_k d^k.$$

Step 5. (**Update**) Let $u^{k+1} = u^k + t_k d^k$. Choose a symmetric positive definite matrix $W_{k+1} \in \Re^{(n+2m+1)\times(n+2m+1)}$. Set $k := k + 1$. Go to Step 1.

Remarks.

(i) If the modified QP (18) is replaced by the QP (16) to generate the search direction in the above algorithm, then our SQP method is very similar to classical SQP methods for smooth nonlinear programming [15, 36]. The difference is that here we anticipate nonsmoothness of ψ. If, further, μ is treated as a parameter rather than a variable, namely, if the last equation in (16) is omitted at each iteration, then the above-modified SQP method begins to look like the explicit smoothing SQP method proposed in Fukushima, Luo, and Pang [12]. See section 6 for an explicit SQP method that has the convergence properties of the explicit smoothing method of Facchinei, Jiang, and Qi [8].

(ii) Since only inequality constraints are relaxed, we use the merit function $\Theta_{(\rho^g, \rho^{\mathrm{NCP}})}$, which has two (likely different) penalty parameters, unlike θ_ρ used in the classical SQP methods. The updates for $\tilde{\rho}$, ρ^g, and ρ^{NCP} are to ensure that the solution of the modified QP (18) is a descent direction of the merit function $\Theta_{(\rho^g, \rho^{\mathrm{NCP}})}$. In the update of ρ, we increase it by a positive constant δ_2 in the case that $\sum \xi_i > 0$ in an attempt to force a decrease in the feasibility gap of the QP (16) at the next iteration.

5.2. QP subproblems and the penalty function.

DEFINITION 5.1. *F is said to be a P_0-function with respect to y if for each $x \in \Re^n$, $F(x, \cdot)$ is a P_0-function; i.e., for any $y, \bar{y} \in \Re^m$ with $y \neq \bar{y}$, there exists an index i*

such that $y_i \neq \bar{y}_i$ *and*

$$(y_i - \bar{y}_i)(F_i(x,y) - F_i(x,\bar{y}) \geq 0.$$

We introduce a new condition on ψ to extend invertibility of the matrix U in (15) to infeasible points.

(A5) For $c \neq 0$, if $(p,q,r) \in \partial\psi(a,b,c)$, then $pq > 0$.

PROPOSITION 5.2. *Suppose F is a P_0-function with respect to y. If the assumptions* (A1)–(A5) *hold, then the matrix given by* (15),

$$U = \begin{pmatrix} F'_y(x,y) & -I \\ A & B \end{pmatrix},$$

is nonsingular for any (x,y,w,μ) with $\mu \neq 0$, where $(A,B,C) \in \partial\Psi(y,w,\mu) = \{\Psi'(y,w,\mu)\}$.

Proof. Since $(A,B,C) \in \partial\Psi(y,w,\mu)$, both A and B are diagonal matrices with nonzero diagonal elements. It turns out that nonsingularity of the matrix U is equivalent to nonsingularity of the matrix $A + BF'_y(x,y)$, or $B^{-1}A + F'_y(x,y)$. Note that $B^{-1}A$ is a diagonal positive definite matrix, and $F'_y(x,y)$ is a P_0-matrix. Therefore, nonsingularity of $B^{-1}A + F'_y(x,y)$ follows; see [6]. This completes the proof. □

The following result concerns the feasibility of the quadratic programs (16) and (18).

PROPOSITION 5.3. *Suppose $F'_y(x,y)$ is a P_0-matrix, the assumptions* (A1)–(A5) *hold, and $\mu \neq 0$. Let U be as defined in Proposition* 5.2. *Then*

(i) *The modified QP* (18) *is always feasible.*

(ii) *The QP* (16) *has a nonempty feasible set if and only if the following system is consistent with respect to dx:*

$$\begin{aligned}&[g'_x(x,y) - g'_y(x,y)(U^{-1})_{yy}F'_x(x,y)]dx \\ &\qquad -g'_y(x,y)[(U^{-1})_{yy}(F(x,y) - w) + (U^{-1})_{yw}(\Psi(y,w,\mu) + Cd\mu)] + g(x,y) \geq 0.\end{aligned}$$

(iii) *If d is a solution of* (16) *or (d,ξ) is a solution of* (18), *then*

$$d\mu = -\frac{e^{\mu} - 1}{e^{\mu}}$$

and (dy, dw) is uniquely determined by dx and $d\mu$, i.e.,

$$(dy, dw) = U^{-1}\begin{pmatrix} -F'_x(x,y)dx - F(x,y) + w \\ -\Psi(y,w,\mu) - Cd\mu \end{pmatrix}.$$

The above proposition not only gives a characterization for nonemptiness of the feasible set of (16), but also shows that a solution of (16) can be found by solving a reduced QP in the variable x and a system of linear equations (a similar argument also applies to the modified QP (18)). This fact can be computationally significant as n is often much smaller than m.

The feasibility of the QP (16) is a serious issue in the context of MPECs. Fukushima and Pang [13] discussed it from a different angle, namely, for mathematical programs with linear complementarity constraints. We remark that the P_0 property assumed in our paper is not necessarily required in [13].

The following is a simple yet important consequence of the above proposition for the case when there are no joint (upper-level) constraints on (x,y).

Corollary 5.4. *Suppose $F'_y(x,y)$ is a P_0-matrix,* (A1)–(A5) *hold, and $\mu \neq 0$. Assume that $g(x,y) = g(x)$. Then* (16) *has a nonempty feasible set if and only if $g'(x)dx + g(x) \geq 0$ is consistent with respect to dx.*

The next result on $d\mu$ is proved in [18]. It basically says that $\{\mu^k\}$ is a positive and monotonically decreasing sequence.

Lemma 5.5. *Suppose $(x,y,w,\mu) \in \Re^{n+2m+1}$, $\mu > 0$, and $d = (dx,dy,dw,d\mu)$ solves the QP* (16) *or (d,ξ) solves the modified QP* (18)*. Then $d\mu \in (-\mu, 0)$ so that*

$$\mu + td\mu \in (0, \mu)$$

for any $t \in (0,1]$.

We now study some properties of the exact penalty function $\Theta_{(\rho^g,\rho^{\text{NCP}})}$. The vector $u^* = (x^*, y^*, w^*, \mu^*)$ is said to be a critical point of (or stationary for) the penalty function $\Theta_{(\rho^g,\rho^{\text{NCP}})}$ for the given positive parameters ρ^g and ρ^{NCP} if for any direction $d \in \Re^{n+2m+1}$,

$$\Theta'_{(\rho^g,\rho^{\text{NCP}})}(u^*; d) \geq 0.$$

In the rest of this subsection, we collect some useful properties that shall be used to improve global convergence of the (modified) implicit smooth SQP in the next subsection. Since all functions are smooth when $\mu \neq 0$, these properties follow directly from the nonlinear programming results presented in the appendix of the manuscript of this paper [20]. Moreover, we are mainly concerned with the properties associated with the modified QP.

Proposition 5.6. *Let $\mu \neq 0$.*

(i) *For $d \in \Re^{n+2m+1}$, $\Theta'_{(\rho^g,\rho^{\text{NCP}})}$ is directionally differentiable at u along the direction d and $\Theta'_{(\rho^g,\rho^{\text{NCP}})}(x,y,w,\mu;d)$ can be easily evaluated.*

(ii) *If (d,ξ) is a solution of the modified QP* (18)*, $\rho^g = \rho$, and $\rho^{\text{NCP}} \geq \max_{1 \leq i \leq l+2m+1} |\lambda_i|$ with λ the KKT multiplier of the modified QP* (18)*, then*

$$\begin{aligned}\Theta'_{(\rho^g,\rho^{\text{NCP}})}(x,y,w,\mu;d) \leq\ & \nabla f(x,y)^T(dx,dy) - (\lambda_g)^T g'(x,y)(dx,dy) \\ & +(\lambda_F)^T(F'(x,y)(dx,dy) - dw) \\ & +(\lambda_\Psi)^T \Psi'(y,w,\mu)(dy,dw,d\mu) + \lambda_\mu e^\mu d\mu\end{aligned}$$

and

$$\Theta'_{(\rho^g,\rho^{\text{NCP}})}(x,y,w,\mu;d) \leq -d^T W d.$$

Proof. When $\mu \neq 0$, $g(x,,y)$, $F(x,y) - w$, $\Psi(y,w,\mu)$, and $e^\mu - 1$ are all continuously differentiable at (x,y,w,μ). Hence the results follow from [20, Proposition A.1] and Lemma 5.5. □

Proposition 5.7. *Let $u^* = (x^*, y^*, w^*, \mu^*)$ be given. Suppose the matrix W^* is symmetric positive definite, $F'_y(x^*, y^*)$ is a P_0-matrix, and Ψ is smooth at (y^*, w^*, μ^*).*

(i) *For given $\rho^g > 0$ and all large $\rho^{\text{NCP}} > 0$, u^* is a critical point of $\Theta_{(\rho^g,\rho\text{NCP})}$ if and only if there exists (d,ξ) with $d = 0$ that is a solution of the modified QP* (18) *with $u = u^*$, $W = W^*$, and $\rho = \rho^g$.*

(ii) *If u^* is a KKT point of* (7) *and λ is its KKT multiplier, then u^* is a critical point of $\Theta_{(\rho^g,\rho^{\text{NCP}})}$ with $\min\{\rho^g, \rho^{\text{NCP}}) \geq \max_{1 \leq i \leq l+2m+1} |\lambda_i|$.*

(iii) *If u^* is a critical point of $\Theta_{(\rho^g,\rho^{\text{NCP}})}$ for some $\rho^g > 0$ and all sufficiently large $\rho^{\text{NCP}} > 0$, and u^* is feasible for* (7)*, then u^* is a KKT point of* (7)*.*

Proof. The desired results can be proved from Proposition 5.3 of this paper and Propositions A.3 and A.5 of [20]. □

5.3. Global convergence under lower-level nondegeneracy.

Lemma 5.8. *Suppose that* $(x, y, w, \mu) \in \Re^{n+2m+1}$ *with* $\mu \neq 0$*, and suppose that* $W \in \Re^{(n+2m+1)\times(n+2m+1)}$ *is symmetric positive definite. Suppose* (d, ξ) *is a solution of the modified QP* (18) *and* λ *is its corresponding KKT multiplier. Then* d *is a descent direction of the merit function* $\Theta_{(\rho^g, \rho^{\text{NCP}})}$ *if* $d \neq 0$*,* $\rho^g = \rho$*, and* $\rho^{\text{NCP}} \geq \max_{1\leq i\leq l+2m+1} |\lambda_i|$*.*

Proof. The lemma follows from the second inequality of (ii) in Proposition 5.6. □

Lemma 5.8 shows that solving the modified QP (18) generates a descent direction of the merit function $\Theta_{(\rho^g, \rho^{\text{NCP}})}$ for sufficiently large ρ^{NCP} when W is symmetric positive definite and $\mu \neq 0$. Furthermore, the line search in Step 4 is well defined; i.e., t_k can be determined in finitely many steps. Therefore, the implicit smooth SQP method is well defined when $\mu \neq 0$ and W is symmetric positive definite at each step. Moreover, since the line search chooses the step size $t_k \in (0, 1)$, Lemma 5.5 can be used to show that $\mu^{k+1} > 0$ if $\mu^k > 0$; hence $\mu^k \neq 0$ for each k if $\mu^0 > 0$.

To present the global convergence of implicit smooth SQP, we assume the following standard conditions.

(B1) There exist two positive numbers $\alpha < \beta$ such that each of the symmetric matrices W_k used in implicit smooth SQP satisfies the following condition for all vectors u of appropriate dimension:

$$\alpha\|u\|^2 \leq u^T W_k u \leq \beta\|u\|^2.$$

(B2) For all large k, $\rho_k = \rho_* > 0$.

Under the assumption (B1) and the feasibility of the modified QP (18) at each iteration, implicit smooth SQP is well defined. The assumption (B2) can be shown to hold under some further conditions. As a consequence of the condition (B2), we obtain that for all sufficiently large k,

$$\rho_k^g = \rho_*, \quad \rho_k^{\text{NCP}} = \rho_*^{\text{NCP}}, \quad \xi^k = 0,$$

where ρ_*^{NCP} is a positive constant. We assume that implicit smooth SQP does not terminate in Step 2. Let $\{u^k\} = \{(x^k, y^k, w^k, \mu^k)\}$ be generated by implicit smooth SQP.

Lemma 5.9. *Suppose* (A1)–(A5) *hold,* (B1)–(B2) *hold, and* F *is a* P_0*-function with respect to* y*. Suppose* $\{u^k\}$ *is the sequence generated by the algorithm,* $\{(d^k, \xi^k)\}$ *is the sequence of solutions of the modified QP* (18)*, and* $\lim_{k\to\infty, k\in K} u^k = u^*$ *for a subset* $K \subseteq \{1, 2, \ldots\}$*. Then the following conclusions hold.*

(i) $\{d^k\}_{k\in K}$ *and* $\{\xi^k\}_{k\in K}$ *are bounded.*

(ii) *Assume* Ψ *is continuously differentiable near* u^**. If* d^**,* ξ^**, and* W^* *are accumulation points of the sequences* $\{d^k\}_{k\in K}$*,* $\{\xi^k\}_{k\in K}$*, and* $\{W_k\}_{k\in K}$*, respectively, then* (d^*, ξ^*) *is a solution of the modified QP* (18) *with* $u = u^*$*,* $W = W^*$*, and* $\rho = \rho_*$*. Furthermore,* $\Theta_{(\rho_*, \rho_*^{\text{NCP}})}$ *is directionally differentiable at* u^* *and it holds that*

$$\Theta'_{(\rho_*, \rho_*^{\text{NCP}})}(u^*; d^*) \leq -(d^*)^T W^* d^*.$$

(iii) *Assume the step size sequence* $\{t_k\}$ *determined by the Armijo line search satisfies* $\lim_{k\to\infty, k\in K} t_k = 0$*. Under the smoothness assumption in* (ii)*, we have*

$$\limsup_{k\to\infty, k\in K} \frac{\Theta_{(\rho_*, \rho_*^{\text{NCP}})}(u^k + t_k d^k) - \Theta_{(\rho_*, \rho_*^{\text{NCP}})}(u^k)}{t_k} \leq \Theta'_{(\rho_*, \rho_*^{\text{NCP}})}(u^*; d^*).$$

Proof. Since $g(x, y)$, $F(x, y) - w$, $\Psi(y, w, \mu)$, and $e^{\mu} - 1$ are smooth at u^*, the desired results follows from Lemmas A.2 and A.3 of [20]. □

The condition (B2) may not hold in general. The following additional conditions ensure that (B2) is satisfied, as shown below. Let H be the function representing the equality constraints of (7), i.e., $H(u) = (F(x, y) - w,\ \Psi(y, w, \mu))$ with $u = (x, y, w, \mu)$.

(B3) $\{u^k\}$ is bounded.

(B4) The generalized Jacobian $\partial H(u^*)$ has full row rank at any accumulation point u^* of $\{u^k\}$.

(B5) For any accumulation point u^* of $\{u^k\}$ and any $V \in \partial H(u^*)$, there exists $d = (dx, dy, dw, d\mu)$ such that $g'(x^*, y^*)(dx, dy) + g(x^*, y^*) > 0$ and $Vd + H(u^*) = 0$.

Note that the conditions (B4) and (B5) together are equivalent to the GMFCQ if u^* is a feasible point of (7).

We are now ready to establish global convergence of implicit smooth SQP under the assumption that Ψ is smooth at the accumulation point. We remark that the smoothness of Ψ at a limit point means that the problem has essentially (asymptotically) been reduced to smooth nonlinear programming.

THEOREM 5.10. *Suppose the assumptions* (A1)–(A5) *hold, the standing assumption* (B1) *holds, and F is a P_0-function with respect to y. Suppose $\mu^0 > 0$ and $\{u^k\}$ is the sequence generated by the algorithm. We obtain the following conclusions.*

(i) *If* (a) *the condition* (B2) *holds,* (b) *u^* is an accumulation point of $\{u^k\}$, and* (c) *Ψ is continuously differentiable at u^*, then u^* is both a critical point of $\Theta_{(\rho_*, \rho_*^{\mathrm{NCP}})}$ and a (classical or primal or piecewise) stationary point of the MPEC* (7).

(ii) *If conditions* (B3), (B4), *and* (B5) *hold, then so does* (B2).

Proof. (i) Since Ψ is smooth at u^*, all results follow from Theorem A.1 of [20] and from Proposition 3.7 of this paper.

(ii) This follows from Theorem A.2 in [20] but with some suitable modifications given that Ψ may be nonsmooth at some accumulation points of the sequence $\{u^k\}$. □

Remark. In the above theorem, global convergence of implicit smooth SQP requires smoothness of the function Ψ at $u^* = (x^*, y^*, w^*, \mu^*)$. As shall be seen in section 7, if ϕ is the Fischer–Burmeister function in Example 7.1, the min function in Example 7.2, or the Kanzow–Kleinmichel function in Example 7.3, then the smoothness condition on Ψ is satisfied at any lower-level nondegenerate point, i.e., Ψ is smooth and is in fact twice continuously differentiable.

As already noted, lower-level nondegeneracy at a limit point u^* often results in smoothness of the function Ψ at this point, which means that we can apply classical theory and obtain classical results. Hence superlinear convergence under the lower-level nondegeneracy condition and the assumption that the stepsize takes the value 1 for all large k would be no surprise though our merit function $\Theta_{(\rho^g, \rho^{\mathrm{NCP}})}$ has two penalty parameters. The unit stepsize assumption is needed in nonlinear programming due to the well-known Maratos effect, which can prevent superlinear convergence of an SQP method that uses an exact penalty merit function unless a second-order correction to the feasibility of the iterate is performed at each iteration. See [11, 36].

In order to study the rate of convergence of implicit smooth SQP, further conditions such as the LICQ, the second-order sufficient condition, careful update rules of the matrix sequence $\{W_k\}$, etc. are needed. We conjecture that superlinear convergence results similar to those of [35, 36] can be obtained.

6. Explicit smooth SQP.

Global convergence of the implicit smooth SQP method requires the lower-level nondegeneracy condition at an accumulation point. This assumption is not unusual for convergence of algorithms for MPECs such as PIPA [28] and also the explicit smoothing SQP method of Fukushima, Luo, and Pang [12], but it is still rather strong in that it essentially reduces the problem to one of nonlinear programming, which is not tenable in general.

As an alternative we propose an explicit smooth SQP algorithm for which global convergence can be established without assuming lower-level nondegeneracy. This method has a similar computational form to the SQP method of [12], although our smoothing parameter update has to be carried out more carefully, like the original smoothing method for MPECs of Facchinei, Jiang, and Qi [8]. Moreover, the method given here weakens the assumptions needed in [8] as explained in remark (ii) following Theorem 6.4.

Note that the term *explicit* refers to the fact that the smoothing parameter μ is not treated as a variable in the QP subproblem at each iteration, nor is it updated in the line search that determines the next iterate $(x^{k+1}, y^{k+1}, w^{k+1})$. In our explicit smooth SQP method, the smoothing parameter tends to be updated less often than once per QP solve, unlike the implicit smooth SQP method of the previous section and the explicit smoothing algorithm of [12].

Recall the definitions of ϕ_μ and Φ_μ from section 3. We approximate the MPEC (1) by the nonlinear programming problem with $\mu \neq 0$,

$$\tag{20} \begin{array}{rl} \displaystyle\min_{x,y,w} & f(x,y) \\ \text{subject to} & g(x,y) \geq 0, \\ & F(x,y) - w = 0, \\ & \Phi_\mu(y,w) = 0, \end{array}$$

which is (4) with

$$\Phi_\mu(y,w) \quad = \quad \begin{pmatrix} \phi_\mu(y_1, w_1) \\ \vdots \\ \phi_\mu(y_m, w_m) \end{pmatrix} \quad = \quad \Psi(y,w,\mu).$$

Obviously, when $\mu = 0$, the above problem reduces to (6). Therefore, our goal is to find approximate solutions of (6) for each $\mu \neq 0$ and then locate a solution or a generalized stationary point of (6) by driving μ to zero.

Similar to implicit smooth SQP, we want to find an approximate solution of (20) by solving a sequence of quadratic programs. More precisely, for any given (x,y,w), $\mu \neq 0$, and $d = (dx, dy, dw)$, we define a modified quadratic program (which is a modified quadratic model of (20) at (x,y) for the fixed $\mu \neq 0$ and $\rho > 0$) as follows:

$$\tag{21} \begin{array}{rl} \displaystyle\min_{d \in \Re^{n+2m}, \xi \in \Re^l} & \nabla f(x,y)^T(dx,dy) + \frac{1}{2} d^T W d + \rho \displaystyle\sum_{i=1}^{l} \xi_i \\ \text{subject to} & g'(x,y)(dx,dy) + g(x,y) \geq -\xi, \\ & F'(x,y)(dx,dy) - dw + (F(x,y) - w) = 0, \\ & A dy + B dw + \Phi_\mu(y,w) = 0, \\ & \xi \geq 0, \end{array}$$

where $[A\ B] = \Phi'_\mu(x,y)$ and the matrix $W \in \Re^{(n+2m)\times(n+2m)}$ is symmetric positive definite.

If the modified QP (21) has a solution (d, ξ), then its KKT condition has the following form:

$$\left(\begin{array}{c} \nabla_x f \\ \nabla_y f \\ 0 \end{array}\right) + Wd - \left(\begin{array}{c} g'_x(x,y)^T \\ g'_y(x,y)^T \\ 0 \end{array}\right)\lambda_g + \left(\begin{array}{c} F'_x(x,y)^T \\ F'_y(x,y)^T \\ -I \end{array}\right)\lambda_F + \left(\begin{array}{c} 0 \\ A^T \\ B^T \end{array}\right)\lambda_{\Phi_\mu} = 0,$$

$$\begin{array}{l} \rho\, \tilde{e} = \lambda_g + \lambda_\xi, \\ 0 \le g'(x,y)(dx,dy) + g(x,y) + \xi \ \perp\ \lambda_g \ge 0, \\ F'(x,y)(dx,dy) - dw + (F(x,y) - w) = 0, \\ Ady + Bdw + \Phi_\mu(y,w) = 0, \\ 0 \le \xi \ \perp\ \lambda_\xi \ge 0, \end{array}$$

(22)

where $\tilde{e}$ is the vector of all ones in $\Re^l$.

We can immediately write down a quadratic model of (20) without the artificial variable ξ:

$$(23) \qquad \begin{array}{rl} \displaystyle\min_{d\in\Re^{n+2m},\xi\in\Re^l} & \nabla f(x,y)^T(dx,dy) + \frac{1}{2}d^TWd \\ \text{subject to} & g'(x,y)(dx,dy) + g(x,y) \ge 0, \\ & F'(x,y)(dx,dy) - dw + (F(x,y) - w) = 0, \\ & Ady + Bdw + \Phi_\mu(y,w) = 0. \end{array}$$

A penalty merit function of (20) is defined by

$$\begin{array}{l} \Theta_{(\rho^g,\rho^{\mathrm{NCP}},\mu)}(x,y,w) \\ \quad = f(x,y) \,+\, \rho^g \displaystyle\sum_{i=1}^{l} \max\{-g_i(x,y), 0\} + \rho^{\mathrm{NCP}} \sum_{j=1}^{m} \left[(|F_j(x,y) - w_j| + |\phi_\mu(y_j,w_j)|)\right], \end{array}$$

where ρ^g and ρ^{NCP} are positive numbers.

Before presenting our second method, we give a result for the case $\mu \neq 0$, when the problem (20) is a smooth NLP that is parallel to Proposition 4.3, which deals with the case $\mu = 0$. This result will not be used directly in the proof of convergence of the explicit smooth SQP method but gives some idea of when the constraints of the nonlinear problem (20) are numerically stable.

Proposition 6.1. *Suppose (x,y,w) is a feasible point of (20) with $\mu \neq 0$, F is a P_0-function with respect to y, and the assumptions (A1)–(A2) and (A5) are satisfied. Define $\mathcal{I}_g = \{i : \ g_i(x,y) = 0\}$,*

$$\begin{array}{rcl} \Gamma & = & g'_x(x,y) - g'_y(x,y)(U^{-1})_{yy}F'_x(x,y), \\ U & = & \left(\begin{array}{cc} F'_y(x,y) & -I \\ A & B \end{array}\right), \quad [A\ B] = \Phi'_\mu(y,w). \end{array}$$

Then the following conclusions hold.

(i) *The CRCQ holds for (20) at (x,y,w) if the row submatrix $\Gamma_{\mathcal{I}_g}$ corresponding to the active indexes of g at (x,y) has a constant rank around (x,y,w). In particular, the CRCQ holds for (20) at (x,y,w) if $g(x,y) = g(x)$ and if the matrix $g'(x)_{\mathcal{I}_g}$ has constant rank around x.*

(ii) *The LICQ holds for (20) at (x,y,w) if the row submatrix $\Gamma_{\mathcal{I}_g}$ has full row rank. In particular, the LICQ holds for (20) at (x,y,w) if $g(x,y) = g(x)$ and if the row submatrix $g'(x)_{\mathcal{I}_g}$ has full row rank.*

(iii) *The MFCQ holds for* (20) *at* (x, y, w) *if there exists a vector* $d \in \Re^n$ *such that*

$$\Gamma_{\mathcal{I}_g} d > 0.$$

In particular, the MFCQ holds for (20) *at* (x, y, w) *if* $g(x, y) = g(x)$ *and there exists a vector* $d \in \Re^n$ *such that*

$$g_i'(x)d > 0 \quad \textit{for } i \in \mathcal{I}_g.$$

Unlike in section 3, in this section we let $u = (x, y, w)$ and $d = (dx, dy, dw)$ because μ is regarded as a parameter but not a variable. For the same reason, we use the subscript k, i.e., μ_k, to denote the value of the parameter μ at the kth iteration.

ALGORITHM: EXPLICIT SMOOTH SQP.

Step 0. (**Initialization**) Let $\rho_{-1} > 0$, $\delta_1 > 0$, $\delta_2 > 0$, $\beta_\mu \in (0, 1)$, $\beta_\varepsilon \in (0, 1)$, $\sigma \in (0, 1)$, $\tau \in (0, 1)$. Choose $u^0 = (x^0, y^0, w^0) \in \Re^{n+2m}$, and choose $\mu_0 > 0$, $\varepsilon_0 > 0$, and a symmetric positive definite matrix $W_0 \in \Re^{(n+2m)\times(n+2m)}$. Set $k := 0$.

Step 1. (**Search direction**) Solve the modified QP (21) with $(x, y, w) = (x^k, y^k, w^k)$, $\mu = \mu_k$, $W = W_k$, and $\rho = \rho_{k-1}$. Let (d^k, ξ^k) be a solution of this QP and $\lambda^k = (\lambda_g, \lambda_F, \lambda_\Phi, \lambda_\xi)$ be its corresponding KKT multiplier.

Step 2. (**Termination check**) If a stopping rule is satisfied, terminate. Otherwise, go to Step 3.

Step 3. (**Penalty update**) Let

$$\tilde{\rho}_k = \begin{cases} \rho_{k-1} & \text{if } \rho_{k-1} \geq \max\limits_{1\leq i\leq l+2m+1} |\lambda_i^k|, \\ \delta_1 + \max\limits_{1\leq i\leq l+2m+1} |\lambda_i^k| & \text{otherwise.} \end{cases}$$

Define $\rho_k^g = \rho_{k-1}$ and $\rho_k^{\mathrm{NCP}} = \tilde{\rho}_k$ and

$$\rho_k = \begin{cases} \tilde{\rho}_k & \text{if } \sum\limits_{1\leq i\leq l} \xi_i^k = 0, \\ \tilde{\rho}_k + \delta_2 & \text{otherwise.} \end{cases}$$

Step 4. (**Line search**) Let $t_k = (\tau)^{i_k}$, where i_k is the smallest nonnegative integer such that $i = i_k$ satisfies

$$\Theta_{(\rho_k^g, \rho_k^{\mathrm{NCP}}, \mu_k)}(u^k + (\tau)^i d^k) - \Theta_{(\rho_k^g, \rho_k^{\mathrm{NCP}}, \mu_k)}(u^k) \leq -\sigma(\tau)^i (d^k)^T W_k d^k.$$

Step 5. (**Update**) Let

$$u^{k+1} = u^k + t_k d^k,$$

$$\mu_{k+1} = \begin{cases} \beta_\mu \, \mu_k & \text{if } \|d^k\| \leq \varepsilon_k, \\ \mu_k & \text{otherwise,} \end{cases}$$

$$\varepsilon_{k+1} = \begin{cases} \beta_\varepsilon \, \varepsilon_k & \text{if } \|d^k\| \leq \varepsilon_k, \\ \varepsilon_k & \text{otherwise.} \end{cases}$$

Choose a symmetric positive definite matrix $W_{k+1} \in \Re^{(n+2m)\times(n+2m)}$. Set $k := k + 1$ and go to Step 1.

We next present some results analogous to those in section 5.2. The proofs are very similar, so we omit all proofs.

PROPOSITION 6.2. *Suppose $F'_y(x,y)$ is a P_0-matrix, the assumptions* (A1)–(A5) *hold, and $\mu \neq 0$. Then* (21) *has a nonempty feasible set. Moreover,* (23) *has a nonempty feasible set if and only if the following system is consistent with respect to dx:*

$$\Gamma dx + g(x,y,w,\mu) \;\geq\; 0,$$

where Γ and U are given by Proposition 6.1 *and $g(x,y,w,\mu)$ is the vector*

$$g(x,y,w,\mu) \;=\; g(x,y) \;-\; g'_y(x,y)[(U^{-1})_{yy}(F(x,y)-w) + (U^{-1})_{yw}\Phi_\mu(y,w)].$$

Furthermore, (dy,dw) is uniquely determined by dx, i.e.,

$$(dy,dw) = U^{-1}\begin{pmatrix} -F'_x(x,y)dx - F(x,y) + w \\ -\Phi_\mu(y,w) \end{pmatrix}.$$

In the case where $g(x,y) = g(x)$, the above consistency condition becomes consistency with respect to dx:

$$g'(x)dx + g(x) \geq 0.$$

PROPOSITION 6.3. *Let $\mu \neq 0$.*

(i) $\Theta'_{(\rho^g,\rho^{\mathrm{NCP}},\mu)}$ *is directionally differentiable at u. Furthermore, if (d,ξ) is a solution of the modified QP (21), $\rho^g = \rho$, and $\rho^{\mathrm{NCP}} \geq \max_{1\leq i\leq l+2m}|\lambda_i|$ with λ its KKT multiplier, then*

$$\begin{aligned}\Theta'_{(\rho^g,\rho^{\mathrm{NCP}},\mu)}(x,y,w;d) \quad \leq \quad & \nabla f(x,y)^T(dx,dy) - (\lambda_g)^T g'(x,y)(dx,dy)\\ & +(\lambda_F)^T(F'(x,y)(dx,dy) - dw)\\ & +(\lambda_{\Phi_\mu})^T\Phi'_\mu(y,w)(dy,dw)\end{aligned}$$

and

$$\Theta'_{(\rho^g,\rho^{\mathrm{NCP}},\mu)}(x,y,w;d) \leq -d^TWd.$$

(ii) *Suppose W is symmetric positive definite. If (d,ξ) is a solution of the modified QP* (21) *with $d \neq 0$, then d is a descent direction of the penalty function $\Theta_{(\rho^g,\rho^{\mathrm{NCP}},\mu)}$ for $\rho^g = \rho$ and any ρ^{NCP} satisfying the condition in* (i).

We need assumptions (B1) and (B2) from section 5.3, although here $u = (x,y,w)$; i.e., μ is omitted, hence the order of each matrix W_k is $n+2m$ rather than $n+2m+1$ as in section 5. As before, we can ensure (B2) by assuming conditions (B3)–(B5). The function H in the conditions (B4) and (B5) now corresponds to the equality constraints of (6); i.e., $H(u) = (F(x,y)-w, \Phi(y,w)) = (F(x,y)-w, \Psi(y,w,0))$.

THEOREM 6.4. *Assume that* (A1)–(A5) *and* (B1) *hold and that F is a P_0-function with respect to y. Let $\mu_0 > 0$ and $\{u^k\}$, $\{\mu_k\}$, and $\{\varepsilon_k\}$ be the sequences generated by the algorithm.*

(i) *If the assumption* (B2) *holds and $\{u^k\}$ has a limit point, then*

$$\lim_{k\to\infty}\mu_k = 0, \qquad \lim_{k\to\infty}\varepsilon_k = 0.$$

(ii) *Let* $K = \{k : \|d^k\| \leq \varepsilon_k\}$*. If we assume that the assumption* (B2) *holds and* $\{u^k\}_{k \in K}$ *has an accumulation point* $u^* = (x^*, y^*, w^*)$*, then* u^* *is a generalized stationary point of* (6)*. Furthermore, if* (x^*, y^*) *is lower-level nondegenerate, then* (x^*, y^*) *is a (classical or primal or piecewise) stationary point of the MPEC.*

(iii) *If conditions* (B1), (B3), (B4), *and* (B5) *hold, then so does* (B2).

Proof. (i) Obviously $\{\mu_k\}$ is bounded. Suppose μ_* is an accumulation point of $\{\mu_k\}$. If $\mu_* > 0$, then $\|d^k\| \leq \varepsilon_k$ occurs only finitely many times. This means that after finitely many iterations, μ_k and ε_k remain unchanged; i.e., for some k_0 and all $k \geq k_0$, $\mu_k = \mu_{k_0} > 0$ and $\varepsilon_k = \varepsilon_{k_0} > 0$. In this case, our smoothing method reduces to the modified SQP method presented in [20, Appendix] for a smooth nonlinear program (20). By [20, Theorem A.1] and its proof, it follows that some subsequence of $\{d^k\}$ approaches 0 as $k \to \infty$, which implies that $\|d^k\| \leq \varepsilon_{k_0}$ will eventually happen, which is a contradiction. Therefore, $\lim_{k\to\infty} \mu_k = 0$. By the update rule in Step 5, it is also true that $\lim_{k\to\infty} \varepsilon_k = 0$.

(ii) By the assumption (B2) and the update rule of the penalty parameter, the KKT multiplier sequence $\{\lambda^k\}_{k\in K}$ is bounded and $\xi^k = 0$ for all large enough k since $\rho_k = \rho_*$ for all sufficiently large k. Note that for each $k \in K$, $\|d^k\| \leq \varepsilon_k$. Hence $\lim_{k\to\infty, k\in K} d^k = 0$. By passing to the limit for $k \in K$, it follows from the KKT condition (22) and the assumption (A3) that u^* is a generalized stationary point of (6). From Proposition 3.7, (x^*, y^*) is a piecewise stationary point of the MPEC if (x^*, y^*) is lower-level nondegenerate.

(iii) This follows, as does part (ii) of Theorem 5.10, by a straightforward extension of a similar result for the smooth case [20, Theorem A.2]. ☐

Remarks.

(i) As discussed in section 5.5, we may find a solution of the modified QP (21) by solving a reduced QP and some systems of linear equations, which may reduce the computational cost significantly, especially if the matrices defining the QP are dense.

(ii) Loosely speaking, the first part of Theorem 6.1 under the assumption (iii) can be viewed as a generalization of Theorem 5.14(b) of [8] when the assumptions (A1)–(A5) in [8] are valid and the smoothing function ψ used in (20) has the form of that defined in Example 7.2 below. To explain further, in [8]: (a) The upper-level constraints have the form $g(x, y) \equiv g(x) \leq 0$; i.e., the MPEC is an implicit program. (b) The upper-level and lower-level feasible sets are assumed to be compact, while in our case compactness is not assumed in either the upper or lower levels (the lower-level feasible set is $\Re^m_+$, corresponding to an NCP). (c) The lower-level objective function F is assumed to be uniformly strongly monotone in [8]; we assume that F is at most a uniform P_0-function in y.

Regarding (b), we should say that nonlinear constraints are allowed in the definition of a lower-level feasible set in [8]. However, the KKT conditions of the lower-level variational inequality problem are a parametric mixed complementarity problem. As mentioned in section 1, this case can be treated as an MPEC of the form (1) with some additional upper-level equality constraints.

7. Special examples of smoothing functions. In this section we give examples of the function ψ satisfying the assumptions (A1)–(A5). Hence these special forms of ψ correspond to particular implementations of smooth SQP methods for MPECs.

Example 7.1.

$$\psi(a,b,c) = \sqrt{a^2+b^2+c^2} - (a+b).$$

This function is used to propose an SQP method in [12]. Corresponding to ψ is the function $\phi(a,b) = \sqrt{a^2+b^2} - (a+b)$, which is now known as the Fischer–Burmeister function [9]. The introduction of ψ originates from [21] for handling linear complementarity problems.

If $(a,b,c) \neq (0,0,0)$, then ψ is smooth at (a,b,c) with $\nabla\psi(a,b,c) = (p,q,r)$ such that

$$p = \frac{a}{\sqrt{a^2+b^2+c^2}} - 1, \quad q = \frac{b}{\sqrt{a^2+b^2+c^2}} - 1, \quad r = \frac{c}{\sqrt{a^2+b^2+c^2}}.$$

If $(a,b,c) = (0,0,0)$, then ψ is locally Lipschitz at (a,b,c) and its generalized Jacobian is the ball [18]

$$\partial\psi(a,b,c) = \{(p,q,r):\ (p+1)^2 + (q+1)^2 + r^2 \leq 1\}.$$

Example 7.2.

$$\psi(a,b,c) = \sqrt{(a-b)^2+c^2} - (a+b).$$

This function is used to propose a smoothing method in [8]. Corresponding to ψ is the function $\phi(a,b) = |a-b| - (a+b) = -2\min\{a,b\}$. The introduction of ψ also originates from [21].

If either $a \neq b$ or $c \neq 0$, then ψ is smooth at (a,b,c) with $\nabla\psi(a,b,c) = (p,q,r)$ such that

$$p = \frac{a-b}{\sqrt{(a-b)^2+c^2}} - 1, \quad q = \frac{b-a}{\sqrt{(a-b)^2+c^2}} - 1, \quad r = \frac{c}{\sqrt{(a-b)^2+c^2}}.$$

If $a = b$ and $c = 0$, then ψ is locally Lipschitz at (a,b,c) and its generalized Jacobian is the intersection of a plane with a box:

$$\partial\psi(a,b,c) = \{(p,q,r):\ p+q=-2,\ p \in [-2,0],\ q \in [-2,0],\ r \in [-1,1]\}.$$

Example 7.3.

$$\begin{aligned}\psi(a,b,c) &= \sqrt{a^2+b^2+\lambda ab+c^2} - (a+b),\\ \phi(a,b) &= \sqrt{a^2+b^2+\lambda ab} - (a+b),\end{aligned}$$

where $\lambda \in [-2,2)$ is a parameter. The function ϕ is introduced in [22] for solving nonlinear complementarity problems. Apparently, when $\lambda = 0$, ϕ reduces to the Fischer–Burmeister function (Example 7.1), and when $\lambda = -2$, ϕ reduces to the min function (Example 7.2).

So we may assume that $\lambda \in (-2,2)$. If $(a,b,c) \neq (0,0,0)$, then ψ is smooth at (a,b,c) and $\nabla\psi(a,b,c) = (p,q,r)$ with

$$p = \frac{a+\lambda b/2}{\sqrt{a^2+b^2+\lambda ab+c^2}} - 1,\ q = \frac{b+\lambda a/2}{\sqrt{a^2+b^2+\lambda ab+c^2}} - 1,\ r = \frac{c}{\sqrt{a^2+b^2+\lambda ab+c^2}}.$$

If $(a,b,c) = (0,0,0)$, then ψ is locally Lipschitz at (a,b,c) and its generalized Jacobian is an ellipsoid:

$$\partial\psi(a,b,c) = \{(p,q,r):\ \alpha(p+1)^2 + \alpha(q+1)^2 + \beta(p-q)^2 + r^2 \leq 1\},$$

where $\alpha = \frac{2}{2+\lambda}$, $\beta = \frac{2\lambda}{4-\lambda^2}$.

Example 7.4.

$$\begin{aligned}
&\psi(a,b,c)=\lambda[\sqrt{a^2+b^2+c^2}-(a+b)]-\tfrac{(1-\lambda)}{4}(\sqrt{a^2+c^2}+a)(\sqrt{b^2+c^2}+b),\\
&\phi(a,b)=\lambda[\sqrt{a^2+b^2}-(a+b)]-(1-\lambda)\max\{a,0\}\ \max\{b,0\},
\end{aligned}$$

where $\lambda\in(0,1]$ is a parameter. The function ϕ is introduced in [3] for solving nonlinear complementarity problems. When $\lambda=1$, ϕ reduces to the Fischer–Burmeister function in Example 7.1.

If $c\neq 0$, then ψ is smooth at (a,b,c) and $\nabla\psi(a,b,c)=(p,q,r)$ with

$$\begin{aligned}
p&=\lambda(\tfrac{a}{\sqrt{a^2+b^2+c^2}}-1)-\tfrac{1-\lambda}{4}(\tfrac{a}{\sqrt{a^2+c^2}}+1)(\sqrt{b^2+c^2}+b),\\
q&=\lambda(\tfrac{b}{\sqrt{a^2+b^2+c^2}}-1)-\tfrac{1-\lambda}{4}(\tfrac{b}{\sqrt{b^2+c^2}}+1)(\sqrt{a^2+c^2}+a),\\
r&=\lambda\tfrac{c}{\sqrt{a^2+b^2+c^2}}-\tfrac{1-\lambda}{4}[\tfrac{c}{\sqrt{a^2+c^2}}(\sqrt{b^2+c^2}+b)+\tfrac{c}{\sqrt{b^2+c^2}}(\sqrt{a^2+c^2}+a)].
\end{aligned}$$

If $(a,b,c)=(0,0,0)$, then ψ is locally Lipschitz at (a,b,c) and its generalized Jacobian is the ball

$$\partial\psi(a,b,c)=\{(p,q,r):\ (p+\lambda)^2+(q+\lambda)^2+r^2\leq\lambda^2\}.$$

If $(a,b)\neq(0,0)$ and $c=0$, then ψ is locally Lipschitz at $(a,b,0)$ and its Jacobian or generalized Jacobian is of the form

$$\partial\psi(a,b,0)=\left\{(p,q,r):\quad
\begin{array}{l}
p=\lambda(\tfrac{a}{\sqrt{a^2+b^2}}-1)-\tfrac{1-\lambda}{4}\alpha(|b|+b);\\[2ex]
q=\lambda(\tfrac{b}{\sqrt{a^2+b^2}}-1)-\tfrac{1-\lambda}{4}\beta(|a|+a);\\[2ex]
r=-\tfrac{1-\lambda}{4}[\gamma_a(|b|+b)+\gamma_b(|a|+a)];\\[2ex]
\qquad\begin{array}{ll}\alpha\in[0,2] & \text{if } a=0,\\ \alpha=\tfrac{a}{|a|}+1 & \text{if } a\neq 0;\end{array}\\[2ex]
\qquad\begin{array}{ll}\beta\in[0,2] & \text{if } b=0,\\ \beta=\tfrac{b}{|b|}+1 & \text{if } b\neq 0;\end{array}\\[2ex]
\qquad\begin{array}{ll}\gamma_a\in[-1,1] & \text{if } a=0,\\ \gamma_a=0 & \text{if } a\neq 0;\end{array}\\[2ex]
\qquad\begin{array}{ll}\gamma_b\in[-1,1] & \text{if } b=0,\\ \gamma_b=0 & \text{if } b\neq 0.\end{array}
\end{array}\right.$$

Note that $a/|a|$ is equal to the sign of a for $a\neq 0$.

Part (i) of the next proposition demonstrates that special explicit smooth SQP methods can be proposed based on these smoothing functions. Its proof is an easy consequence of the above formulae for $\partial\psi(a,b,c)$. Part (ii), which is evident, says that the smoothing functions in Examples 7.1, 7.2, and 7.3 satisfy the smoothness assumption needed for global convergence in Theorem 5.10.

Proposition 7.1.

(i) *Each function ψ defined in Examples* 7.1, 7.2, 7.3, *and* 7.4 *satisfies the assumptions* (A1)–(A5).

(ii) *Each function ϕ defined in Examples* 7.1, 7.2, *and* 7.3 *is twice continuously differentiable at any nondegenerate point (a,b), i.e., $a \neq b$.*

8. Concluding remarks. In this article, mathematical programs with equilibrium constraints are reformulated as better posed nonsmooth programs and then, by means of so-called smoothing functions, approximated by (smooth) nonlinear programs. Consequently, some techniques that are well known in the context of nonlinear programming can be used for solving MPECs. In particular, we have developed two classes of SQP methods. Some global convergence results of these methods have been established. Numerical experience is yet to be established.

The extent of these convergence results depends critically on the convergence theory available for the corresponding nonlinear programming algorithm. So we expect that the future application of different nonlinear programming methods in the context of smoothing for MPECs and other nonsmooth optimization problems will give rise to different global convergence results.

We have also given concrete examples of smoothing functions motivated by the literature on complementarity problems. It would be interesting to find other smoothing functions to satisfy the assumptions (A1)–(A5) and other smoothing functions which may not satisfy those assumptions but may play similar roles in other algorithms.

Acknowledgments. The authors thank Francisco Facchinei for suggesting the reference [1] and for subsequent discussions on nonlinear programming, Jong-Shi Pang for comments on the meaning of "well-posedness," and two anonymous referees for their insightful comments.

REFERENCES

[1] D.P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, Sydney, Australia, 1982.

[2] J.V. Burke and S.P. Han, *A robust sequential quadratic programming method*, Math. Programming, 43 (1989), pp. 277–303.

[3] B. Chen, X. Chen, and C. Kanzow, *A Penalized Fischer-Burmeister NCP-function: Theoretical Investigation and Numerical Results*, Preprint 126, Institute of Applied Mathematics, University of Hamburg, Hamburg, Germany, 1997.

[4] Y. Chen and M. Florian, *The nonlinear bilevel programming problem: Formulations, regularity and optimality conditions*, Optimization, 32 (1995), pp. 193–209.

[5] F.H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.

[6] R.W. Cottle, J.S. Pang, and R.E. Stone, *The Linear Complementarity Problem*, Academic Press, New York, 1992.

[7] S. Dirkse and M.C. Ferris, *Modeling and solution environments for MPEC: GAMS & MATLAB*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Norwell, MA, 1999, pp. 127–148.

[8] F. Facchinei, H. Jiang, and L. Qi, *A smoothing method for mathematical programs with equilibrium constraints*, Math. Programming, 85 (1999), pp. 107–134.

[9] A. Fischer, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.

[10] A. Fischer and C. Kanzow, *On finite termination of an iterative method for linear complementarity problems*, Math. Programming, 74 (1996), pp. 279–292.

[11] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley, New York, 1987.

[12] M. Fukushima, Z.-Q. Luo, and J.S. Pang, *A globally convergent sequential quadratic programming algorithm for mathematical programs with linear complementarity constraints*, Comput. Optim. Appl., 10 (1998), pp. 5–34.

[13] M. Fukushima and J.S. Pang, *Some feasibility issues in mathematical programs with equilibrium constraints*, SIAM J. Optim., 8 (1998), pp. 673–681.

[14] J. Gauvin, *A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming*, Math. Programming, 12 (1977), pp. 136–138.

[15] S.P. Han, *A globally convergent method for nonlinear programming*, J. Optim. Theory Appl., 22 (1977), pp. 297–309.

[16] J.-B. Hiriart-Urruty, *Refinements of necessary optimality conditions in nondifferentiable programming* I, Appl. Math. Optim., 5 (1979), pp. 63–82.

[17] R. Janin, *Directional derivative of the marginal function in nonlinear programming*, Math. Programming Stud., 21 (1984), pp. 110–126.

[18] H. Jiang, *Smoothed Fischer-Burmeister Equation Methods for the Complementarity Problem*, Department of Mathematics and Statistics, The University of Melbourne, Australia, June, 1997.

[19] H. Jiang and D. Ralph, *QPECgen, a MATLAB generator for mathematical programs with quadratic objectives and affine variational inequality constraints*, Comput. Optim. Appl., 13 (1999), pp. 25–59.

[20] H. Jiang and D. Ralph, *Smooth SQP Methods for Mathematical Programs with Equilibrium Constraints*, Department of Mathematics and Statistics, The University of Melbourne, Australia, June, 1999 (revised), http://www.ms.unimelb.edu.au/~danny/smooth-mpec.ps.

[21] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 851–868

[22] C. Kanzow and H. Kleinmichel, *A new class of semismooth Newton-type methods for nonlinear complementarity problems*, Comput. Optim. Appl., 11 (1998), pp. 227–251.

[23] M. Kočvara and J.V. Outrata, *On optimization of systems governed by implicit complementarity problems*, Numer. Funct. Anal. Optim., 15 (1994), pp. 869–887.

[24] M. Kočvara and J.V. Outrata, *On the solution of optimum design problems with variational inequalities*, in Recent Advances in Nonsmooth Optimization, D.Z. Du, L. Qi, and R.S. Womersley, eds., World Scientific, Singapore, 1995, pp. 172–192.

[25] A. Kuntsevich and F. Kappel, *SolvOpt: The Solver for Local Nonlinear Optimization Problems*, Institute for Mathematics, Karl-Franzens University of Graz, Austria, 1997.

[26] M.B. Lignola and J. Morgan, *Stability of regularized bilevel programming problem*, J. Optim. Theory Appl., 93 (1997), pp. 575–596.

[27] M.B. Lignola and J. Morgan, *Existence and solutions to generalized bilevel programming problems*, in Multilevel Optimization: Algorithms, Complexity and Applications, A. Migdalas, P.M. Pardalos, and P. Värbrand, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 315–332.

[28] Z.-Q. Luo, J.S. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge, UK, 1996.

[29] Z.-Q. Luo, J.S. Pang, and D. Ralph, *Piecewise sequential quadratic programming for mathematical programs with nonlinear complementarity constraints*, in Multilevel Optimization: Algorithms, Complexity and Applications, A. Migdalas, P.M. Pardalos, and P. Värbrand, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 209–229.

[30] Z.-Q. Luo, J.S. Pang, D. Ralph, and S.-Q. Wu, *Exact penalization and stationarity conditions of mathematical programs with equilibrium constraints*, Math. Programming, 75 (1996), pp. 19–76.

[31] O.L. Mangasarian, *Nonlinear Programming*, McGraw–Hill, New York, 1969.

[32] O.L. Mangasarian, *Mathematical programming in machine learning*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum, New York, 1996, pp. 283–295.

[33] J.V. Outrata, *On optimization problems with variational inequality constraints*, SIAM J. Optim., 4 (1994), pp. 340–357.

[34] J.V. Outrata and J. Zowe, *A numerical approach to optimization problems with variational inequality constraints*, Math. Programming, 68 (1995), pp. 105–130.

[35] M.J.D. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in Nonlinear Programming 3, O.L. Mangasarian, R.R. Meyer, and S.M. Robinson, eds., Academic Press, New York, 1978, pp. 27–63.

[36] M.J.D. Powell, *Variable metric methods for constrained optimization*, in Mathematical Programming: The State of the Art, A. Bachem, M. Grötschel, and B. Korte, eds., Springer-Verlag, Berlin, 1983, pp. 288–311.

[37] D. Ralph, *Sequential quadratic programming for mathematical programs with linear complementarity constraints*, in CTAC95 Computational Techniques and Applications, R.L. May and A.K. Easton, eds., World Scientific, Singapore, 1996, pp. 663–668.

[38] S.M. Robinson, *Stability theory for systems of inequalities, part* II: *Differentiable nonlinear systems*, SIAM J. Numer. Anal., 13 (1976), pp. 497–513.

[39] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[40] H. Scheel and S. Scholtes, *Mathematical programs with equilibrium constraints: Stationarity, optimality, and sensitivity*, Math. Oper. Res., to appear.

[41] S. Scholtes and M. Stöhr, *Exact penalization of mathematical programs with equilibrium constraints*, SIAM J. Control Optim., 37 (1999), pp. 617–652.

[42] N. Yamashita and M. Fukushima, *Modified Newton methods for solving a semismooth reformulation of monotone complementarity problems*, Math. Programming, 76 (1997), pp. 469–497.

[43] J.J. Ye and D.L. Zhu, *Optimality conditions for bilevel programming problems*, Optimization, 33 (1995), pp. 9–27.

SIAM J. Optim.
Vol. 10, No. 3, pp. 809–835

DUALIZATION OF GENERALIZED EQUATIONS OF MAXIMAL MONOTONE TYPE[*]

TEEMU PENNANEN[†]

Abstract. This paper develops a simple duality framework for generalized equations defined by set-valued mappings from a linear space to another. The original problem is related to two auxiliary problems of the similar form, corresponding to Lagrangian and dual problems in the theory of convex programming. As in convex programming, the alternative formulations can be used to obtain information about a given problem and then used to solve it numerically. In particular, dualization can be used in deriving existence criteria for a given problem indirectly by considering one of the alternative formulations. Also, a given problem can often be solved more easily by way of a "dual method." The strongest results of this paper concern monotone mappings. In this context, the duality framework yields several new criteria for maximal monotonicity of composite mappings. These results are useful theoretically as well as in numerical solution of generalized equations. The duality framework can also be used in problem decomposition since dualization can lead to reformulations to which operator-splitting methods and other special methods can be applied.

Key words. duality, inclusions, monotone mappings, composition, maximal monotonicity

AMS subject classifications. 47H05, 47H15, 49N15, 65K10

PII. S1052623498340448

1. Introduction.

A fundamental problem in many branches of applied mathematics is that of finding a point x of a linear space X, such that

$$(\mathcal{P}) \qquad 0 \in F_0(x),$$

where $F_0 : X \rightrightarrows X^*$ is a set-valued mapping assigning to each point x of X a (possibly empty) subset $F_0(x)$ of an associated dual space X^*. Set-valuedness of F_0 allows a wide range of problems in physics, economics, operations research, and other fields to be cast in this form. In addition to all systems of equations and inequalities, a typical example is the case where F_0 is the subdifferential mapping of an extended-real-valued function f_0 [39, Chapter 8]. Then $0 \in F_0(x)$ means that x is a stationary point of f_0, e.g., a local minimizer. Another example is the variational inequality

$$x \in C, \qquad \langle T(x), y - x\rangle \geq 0 \quad \forall y \in C,$$

where $T : X \to X^*$ is single-valued and C is a convex subset of X. This may be written as $(\mathcal{P})$ by defining $F_0 = T + N_C$, where N_C is the set-valued *normal cone mapping* of C

$$N_C(x) = \{x^* \mid \langle x^*, y - x\rangle \leq 0 \quad \forall y \in C\}.$$

An advantage of the inclusion form $0 \in T(x) + N_C(x)$ over the variational form is that, by using more general definitions of normal cones [39, Chapter 6], it allows generalizations where C may be nonconvex. For example, if we replace N_C by the Clarke normal cone mapping, we obtain the *hemivariational inequality problem* studied in

[*] Received by the editors June 15, 1998; accepted for publication (in revised form) September 21, 1999; published electronically May 2, 2000.
http://www.siam.org/journals/siopt/10-3/34044.html

[†] Department of Economics, Helsinki School of Economics and Business Administration, PL 1210, 00101 Helsinki, Finland (pennanen@hkkk.fi).

Goeleven, Stavroulakis, and Panagiotopoulos [18]. The inclusion form is more natural also when T is set-valued.

Problems of the form $(\mathcal{P})$ have been referred to by many names, each having its advantages and disadvantages. For example, the names "generalized equation," "inclusion," and "equation" were used in [27], [6], and [4], respectively. As a compromise between simplicity and accuracy, we will use the term *inclusion.* This is not as descriptive as "generalized equation," and it may be misleading unless clearly defined. We will use it to refer to *problems* of the form $(\mathcal{P})$ as well as to *relations* of the form $x \in C$. The meaning will be clear from the context. This is similar to using the word "equation" to refer to both problems and relations. Also, the term inclusion is consistent with the now standard term "differential inclusion," which refers to generalized differential equations of the form $\dot{x}(t) \in S(x(t))$.

A wealth of theory and algorithms for convex programs is built on the notion of duality. It is thus natural to look for a corresponding theory applicable to a more general class of problems. The main purpose of this paper is to derive a duality framework for problems of the general form $(\mathcal{P})$, with emphasis on the case where the mapping F_0 is *monotone*

$$x_1^* \in F_0(x_1),\ x_2^* \in F_0(x_2) \quad \Longrightarrow \quad \langle x_1 - x_2, x_1^* - x_2^* \rangle \geq 0.$$

An inclusion involving a monotone mapping will be said to be *monotone.* Any convex minimization problem is equivalent to the monotone inclusion where F_0 is the subdifferential mapping of an extended-real-valued objective function. For such problems, the developed duality scheme reduces to familiar duality relations in the theory of convex programming. Viewing the duality relations in terms of subdifferentials leads to a natural duality framework for nonconvex minimization problems as well.

Many authors have presented duality schemes for problems more general than convex programming. In [19], McLinden generalized the *conjugate duality* scheme of Rockafellar [30, 34, 39] to saddle-functions and minimax problems. Mosco [23] and Gabay [17] studied dual pairs of inclusions for mappings that are sums of general monotone mappings and subdifferentials of convex functions. Spingarn [42] introduced a duality framework for general monotone mappings defined on a direct sum of closed Hilbert spaces. In [38], Rockafellar gave a duality scheme for network-type problems involving general set-valued mappings. Eckstein and Ferris [16] presented a duality framework for sums of general monotone mappings. In [4], Attouch and Théra proposed a duality framework for sums of general set-valued mappings from a general linear space to another. In [28], Robinson extended the Attouch–Théra framework for mappings in a composite form.

The duality frameworks of [23, 17, 38, 4, 16, 28] may be regarded as generalizations of the Fenchel–Rockafellar duality for convex programming [30, section 31], whereas [42] generalizes the *complementary duality* model presented in [34, Example 12]. What is missing from this list is a general framework corresponding to the conjugate duality framework, which is the most general and, in many respects, the most useful model for convex programming. In the next section, we extend the conjugate duality scheme to arbitrary set-valued mappings from a linear space to another. This yields a framework where the original inclusion is related to "Lagrangian" and "dual" inclusions. It will be shown that our model inherits many of the powerful features of conjugate duality. Just as the conjugate duality framework may be used to obtain other duality schemes in convex programming, our framework can easily be used to derive the duality frameworks mentioned above.

An important property of the duality scheme is that the original inclusion has a solution if and only if the Lagrangian inclusion and the dual inclusion have one. This can be valuable because the mappings in one of the alternative inclusions may be better behaved, as pointed out by Attouch and Théra in the case of their duality framework. For example, this idea leads to simple proofs of the Brezis–Crandall–Pazy theorem [4] and a nonlinear Hille–Yosida theorem [5]. It also gives a simple way to derive maximality criteria for the sum of monotone mappings [4, 25]. As a broadening of the Attouch–Théra duality framework, our framework has potential for even a wider range of applications. Some examples will be supplied in the following sections.

An essential part of our duality framework is the Lagrangian inclusion, which was not considered explicitly in [4, 28]. In the applications of the next sections, the Lagrangian inclusion will have the key role in studying various monotone mappings.

A monotone mapping is said to be *maximal* if its graph is not properly contained in the graph of another monotone mapping. This is a fundamental property in the general theory of monotone mappings, and it has an essential role in various existence criteria for monotone inclusions as well as in their numerical solution. In section 3, we derive simple maximality criteria for the mappings in a monotone case of the general duality framework in a reflexive Banach space setting. These will be useful for deriving more concrete maximality criteria for mappings with special structure.

In section 4, we specialize the general duality scheme to a generalization of the Fenchel–Rockafellar convex programming model. We obtain simple maximality criteria for mappings in a duality framework associated with inclusions of the form

$$0 \in S(x) + A^*T(Ax),$$

where S and T are monotone mappings and A is a linear set-valued mapping. Our maximality results apply in reflexive Banach spaces, and they unify many earlier results about more special versions of this model. As a corollary, we obtain a generalization of the recent maximality results of Rockafellar and Wets [39, Theorem 12.43] and Robinson [28] for monotone mappings of the form A^*TA. In section 5, we study finite-dimensional problems of the form

$$0 \in S(x) + D_K^*h(x)T(h(x)),$$

where h is a K-convex function and $D_K^*h(x) : U^* \rightrightarrows X^*$ is the adjoint sublinear mapping of the "K-Jacobian" of h at x [24]. This model generalizes the "composite model" in convex programming, and it gives a convenient format for numerous variational problems arising in practice.

Many efficient algorithms for convex programming are obtained by formulating the dual or Lagrangian problem as an inclusion and applying an algorithm designed for general monotone mappings. Such derivations are simplified when one has a duality framework directly in terms of the subdifferential mappings or, more generally, in terms of monotone mappings. Also, this approach yields generalizations of convex programming algorithms to general monotone inclusions [16, 26]. In the last section, we state and prove a decomposition principle for monotone inclusions. It shows how in a certain class of (large-scale) problems, the structure of the dual problem can be employed to obtain decomposition algorithms. Crucial to most numerical methods for monotone inclusions is the maximality of the associated mappings, so that the maximality criteria of section 4 and section 5 will be essential in proving the convergence of these methods.

2. Dualization. In dualizing a convex minimization problem, one is dealing with dual pairs of topological vector spaces. Although our main concern too is with topological vector spaces, we begin more abstractly, with X a general linear space that is associated with another linear space X^*, similarly for Y and Y^*, and so forth. The way X is related to X^* is irrelevant in this section, but it might be helpful to think of them as dual pairs of topological vector spaces. This level of generality is intended to emphasize the simplicity of the duality framework introduced below.

For any two spaces Y and Z, $P_Y : Y \times Z \to Y$ denotes the projection mapping. We define a formal adjoint $P_Y^* : Y^* \to Y^* \times Z^*$ of P_Y by $P_Y^*(y^*) = (y^*, 0)$. With a slight misuse of notation we will use the same symbols for the projection and its adjoint, independent of the space Z. The *graph* of a set-valued mapping $S : X \rightrightarrows X^*$ is defined by $\operatorname{gph} S = \{(x, x^*) \mid x^* \in S(x)\}$. The *domain* $\operatorname{dom} S$ and the *range* $\operatorname{rge} S$ of S are the projections of $\operatorname{gph} S$ to X and X^*, respectively:

$$\begin{aligned}\operatorname{dom} S &= \{x \in X \mid S(x) \neq \emptyset\},\\ \operatorname{rge} S &= \left\{x^* \in X^* \;\middle|\; S^{-1}(x^*) \neq \emptyset\right\},\end{aligned}$$

where S^{-1} is the *inverse* of S, defined by $S^{-1}(x^*) = \{x \in X \mid x^* \in S(x)\}$.

The duality scheme below is based on *partial inversion* of mappings on product spaces.

Definition 2.1. *For a mapping $F : X \times U \rightrightarrows X^* \times U^*$, the* partial inverses *$F^{(-1,1)} : X^* \times U \rightrightarrows X \times U^*$ and $F^{(1,-1)} : X \times U^* \rightrightarrows X^* \times U$ are given by*

$$\begin{aligned}F^{(-1,1)}(x^*, u) &= \{(x, u^*) \mid (x^*, u^*) \in F(x, u)\},\\ F^{(1,-1)}(x, u^*) &= \{(x^*, u) \mid (x^*, u^*) \in F(x, u)\},\end{aligned}$$

that is

$$(x^*, u^*) \in F(x, u) \iff (x, u^*) \in F^{(-1,1)}(x^*, u) \iff (x^*, u) \in F^{(1,-1)}(x, u^*).$$

We have the obvious relations

$$\begin{aligned}(F^{(-1,1)})^{(-1,1)} &= (F^{(1,-1)})^{(1,-1)} = F,\\ (F^{(-1,1)})^{(1,-1)} &= (F^{(1,-1)})^{(-1,1)} = F^{-1},\end{aligned}$$

or more generally, $(F^{(i,j)})^{(k,l)} = F^{(ik,jl)}$ for all $i, j, k, l \in \{-1, 1\}$, where $F^{(1,1)} = F$ and $F^{(-1,-1)} = F^{-1}$.

The idea in generalizing a convex programming duality framework is to express the primal, Lagrangian, and dual problems in terms of the corresponding subdifferential mappings and to replace these by more general set-valued mappings. The main observation of this paper is that doing this for the conjugate duality framework leads to a very general duality scheme. Therefore, we very briefly outline here the conjugate duality scheme for convex programming in the finite-dimensional case. The aim is to interpret the duality relations and optimality conditions in terms of subdifferential mappings of the functions involved. For general and detailed treatments of conjugate duality, we refer the reader to [30], [34], and [39, section 12H].

Let $X = X^* = \mathbb{R}^n$, $U = U^* = \mathbb{R}^m$, and consider the problem of minimizing an extended-real-valued convex function f_0 on X. We assume that a *parameterization* of f_0 has been specified, i.e., f_0 may be expressed as

$$f_0(x) = f(x, 0),$$

for some extended-real-valued convex function f on $X \times U$. The *primal problem* associated with f is to find a minimizer $\bar{x}$ of f_0.

The *Lagrangian* associated with f is the extended-real-valued convex-concave function l on $X \times U^*$ defined by

$$l(x, u^*) = \inf_{u} \{f(x, u) + \langle u, u^* \rangle\}.$$

The *Lagrangian problem* is to find a saddle-point $(\bar{x}, \bar{u}^*)$ of l, with respect to minimizing in x and maximizing in u^*.

The function g on $X^* \times U^*$ defined by

$$\begin{aligned} g(x^*, u^*) &= \inf_{x} \{l(x, u^*) - \langle x, x^* \rangle\} \\ &= \inf_{x,u} \{f(x, u) - \langle x, x^* \rangle + \langle u, u^* \rangle\} = -f^*(x^*, -u^*) \end{aligned}$$

gives a parameterized family of extended-real-valued concave functions $g_{x^*}(u^*) := g(x^*, u^*)$ on U^*. The *dual problem* is to find a maximizer $\bar{u}^*$ of the *dual objective* defined by

$$g_0(u^*) = g(0, u^*).$$

The primal, Lagrangian, and dual problems are equivalent to the inclusions

$$\begin{aligned} 0 &\in \partial f_0(\bar{x}), \\ (0, 0) &\in \partial l(\bar{x}, \bar{u}^*), \\ 0 &\in \partial g_0(\bar{u}^*), \end{aligned} \tag{2.1}$$

respectively [30]. If f is closed, then by [30, Theorem 37.5] the inclusions

$$\begin{aligned} (x^*, -u^*) &\in \partial f(x, u), \\ (x^*, u) &\in \partial l(x, u^*), \\ (-x, u) &\in \partial g(x^*, u^*) \end{aligned} \tag{2.2}$$

are equivalent. That is, ∂l and ∂g are obtained from ∂f by successive partial inversions and some sign changes. Because

$$\partial f_0 = \partial(f P_{X^*}^*) \supset P_{X^*} \partial f P_{X^*}^* \tag{2.3}$$

and

$$\partial g_0 = \partial(g P_U^*) \supset P_U \partial g P_U^*, \tag{2.4}$$

by [30, Theorem 23.9], the equivalent conditions

$$\begin{aligned} (0, -\bar{u}^*) &\in \partial f(\bar{x}, 0), \\ (0, 0) &\in \partial l(\bar{x}, \bar{u}^*), \\ (-\bar{x}, 0) &\in \partial g(0, \bar{u}^*) \end{aligned}$$

guarantee that $0 \in \partial f_0(\bar{x})$ and $0 \in \partial g_0(\bar{u}^*)$. The middle inclusion is an abstract version of the Karush–Kuhn–Tucker (KKT) condition in convex programming [30, Theorem 36.6]. If one has equality in (2.3), then the KKT condition is also necessary for primal optimality; if one has equality in (2.4) it is necessary for dual optimality. By [30, Theorem 23.9], (2.3) holds as an equality if $\operatorname{rge} P^*_{X^*} \cap \operatorname{ri} \operatorname{dom} f \neq \emptyset$, which may be expressed equivalently as $0 \in \operatorname{ri} P_U(\operatorname{dom} f)$. Similarly, a sufficient condition for (2.4) to hold as an equality is $0 \in \operatorname{ri} P_{X^*}(\operatorname{dom} g)$.

The generalization of the conjugate duality framework is now straightforward. We replace ∂f by an arbitrary set-valued mapping $F : X \times U \rightrightarrows X^* \times U^*$ and, as suggested by the relations (2.2), we define mappings $L = F^{1,-1}$ and $G = F^{-1}$, corresponding to ∂l and ∂g, respectively. For simplicity, we omit the sign changes involved in (2.2). The relations (2.3) and (2.4) suggest to define mappings $F_0 = P_{X^*} F P^*_{X^*}$ and $G_0 = P_U G P^*_U$, which correspond to ∂f_0 and ∂g_0. The generalizations of the primal, Lagrangian, and dual problems will be obtained from (2.1) by replacing ∂f_0, ∂l, and ∂g_0 with F_0, L, and G_0, respectively.

DEFINITION 2.2 (duality framework). *Let $F_0 : X \rightrightarrows X^*$ be an arbitrary set-valued mapping, and assume that a* parameterization $F : X \times U \rightrightarrows X^* \times U^*$ *of F_0 has been specified: $F_0 = P_{X^*} F P^*_{X^*}$. The* Lagrangian *corresponding to F is defined by $L = F^{(1,-1)}$. We also define the mappings $G = F^{-1}$ and $G_0 = P_U G P^*_U$. The problems*

$$(\mathcal{P}) \qquad 0 \in F_0(x),$$

$$(\mathcal{L}) \qquad (0,0) \in L(x, u^*),$$

$$(\mathcal{D}) \qquad 0 \in G_0(u^*)$$

are called the primal, Lagrangian, *and* dual inclusion, *respectively.*

The parameterization F may be regarded as describing a family of mappings

$$F_u(x) = P_{X^*}(F(x,u))$$

from X to X^*, each corresponding to a perturbed primal problem $0 \in F_u(x)$. Dually, the mapping G gives rise to the family of mappings

$$G_{x^*}(u^*) = P_U(G(x^*, u^*))$$

from U^* to U, each corresponding to a perturbed dual problem $0 \in G_{x^*}(u^*)$. For a given F_0, there are arbitrarily many ways to introduce a parameterization F, such that $F_0 = P_{X^*} F P^*_{X^*}$. Any such F defines the same primal inclusion, but the forms of the Lagrangian and dual inclusions depend crucially on the particular choice.

Any one of the mappings F, L, or G is enough for defining the three problems uniquely, since the other two may be obtained by partial inversions from the given one:

$$(x^*, u^*) \in F(x,u) \iff (x^*, u) \in L(x, u^*) \iff (x, u) \in G(x^*, u^*).$$

This reveals the perfect symmetry between the primal and dual inclusions. The dual—with respect to the parameterization G—of the dual inclusion is the primal inclusion. It will sometimes be convenient to express both the primal and dual mappings in terms of the Lagrangian:

$$F_0(x) = \{x^* \mid \exists u^* \in U^* : \ (x^*, 0) \in L(x, u^*)\}$$

and

$$G_0(u^*) = \{u \mid \exists x \in X : \ (0, u) \in L(x, u^*)\}\,. \tag{2.5}$$

This corresponds to the expressions $f_0(x) = \sup_{u^*} l(x, u^*)$ and $g_0(u^*) = \inf_x l(x, u^*)$ of the primal and dual objectives in the conjugate duality framework.

By the definition of F_0, a vector $\bar{x}$ is a solution of the primal inclusion if and only if there exists a $\bar{u}^*$ such that $(0, \bar{u}^*) \in F(\bar{x}, 0)$ or, equivalently, $(0, 0) \in L(\bar{x}, \bar{u}^*)$. Dually, a vector $\bar{u}^*$ is a solution of the dual inclusion if and only if there exists an $\bar{x}$ such that $(0, 0) \in L(\bar{x}, \bar{u}^*)$. This is formalized in the following theorem corresponding to the KKT theorem in convex programming.

Theorem 2.3. *The solution sets of $(\mathcal{P})$ and $(\mathcal{D})$ are the projections of the solution set of $(\mathcal{L})$ to X and U^*, respectively. In particular, each of the solution sets is nonempty if and only if the other two are nonempty.*

Note that no particular structure is needed for any of the spaces or mappings above. All the spaces could even be arbitrary sets, as long as the zero elements in X^* and U have been specified. The relations are based solely on the definitions of partial inversion and projection, and the perfect duality for the solution sets is achieved without any conditions.

Consider again the conjugate duality framework in the finite-dimensional case. If f is closed and $F = \partial f$, we have by [30, Theorem 37.5] that

$$L(x, u^*) = \{(x^*, u) \mid (x^*, -u) \in \partial l(x, u^*)\}\,;$$

and if $0 \in \operatorname{ri} P_U(\operatorname{dom} F)$, then $F_0 = \partial f_0$ by [30, Theorem 23.9]. In this case Theorem 2.3 gives the generalized KKT theorem for convex programming [30, Corollary 29.3.1]: under the condition $0 \in \operatorname{ri} P_U(\operatorname{dom} F)$, $\bar{x}$ minimizes f_0 if and only if there exists a $\bar{u}^*$ such that $(\bar{x}, \bar{u}^*)$ is a saddle-point of l. Similarly, if $0 \in \operatorname{ri} P_{X^*}(\operatorname{dom} G)$, we have $G_0 = \partial g_0$ and obtain the dual part. When F is the subdifferential mapping of a parameterized saddle-function, we obtain a KKT theorem for concave-convex saddle-point problems, considered in [19]. Without the constraint qualification, $\operatorname{gph} F_0$ may be properly contained in $\operatorname{gph} \partial f_0$, so that $0 \in F_0(\bar{x})$ is only a sufficient condition for optimality of $\bar{x}$. In any case, Theorem 2.3 is always valid for the inclusions defined directly in terms of ∂f.

In convex programming, the convex-concave form of the Lagrangian function causes the solution set of the Lagrangian problem to be a Cartesian product of two convex sets, namely, the solution sets of the primal and dual problems, respectively. This means that any pair of primal and dual solutions is a solution of the Lagrangian problem. This property does not hold for an arbitrary problem triplet, as noted by McLinden in the case of dual saddle-point problems in [19] and by Spingarn for the duality framework in [42]. However, we still have relations like

$$G_0^{-1}(u) = \{u^* \mid \exists x : \ (0, u^*) \in F(x, u)\}\,,$$

which corresponds to the conjugacy relation between the dual objective and the optimal-value function of a primal minimization problem.

By allowing general set-valued mappings, our framework goes far beyond the field of convex programming or saddle-point problems. Some aspects of the general monotone case will be studied in the following sections. One could also study duality relations for inclusions where F is the generalized subdifferential mapping for a possibly nonconvex extended-real-valued function [39, Chapter 8]. This approach opens new possibilities for generalizing some convex programming algorithms beyond the convex case.

Inspired by the Fenchel–Rockafellar duality framework for convex programming [30, section 31] and its generalization to monotone inclusions [16], we consider the following example studied in Robinson [28].

COROLLARY 2.4 (Fenchel–Rockafellar model). *Let* $S : X \rightrightarrows X^*$, $T : U \rightrightarrows U^*$, $A : X \rightrightarrows U$, *and* $B : U^* \rightrightarrows X^*$ *be set-valued mappings. An* x *solves*

$$(\mathcal{P}_{FR}) \qquad 0 \in S(x) + B(T(A(x)))$$

if and only if there is a u^* *such that*

$$(\mathcal{L}_{FR}) \qquad (0,0) \in [S(x) \times T^{-1}(u^*)] + [B(u^*) \times (-A(x))],$$

in which case u^* *solves*

$$(\mathcal{D}_{FR}) \qquad 0 \in -A(S^{-1}(-B(u^*))) + T^{-1}(u^*).$$

Dually, a u^* *solves* $(\mathcal{D}_{FR})$ *if and only if there exists an* x *such that* (x, u^*) *solves* $(\mathcal{L}_{FR})$, *in which case* x *solves* $(\mathcal{P}_{FR})$.

Proof. We parameterize the mapping $F_0 = S + BTA$ by

$$F(x,u) = S(x) \times \{0\} + \bigcup_{u^* \in T(A(x)+u)} B(u^*) \times \{u^*\}.$$

Then

$$\begin{aligned}(x^*,u^*) \in F(x,u) &\iff x^* \in S(x) + B(u^*), \quad u^* \in T(A(x)+u)\\ &\iff \exists v \in A(x): \quad x^* \in S(x) + B(u^*), \quad u^* \in T(v+u)\\ &\iff \exists v \in A(x): \quad x^* \in S(x) + B(u^*), \quad u \in T^{-1}(u^*) - v\\ &\iff x^* \in S(x) + B(u^*), \quad u \in T^{-1}(u^*) - A(x),\end{aligned}$$

so that

$$L(x,u^*) = F^{(1,-1)}(x,u^*) = S(x) \times T^{-1}(u^*) + B(u^*) \times (-A(x)).$$

Similarly,

$$G(x^*,u^*) = L^{(-1,1)}(x^*,u^*) = \{0\} \times T^{-1}(u^*) + \bigcup_{x \in S^{-1}(x^* - B(u^*))} \{x\} \times (-A(x)),$$

so that $G_0 = -AS^{-1}(-B) + T^{-1}$. Thus, this model is a special case of the general duality scheme, and everything follows from Theorem 2.3. ▯

If A and B are at most single-valued, the Lagrangian inclusion above can be rewritten as

$$-B(u^*) \in S(x) \quad \text{and} \quad A(x) \in T^{-1}(u^*),$$

which corresponds to the KKT conditions in [30, Theorem 31.1]. When $X = X^* = \mathbb{R}^n$, $U = U^* = \mathbb{R}^m$, and $B = A^*$, where A is the incidence matrix of a network, we obtain the network equilibrium model of Rockafellar [38] by setting $S = S_1 \times \cdots \times S_n$ and $T = T_1 \times \cdots \times T_m$, where n is the number of arcs and m is the number of nodes.

When $X = U$, $X^* = U^*$, and A and B are the identity mappings on X and X^*, respectively, we obtain the duality framework of Attouch and Théra [4] for the sum of set-valued mappings. As noted in [4], the underlying relations in the Singer–Toland duality scheme as well as in the Clarke–Ekeland least dual action principle can be viewed as special cases of this model.

By Theorem 2.3, the existence of solutions to an inclusion can be guaranteed by proving the existence of a dual solution. In the dual inclusion of Corollary 2.4, one is dealing with the inverses of the original mappings. As pointed out by Attouch and Théra, this can be very useful since the inverses may behave better than the mappings themselves; see, for example, [4, 5].

The above model can be written compactly in terms of the following notation. For any mappings $S_{11} : X_1 \rightrightarrows X_1^*$, $S_{12} : X_2 \rightrightarrows X_1^*$, $S_{21} : X_1 \rightrightarrows X_2^*$, and $S_{22} : X_2 \rightrightarrows X_2^*$, we define the mapping

$$\begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} : X_1 \times X_2 \rightrightarrows X_1^* \times X_2^*$$

by

$$\begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} (x_1, x_2) = (S_{11}(x_1) + S_{12}(x_2)) \times (S_{21}(x_1) + S_{22}(x_2)).$$

The proof of Corollary 2.4 shows that the Fenchel–Rockafellar model corresponds to the general duality framework with

$$\begin{aligned} F_0 &= \begin{bmatrix} I & B \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} I \\ A \end{bmatrix}, \\ F &= \begin{bmatrix} I & B \\ 0 & I \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} I & 0 \\ A & I \end{bmatrix}, \\ L &= \begin{bmatrix} S & 0 \\ 0 & T^{-1} \end{bmatrix} + \begin{bmatrix} 0 & B \\ -A & 0 \end{bmatrix} = \begin{bmatrix} S & B \\ -A & T^{-1} \end{bmatrix}, \\ G &= \begin{bmatrix} I & 0 \\ -A & I \end{bmatrix} \begin{bmatrix} S^{-1} & 0 \\ 0 & T^{-1} \end{bmatrix} \begin{bmatrix} I & -B \\ 0 & I \end{bmatrix}, \\ G_0 &= \begin{bmatrix} -A & I \end{bmatrix} \begin{bmatrix} S^{-1} & 0 \\ 0 & T^{-1} \end{bmatrix} \begin{bmatrix} -B \\ I \end{bmatrix}. \end{aligned}$$

A straightforward generalization of this model is obtained by replacing the mappings

$$\begin{bmatrix} S & 0 \\ 0 & T \end{bmatrix}, \qquad \begin{bmatrix} S & 0 \\ 0 & T^{-1} \end{bmatrix}, \qquad \begin{bmatrix} S^{-1} & 0 \\ 0 & T^{-1} \end{bmatrix}$$

by R, $R^{(1,-1)}$, and R^{-1}, respectively, where $R : X \times U \rightrightarrows X^* \times U^*$ is a general set-valued mapping.

Corollary 2.5. *Let $R : X \times U \rightrightarrows X^* \times U^*$, $A : X \rightrightarrows U$, and $B : U^* \rightrightarrows X^*$ be set-valued mappings. An x solves*

$$(\mathcal{P}_H) \qquad 0 \in \begin{bmatrix} I & B \end{bmatrix} R(x, Ax)$$

if and only if there is a u^* such that

$$(\mathcal{L}_H) \qquad (Bu^*, -Ax) \in R^{(1,-1)}(x, u^*),$$

in which case u^* solves

$$(\mathcal{D}_H) \qquad 0 \in \begin{bmatrix} -A & I \end{bmatrix} R^{-1}(-Bu^*, u^*).$$

Dually, a u^* solves $(\mathcal{D}_H)$ if and only if there exists an x such that (x, u^*) solves $(\mathcal{L}_H)$, in which case x solves $(\mathcal{P}_H)$.

Proof. As suggested by the preceding discussion, we introduce the parameterization

$$F = \begin{bmatrix} I & B \\ 0 & I \end{bmatrix} R \begin{bmatrix} I & 0 \\ A & I \end{bmatrix}.$$

Much as in the proof of Corollary 2.4, it follows that

$$L = R^{(1,-1)} + \begin{bmatrix} 0 & B \\ -A & 0 \end{bmatrix}$$

and

$$G = \begin{bmatrix} I & 0 \\ -A & I \end{bmatrix} R^{-1} \begin{bmatrix} I & -B \\ 0 & I \end{bmatrix},$$

so that $G_0(u^*) = \begin{bmatrix} -A & I \end{bmatrix} R^{-1}(-Bu^*, u^*)$. The result now follows from Theorem 2.3. □

The Lagrangian inclusion in the above model may be regarded as a generalized Hamiltonian system, where the mapping $R^{(1,-1)}$ corresponds to the gradient of the Hamiltonian and A and B play the role of differential operators. Problem $(\mathcal{P}_H)$ corresponds to minimization problems of the form

$$\text{minimize} \quad f(x, Ax),$$

where $A : X \to U$ is linear and f is an extended-real-valued function on the product space $X \times U$; see [34, p. 28]. As pointed out in [34], such problems can be viewed as problems of minimizing the function f over the subspace $\operatorname{gph} A$ of $X \times U$. In this sense, the following model due to Spingarn [42, section 3] (only monotone mappings were considered in [42]) is more general.

Corollary 2.6 (Spingarn's model). *Let $H = A \otimes B$ be an orthogonal decomposition of a Hilbert space H into closed subspaces A and B, and let $T : H \rightrightarrows H$ be a set-valued mapping. Let $S = T^{-1}$, and define $T_0 : A \rightrightarrows A$ and $S_0 : B \rightrightarrows B$ by*

$$\operatorname{gph} T_0 = \{(x, y_A) \mid y \in T(x),\ x \in A\},$$
$$\operatorname{gph} S_0 = \{(y, x_B) \mid x \in S(y),\ y \in B\},$$

where y_A is the projection of y to A and x_B is the projection of x to B. An x solves

$$(\mathcal{P}_S) \qquad 0 \in T_0(x)$$

if and only if there is a y such that

$$(\mathcal{L}_S) \qquad x \in A, \qquad y \in B, \qquad y \in T(x),$$

in which case y solves

$$(\mathcal{D}_S) \qquad 0 \in S_0(y).$$

Dually, a y solves $(\mathcal{D}_S)$ if and only if there exists an x such that (x, y) solves $(\mathcal{L}_S)$, in which case x solves $(\mathcal{P}_S)$.

Proof. This can be written in the format of Definition 2.2 by setting $X = X^* = A$, $U = U^* = B$, and

$$F(x,u) = \{(y_A, y_B) \mid y \in T(x+u)\}.$$

Indeed, $(\mathcal{P}_S)$ is equivalent to finding an $x \in X$ such that $(0, u^*) \in F(x, 0)$ for some $u^* \in U^*$, which means that $0 \in F_0(x)$, where $F_0 = P_{X^*} F P^*_{X^*}$. The problem $(\mathcal{L}_S)$ is equivalent to finding $x \in X$ and $u^* \in U^*$ such that $(0, u^*) \in F(x, 0)$, which means that $(0, 0) \in L(x, u^*)$. Similarly, $(\mathcal{D}_S)$ can be written as $0 \in G_0(u^*)$ for the mapping $G_0 = P_U G P^*_U$, where $G = F^{-1}$. Thus, everything follows from Theorem 2.3. □

Whereas the general model of Definition 2.2 corresponds to the conjugate duality scheme, Spingarn's model corresponds to the complementary duality model for convex programming [34, Example 12]. In our model the expressions for F_0 and G_0 are more explicit than the expressions for the corresponding mappings T_0 and S_0 and the parameter u may have practical significance representing data perturbations. For applications such as those considered below, our model seems to be more convenient. However, there are applications in the derivation of numerical algorithms for monotone inclusions for which Spingarn's model is more suitable [42].

A reader familiar with [34] has probably recognized the similarity of the above examples with some of the examples of the conjugate duality framework, considered in [34, section 5]. A model corresponding to the ordinary convex programming model will be studied at the end of section 5, as a special case of the "composite model," which is yet another instance of the general duality framework.

3. Monotonicity in duality. In this section X and U are reflexive Banach spaces with duals X^* and U^*, respectively. We will use the inner-product notation for the pairing between a space and its dual. The product spaces $X \times U$, $X \times U^*$, etc. are reflexive Banach spaces with the obvious duals and pairings.

An important example of a monotone mapping is the subdifferential ∂f of a convex function f. If f is closed, then ∂f is maximal [33]. Of specific interest is the indicator function δ_C of a convex set $C \subset X$:

$$\delta_C(x) = \begin{cases} 0 & \text{if } x \in C, \\ \infty & \text{if } x \notin C. \end{cases}$$

The closedness of δ_C is equivalent to the closedness of C, which thus implies the maximality of the *normal cone mapping* $N_C = \partial \delta_C$ of C. Another important class of monotone mappings arises from saddle-functions; see [32]. If l is a convex-concave function on $X \times U^*$, then the mapping $\tilde{\partial} l$ defined by

$$\tilde{\partial} l(x, u^*) = \partial_x l(x, u^*) \times \partial_{u^*}[-l](x, u^*)$$

is monotone. If l can be expressed as $l(x, u^*) = \inf_u\{\langle u, u^* \rangle + f(x, u)\}$ for a closed convex function f, then $\tilde{\partial} l$ is maximal monotone.

The following two results are our main tools for studying maximal monotonicity in the duality framework.

LEMMA 3.1. *For a mapping $F : X \times U \rightrightarrows X^* \times U^*$, the monotonicity and maximality of F, $F^{(-1,1)}$, $F^{(1,-1)}$, and F^{-1} are equivalent.*

Proof. Monotonicity of F means that

$$\langle (x_1,u_1)-(x_2,u_2),(x_1^*,u_1^*)-(x_2^*,u_2^*)\rangle \geq 0 \quad \forall (x_i,u_i,x_i^*,u_i^*)\in \operatorname{gph} F. \tag{3.1}$$

This can be written in terms of $F^{(-1,1)}$ and the pairing between $X\times U^*$ and $X^*\times U$ as

$$\langle (x_1^*,u_1)-(x_2^*,u_2),(x_1,u_1^*)-(x_2,u_2^*)\rangle \geq 0 \quad \forall (x_i^*,u_i,x_i,u_i^*)\in \operatorname{gph} F^{(-1,1)},$$

which means that $F^{(-1,1)}$ is monotone. Maximality of F means that if (x_1,u_1,x_1^*,u_1^*) satisfies the inequality in (3.1) for every $(x_2,u_2,x_2^*,u_2^*)\in \operatorname{gph} F$, then $(x_1,u_1,x_1^*,u_1^*)\in \operatorname{gph} F$. Writing this in terms of $F^{(-1,1)}$ and the pairing between $X\times U^*$ and $X^*\times U$, we see that it is equivalent to the maximality of $F^{(-1,1)}$. The other equivalences are proved similarly. □

The following maximality result for the sum of two monotone mappings was obtained in Attouch, Riahi, and Théra [3] and Chu [11]. For a set C in a topological vector space the *relative interior* $\operatorname{ri} C$ of C is the interior of C with respect to the relative topology on $\operatorname{cl}\operatorname{aff} C$.

THEOREM 3.2. *Let $T_1,T_2 : U \rightrightarrows U^*$ be maximal monotone. If*

$$0\in \operatorname{ri}(\operatorname{dom} T_1-\operatorname{dom} T_2),$$

then T_1+T_2 is maximal monotone.

In [3], the constraint qualification is written in the form

$$\bigcup_{\alpha>0}\alpha(\operatorname{dom} T_1-\operatorname{dom} T_2) \text{ is a closed subspace,}$$

which is clearly implied by $0\in \operatorname{ri}(\operatorname{dom} T_1-\operatorname{dom} T_2)$. However, by Simons [40, Theorem 23.2] these conditions are actually equivalent. For simplicity, we will use the first condition, although in practice, the second might be easier to verify. We refer the reader to [40] for an exposition of various recent results on monotone mappings.

One of the major drawbacks of the above constraint qualification in infinite dimensions is that, in general, the relative interior does not obey the same rules as in finite dimensions [30, section 6] and, more seriously, it may well be empty, even for convex sets. However, in a product space, we still have

$$\operatorname{ri}(C_1\times C_2)=\operatorname{ri} C_1\times \operatorname{ri} C_2.$$

This fact will be used without further mention.

Our main result now follows easily.

THEOREM 3.3. *Monotonicity and maximality of F, L, and G are equivalent, and monotonicity of F implies the monotonicity of F_0 and G_0. If F is maximal monotone, the following hold:*

(a) *If $0\in \operatorname{ri} P_U(\operatorname{dom} F)$, then F_0 is maximal monotone.*

(b) *If $0\in \operatorname{ri} P_{X^*}(\operatorname{dom} G)$, then G_0 is maximal monotone.*

Moreover, $P_U(\operatorname{dom} F)$ and $P_{X^*}(\operatorname{dom} G)$ may be expressed as

$$P_U(\operatorname{dom} F)=P_U(\operatorname{rge} L)=P_U(\operatorname{rge} G)$$

and

$$P_{X^*}(\operatorname{dom} G)=P_{X^*}(\operatorname{rge} L)=P_{X^*}(\operatorname{rge} F).$$

Proof. The first facts follow from Lemma 3.1 and from the fact that A^*TA is monotone for any continuous linear mapping A and monotone T. The expressions for $P_U(\operatorname{dom} F)$ and $P_{X^*}(\operatorname{dom} G)$ follow directly from the definitions of partial inverses. To finish, it suffices to prove (a) since (b) then follows by symmetry.

Let $N_0 : U \rightrightarrows U^*$ and $N : X \times U \rightrightarrows X^* \times U^*$ denote the normal cone mappings of $\{0\}$ and $X \times \{0\}$, respectively:

$$N_0(u) = \begin{cases} U^* & \text{if } u = 0, \\ \emptyset & \text{if } u \neq 0, \end{cases} \qquad N(x,u) = \begin{cases} \{0\} \times U^* & \text{if } u = 0, \\ \emptyset & \text{if } u \neq 0. \end{cases}$$

Since N_0 is maximal, F_0 is maximal if and only if $F_0 \times N_0$ is maximal. Since $F_0 \times N_0 = F + N$, where both F and N are maximal, Theorem 3.2 implies that F_0 is maximal if

$$(0,0) \in \operatorname{ri}(\operatorname{dom} F - \operatorname{dom} N) = \operatorname{ri}(\operatorname{dom} F - X \times \{0\}) = \operatorname{ri}[X \times P_U(\operatorname{dom} F)]$$

or, equivalently, if $0 \in \operatorname{ri} P_U(\operatorname{dom} F)$. □

Consider the conjugate duality framework, with $F = \partial f$. If f is closed, the condition $0 \in \operatorname{ri} P_U(\operatorname{dom} F)$ guarantees that F_0 is maximal monotone, so $\operatorname{gph} F_0$ cannot be properly contained in $\operatorname{gph} \partial f_0$. As noted in the previous section, this implies that the KKT condition is necessary and sufficient for primal optimality. An analogous statement applies to the dual problem. The same argument can be used in the case of saddle-point problems with closed saddle-functions.

Theorem 3.3 allows us to deduce the maximality of F_0 and G_0 indirectly by first establishing the maximality of L, then forming expressions for $P_U(\operatorname{rge} L)$ and $P_{X^*}(\operatorname{rge} L)$, and checking whether the conditions in (a) and (b) hold. The simplest application is the following.

COROLLARY 3.4. *Let F be maximal monotone.*

(a) *If $0 \in \operatorname{int} \operatorname{rge} F_0$, then G_0 is maximal monotone.*

(b) *If $0 \in \operatorname{int} \operatorname{rge} G_0$, then F_0 is maximal monotone.*

Proof. We have

$$\operatorname{rge} F_0 = \{x^* \mid \exists x, u^* : \; (x^*, u^*) \in F(x,0)\} \subset P_{X^*}(\operatorname{rge} F) = P_{X^*}(\operatorname{dom} G),$$

so that $\operatorname{int} \operatorname{rge} F_0 \subset \operatorname{int} P_{X^*}(\operatorname{dom} G)$. Theorem 3.3(b) now implies (a), and (b) is verified similarly. □

The condition $0 \in \operatorname{int} \operatorname{rge} F_0$ is trivially satisfied if $\operatorname{rge} F_0 = X^*$. If F_0 is maximal, this holds if F_0 is *weakly coercive,* i.e., if $\|x_k^*\|$ is unbounded for any unbounded sequence (x_k, x_k^*) in $\operatorname{gph} F_0$ [43, Corollary 32.35]. This happens in particular when $\operatorname{dom} F_0$ is bounded, which is the case in many situations arising in practice. Note that we cannot weaken the condition $0 \in \operatorname{int} \operatorname{rge} F_0$ to $0 \in \operatorname{ri} \operatorname{rge} F_0$, since $C_1 \subset C_2$ does not, in general, imply $\operatorname{ri} C_1 \subset \operatorname{ri} C_2$.

If $F = \partial f$, the condition in Theorem 3.3(a) also guarantees the existence of a dual solution and the condition in (b) guarantees the existence of a primal solution [30, Corollary 30.5.2]. This is not true in the general monotone case. This is related to the fact that the solution set to $(\mathcal{L})$ is not, in general, the Cartesian product of the solution sets to $(\mathcal{P})$ and $(\mathcal{D})$. The following class of mappings, first considered in [8], behaves much like subdifferentials in this respect.

DEFINITION 3.5. *A monotone mapping T is* star-monotone *if $(\bar{u}, \bar{u}^*) \in \operatorname{dom} T \times \operatorname{rge} T$ implies*

$$\inf_{u^* \in T(u)} \langle u - \bar{u}, u^* - \bar{u}^* \rangle > -\infty.$$

The following is proved in [25].

THEOREM 3.6. *Let $A : X \to U$ be linear and continuous, and let $T : U \rightrightarrows U^*$ be star-monotone. If A^*TA is maximal monotone, then*

$$A^*(\operatorname{rge} T) \subset \operatorname{cl}\operatorname{rge}(A^*TA)$$

and

$$\operatorname{ri}\operatorname{co} A^*(\operatorname{rge} T) \subset \operatorname{rge}(A^*TA).$$

As shown in [8], strongly monotone mappings and subdifferentials of convex functions are special cases of star-monotone mappings. The following may thus be considered as a generalization of the existence result [30, Corollary 35.5.2] for the conjugate duality framework; see also [34, section 7].

PROPOSITION 3.7. *If F is star-monotone, any one of the following conditions guarantees that $(\mathcal{P})$, $(\mathcal{L})$, and $(\mathcal{D})$ are all solvable.*

(a) *F_0 is maximal, and $0 \in \operatorname{ri}\operatorname{co} P_{X^*}(\operatorname{dom} G)$.*

(b) *G_0 is maximal, and $0 \in \operatorname{ri}\operatorname{co} P_U(\operatorname{dom} F)$.*

(c) *F is maximal, $0 \in \operatorname{ri} P_{X^*}(\operatorname{dom} G)$, and $0 \in \operatorname{ri} P_U(\operatorname{dom} F)$.*

Proof. By applying Theorem 3.6 to the mapping $F_0 = P_{X^*}FP_{X^*}^*$, we obtain $\operatorname{ri}\operatorname{co} P_{X^*}(\operatorname{rge} F) \subset \operatorname{rge} F_0$, where by Theorem 3.3 $P_{X^*}(\operatorname{dom} G) = P_{X^*}(\operatorname{rge} F)$. This proves part (a), and part (b) is verified similarly after noting that star-monotonicity is preserved under inversion. By Theorem 3.3, condition (c) implies both (a) and (b). □

A major difficulty with star-monotonicity is that it is generally not preserved under operations such as addition, composition, or partial inversion. As a result, it may be hard to check in practice for a mapping that has been constructed from other mappings.

4. The Fenchel–Rockafellar model.

In this section we specialize the results of the previous section to a monotone case of the model of Corollary 2.4. We continue to assume that all the spaces are reflexive Banach spaces, unless otherwise specified.

We will say that a set-valued mapping $A : X \rightrightarrows U$ is *linear* if $\operatorname{gph} A$ is a linear subspace of $X \times U$. These mappings were introduced in [1] under the name *linear relation,* and they have been studied also in [9] and [7] under the names of *linear operator* and *linear process,* respectively. See also the recent monograph [13]. Linear set-valued mappings are special cases of *sublinear mappings,* which are set-valued mappings whose graphs are convex cones [29, 30, 39].

It is easily seen that sums, compositions, and inverses of linear set-valued mappings are linear. The *adjoint* of $A : X \rightrightarrows U$ is the linear mapping $A^* : U^* \rightrightarrows X^*$ given by

$$A^*(u^*) = \left\{x^* \;\middle|\; (x^*, -u^*) \in (\operatorname{gph} A)^\perp \right\} = \{x^* \mid \langle x, x^*\rangle = \langle u, u^*\rangle \;\; \forall (x, u) \in \operatorname{gph} A\}.$$

Linear set-valued mappings arise naturally as inverses and adjoints of single-valued linear mappings when these fail to exist as single-valued mappings. More precisely, consider a linear single-valued mapping A_0 from a subspace $\operatorname{dom} A_0 \subset X$ to U. Then the set-valued mapping

$$A(x) = \begin{cases} \{A_0(x)\} & \text{if } x \in \operatorname{dom} A_0, \\ \emptyset & \text{if } x \notin \operatorname{dom} A_0, \end{cases}$$

as well as its set-valued inverse and adjoint are linear. For a linear mapping $A : X \rightrightarrows U$, $A^*(0) = (\operatorname{dom} A)^\perp$, so $\operatorname{cl}\operatorname{dom} A = X$ if and only if $A^*(0) = \{0\}$, which is equivalent to A^* being at most single-valued. Dually, if A is closed we have $A^{**} = A$, and A is at most single-valued if and only if $\operatorname{cl}\operatorname{dom} A^* = U^*$.

Now, consider the model of Corollary 2.4 and assume that $S : X \rightrightarrows X^*$ and $T : U \rightrightarrows U^*$ are monotone, $A : X \rightrightarrows U$ is linear, and $B = A^*$. Then

$$\begin{aligned}
F_0 &= S + A^*TA,\\
F &= \begin{bmatrix} I & A^* \\ 0 & I \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & T \end{bmatrix} \begin{bmatrix} I & 0 \\ A & I \end{bmatrix},\\
L &= \begin{bmatrix} S & 0 \\ 0 & T^{-1} \end{bmatrix} + \begin{bmatrix} 0 & A^* \\ -A & 0 \end{bmatrix},\\
G &= \begin{bmatrix} I & 0 \\ -A & I \end{bmatrix} \begin{bmatrix} S^{-1} & 0 \\ 0 & T^{-1} \end{bmatrix} \begin{bmatrix} I & -A^* \\ 0 & I \end{bmatrix},\\
G_0 &= T^{-1} + (-A)S^{-1}(-A^*).
\end{aligned}$$

The case where X and U are Hilbert spaces and A is continuous has been studied by Robinson [28]. The finite-dimensional case, with $A : X \to U$ single-valued, was studied in Eckstein and Ferris [16]. The primal inclusion $0 \in F_0(x)$ in the Fenchel–Rockafellar model may be written as

$$\exists x^* \in S(x) : \quad -x^* \in (A^*TA)(x),$$

and the dual inclusion $0 \in G_0(u^*)$ may be written as

$$\exists u \in [(-A)S^{-1}(-A^*)](u^*) : \quad -u \in T^{-1}(u^*).$$

If $T = \partial g$ for a closed convex function g on U, we have $T^{-1} = \partial g^*$ and $A^*T(Ax) \subset \partial(gA)(x)$. If the second formula holds as an equality (see [34, Theorem 19] or Corollary 4.4 below) $(\mathcal{P})$ and $(\mathcal{D})$ can be written in the variational form as

$$\exists x^* \in S(x) : \quad \langle x^*, y - x\rangle \geq (gA)(x) - (gA)(y) \quad \forall y \in X$$

and

$$\exists u \in -A(S^{-1}(-A^*(u^*))) : \quad \langle u, v^* - u^*\rangle \geq g^*(u^*) - g^*(v^*) \quad \forall v^* \in U^*.$$

This is the duality framework studied by Gabay [17]. When $U = X$ and $A = I$, we obtain the duality scheme introduced by Mosco [23].

Under special assumptions, maximality criteria for the mappings F_0 and G_0 were given in [23, 17, 16, 28]. Using dualization and the abstract results of the previous section, the question of the maximality of a composite mapping can be reduced to a question of the maximality of a sum. This yields the following maximality result for the Fenchel–Rockafellar model.

THEOREM 4.1. *All the mappings F, L, G, F_0, and G_0 are monotone. If S and T are maximal monotone, $A : X \rightrightarrows U$ is closed, $0 \in \operatorname{ri}(\operatorname{dom} S - \operatorname{dom} A)$, and $0 \in \operatorname{ri}(\operatorname{rge} T - \operatorname{dom} A^*)$, then F, L, and G are maximal monotone. When this happens, then also*

(a) *if $0 \in \operatorname{ri}[T^{-1}(\operatorname{dom} A^*) - A(\operatorname{dom} S)]$, then*

$$S + A^*TA$$

is maximal monotone;

(b) *if* $0 \in \operatorname{ri}[S(\operatorname{dom} A) + A^*(\operatorname{rge} T)]$, *then*

$$T^{-1} + (-A)S^{-1}(-A^*)$$

is maximal monotone.

Proof. The Lagrangian may be written as the sum $L = F_1^{(1,-1)} + F_2^{(1,-1)}$, where $F_1 = S \times T$, and

$$F_2(x,u) = \{(x^*,u^*) \mid (x^*,-u^*) \in N_{\operatorname{gph} A}(x,-u)\}.$$

The mapping F_1 is monotone since S and T are, and the mapping F_2 is monotone by the monotonicity of $N_{\operatorname{gph} A}$. By Lemma 3.1, $F_1^{(1,-1)}$ and $F_2^{(1,-1)}$ are monotone, which implies the monotonicity of L, and Theorem 3.3 then implies the monotonicity of the other mappings.

Similarly, the maximality of S and T implies that of $F_1^{(1,-1)}$, and the closedness of A implies the maximality of F_2, which by Lemma 3.1 is equivalent to the maximality of $F_2^{(1,-1)}$. To prove the maximality of L, it suffices by Theorem 3.2 to show that $(0,0) \in \operatorname{ri}(\operatorname{dom} F_1^{(1,-1)} - \operatorname{dom} F_2^{(1,-1)})$. Since

$$\begin{aligned}\operatorname{dom} F_1^{(1,-1)} - \operatorname{dom} F_2^{(1,-1)} &= \operatorname{dom} S \times \operatorname{rge} T - \operatorname{dom} A \times \operatorname{dom} A^* \\ &= (\operatorname{dom} S - \operatorname{dom} A) \times (\operatorname{rge} T - \operatorname{dom} A^*),\end{aligned}$$

this is implied by the conditions $0 \in \operatorname{ri}(\operatorname{dom} S - \operatorname{dom} A)$ and $0 \in \operatorname{ri}(\operatorname{rge} T - \operatorname{dom} A^*)$. Maximality of F and G now follows from Theorem 3.3. Parts (a) and (b) follow from parts (a) and (b) of Theorem 3.3 by noting that $P_U(\operatorname{rge} L) = T^{-1}(\operatorname{dom} A^*) - A(\operatorname{dom} S)$ and $P_{X^*}(\operatorname{rge} L) = S(\operatorname{dom} A) + A^*(\operatorname{rge} T)$. □

In the important special case where A is continuous (and single-valued), we have $\operatorname{dom} A = X$ and $\operatorname{dom} A^* = U^*$, so that the conditions in the above result simplify considerably.

Corollary 4.2. *Assume that* S *and* T *are maximal monotone, and* $A : X \to U$ *is continuous. Then* F, L, *and* G *are maximal monotone, and*

(a) *if* $0 \in \operatorname{ri}[\operatorname{dom} T - A(\operatorname{dom} S)]$, *then* $S + A^*TA$ *is maximal monotone;*

(b) *if* $0 \in \operatorname{ri}[\operatorname{rge} S + A^*(\operatorname{rge} T)]$, *then* $T^{-1} + (-A)S^{-1}(-A^*)$ *is maximal monotone.*

If X and U are finite-dimensional, we can use the calculus of relative interiors in [30, section 6] to write the above conditions more explicitly[1]. Using Theorem 6.6 and Corollary 6.6.2 of [30], we obtain the following result.

Corollary 4.3. *Consider the Fenchel–Rockafellar model in the finite-dimensional case, and assume that* S *and* T *are maximal monotone and* $A : X \to U$ *is single-valued. Then the mappings* F, L, *and* G *are maximal monotone, and*

(a) *if there is an* $x \in \operatorname{ri} \operatorname{dom} S$, *such that* $A(x) \in \operatorname{ri} \operatorname{dom} T$, *then the mapping* $S + A^*TA$ *is maximal monotone;*

(b) *if there is a* $u^* \in \operatorname{ri} \operatorname{rge} T$, *such that* $-A^*(u^*) \in \operatorname{ri} \operatorname{rge} S$, *then the mapping* $T^{-1} + (-A)S^{-1}(-A^*)$ *is maximal monotone.*

If $S = \partial f$ and $T = \partial g$ for convex functions on X and U, respectively, these conditions become the "strong consistency" conditions for the Fenchel–Rockafellar

[1] In [22], Minty showed that the domain C of a maximal monotone mapping in $\mathbb{R}^n$ is *almost convex,* in the sense that it contains the relative interior of its convex hull: $C \supset \operatorname{ri}(\operatorname{co} C)$. Because $\operatorname{aff} C = \operatorname{aff}(\operatorname{co} C)$, this may be expressed as $\operatorname{ri} C = \operatorname{ri}(\operatorname{co} C)$. Consequently, one can easily verify that the calculus rules for relative interiors of convex sets in [30, section 6] are valid for almost convex sets as well.

convex programming model in [30, Theorem 31.2]. As noted earlier, the maximality of the mappings would imply that equalities hold in (2.3) and (2.4), in which case the KKT conditions become necessary for optimality; see also [30, Corollary 31.2.1]. Similarly, if S and T are monotone mappings associated with saddle-functions, we obtain a KKT theorem for McLinden's saddle-function version [21] of the Fenchel–Rockafellar model.

If $U = X$ and $A = I$, we have $\operatorname{dom} A = X$ and $\operatorname{dom} A^* = X^*$, so that we recover Theorem 3.2. In the case $S = 0$, we obtain the following generalization of [39, Theorem 12.43] and [28, Theorem 5].

Corollary 4.4. *Let $A : X \rightrightarrows U$ be closed and linear, and let $T : U \rightrightarrows U^*$ be maximal monotone. The mapping A^*TA is maximal monotone if any one of the following holds:*

(a) $0 \in \operatorname{ri}(\operatorname{rge} A - \operatorname{dom} T)$ *and* $0 \in \operatorname{ri}[T(\operatorname{rge} A) - \operatorname{dom} A^*]$;

(b) $0 \in \operatorname{ri}[\operatorname{rge} A - T^{-1}(\operatorname{dom} A^*)]$ *and* $0 \in \operatorname{ri}(\operatorname{rge} T - \operatorname{dom} A^*)$;

(c) *A is continuous and* $0 \in \operatorname{ri}(\operatorname{rge} A - \operatorname{dom} T)$.

Proof. The condition (b) follows from Theorem 4.1(a) by setting $S = 0$ so that $\operatorname{dom} S = X$. The condition (a) is obtained by applying (b) to $(A^*TA)^{-1} = A^{-1}T^{-1}(A^*)^{-1}$ whose maximality is equivalent to the maximality of A^*TA. When A is continuous, the conditions (a) and (b) both reduce to (c). □

Part (c) above can be applied to partial differential equations in "divergence form." As an example, one may consider the case where $A = \nabla : H_0^1(\Omega) \to L_2(\Omega)^n$ ($n = \dim \Omega$) and T is a monotone mapping from $L_2(\Omega)^n$ to itself, say conductivity. The mapping ∇ is continuous and has the adjoint $\nabla^* = -\operatorname{div} : L_2(\Omega) \to H^{-1}(\Omega)$, the *divergence.*

For additional insights on Corollary 4.4, consider the case where $T = \partial f$ for a convex function f on U and A is closed and single-valued on a dense subspace $\operatorname{dom} A \subset X$. Define the convex function h on X by

$$h(x) = \begin{cases} f(A(x)) & \text{if } x \in \operatorname{dom} A, \\ \infty & \text{if } x \notin \operatorname{dom} A. \end{cases}$$

If f is closed and $0 \in \operatorname{ri}(\operatorname{rge} A - \operatorname{dom} f)$, then $\partial h = A^*\partial f A$ by [34, Theorem 19]. If h is closed, this implies the maximality of $A^*\partial f A$. However, when A is not continuous, h need not be closed, and thus, without further conditions, we cannot guarantee the maximality of $A^*\partial f A$. It can be shown that if $0 \in \operatorname{ri}(\operatorname{dom} f^* - \operatorname{dom} A^*)$, then h is closed, so that $A^*\partial f A$ is maximal. A simple example where the closedness of h fails is obtained by setting $f = 0$. Then $h = \delta_{\operatorname{dom} A}$, which is not closed unless $\operatorname{dom} A$ is.

For general $A : X \rightrightarrows U$, it is easily shown that we always have $\partial(fA) \supset A^*\partial f A$, where

$$(fA)(x) = \inf_{u \in A(x)} f(u).$$

If f and A are closed, we have $(\partial f)^{-1} = \partial f^*$, and Corollary 4.4 shows that if $0 \in \operatorname{ri}(\operatorname{dom} \partial f^* - \operatorname{dom} A^*)$ and $0 \in \operatorname{ri}[\partial f^*(\operatorname{dom} A^*) - \operatorname{rge} A]$, then $A^*\partial f A$ is maximal monotone, so that the inclusion must hold as an equality. This corresponds to a composition version of the subdifferential rule in [2, Corollary 2.1] for general Banach spaces. Similarly, one could use Theorem 4.1 and its corollaries to derive calculus rules for subdifferentials of closed saddle-functions on reflexive Banach spaces.

When U is a Hilbert space and T is the identity mapping, Corollary 4.4 yields the following.

COROLLARY 4.5. *If U is a Hilbert space and $A : X \rightrightarrows U$ is closed and linear, then the linear mapping $A^*A : X \rightrightarrows X^*$ is maximal monotone and, in particular, closed.*

Proof. By Corollary 4.4, it suffices to show that $\operatorname{dom} A^* - \operatorname{rge} A = U$. The closedness of A implies the maximality of $N_{\operatorname{gph} A}$, so that, since $N_{\operatorname{gph} A}(x, u) = \operatorname{gph} A^*$ for any $(x, u) \in \operatorname{gph} A$, we have by Rockafellar's generalization [31, Proposition 1] of Minty's lemma that

$$X^* \times U^* = \operatorname{rge}(N_{\operatorname{gph} A} + J_{X\times U}) = \operatorname{gph} A^* + J_{X\times U}(\operatorname{gph} A),$$

where $J_{X\times U}(x, u) = \partial(1/2\|(x, u)\|^2)$. Since $\|(x, u)\|^2 = \|x\|^2 + \|u\|^2$, we have $J_{X\times U} = J_X \times I$, where $J_X(x) = \partial(1/2\|x\|^2)$, so the result follows by projecting on $U^* = U$. □

If we choose $T = I$ in the situation discussed after Corollary 4.4, we obtain the well-known fact that the negative *Laplacian* $-\Delta = -\operatorname{div}\circ\nabla$ is maximal monotone.

5. The composite model.

In this section we develop a generalization of the convex programming composite model that was studied in [24]. From now on all the spaces are finite-dimensional Euclidean spaces with the usual inner-products. This enables us to use the strong calculus rules in [30, section 76] for computing relative interiors of (almost) convex sets.

For a convex cone $K \subset U$, a function h from a subset $\operatorname{dom} h$ of X to U is said to be *K-convex* if $\operatorname{dom} h$ is convex, and for any $x_1, x_2 \in \operatorname{dom} h$

$$h(\alpha_1 x_1 + \alpha_2 x_2) - \alpha_1 h(x_1) - \alpha_2 h(x_2) \in K$$

whenever $\alpha_1, \alpha_2 \in [0, 1]$ and $\alpha_1 + \alpha_2 = 1$. Just as in the extended-real-valued case, it is easily shown that K-convexity of h is equivalent to the convexity of the *K-epigraph*

$$\operatorname{epi}_K h = \{(x, u) \mid x \in \operatorname{dom} h,\ h(x) - u \in K\}$$

of h. We say that h is *K-closed* if $\operatorname{epi}_K h$ is closed. The *K-range* $\operatorname{rge}_K h$ of h is defined as the projection of $\operatorname{epi}_K h$ to U. It follows from the definition that h is $(\operatorname{cl} K)$-convex if and only if the extended-real-valued functions $\langle u^*, h\rangle (x) := \langle u^*, h(x)\rangle$, with domain $\operatorname{dom} h$, are convex for every u^* in the *polar cone*

$$K^* = \{u^* \in U^* \mid \langle u, u^*\rangle \le 0 \ \forall u \in K\}$$

of K. For $x \in \operatorname{dom} h$, we define the sublinear mapping $D^*_K h(x) : U^* \rightrightarrows X^*$ by

$$D^*_K h(x)(u^*) = \begin{cases} \partial \langle u^*, h\rangle (x) & \text{if } u^* \in K^*, \\ \emptyset & \text{if } u^* \notin K^*. \end{cases}$$

This is the adjoint of the "K-Jacobian" of h. The *K-recession cone* $\operatorname{rc}_K h$ of h is the set of directions in which h is nonincreasing with respect to the partial order induced by K

$$\operatorname{rc}_K h = \{y \mid h(x + \lambda y) - h(x) \in K \quad \forall x \in \operatorname{dom} h,\ \forall \lambda \ge 0\}.$$

The convex programming composite model is the minimization problem

$$\text{minimize} \quad f_0 := f + g\circ h \quad \text{over} \quad \operatorname{dom}(f + g\circ h),$$

where f and g are extended-real-valued convex functions on X and U, respectively, and h is a K-convex function from X to U, such that $K \subset \operatorname{rc}_{\mathbb{R}_-} g$. As pointed out in

[24], this model subsumes many standard models in convex programming. Note that, since $\operatorname{dom}(f + g\circ h) = \{x \in \operatorname{dom} f \cap \operatorname{dom} h \mid h(x) \in \operatorname{dom} g\}$, any convex constraints can be represented by restricting the domain of f or h or both. If there exists an $x \in \operatorname{ri}\operatorname{dom} f \cap \operatorname{ri}\operatorname{dom} h$ such that $h(x) \in \operatorname{ri}\operatorname{dom} g$, then by [24, Proposition 23] the subdifferential of f_0 may be expressed as

$$\partial f_0(x) = \partial f(x) + D_K^* h(x)(\partial g(h(x))).$$

A generalization of the optimization model is obtained by replacing ∂f and ∂g in this formula by general monotone mappings. We will use the notation $\tilde{\partial}$ introduced at the beginning of section 3.

PROPOSITION 5.1 (composite model). *Let $S : X \rightrightarrows X^*$ and $T : U \rightrightarrows U^*$ be maximal monotone, and let h be a K-convex K-closed function from X to U such that* $\operatorname{rge} T \subset K^*$. *An x solves*

$$(\mathcal{P}_{comp}) \qquad 0 \in S(x) + D_K^* h(x)(T(h(x)))$$

if and only if there is a u^* such that

$$(\mathcal{L}_{comp}) \qquad (0, 0) \in S(x) \times T^{-1}(u^*) + \tilde{\partial} H(x, u^*),$$

where H is the saddle-function

$$H(x, u^*) = \begin{cases} \langle u^*, h(x)\rangle - \delta_K(u^*) & \textit{if } x \in \operatorname{dom} h, \\ +\infty & \textit{otherwise.} \end{cases}$$

Proof. We parameterize the mapping $F_0(x) = S(x) + D_K^* h(x)(T(h(x)))$ by

$$F(x, u) = \begin{bmatrix} S(x) \\ 0 \end{bmatrix} + \begin{bmatrix} D_K^* h(x) \\ I \end{bmatrix} (T(h(x) + u)).$$

For any maximal monotone T and a closed convex set $C \supset \operatorname{dom} T$, we have $T(u) = T(u) + N_C(u)$ for all u. Thus, the condition $\operatorname{rge} T \subset K^*$ implies $T = (T^{-1} + N_{K^*})^{-1}$, so that

$$\begin{aligned} (x^*, u^*) \in F(x, u) \iff & \quad x^* \in S(x) + D_K^* h(x)(u^*), \quad u^* \in (T^{-1} + N_{K^*})^{-1}(h(x) + u) \\ \iff & \quad x^* \in S(x) + \partial \langle u^*, h\rangle (x), \quad h(x) + u \in T^{-1}(u^*) + N_{K^*} \\ \iff & \quad \begin{bmatrix} x^* \\ u \end{bmatrix} \in \begin{bmatrix} S(x) \\ T^{-1}(u^*) \end{bmatrix} + \begin{bmatrix} \partial \langle u^*, h\rangle (x) \\ N_{K^*}(u^*) - h(x) \end{bmatrix}, \end{aligned}$$

which shows that $(\mathcal{L}_{comp})$ is the Lagrangian problem corresponding to $(\mathcal{P}_{comp})$. The result follows from Theorem 2.3. ∎

If $T = \partial g$ for a closed convex function g, the condition $\operatorname{rge} T \subset K^*$ is equivalent to $K \subset \operatorname{rc}_{\mathbb{R}_-} g$. It is easily checked that this implies the convexity of the composition $g\circ h$, so that $\partial(g\circ h)$ is monotone. The condition $\operatorname{rge} T \subset K^*$ could be replaced by the weaker condition

$$(T^{-1} + N_{K^*})^{-1}(u) = T(u) \cap K^* \quad \forall u \in \operatorname{rge} h,$$

which corresponds to the convexity condition for $g\circ h$ in [24, section 7]. All the results in this section can be generalized accordingly, but for simplicity we will consider only the condition $\operatorname{rge} T \subset K^*$. Note also that neither maximality of S nor K-closedness of

h was used in the above proof. However, since the main results of this section require those properties anyway, we have chosen to simplify things by assuming them from the start.

The proof of Proposition 5.1 shows that the composite model corresponds to the general duality framework with $F_0(x) = S(x) + D_K^* h(x)(T(h(x)))$,

$$F(x,u) = \begin{bmatrix} S(x) \\ 0 \end{bmatrix} + \begin{bmatrix} D_K^* h(x) \\ I \end{bmatrix} (T(h(x)+u)),$$

and

$$L = S \times T^{-1} + \tilde{\partial} H.$$

Note that since $\operatorname{rge} T \subset K^*$ implies that $T^{-1}(u^*) + N_{K^*}(u^*) = T^{-1}(u^*)$ for all u^*, the Lagrangian can also be written as

$$L(x,u^*) = \begin{bmatrix} S(x) \\ T^{-1}(u^*) \end{bmatrix} + \begin{bmatrix} \partial \langle u^*, h \rangle (x) \\ -h(x) \end{bmatrix}.$$

If $h = A$ for a linear $A : X \to U$, we may choose $K = \{0\}$, so that $K^* = U^*$, and $D_K^* h(x) = A^*$ for any $x \in \operatorname{dom} h$. In this case the composite model reduces to the finite-dimensional case of the Fenchel–Rockafellar model. In the general case, the mappings G and G_0 do not have explicit expressions in terms of S, T, and h, but in practice it may be possible to calculate G_0 according to the formula (2.5). That is, for a fixed $u^* \in K^*$, one first finds the solutions to the inclusion

$$0 \in S(x) + \partial \langle u^*, h \rangle (x)$$

and then forms the union of the sets $T^{-1}(u^*) - h(x)$ over all such solutions x. This approach corresponds to "dual methods" in convex programming, and it might be useful in numerical solution of monotone inclusions. Such considerations depend on monotonicity and maximality of the mapping G_0. The first step is to guarantee these properties for F, L, and G.

PROPOSITION 5.2. *All the mappings F, L, G, F_0, and G_0 in the composite model are monotone; and if* $\operatorname{ri} \operatorname{dom} S \cap \operatorname{ri} \operatorname{dom} h \neq \emptyset$ *and* $\operatorname{ri} \operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$, *then F, L, and G are maximal monotone.*

Proof. As a sum of two monotone mappings, L is monotone, which by the first part of Theorem 3.3 implies the monotonicity of the other mappings. The first term in the expression $L = S \times T^{-1} + \tilde{\partial} H$ is maximal if and only if S and T are. The function H can be expressed as $H(x,u^*) = \inf_u \{\langle u, u^* \rangle + \delta_{\operatorname{epi}_K h}(x,u)\}$, where the closedness of $\delta_{\operatorname{epi}_K h}$ is equivalent to the K-closedness of h. Thus, $\tilde{\partial} H$ is maximal monotone [32].

Obviously, $\operatorname{dom}(S \times T^{-1}) = \operatorname{dom} S \times \operatorname{rge} T$. By [30, Theorem 34.2], the domain of H is $\operatorname{dom} h \times K^*$, so that by [30, Theorem 37.4], $\operatorname{ri} \operatorname{dom} \tilde{\partial} H = \operatorname{ri} \operatorname{dom} h \times \operatorname{ri} K^*$. Thus, by Theorem 3.2, the conditions $\operatorname{ri} \operatorname{dom} S \cap \operatorname{ri} \operatorname{dom} h \neq \emptyset$ and $\operatorname{ri} \operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ guarantee the maximality of L, which is equivalent to the maximality of F and G. □

Note that since $\operatorname{rge} T \subset K^*$, the condition $\operatorname{ri} \operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ holds by [30, Corollary 6.5.2] whenever $\operatorname{rge} T$ is not entirely contained in the relative boundary $K^* \setminus \operatorname{ri} K^*$ of K^*.

To derive maximality criteria for F_0 and G_0 in the composite model, we need some preliminary results that are of interest on their own. In [8], the following result is obtained by invoking Lemma $1'$ instead of Lemma 1 in the proof of the third variation of Theorem 4 and using the remark on page 167; see also [25].

THEOREM 5.3 (Brezis–Haraux). *Let S be a monotone mapping, and let f be a convex function such that $S + \partial f$ is maximal. If* $\operatorname{dom} S \subset \operatorname{dom} f$, *then*

$$\operatorname{ri}\operatorname{rge}(S + \partial f) = \operatorname{ri}(\operatorname{rge} S + \operatorname{rge} \partial f).$$

The following corresponds to the subdifferential chain rule [24, Corollary 22.1(d)].

THEOREM 5.4. *Let $T : U \rightrightarrows U^*$ be maximal monotone, and let h be a K-convex K-closed function from X to U, such that* $\operatorname{rge} T \subset K^*$ *and* $\operatorname{ri}\operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$. *If* $\operatorname{ri}\operatorname{rge}_K h \cap \operatorname{ri}\operatorname{dom} T \neq \emptyset$, *then the mapping $F_0 : X \rightrightarrows X^*$, defined by*

$$F_0(x) = D_K^* h(x) T(h(x)),$$

is maximal monotone.

Proof. We have $F_0 = P_{X^*} F P_{X^*}^*$, where F corresponds to the composite model in the special case $S = 0$. Hence, by Proposition 5.2, the condition $\operatorname{ri}\operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ guarantees the maximality of F. To guarantee the maximality of F_0, it suffices by Theorem 3.3(a) to show that $\operatorname{ri} P_U(\operatorname{rge} L) = \operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h)$, since then the condition $\operatorname{ri}\operatorname{rge}_K h \cap \operatorname{ri}\operatorname{dom} T \neq \emptyset$ implies $0 \in \operatorname{ri} P_U(\operatorname{rge} L)$.

The closedness of $\operatorname{epi}_K h$ implies that of K, so that $K = K^{**} = \operatorname{rge} N_{K^*}$. By [30, Corollary 6.6.2] and [24, Proposition 18] we have

$$\begin{aligned}\operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h) &= \operatorname{ri}\operatorname{dom} T - \operatorname{ri}\operatorname{rge}_K h = \operatorname{ri}\operatorname{dom} T + \bigcup_{x \in \operatorname{ri}\operatorname{dom} h} (\operatorname{ri} K - h(x)) \\ &= \bigcup_{x \in \operatorname{ri}\operatorname{dom} h} [\operatorname{ri}(\operatorname{rge} T^{-1} + \operatorname{rge} N_{K^*}) - h(x)].\end{aligned}$$

Since $\operatorname{ri}\operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$, the sum $T^{-1} + N_{K^*}$ is maximal, so that the condition $\operatorname{rge} T \subset K^*$ implies by Theorem 5.3 that

$$\begin{aligned}\operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h) &= \bigcup_{x \in \operatorname{ri}\operatorname{dom} h} [\operatorname{ri}\operatorname{rge}(T^{-1} + N_{K^*}) - h(x)] \\ &\subset \bigcup_{\substack{x \in \operatorname{ri}\operatorname{dom} h \\ u^* \in K^*}} [T^{-1}(u^*) + N_{K^*}(u^*) - h(x)].\end{aligned}$$

By [30, Theorem 23.4], $x \in \operatorname{ri}\operatorname{dom} h$ and $u^* \in K^*$ imply $\partial \langle u^*, h \rangle (x) \neq \emptyset$, so that

$$\operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h) \subset \bigcup_{D_K^* h(x)(u^*) \neq \emptyset} [T^{-1}(u^*) + N_{K^*}(u^*) - h(x)] = P_U(\operatorname{rge} L).$$

On the other hand, $\operatorname{rge} L \subset \operatorname{rge}[\,0 \times T^{-1}] + \operatorname{rge} \tilde{\partial} H$, so that $P_U(\operatorname{rge} L) \subset \operatorname{dom} T + P_U(\operatorname{rge} \tilde{\partial} H)$, where

$$P_U(\operatorname{rge} \tilde{\partial} H) \subset \bigcup_{x \in \operatorname{dom} h} (K - h(x)) = -\operatorname{rge}_K h.$$

We have thus obtained the inclusions

$$\operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h) \subset P_U(\operatorname{rge} L) \subset \operatorname{dom} T - \operatorname{rge}_K h,$$

which by [30, Theorem 6.3] imply $\operatorname{cl} P_U(\operatorname{rge} L) = \operatorname{cl}(\operatorname{dom} T - \operatorname{rge}_K h)$ and $\operatorname{ri} P_U(\operatorname{rge} L) = \operatorname{ri}(\operatorname{dom} T - \operatorname{rge}_K h)$. □

If we choose $h(x) = Ax$ and $K = \{0\}$, we have $F_0 = A^*TA$ and $K^* = U^*$, so that the conditions $\operatorname{rge} T \subset K^*$ and $\operatorname{ri} \operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ hold trivially. Since now $\operatorname{rge}_K h = \operatorname{rge} A$, the above theorem reduces to the finite-dimensional version of Corollary 4.4.

When we combine the mapping $F_0(x) = D_K^* h(x) T(h(x))$ with other mappings, the following lemma will be useful.

Lemma 5.5. *Let F_0 be as in Theorem 5.4. The condition $\operatorname{ri} \operatorname{rge}_K h \cap \operatorname{ri} \operatorname{dom} T \neq \emptyset$ holds if and only if there exists an $x \in \operatorname{ri} \operatorname{dom} h$ such that $h(x) \in \operatorname{ri} \operatorname{dom} T$. When this happens,*

$$\operatorname{ri} \operatorname{dom} F_0 = \{x \in \operatorname{ri} \operatorname{dom} h \mid h(x) \in \operatorname{ri} \operatorname{dom} T\}.$$

Proof. By [25, Corollary 11.3], the maximality of T and the condition $\operatorname{rge} T \subset K^*$ imply that $K \subset \operatorname{rc} \operatorname{cl} \operatorname{dom} T$. Since $\operatorname{dom} T$ is almost convex, the first part follows from [24, Lemma 21]. To obtain the expression for $\operatorname{ri} \operatorname{dom} F_0$, we first note that

$$\operatorname{dom} F_0 = \{x \in \operatorname{dom} h \mid h(x) \in \operatorname{dom} T,\ \operatorname{dom} D_K^* h(x) \cap T(h(x)) \neq \emptyset\},$$

where $\operatorname{dom} D_K^* h(x) = \{u^* \in K^* \mid \partial \langle u^*, h \rangle (x) \neq \emptyset\}$. Defining

$$C = \{x \in \operatorname{dom} h \mid h(x) \in \operatorname{cl} \operatorname{dom} T\},$$

we have $\operatorname{dom} F_0 \subset C$. Thus, by [24, Lemma 21], the condition $\operatorname{ri} \operatorname{rge}_K h \cap \operatorname{ri} \operatorname{dom} T \neq \emptyset$ implies

$$\operatorname{ri} C = \{x \in \operatorname{ri} \operatorname{dom} h \mid h(x) \in \operatorname{ri} \operatorname{dom} T\}.$$

By [30, Theorem 23.4], $\operatorname{dom} D_K^* h(x) = K^*$ for any $x \in \operatorname{ri} \operatorname{dom} h$, so that $\operatorname{rge} T \subset K^*$ implies

$$\operatorname{ri} C \subset \operatorname{dom} F_0 \subset C$$

and, hence, $\operatorname{ri} \operatorname{dom} F_0 = \operatorname{ri} C$ by [30, Theorem 6.3]. □

The maximality condition for G_0 will be stated in terms of the adjoint $R_K^* h : U^* \rightrightarrows X^*$ of the "K-recession mapping" of h [24]. By [24, Proposition 15], this is the closure of the mapping

$$H(u^*) = \begin{cases} \operatorname{dom} \langle u^*, h \rangle^* & \text{for } u^* \in K^*, \\ \emptyset & \text{for } u^* \notin K^*. \end{cases}$$

Calculus rules for computing K-recession mappings are given in [24, section 7].

Theorem 5.6. *Consider the composite model, and assume that $\operatorname{ri} \operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ and $\operatorname{ri} \operatorname{dom} S \cap \operatorname{ri} \operatorname{dom} h \neq \emptyset$. Then the following hold:*

(a) *If there exists an $x \in \operatorname{ri} \operatorname{dom} S \cap \operatorname{ri} \operatorname{dom} h$ such that $h(x) \in \operatorname{ri} \operatorname{dom} T$, then F_0 is maximal.*

(b) *Assume that $\operatorname{dom} S \subset \operatorname{dom} h$ or that S is star-monotone. If $0 \in \operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T))$, then G_0 is maximal.*

Proof. The condition in (a) guarantees by Theorem 5.4 that the second term in F_0 is maximal and by Lemma 5.5 that the relative interior of its domain intersects $\operatorname{ri} \operatorname{dom} S$. Hence, Theorem 3.2 implies the maximality of F_0. By Proposition 5.2, F is maximal, so to prove (b), it suffices by Theorem 3.3(b) to show that $\operatorname{dom} S \subset$

$\operatorname{dom} h$ or, alternatively, star-monotonicity of S implies that $\operatorname{ri} P_{X^*}(\operatorname{dom} G) = \operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T))$.

By [30, Corollary 6.6.2] and [24, Corollary 11.1]

$$\begin{aligned}\operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T)) &= \operatorname{ri}\operatorname{rge} S + \operatorname{ri} R_K^* h(\operatorname{rge} T)\\ &= \operatorname{ri}\operatorname{rge} S + \bigcup \left\{\operatorname{ri} R_K^* h(u^*) \;\middle|\; u^* \in \operatorname{ri}\operatorname{rge} T\right\},\end{aligned}$$

where $\operatorname{ri} R_K^* h(u^*) = \operatorname{ri}\operatorname{dom} \langle u^*, h\rangle^*$. The conditions $\operatorname{rge} T \subset K^*$ and $\operatorname{ri}\operatorname{rge} T \cap \operatorname{ri} K^* \neq \emptyset$ imply by [30, Corollary 6.5.2] that $\operatorname{ri}\operatorname{rge} T \subset \operatorname{ri} K^*$, so that by [24, Corollary 20.1(b)], $\langle u^*, h\rangle$ is a closed convex function for any $u^* \in \operatorname{ri}\operatorname{rge} T$. By Theorem 23.4 and Corollary 23.5.1 of [30], $\operatorname{ri}\operatorname{dom} \langle u^*, h\rangle^* = \operatorname{ri}\operatorname{dom} \partial \langle u^*, h\rangle^* = \operatorname{ri}\operatorname{rge} \partial \langle u^*, h\rangle$, so that

$$\begin{aligned}\operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T)) &= \operatorname{ri}\operatorname{rge} S + \bigcup \left\{\operatorname{ri}\operatorname{rge} \partial \langle u^*, h\rangle \;\middle|\; u^* \in \operatorname{ri}\operatorname{rge} T\right\}\\ &= \bigcup_{u^* \in \operatorname{ri}\operatorname{rge} T} (\operatorname{ri}\operatorname{rge} S + \operatorname{ri}\operatorname{rge} \partial \langle u^*, h\rangle).\end{aligned}$$

Since $\operatorname{ri}\operatorname{dom} S \cap \operatorname{ri}\operatorname{dom} \partial \langle u^*, h\rangle \neq \emptyset$, $S + \partial \langle u^*, h\rangle$ is maximal by Theorem 3.2. Thus, the condition $\operatorname{dom} S \subset \operatorname{dom} h$ and Theorem 5.3 or, alternatively, star-monotonicity of T and the third variation of [8, Theorem 3] (see also [25, Corollary 11.1]) imply that

$$\begin{aligned}\operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T)) &= \bigcup_{u^* \in \operatorname{ri}\operatorname{rge} T} \operatorname{ri}\operatorname{rge}(S + \partial \langle u^*, h\rangle)\\ &\subset \bigcup_{u^* \in \operatorname{rge} T} \operatorname{rge}(S + \partial \langle u^*, h\rangle) = P_{X^*}(\operatorname{rge} L).\end{aligned}$$

On the other hand,

$$P_{X^*}(\operatorname{rge} L) \subset \operatorname{rge} S + \bigcup_{u^* \in \operatorname{rge} T} \operatorname{dom} \langle u^*, h\rangle^* \subset \operatorname{rge} S + R_K^* h(\operatorname{rge} T),$$

so that [30, Theorem 6.3] implies $\operatorname{ri} P_{X^*}(\operatorname{rge} L) = \operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T))$. □

As in the proof of [24, Proposition 23], one may show that, under the assumptions of Proposition 5.2, the condition $0 \in \operatorname{ri}(\operatorname{rge} S + R_K^* h(\operatorname{rge} T))$ in part (b) is equivalent to the existence of a $\bar{u}^* \in \operatorname{ri}\operatorname{rge} T$ such that $\operatorname{ri}\operatorname{dom} S \cap - \operatorname{ri}\operatorname{dom} \langle u^*, h\rangle^* \neq \emptyset$, where $\langle u^*, h\rangle^*$ is the convex conjugate of $\langle u^*, h\rangle$. If $h = A$ for a linear $A : X \to U$ and $K = \{0\}$, the conditions of Proposition 5.2 hold trivially and $D_K^* h(x) = R_K^* h = A^*$ for any $x \in X$, so that Theorem 5.6 reduces to Corollary 4.3.

The following corresponds to minimization problems of the form

$$\text{minimize} \quad f(x) \quad \text{subject to} \quad h(x) \in K.$$

Again, by restricting $\operatorname{dom} f$ or $\operatorname{dom} h$, any convex constraints can be represented by the feasible set $\{x \in \operatorname{dom} f \cap \operatorname{dom} h \mid h(x) \in K\}$.

Corollary 5.7. *Consider the composite model in the case $T = N_K$. Then x solves $(\mathcal{P}_{comp})$ if and only if there exists a u^* such that*

$$\begin{gathered}0 \in S(x) + \partial \langle u^*, h\rangle (x),\\ h(x) \in K, \qquad u^* \in K^*, \qquad \langle u^*, h(x)\rangle = 0.\end{gathered}$$

If $\operatorname{ri}\operatorname{dom} S \cap \operatorname{ri}\operatorname{dom} h \neq \emptyset$, then F, L, and G are maximal and the following hold:

(a) *If there exists an* $x \in \operatorname{ri}\operatorname{dom} h \cap \operatorname{ri}\operatorname{dom} S$ *such that* $h(x) \in \operatorname{ri} K$, *then* F_0 *is maximal.*

(b) *If* $\operatorname{dom} S \subset \operatorname{dom} h$ *(or if* S *is star-monotone) and* $0 \in \operatorname{ri}(\operatorname{rge} S + (\operatorname{rc}_K h)^*)$, *then* G_0 *is maximal.*

Proof. The first claim follows by noting that $u^* \in N_K(h(x))$ is equivalent to the given complementarity condition. When h is K-closed, K has to be closed too so that N_K is maximal and $\operatorname{rge} T = K^*$. The maximality of F, L, and G now follows from Proposition 5.2, and the maximality condition for F_0 is obtained from Theorem 5.6(a). Part (b) follows from Theorem 5.6(b), since $R_K^* h(K^*) = \operatorname{rge} R_K^* h = (\operatorname{rc}_K h)^*$, by [24, Lemma 3]. □

The special case $K = \mathbb{R}^n_-$ was considered in [31, 37]. If S is (at most) single-valued and there exists an $x \in \operatorname{ri}\operatorname{dom} h$ such that $h(x) \in \operatorname{ri} K$, then by [24, Proposition 24] the primal inclusion is equivalent to the variational inequality

$$\langle S(x), y - x\rangle \geq 0 \qquad \forall y \in C,$$

where $C = \{x \in X \mid h(x) \in K\}$. When $U = X$ and $h = I$, it is equivalent to the complementarity problem

$$x \in K, \qquad -S(x) \in K^*, \qquad \langle S(x), x\rangle = 0.$$

Now that maximality criteria for the composite model have been established, one can apply proximal point algorithms [35, 36, 15] to the Lagrangian or dual inclusion. This leads to "multiplier methods" for the composite model [26]. In [16], Eckstein and Ferris used this approach in a special case of the Fenchel–Rockafellar model, obtaining efficient numerical algorithms for variational inequalities. As mentioned earlier, since the general duality framework of section 2 does not depend on monotonicity, it has potential in deriving numerical methods for nonmonotone variational problems as well. One could also apply operator splitting methods [14] to the Lagrangian $L = S \times T^{-1} + \tilde{\partial} H$ in Proposition 5.1.

6. Decomposition.

This section generalizes the classical decomposition principle of convex programming and McLinden's decomposition principle for saddle-point problems [20]. The setting is finite-dimensional.

Consider the composite model in the separable case: $X = X_1 \times \cdots \times X_n$, $S = S_1 \times \cdots \times S_n$, $h = h_1 + \cdots + h_n$, and $K = K_1 + \cdots + K_n$, where S_i is a monotone mapping on X_i and h_i is a K_i-convex function from X_i to U. The mapping F becomes

$$F(x,u) = \begin{bmatrix} S_1(x_1) \\ \vdots \\ S_n(x_n) \\ 0 \end{bmatrix} + \begin{bmatrix} D^*_{K_1} h_1(x_1) \\ \vdots \\ D^*_{K_n} h_n(x_n) \\ I \end{bmatrix} T\left(\sum_{i=1}^n h_i(x_i) + u\right),$$

and the problems $(\mathcal{P}_{comp})$ and $(\mathcal{L}_{comp})$ specialize to

$$(\mathcal{P}_{sep}) \qquad 0 \in S_i(x_i) + D^*_{K_i} h_i(x_i) T\left(\sum_{i=1}^n h_i(x_i)\right) \quad \forall i = 1, \ldots, n,$$

$$(\mathcal{L}_{sep}) \qquad \begin{array}{l} 0 \in S_i(x_i) + D^*_{K_i} h_i(x_i)(u^*) \quad \forall i = 1, \ldots, n, \\ 0 \in T^{-1}(u^*) - \sum_{i=1}^n h_i(x_i). \end{array}$$

In ($\mathcal{L}_{sep}$), we have used the fact that $T^{-1} + N_{K^*} = T^{-1}$, since $\operatorname{rge} T \subset K^*$. This model corresponds to minimization problems of the form

$$\text{minimize} \quad \sum_{i=1}^{n} f_i(x_i) + g\left(\sum_{i=1}^{n} h_i(x_i)\right),$$

where f_i and g are extended-real-valued convex functions. When g is the indicator function of K, one obtains a more conventional form to which the convex programming decomposition principle is usually applied. In the case of saddle-point problems, f_i and g would be saddle-functions in appropriate product spaces, h_i would map to the domain-space of g, and K would be of the form $K_1 \times K_2$ for K_1 and K_2 in the spaces of the convex and concave argument of g, respectively.

Again, we are unable to obtain explicit expressions for G or G_0 in terms of S, T, and h_is. However, the mapping G_0 decomposes into a sum that can be expressed in terms of mappings G_0^i that correspond to *independent* composite models in the subspaces X_i.

PROPOSITION 6.1. *Consider the composite model in the separable case. If the sets* $\operatorname{ri} K_i^*$ *have a point in common, then* G_0 *may be written as the sum*

$$G_0 = T^{-1} + G_0^1 + \cdots + G_0^n, \tag{6.1}$$

where, for $i = 1, \ldots, n$, G_0^i *is the monotone mapping in the dual inclusion* ($\mathcal{D}_i$) *corresponding to the parameterization*

$$F_i(x_i, u) = \begin{bmatrix} S_i(x_i) \\ 0 \end{bmatrix} + \begin{bmatrix} D_{K_i}^* h_i(x_i) \\ I \end{bmatrix} N_{K_i}(h_i(x_i) + u)$$

of the inclusion

$$0 \in S_i(x_i) + D_{K_i}^* h_i(x_i) N_{K_i}(h_i(x_i)). \tag{$\mathcal{P}_i$}$$

Proof. The maximality of T and the inclusion $\operatorname{rge} T \subset K^*$ imply $T^{-1} = T^{-1} + N_{K^*}$, so that by Proposition 5.1, the expression (2.5) may be written as

$$G_0(u^*) = T^{-1}(u^*) + \bigcup_x \left\{ N_{K^*}(u^*) - h(x) \ \mid 0 \in S(x) + \partial \langle u^*, h \rangle (x) \right\}. \tag{6.2}$$

In the separable case we have $K^* = K_1^* \cap \cdots \cap K_n^*$, and if the sets $\operatorname{ri} K_i^*$ have a point in common, $N_{K^*} = N_{K_1^*} + \cdots + N_{K_n^*}$ by [30, Corollary 23.8.1]. Hence, (6.2) may be written as the sum

$$G_0 = T^{-1} + G_0^1 + \cdots + G_0^n,$$

where

$$G_0^i(u^*) = \bigcup_x \left\{ N_{K_i^*}(u^*) - h_i(x_i) \ \mid 0 \in S_i(x_i) + \partial \langle u^*, h_i \rangle (x_i) \right\}.$$

Inverting the logic that took us to (6.2), we find that G_0^i corresponds to F_i given above. □

We see that the subproblems ($\mathcal{P}_i$) are exactly in the form of Corollary 5.7. In the separable case, $G_0(u^*)$ may be evaluated by computing $T^{-1}(u^*)$ and $G_0^i(u^*)$ for $i = 1, \ldots, n$ independently of each other.

Recently, Burachik, Sagastizábal, and Svaiter [10] obtained bundle methods that can be used to solve inclusions for general maximal monotone mappings. These methods require only that one is able to compute a single element in the image of any given point. Currently, this algorithm seems to be the only "direct" algorithm for general monotone mappings, i.e., it does not require the computation of resolvants; see also the paper of Solodov and Svaiter [41], which develops a general algorithmic framework with potential applications in this direction. Combining the above decomposition principle and the algorithm in [10], one obtains implementable decomposition algorithms for monotone inclusions, much as for convex minimization when using ordinary bundle methods or other algorithms for nonsmooth convex minimization. The separable form could also be utilized by applying operator splitting methods [42, 14, 12] for finding a solution of $(\mathcal{D})$. The convergence of all of these methods depends on the maximality of the involved mappings, so the maximality results in the previous section may prove to be useful in this context.

REFERENCES

[1] R. Arens, *Operational calculus of linear relations*, Pacific J. Math., 11 (1961), pp. 9–23.

[2] H. Attouch and H. Brezis, *Duality for the sum of convex functions in general Banach spaces*, in Aspects of Mathematics and its Applications, North-Holland Math. Library 34, North-Holland, Amsterdam, 1986, pp. 125–133.

[3] H. Attouch, H. Riahi, and M. Théra, *Somme ponctuelle d'operateurs maximaux monotones*, Serdica Math. J., 22 (1996), pp. 267–292.

[4] H. Attouch and M. Théra, *A general duality principle for the sum of two operators*, J. Convex Anal., 3 (1996), pp. 1–44.

[5] H. Attouch and M. Théra, *A duality proof of the Hille-Yosida theorem*, in Progress in Partial Differential Equations: The Metz surveys, 4, Pitman Res. Notes in Math. 345, Longman, Harlow, UK, 1996, pp. 18–35.

[6] J.-P. Aubin and I. Ekeland, *Applied Nonlinear Analysis*, John Wiley, New York, 1984.

[7] J.-P. Aubin and H. Frankowska, *Set-Valued Analysis*, Birkhäuser, Basel, 1990.

[8] H. Brezis and A. Haraux, *Image d'une somme d'operateurs monotones et applications*, Israel J. Math., 23 (1976), pp. 165–186.

[9] H. Brezis and F. Browder, *Linear maximal monotone operators and singular nonlinear integral equation of Hammerstein type*, in Nonlinear Analysis: A Collection of Papers in Honor of Erich H. Rothe, L. Cesari, R. Kannan, and H. Weinberger, eds., Academic Press, New York, 1978, pp. 31–42.

[10] R. S. Burachik, C. A. Sagastizábal, and B. F. Svaiter, *Bundle methods for maximal monotone operators*, in Ill-Posed Variational Problems and Regularization Techniques, M. Théra and R. Tichatschke, eds., Lecture Notes in Economics and Mathematical Systems 477, Springer-Verlag, Berlin, 1999, pp. 49–64.

[11] L.-J. Chu, *On the sum of monotone operators*, Michigan Math. J., 43 (1996), pp. 273–289.

[12] G. H.-G. Chen and R. T. Rockafellar, *Convergence rates in forward-backward splitting*, SIAM J. Optim., 7 (1997), pp. 421–444.

[13] R. Cross, *Multivalued Linear Operators*, Marcel Dekker, New York, 1998.

[14] J. Eckstein, *Splitting Methods for Monotone Operators with Applications to Parallel Optimization*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1989.

[15] J. Eckstein, *Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming*, Math. Oper. Res., 18 (1993), pp. 203–226.

[16] J. Eckstein and M. C. Ferris, *Smooth methods of multipliers for complementarity problems*, Math. Program., 86 (1999), pp. 65–90.

[17] D. Gabay, *Applications of the method of multipliers to variational inequalities*, in Augmented Lagrangian Methods: Applications to the Solution of Boundary-Value Problems, M. Fortin and R. Glowinski, eds., North-Holland, Amsterdam, 1983, pp. 299–331.

[18] D. Goeleven, G. E. Stavroulakis, and P. D. Panagiotopoulos, *Solvability theory for a class of hemivariational inequalities involving copositive plus matrices. Applications in robotics*, Math. Programming, 75 (1996), pp. 441–465.

[19] L. McLinden, *Minimax Problems, Saddle-Functions and Duality*, Ph.D. thesis, University of Washington, Seattle, WA, 1971.

[20] L. McLinden, *Decomposition principle for minimax problems*, in Decomposition of Large-Scale Problems, D. M. Himmelblau, ed., North-Holland, Amsterdam, 1973, pp. 427–435.

[21] L. McLinden, *An extension of Fenchel's duality theorem to saddle-functions and dual minimax-problems*, Pacific J. Math., 50 (1974), pp. 135–158.

[22] G. J. Minty, *On the maximal domain of a "monotone" function*, Michigan Math. J., 8 (1961), pp. 135–137.

[23] U. Mosco, *Dual variational inequalities*, J. Math. Anal. Appl., 40 (1972), pp. 202–206.

[24] T. Pennanen, *Graph convex mappings and K-convex functions*, J. Convex Anal., 6 (1999).

[25] T. Pennanen, *On the Range of Monotone Composite Mappings*, manuscript, 1998.

[26] T. Pennanen, *Multiplier Methods for Monotone Inclusions*, manuscript, 1998.

[27] S. M. Robinson, *Generalized equations and their solutions.* I. *Basic theory*, Math. Progr. Stud., 10 (1979), pp. 128–141.

[28] S. M. Robinson, *Composition duality and maximal monotonicity*, Math. Program., 85 (1999), pp. 1–13.

[29] R. T. Rockafellar, *A monotone convex analogue of linear algebra*, in Proceedings of the Colloquium on Convexity, W. Fenchel, ed., University of Copenhagen, Copenhagen, 1965, pp. 261–276.

[30] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[31] R. T. Rockafellar, *On the maximality of sums of nonlinear monotone operators*, Trans. Amer. Math. Soc., 149 (1970), pp. 75–88.

[32] R. T. Rockafellar, *Monotone operators associated with saddle-functions and minimax problems*, in Nonlinear Functional Analysis: Part 1, F. Browder, ed., Proc. Sympos. Pure Math. 18, Amer. Math. Soc., Providence, RI, 1970, pp. 241–250.

[33] R. T. Rockafellar, *On the maximal monotonicity of subdifferential mappings*, Pacific J. Math., 33 (1970), pp. 209–216.

[34] R. T. Rockafellar, *Conjugate Duality and Optimization*, SIAM, Philadelphia, PA, 1974.

[35] R. T. Rockafellar, *Augmented Lagrangians and applications of the proximal point algorithm in convex programming*, Math. Oper. Res., 1 (1976), pp. 97–116.

[36] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.

[37] R. T. Rockafellar, *Lagrange multipliers and variational inequalities*, in Variational Inequalities and Complementarity Problems, R. W. Cottle, F. Giannessi, and J.-L. Lions, eds., John Wiley, New York, 1980, pp. 303–322.

[38] R. T. Rockafellar, *Monotone relations and network equilibrium*, in Variational Inequalities and Network Equilibrium Problems, F. Giannessi and A. Maugeri, eds., Plenum Press, New York, 1995, pp. 271–288.

[39] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer-Verlag, New York, 1998.

[40] S. Simons, *Minimax and Monotonicity*, Lecture Notes in Math. 1693, Springer-Verlag, New York, 1998.

[41] M. V. Solodov and B. F. Svaiter, *A hybrid approximate extragradient–proximal point algorithm using the enlargement of a maximal monotone operator*, Set-Valued Anal., to appear.

[42] J. E. Spingarn, *Partial inverse of a monotone operator*, Appl. Math. Optim., 10 (1983), pp. 247–265.

[43] E. Zeidler, *Nonlinear Functional Analysis and Its Applications* II/*B, Nonlinear Monotone Operators*, Springer-Verlag, New York, 1990.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 836–851

GLOBAL ERROR BOUNDS FOR CONVEX CONIC PROBLEMS[*]

SHUZHONG ZHANG[†]

Abstract. This paper aims at deriving and proving some Lipschitzian-type error bounds for convex conic problems in a simple way. First, it is shown that if the recession directions satisfy Slater's condition, then a global Lipschitzian-type error bound holds. Alternatively, if the feasible region is bounded, then the ordinary Slater condition guarantees a global Lipschitzian-type error bound. These can be considered as generalizations of previously known results for inequality systems, which also follow from general results by Bauschke and Borwein in [*SIAM Review*, 38 (1996), pp. 367–426] and Bauschke, Borwein, and Li in a 1997 report. However, the proofs in the current paper are considerably simpler. Some of the results are generalized to the intersection of multiple shifted cones (with different shifts). Under Slater's condition alone, a global Lipschitzian-type error bound does not hold. It is shown, however, that such an error bound holds for a specific conic region. For linear systems we establish that the sharp constant involved in Hoffman's error bound is nothing but the condition number for linear programming as used by Vavasis and Ye in [*Math. Programming*, 74 (1996), pp. 79–120].

Key words. error bound, convex conic problems, condition number

AMS subject classifications. 52A20, 65G99, 90C31

PII. S105262349834429X

1. Introduction.

In optimization theory it is often desirable to measure the distance to the solution set from a certain given point. In general, this distance can be difficult to assess, since one may not have a complete knowledge about the solution set. However, if the form of the solution set is explicitly given, then in some cases it is possible to estimate the distance to the solution set by the so-called *constraint violation*, which is computable. This kind of estimation is termed *error bound relation*. The first such result was obtained by Hoffman [8] for systems of linear equalities and inequalities. We shall discuss Hoffman's error bound in this paper too. A recent extensive survey on various types of error bound results can be found in Pang [22].

Most papers discussing error bound results assume that the solution set is given by equations and inequalities, e.g.,

$$S = \{x \mid f_i(x) = 0 \text{ for } i = 1, \dots, m \text{ and } g_j(x) \le 0 \text{ for } j = 1, \dots, l\}.$$

For a given point x the amount of constraint violation can be measured by the quantity

$$v(x) = \|f(x)\| + \|(g(x))_+\|,$$

where $f(x) = (f_1(x), \dots, f_m(x))$ and $(g(x))_+ = ((g_1(x))_+, \dots, (g_l(x))_+)$ with the notation $(y)_+ = \max(y, 0)$.

A measure for constraint violation is similar to a penalty function in the sense that it takes a positive value for points outside the set and zero otherwise. Note that a measure for constraint violation should be easily computable, such as the case for

[*] Received by the editors September 8, 1998; accepted for publication (in revised form) October 4, 1999; published electronically May 2, 2000.

http://www.siam.org/journals/siopt/10-3/34429.html

[†] Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong (zhang@se.cuhk.edu.hk); on leave from Econometric Institute, Erasmus University, Rotterdam, The Netherlands.

the above defined function $v(x)$. Hoffman's lemma [8] states that if $S \neq \emptyset$ and f_i and g_j are all affine linear functions, then there is a positive constant $\kappa > 0$ such that

$$\operatorname{dist}(x, S) \leq \kappa v(x) \tag{1.1}$$

for all $x \in \Re^n$. This means that the distance to S is of the same magnitude as $v(x)$. Such a relation is known as a Lipschitzian-type error bound.

In the case that f_i and g_j are not affine linear, inequality (1.1) does not hold in general. Early results concerning nonlinear functions are due to Robinson [24] and Mangasarian [19]. Robinson [24] showed that for inequality systems if all functions are convex and differentiable, S is bounded, and the Slater condition holds, i.e., there is a $\hat{x}$ such that $g_j(\hat{x}) < 0$ for all j, then relation (1.1) holds. Mangasarian [19] removed the assumption that S is bounded by assuming an additional asymptotic constraint qualification condition, which, however, can be difficult to verify in general. For an extensive discussion of the related results we refer to [1] and [2] and the references therein.

In this paper we consider the convex conic set

$$\mathcal{F} = (b + \mathcal{L}) \cap \mathcal{K}, \tag{1.2}$$

where $b \in \Re^n$, $\mathcal{L}$ is a subspace of $\Re^n$, and $\mathcal{K} \subseteq \Re^n$ is a closed convex cone. Polynomial-time interior-point algorithms for solving convex optimization problems with a convex conic feasible set were introduced in a systematic manner by Nesterov and Nemirovskii [21]. It turns out that many important classes of optimization problems, such as linear programming and semidefinite programming, can be cast in this form. The focus of this paper is to discuss how an error bound–type relation can be established for such problems. Throughout this paper we make the following assumption.

Assumption 1. $\mathcal{F} \neq \emptyset$.

The organization of this paper is as follows. In the next section we prove that with a proper definition of constraint violation a Lipschitzian-type error bound (1.1) can be established for general convex conic problems, under various conditions on the relations between $\mathcal{L}$ and $\mathcal{K}$, including Slater-type conditions. In section 3 we discuss a link between the constant in Hoffman's error bound and the so-called condition number for linear programming. Finally, we conclude the paper in section 4.

We use the following notation in this paper. Matrices are denoted by capital letters, e.g., X. For a given index set I, X_I is composed of columns of X whose indices belong to I. We denote n-dimensional Euclidean space by $\Re^n$ and its nonnegative quadrant by $\Re^n_+$. The space of all symmetric n by n matrices is denoted by $\mathcal{S}^{n\times n}$ and the cone of all symmetric positive semidefinite n by n matrices by $\mathcal{S}^{n\times n}_+$. Vector e represents a vector of all ones with appropriate dimension. For a vector $v \in \Re^n$, we use the capitalized letter V to denote the diagonal matrix that takes v as its diagonal elements. We use the Euclidean norm $\|v\|$ for a vector v and the spectral norm $\|M\|$ for a matrix M. A vector $a \geq 0$ means that each component of a is nonnegative.

2. Convex conic systems.

Consider the convex conic set (1.2). For convenience we further assume that $\mathcal{K}$ is a closed, pointed, and solid cone, i.e., $\mathcal{K} \cap (-\mathcal{K}) = \{0\}$ and $\dim \mathcal{K} = n$.

The dual of $\mathcal{K}$ is

$$\mathcal{K}^* = \{x \mid x^T y \geq 0 \text{ for all } y \in \mathcal{K}\}.$$

Since $\mathcal{K}$ is pointed and solid, $\mathcal{K}^*$ too is a closed, convex, pointed, and solid cone.

An immediate next question is, How can we define a constraint violation function for $\mathcal{F}$? For this purpose we note the following lemma, due to Moreau (see Theorem 31.5 in [25]).

LEMMA 2.1. *For any $x \in \Re^n$ there is a unique $x_p \in \mathcal{K}$ and $x_d \in \mathcal{K}^*$ such that $x = x_p - x_d$ and $x_p^T x_d = 0$.*

In fact, x_p is simply the projection of x onto $\mathcal{K}$ and $\|x_d\|$ measures the distance from x to $\mathcal{K}$. A natural definition for the constraint violation for $\mathcal{F}$ is now in order.

DEFINITION 2.2. *For any $x \in \Re^n$ define*

$$v(x; \mathcal{F}) := \operatorname{dist}(x, b + \mathcal{L}) + \|x_d\|$$

as the constraint violation function for $\mathcal{F}$.

It is readily seen that $v(x, \mathcal{F}) = 0$ iff $x \in \mathcal{F}$.

It is, however, not immediately clear how the function $v(x; \mathcal{F})$ can be computed. Below we shall see some examples in which this function is explicitly derived. First we consider the case $\mathcal{K} = \Re^n_+$, the nonnegative quadrant of $\Re^n$. Clearly, $x = x_+ + x_-$, where $x_+ = ((x_1)_+, \ldots, (x_n)_+)$ and $x_- = (-(-x_1)_+, \ldots, -(-x_n)_+)$. Obviously, $x_+ \in \mathcal{K}$, $-x_- \in \mathcal{K}$, and $x_+^T x_- = 0$. Therefore, $\|x_d\| = \|(x)_-\|$, which is exactly the usual definition of the violation for nonnegativity constraints.

Another example is $\mathcal{K} = \mathcal{S}^{n \times n}_+$, the cone of n by n symmetric positive semidefinite matrices. Consider a given n by n symmetric matrix X. Following Lemma 2.1 we know that there are unique positive semidefinite matrices X_p and X_d such that $X = X_p - X_d$ and $\operatorname{tr}(X_p X_d = 0)$. Matrices X_p and X_d can be computed as follows. Let $X = Q\Lambda Q^T$ with Q as an orthonormal matrix and Λ as a diagonal matrix with eigenvalues of X as its components. Splitting $\Lambda = \Lambda_+ + \Lambda_-$, where Λ_+ and Λ_- denote the nonnegative and nonpositive parts of Λ, respectively, it follows that $X_p = Q\Lambda_+ Q^T$, $X_d = -Q\Lambda_- Q^T$, and $X_p X_d = 0$. Hence, $\|X_d\| = \|\Lambda_-\|$.

Finally, we consider another popular convex cone: the second order cone $\mathcal{K} \in \Re^{n+1}$ defined as

$$\mathcal{K} = \{(x_0, x) \mid x \in \Re^n \text{ and } x_0 \geq \|x\|\}.$$

It can be shown that in this case

$$\|x_d\| = (\|x\| - x_0)_+ / \sqrt{2}.$$

In general, Definition 2.2 is only related to the geometry of the object under consideration.

Consider now an arbitrary point $z \in \Re^n$. Assume that $z \notin \mathcal{F}$. The following problem yields a unique point in $\mathcal{F}$ with the shortest Euclidean distance to z:

$$\text{(Proj)} \qquad \begin{array}{ll} \text{minimize} & \frac{1}{2}\|x - z\|^2 \\ \text{subject to} & x \in b + \mathcal{L}, \\ & x \in \mathcal{K}. \end{array}$$

Let this optimal solution be $\bar{x}$. The Karush–Kuhn–Tucker (KKT) optimality condition for (Proj) is given as

$$\text{(2.1)} \qquad \text{(KKT)} \left\{ \begin{array}{l} \bar{x} - z + \lambda - \mu = 0, \\ \bar{x}^T \mu = 0, \\ \bar{x} \in (b + \mathcal{L}) \cap \mathcal{K}, \\ \mu \in \mathcal{K}^*, \\ \lambda \in \mathcal{L}^\perp. \end{array} \right.$$

Hence,

$$\begin{aligned}\|\bar{x}-z\|^2 &= (\bar{x}-z)^T(\bar{x}-z)\\ &= (\bar{x}-z)^T(\mu-\lambda)\\ &= -(z_p-z_d)^T\mu+(z-\bar{x})^T\lambda\\ &\le z_d^T\mu+(z-\bar{x})^T\lambda\\ &\le \|z_d\|\|\mu\|+(z-\bar{x})^T\lambda, \end{aligned}\tag{2.2}$$

where the first inequality follows from the fact that $z_p \in \mathcal{K}$ and $\mu \in \mathcal{K}^*$.

Let the projection of z onto the affine subspace $b+\mathcal{L}$ be z_l. Then

$$\begin{aligned}(z-\bar{x})^T\lambda &= (z-z_l+z_l-\bar{x})^T\lambda\\ &= (z-z_l)^T\lambda\\ &\le \|z-z_l\|\|\lambda\|\\ &= \operatorname{dist}(z, b+\mathcal{L})\|\lambda\|.\end{aligned}$$

Substituting this relation into (2.2) we obtain

$$(\operatorname{dist}(z,\mathcal{F}))^2 = \|\bar{x}-z\|^2 \le \|z_d\|\|\mu\| + \operatorname{dist}(z, b+\mathcal{L})\|\lambda\|.\tag{2.3}$$

In section 3 we shall discuss how to further bound the errors when $\mathcal{K}$ is a polyhedral cone, which is the situation when the original Hoffman lemma applies. In the rest of this section we assume that $\mathcal{K}$ is a general closed convex cone. In addition to this we assume that the Slater condition is satisfied.

Assumption 2. $(b+\mathcal{L})\cap \operatorname{int}\mathcal{K} \neq \emptyset$.

The following lemma is well known. For completeness we provide a short proof.

LEMMA 2.3. *Suppose that Assumption* 2 *holds. Then for any* $y \in \mathcal{L}^\perp \cap \mathcal{K}^*$ *with* $y \neq 0$ *it must follow that* $b^Ty > 0$.

Proof. Suppose, for the sake of deriving a contradiction, that there is $y \neq 0$ such that $y \in \mathcal{L}^\perp \cap \mathcal{K}^*$ and $b^Ty \le 0$.

Consider the hyperplane

$$H_y = \{x \mid y^Tx = 0\}.$$

For any $x \in b+\mathcal{L}$ we have $y^Tx = b^Ty \le 0$, while for any $x \in \mathcal{K}$, since $y \in \mathcal{K}^*$ we have $y^Tx \ge 0$. This means that H_y separates $b+\mathcal{L}$ and $\mathcal{K}$, yielding a contradiction to the fact that $b+\mathcal{L}$ intersects with the interior of $\mathcal{K}$. □

For fixed $\bar{x}$ we consider again the system (KKT) in terms of μ and λ. After some rearranging this yields

$$\begin{cases}\mu-\lambda = \bar{x}-z,\\ \bar{x}^T\mu = 0,\\ \mu \in \mathcal{K}^*,\\ \lambda \in \mathcal{L}^\perp.\end{cases}\tag{2.4}$$

Define

$$\bar{\mathcal{K}}^* = \mathcal{K}^* \cap \{\mu \mid \bar{x}^T\mu = 0\},$$

which is a closed convex cone as well; see Figure 2.1.

Note that $\bar{x} = 0$ is a trivial case and can only happen when $b = 0$.

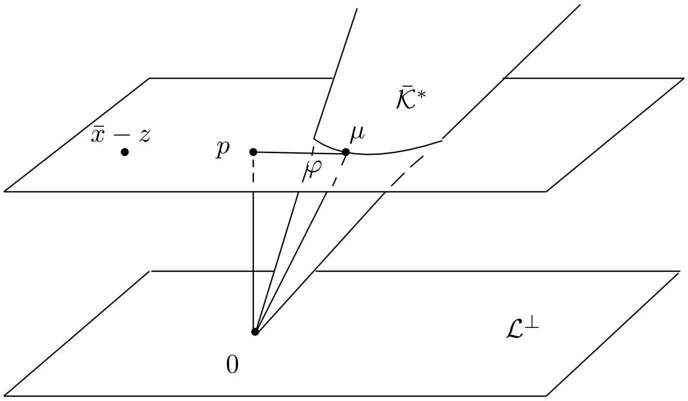


FIG. 2.1. *Subspace $\mathcal{L}^\perp$ and the cone $\bar{\mathcal{K}}^*$.*

We shall mention another easy case, i.e., if $\bar{x}$ lies in the interior of $\mathcal{K}$, then $\bar{\mathcal{K}} = \{0\}$. In this case $\mu = 0$ and $\lambda = z - \bar{x}$, and therefore

$$\text{dist}\,(z, \mathcal{F}) \leq \text{dist}\,(z, b + \mathcal{L})$$

due to (2.3). In what remains we shall concentrate only on the situation when $\bar{x} \notin \text{int}\,\mathcal{K}$.

Remark that for $\bar{x} \in \mathcal{K}$, the cone $\bar{\mathcal{K}}^*$ is known as a *face* of $\mathcal{K}^*$.

The condition (2.4) is equivalent to

$$\mu \in (\bar{x} - z + \mathcal{L}^\perp) \cap \bar{\mathcal{K}}^*.$$

The following fact is readily seen using Lemma 2.3.

LEMMA 2.4. *If Assumption* 2 *holds, then $\mathcal{L}^\perp \cap \bar{\mathcal{K}}^* = \{0\}$.*

Now we define the minimum angle between $\mathcal{L}^\perp$ and $\bar{\mathcal{K}}^*$ as

$$\angle(\mathcal{L}^\perp, \bar{\mathcal{K}}^*) := \min\{\arccos(u^T v/(\|u\|\|v\|)) \mid u \in \mathcal{L}^\perp \setminus \{0\},\, v \in \bar{\mathcal{K}}^* \setminus \{0\}\}.$$

Note that both $\mathcal{L}^\perp$ and $\bar{\mathcal{K}}^*$ are closed cones. According to Lemma 2.4, it follows that

$$\angle(\mathcal{L}^\perp, \bar{\mathcal{K}}^*) > 0$$

for any $\bar{x} \in (b + \mathcal{L}) \cap \mathcal{K}$.

In order to pursue our analysis further, one of the following two mutually exclusive cases will be considered.

Assumption 3. *The set $\mathcal{F} = (b + \mathcal{L}) \cap \mathcal{K}$ is compact.*

Assumption 4. $\mathcal{L} \cap \text{int}\,\mathcal{K} \neq \emptyset$.

Let us first consider the situation when Assumption 3 holds. In that case we know that there exists $\theta > 0$ such that for any $\bar{x} \in \mathcal{F}$ we always have

$$\angle(\mathcal{L}^\perp, \bar{\mathcal{K}}^*) \geq \theta > 0.$$

Now take $\mu \in (\bar{x} - z + \mathcal{L}^\perp) \cap \bar{\mathcal{K}}^*$. Let the projection of 0 onto $\bar{x} - z + \mathcal{L}^\perp$ be p.

Let the angle between μ and $\mu - p$ be φ. Clearly, $\theta \leq \varphi \leq \pi/2$. Moreover,

$$\|\mu\| = \|p\|/\sin\varphi \leq \|p\|/\sin\theta \leq \|\bar{x} - z\|/\sin\theta. \tag{2.5}$$

Denote

$$\kappa = 1 + 1/\sin\theta.$$

We are now in a position to prove the following error bound result.

THEOREM 2.5. *If Assumption* 2 *and Assumption* 3 *hold, then*

$$\text{dist}\,(z, \mathcal{F}) \le \kappa v(z; \mathcal{F})$$

for all $z \in \Re^n$.

Proof. By (2.5) we have

$$\|\mu\| \le \text{dist}\,(z, \mathcal{F})/\sin\theta \le \kappa\,\text{dist}\,(z, \mathcal{F}).$$

Using the first equation in (2.4) we also have

$$\|\lambda\| \le \|\mu\| + \|\bar{x} - z\| \le (1 + 1/\sin\theta)\text{dist}\,(z, \mathcal{F}) = \kappa\,\text{dist}\,(z, \mathcal{F}).$$

Recall relation (2.3). By the above estimations on $\|\mu\|$ and $\|\lambda\|$, it follows from (2.3) that

$$(\text{dist}\,(z, \mathcal{F}))^2 \le \kappa\,\text{dist}\,(z, \mathcal{F})(\|z_d\| + \text{dist}\,(z, b + \mathcal{L}))$$

and consequently

$$\text{dist}\,(z, \mathcal{F}) \le \kappa v(z; \mathcal{F}). \qquad \square$$

In the other situation, namely if Assumption 4 holds, then a similar result can be shown.

THEOREM 2.6. *If Assumption* 4 *holds, then for any* $b \in \Re^n$ *we must have* $(b + \mathcal{L}) \cap \text{int}\,\mathcal{K} \neq \emptyset$. *Moreover, there is a constant* $\kappa > 0$, *independent of* b, *such that*

$$\text{dist}\,(z, \mathcal{F}) \le \kappa v(z; \mathcal{F}),$$

where $\mathcal{F} = (b + \mathcal{L}) \cap \mathcal{K}$.

Proof. First we show that $(b + \mathcal{L}) \cap \text{int}\,\mathcal{K} \neq \emptyset$ for all b. Suppose otherwise that there is b with

$$(b + \mathcal{L}) \cap \text{int}\,\mathcal{K} = \emptyset.$$

Then, there will be a hyperplane separating $b + \mathcal{L}$ and $\mathcal{K}$, say with $0 \neq y \in \Re^n$ and $c \in \Re$ such that

$$y^T(b + x) \le c \quad \text{for all } x \in \mathcal{L},$$

$$y^T x \ge c \quad \text{for all } x \in \mathcal{K}.$$

Since $\mathcal{K}$ is a closed cone, the above separation implies that $y^T x \ge 0$ for all $x \in \mathcal{K}$ and $c = 0$. Moreover, we also have $y^T x = 0$ for all $x \in \mathcal{L}$. This is in contradiction with the condition $\mathcal{L} \cap \text{int}\,\mathcal{K} \neq \emptyset$.

Compared with Lemma 2.4, we have now a stronger relation: $\mathcal{L}^\perp \cap \mathcal{K}^* = \{0\}$. This means that the proof of Theorem 2.5 can remain exactly the same, except that now

$\theta > 0$ can be taken as the minimum angle between $\mathcal{L}^{\perp}$ and $\mathcal{K}^*$, which is independent of b. □

We remark that both Theorem 2.5 and Theorem 2.6 easily extend to the case when $\mathcal{L}$ is a closed cone.

Theorem 2.7. *Suppose that $\mathcal{K}_1$ is a closed convex cone and $\mathcal{K}_2$ is a closed, convex, solid, and pointed cone. Furthermore, suppose that $(b + \mathcal{K}_1) \cap \operatorname{int} \mathcal{K}_2 \neq \emptyset$ and $(b + \mathcal{K}_1) \cap \mathcal{K}_2$ is compact. Then there is a constant $\kappa > 0$ such that*

$$\operatorname{dist}(z, \mathcal{F}) \leq \kappa(\operatorname{dist}(z, b + \mathcal{K}_1) + \operatorname{dist}(z, \mathcal{K}_2))$$

for all $z \in \Re^n$, where $\mathcal{F} = (b + \mathcal{K}_1) \cap \mathcal{K}_2$.

Proof. We follow similar lines as in the proof of Theorem 2.5. Consider

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|x - z\|^2 \\ \text{subject to} & x \in b + \mathcal{K}_1, \\ & x \in \mathcal{K}_2. \end{array}$$

Let the optimal solution be $\bar{x}$. Since the Slater regularity condition is assumed, the KKT optimality condition holds at optimality, yielding

$$\left\{ \begin{array}{l} \bar{x} - z - \mu_1 - \mu_2 = 0, \\ (\bar{x} - b)^T \mu_1 = 0, \\ \bar{x}^T \mu_2 = 0, \\ \bar{x} \in (b + \mathcal{K}_1) \cap \mathcal{K}_2, \\ \mu_1 \in \mathcal{K}_1^*, \\ \mu_2 \in \mathcal{K}_2^*. \end{array} \right.$$

Let

$$\bar{\mathcal{K}}_1^* = \mathcal{K}_1^* \cap \{\mu \mid (\bar{x} - b)^T \mu = 0\}$$

and

$$\bar{\mathcal{K}}_2^* = \mathcal{K}_2^* \cap \{\mu \mid \bar{x}^T \mu = 0\}.$$

Both $\bar{\mathcal{K}}_1^*$ and $\bar{\mathcal{K}}_2^*$ are closed convex cones.

Now we claim that

$$(-\bar{\mathcal{K}}_1^*) \cap \bar{\mathcal{K}}_2^* = \{0\}. \tag{2.6}$$

Suppose such is not the case. Then, one should be able to find $\mu \neq 0$ satisfying

$$\left\{ \begin{array}{l} \mu \in (-\mathcal{K}_1^*) \cap \mathcal{K}_2^*, \\ (\bar{x} - b)^T \mu = 0, \\ \bar{x}^T \mu = 0. \end{array} \right.$$

Hence, $b^T \mu = 0$. Therefore, $\mu^T(b + x) \leq 0$ for all $x \in \mathcal{K}_1$ and $\mu^T x \geq 0$ for all $x \in \mathcal{K}_2$. This implies that $\{x \mid \mu^T x = 0\}$ separates $b + \mathcal{K}_1$ from $\mathcal{K}_2$, contradicting the Slater condition.

Since $-\bar{\mathcal{K}}_1^*$ and $\bar{\mathcal{K}}_2^*$ are closed convex cones and since, moreover, $\bar{\mathcal{K}}_2^*$ is a solid pointed cone, we derive from (2.6) that $\bar{\mathcal{K}}_2^*$ can be strictly separated from $-\bar{\mathcal{K}}_1^*$. Due to compactness of $\mathcal{F}$ we may let θ be a positive lower bound on the minimum angle between this separating hyperplane and $\bar{\mathcal{K}}_2^*$. Then we have

$$\|\mu_2\| \leq \|\bar{x} - z\| / \sin\theta$$

and consequently

$$\|\mu_1\| \le (1 + 1/\sin\theta)\|\bar{x} - z\|.$$

Now,

$$\begin{aligned}\|\bar{x} - z\|^2 &= (\bar{x} - z)^T(\mu_1 + \mu_2)\\ &= (b - z)^T\mu_1 - z^T\mu_2\\ &\le \operatorname{dist}(z, b + \mathcal{K}_1)\|\mu_1\| + \operatorname{dist}(z, \mathcal{K}_2)\|\mu_2\|\\ &\le (1 + 1/\sin\theta)(\operatorname{dist}(z, b + \mathcal{K}_1) + \operatorname{dist}(z, \mathcal{K}_2))\|\bar{x} - z\|.\end{aligned}$$

The desired result thus follows. □

Similarly, we have the following result, the proof of which is pretty much the same as that of Theorem 2.6 and Theorem 2.7 and is omitted here.

THEOREM 2.8. *Suppose that $\mathcal{K}_1$ is a closed convex cone and $\mathcal{K}_2$ is a closed, convex, solid, and pointed cone. Furthermore, suppose that $\mathcal{K}_1 \cap \operatorname{int} \mathcal{K}_2 \neq \emptyset$. Then for any $b \in \Re^n$ there is a constant $\kappa > 0$, independent of b, such that*

$$\operatorname{dist}(z, \mathcal{F}) \le \kappa(\operatorname{dist}(z, b + \mathcal{K}_1) + \operatorname{dist}(z, \mathcal{K}_2))$$

for all $z \in \Re^n$, where $\mathcal{F} = (b + \mathcal{K}_1) \cap \mathcal{K}_2$.

When more than two cones are concerned, a similar result holds under Slater's condition. First we note the following lemma; see, e.g., [17].

LEMMA 2.9. *Let $\mathcal{K}$ be a convex cone and* int $\mathcal{K} \neq \emptyset$. *Then, $x \in$* int $\mathcal{K}$ *if and only if for any $0 \neq \mu \in \mathcal{K}^*$ it holds that $\angle(x, \mu) \ge \theta > 0$.*

THEOREM 2.10. *Let $\mathcal{K}_i$ be closed convex cones, $i = 1, \ldots, m$. Suppose that*

$$\bigcap_{i=1}^{m} \operatorname{int} \mathcal{K}_i \neq \emptyset.$$

Then there is $\kappa > 0$ such that

$$\operatorname{dist}\left(z, \bigcap_{i=1}^{m}(b_i + \mathcal{K}_i)\right) \le \kappa \sum_{i=1}^{m} \operatorname{dist}(z, b_i + \mathcal{K}_i)$$

for any $z \in \Re^n$.

Proof. First note that if $\bigcap_{i=1}^m \operatorname{int} \mathcal{K}_i \neq \emptyset$, then necessarily $\bigcap_{i=1}^m (b_i + \mathcal{K}_i) \neq \emptyset$.

Consider

$$\begin{array}{ll}\text{minimize} & \frac{1}{2}\|x - z\|^2\\ \text{subject to} & x \in b_i + \mathcal{K}_i,\ i = 1, \ldots, m.\end{array}$$

For the optimal solution $\bar{x}$, the KKT condition yields

$$\bar{x} - z = \sum_{i=1}^{m} \mu_i$$

with $\mu_i \in \mathcal{K}_i^*$ and $(\bar{x} - b_i)^T\mu_i = 0$ for $i = 1, \ldots, m$.

Let

$$d \in \bigcap_{i=1}^{m} \operatorname{int} \mathcal{K}_i.$$

By Lemma 2.9 there exists $g_i > 0$ satisfying

$$d^T \mu_i \geq g_i \|\mu_i\|$$

for $i = 1, \ldots, m$.

Let

$$z - b_i = z_{ip} - z_{id}$$

with $z_{ip} \in \mathcal{K}_i$, $z_{id} \in \mathcal{K}_i^*$, and $z_{ip}^T z_{id} = 0$ due to Lemma 2.1. Moreover, $\|z_{id}\| = \operatorname{dist}(z, b_i + \mathcal{K}_i)$, $i = 1, \ldots, m$.

Therefore,

$$\begin{aligned} \|\bar{x} - z\|^2 &= (\bar{x} - z)^T \sum_{i=1}^m \mu_i \\ &= \sum_{i=1}^m (b_i - z)^T \mu_i \\ &= -\sum_{i=1}^m (z_{ip} - z_{id})^T \mu_i \\ &\leq \sum_{i=1}^m z_{id}^T \mu_i \\ &\leq \sum_{i=1}^m \|z_{id}\| \|\mu_i\|. \end{aligned}$$

On the other hand, since

$$\|d\| \|\bar{x} - z\| \geq d^T (\bar{x} - z) = \sum_{i=1}^m d^T \mu_i \geq g_i \|\mu_i\|$$

for $i = 1, \ldots, m$, it follows that

$$\|\bar{x} - z\|^2 \leq \sum_{i=1}^m \operatorname{dist}(z, \mathcal{K}_i)(\|d\|/g_i)\|\bar{x} - z\|,$$

and so by letting

$$\kappa = \max_{i=1,\ldots,m} \|d\|/g_i$$

it follows that

$$\operatorname{dist}\left(z, \bigcap_{i=1}^m (b_i + \mathcal{K}_i)\right) \leq \kappa \sum_{i=1}^m \operatorname{dist}(z, b_i + \mathcal{K}_i). \qquad \square$$

Theorem 2.5 can be viewed as an analogue to Robinson's result for convex inequality systems. In the form of convex inequality systems, Theorem 2.6 can be found in Hu and Wang [11] and Deng and Hu [5]. In particular, Deng and Hu [5] investigated the case when $\mathcal{K}$ is the cone of positive semidefinite matrices. This case is known as *linear matrix inequalities* (LMIs). In its optimization version it is also called *semidefinite programming* and has received intensive research attention recently. Sturm [27] mainly investigated error bounds for LMIs in the absence of Slater's condition. In fact, in the context of LMIs, both Theorem 2.5 and Theorem 2.6 also follow from the analysis in [27]. Moreover, an example was given in Sturm [27] showing that Assumption 2 alone cannot guarantee a global Lipschitzian-type error bound even for LMIs. Such an error bound is only possible when an additional scaling factor is present. Error bounds (with a scaling factor) for general conic problems under Assumption 2 can be found in Renegar [23], where the notion of *distance to ill-posedness* is used in estimating the Lipschitz constant.

Theorem 2.5 can also be derived from a result established by Bauschke, Borwein, and Li [2, section 5], though the two approaches are quite different in motivation. In case all shifting vectors b_is are identical, Theorem 2.10 can be found in [1], in which many other interesting error bound–type results are discussed. See also [9] and [12] for related results.

Below we shall discuss how to derive some conditioned error bound relations for the convex conic problem (1.2) under Assumption 2, without assuming Assumption 3 and Assumption 4.

In this situation the recession direction set $\mathcal{L} \cap \mathcal{K} \setminus \{0\}$ must be nonempty and is not contained in the interior of $\mathcal{K}$.

In a similar spirit as the angle between two cones, we define the angle between a vector and a closed convex cone as the minimum angle between the vector and *any* nontrivial direction in the cone. We make a convention here that the angle between the zero vector and a cone is π. Now, for a fixed positive angle $0 < \theta < \pi/2$, consider the cone

$$\mathcal{C}_\theta = \{z \mid \text{ the projection of } z \text{ onto } \mathcal{L} \text{ and the cone } \mathcal{L} \cap \mathcal{K} \text{ has an angle at least } \pi/2 + \theta\}.$$

Roughly speaking, $\mathcal{C}_\theta$ contains the points that do not lean too much toward the recession cone $\mathcal{L} \cap \mathcal{K}$.

THEOREM 2.11. *Suppose that Assumption* 2 *holds. Then, for any* $0 < \theta < \pi/2$, *there exists a constant* $\kappa_\theta > 0$ *such that*

$$\operatorname{dist}(z, \mathcal{F}) \leq \kappa_\theta v(z; \mathcal{F})$$

for all $z \in \mathcal{C}_\theta$.

Proof. Observe that if $\bar{x}$ is the projection of z on $\mathcal{F}$, then it must also be the projection of $z + y$ on $\mathcal{F}$ for any $y \in \mathcal{L}^\perp$. This can be seen as follows. The fact that $\bar{x} \in (b + \mathcal{L}) \cap \mathcal{K}$ is the projection of z is equivalent to the existence of $\lambda \in \mathcal{L}^\perp$ and $\mu \in \mathcal{K}^*$ such that

$$\bar{x} - z = \mu - \lambda \quad \text{and} \quad \bar{x}^T \mu = 0.$$

(See also (2.1).) Now if z is changed to $z + y$, then we need only to change λ to $\lambda + y \in \mathcal{L}^\perp$ to satisfy the same set of KKT conditions.

Remark also that to prove the theorem it is sufficient to show that, for any $z \in \mathcal{C}_\theta$, its projection onto $\mathcal{F}$ is contained in a compact set.

Suppose that the theorem is false and that there is a sequence $\{z^{(k)} \in \mathcal{C}_\theta \mid k = 1, 2, \ldots\}$ such that the corresponding projection on $\mathcal{F}$, $\{\bar{x}^{(k)} \in \mathcal{F} \mid k = 1, 2, \ldots\}$, is unbounded. Due to the above remarks, we need only to consider the projection of $z^{(k)}$ onto the subspace $\mathcal{L}$. Without loss of generality, assume that $z^{(k)} \in \mathcal{L} \cap \mathcal{C}_\theta$ for all k.

For sufficiently large k we have

$$\begin{aligned}\|z^{(k)}-\bar{x}^{(k)}\|^2 &= \|z^{(k)}\|^2+\|\bar{x}^{(k)}\|^2-2\langle z^{(k)},\bar{x}^{(k)}\rangle\\ &\geq \|z^{(k)}\|^2+\|\bar{x}^{(k)}\|^2+\cos(\theta/2)\|z^{(k)}\|\|\bar{x}^{(k)}\|\\ &> \|z^{(k)}-\bar{x}^{(1)}\|^2,\end{aligned}$$

where the first inequality is because $\bar{x}^{(k)}/\|\bar{x}^{(k)}\|$ must be pointing asymptotically toward the cone of recession directions $\mathcal{L}\cap\mathcal{K}$ and the last inequality is due to the fact that $\|\bar{x}^{(k)}\|\to\infty$. This contradicts $\bar{x}^{(k)}$ being the closest point in $\mathcal{F}$ to $z^{(k)}$. □

For any given point $z\in\Re^n$, we may decompose $z=z_1+z_2$ with $z_1\in\mathcal{L}\cap\mathcal{K}$ and $z_2\in\mathcal{C}_\theta$. Let the projection of z_2 onto $\mathcal{F}$ be $\bar{x}_2$. Then

$$\text{dist}\,(z,\mathcal{F})\leq\|z_2+z_1-(\bar{x}_2+z_1)\|=\text{dist}\,(z_2,\mathcal{F}),\tag{2.7}$$

where we used the fact that $z_1\in\mathcal{L}\cap\mathcal{K}$ and so $\bar{x}_2+z_1\in\mathcal{F}$.

Combining (2.7) with Theorem 2.11 we have the following theorem.

THEOREM 2.12. *Suppose that Assumption 2 holds. Then*

$$\text{dist}\,(z,\mathcal{F})\leq\kappa_\theta v(z_2;\mathcal{F})$$

for all $z\in\Re^n$ with $z=z_1+z_2$, $z_1\in\mathcal{L}\cap\mathcal{K}$ and $z_2\in\mathcal{C}_\theta$.

3. Hoffman's error bound and the condition number for LP.

In this section we shall discuss error bounds for the linear system $\{x\mid A^Tx\leq b\}$ with $A\in\Re^{m\times n}$ and rank $(A)=m$. This is the setting for which Hoffman's error bound result applies [8]. Our purpose is to see how the constant in Hoffman's bound is related to other known quantities for the linear system. Previous results on the constant of Hoffman's bound can be found, e.g., in [18, 20, 3, 15, 13, 14, 7]. Extensions of Hoffman's result in infinite dimensional space can be found in [4, 16].

By introducing a slack $s(x)=b-A^Tx$ we confine ourselves to the range space of A^T, i.e.,

$$\mathcal{L}=\{s\mid\exists x\in\Re^m:\ s=A^Tx\}.$$

Accordingly, $\mathcal{K}=\Re^n_+$.

For a given $z\in\Re^n$ with $s(z)\notin\Re^n_+$, let $\bar{x}$ be such that $s(\bar{x})=b-A^T\bar{x}\in(b+\mathcal{L})\cap\mathcal{K}$ and that the distance between $\bar{x}$ and z is minimal.

Let

$$K=\{i\mid s(\bar{x})_i>0\}\quad\text{and}\quad J=\{1,\ldots,n\}\setminus K.$$

Then for this given $s(\bar{x})\geq0$ we can rewrite (2.1) as

$$\begin{cases}A_J\mu_J=z-\bar{x},\\ \mu_K=0,\\ \mu_J\geq0.\end{cases}\tag{3.1}$$

As (3.1) is a necessary condition for optimality, it is certain that (3.1) is feasible. What remains to be analyzed is the size of the solution. A key ingredient in our analysis is the following lemma.

Lemma 3.1. *Suppose that A has full row rank. Then*

$$\chi(A) := \sup\{\|DA^T(ADA^T)^{-1}\| \mid D \textit{ diagonal and } D \succ 0\} < \infty.$$

Lemma 3.1 was first shown by Dikin [6] and was used in his convergence analysis for affine scaling methods. Among others, Stewart [26] and Todd [29] rediscovered this result later.

The meaning of Lemma 3.1 can be interpreted as follows. It is well known that $\mathrm{Null}(A) = \{x \mid Ax = 0\}$ and $\mathrm{Range}(A^T) = \{x \mid \exists y \in \Re^m,\ x = A^Ty\}$ are orthocomplements to each other. Obviously, for a given positive diagonal matrix D, $\mathrm{Null}(A)$ can only intersect with $D\,\mathrm{Range}(A^T)$ at the origin; hence, there must be a positive angle between them. Lemma 3.1 further states that the minimum angle between $\mathrm{Null}(A)$ and $D\,\mathrm{Range}(A^T)$ is uniformly bounded from below by a positive constant that is independent of D.

To understand this fact we may consider the following example. Let $A = [1, 1]$. Then $\mathrm{Null}(A)$ is simply the line $x_1 + x_2 = 0$. For a given positive diagonal matrix D, $D\,\mathrm{Range}(A^T)$ is contained in the first and the third quadrants. The angle between these two subspaces never drops below $\pi/4$. Another closely related quantity is

$$\bar{\chi}(A) := \sup\{\|DA^T(ADA^T)^{-1}A\| \mid D \text{ diagonal and } D \succ 0\}.$$

An important property of the constant $\bar{\chi}(A)$ is that it reflects an intrinsic, geometric relationship of the spaces. Vavasis and Ye [31] used $\bar{\chi}(A)$ and $\chi(A)$ as a measure of complexity for solving the related linear programming problem. Their results showed that, in a real-number computation model, the linear program is solvable in polynomial time, in terms of the total number of basic operations, with respect to the dimension n and the complexity measure $\log \bar{\chi}(A)$. For problems with integral input data, this result yields the usual polynomiality complexity result for linear programs in terms of the input length.

Holder, Sturm, and Zhang [10] showed that $\chi(A)$ and $\bar{\chi}(A)$ play an important role in sensitivity analysis for linear programming. Furthermore, Sturm and Zhang [28] extended some of the results in [10] to semidefinite programming. It is known, however, that Lemma 3.1 cannot extend to general semidefinite programming for arbitrary invariant scaling of the cone $\mathcal{S}_+^{n\times n}$; see [28].

Fortunately, in analyzing (3.1) we need only to deal with a polyhedral cone. To see how the condition number $\chi(A)$ ($\bar{\chi}(A)$) is relevant in error bound analysis, we need to introduce a number of technical lemmas.

First we note the following equivalent definition of $\chi(A)$ for arbitrary matrix A due to Vavasis and Ye [31].

Lemma 3.2. *It holds that*

$$\chi(A) = \sup\left\{\frac{\|y\|}{\|c\|} \mid y \textit{ minimizes } \|D^{1/2}(A^Ty - c)\| \right.$$
$$\left. \textit{for } 0 \neq c \in \Re^n \textit{ and } D \textit{ positive diagonal}\right\}.$$

For our analysis it is important to know the size of a solution for a linear system. To this end, we note the following two lemmas. Renegar [23] studied similar problems in a quite general framework using a different quantity known as the *distance to ill-posedness.*

LEMMA 3.3. *Suppose that A has full row rank. Further assume that $\{x \mid Ax = b,\ x > 0\} \neq \emptyset$. Then there is a solution $\bar{x}$ in $\mathcal{F} = \{x \mid Ax = b,\ x \geq 0\}$ such that*

$$\|\bar{x}\| \leq \chi(A)\|b\|.$$

Proof. Consider a linear program

$$\text{(P)} \qquad \begin{array}{ll} \text{minimize} & e^T x \\ \text{subject to} & Ax = b, \\ & x \geq 0, \end{array}$$

and its dual

$$\text{(D)} \qquad \begin{array}{ll} \text{maximize} & b^T y \\ \text{subject to} & A^T y + s = e, \\ & s \geq 0. \end{array}$$

Both (P) and (D) satisfy Slater's condition. Therefore their respective analytic central paths $\{x(\mu) \mid \mu > 0\}$ and $\{(y(\mu), s(\mu)) \mid \mu > 0\}$ exist, satisfying the relation

$$\text{(3.2)} \qquad \left\{ \begin{array}{l} Ax(\mu) = b, \\ A^T y(\mu) + s(\mu) = e, \\ x(\mu)s(\mu) = \mu e. \end{array} \right.$$

Multiplying the second equation in (3.2) with $X(\mu)$, multiplying the diagonal matrix with $x(\mu)$ as its diagonal components, and applying the first equation in (3.2) we obtain

$$y(\mu) = (AX(\mu)A^T)^{-1}b - \mu(AX(\mu)A^T)^{-1}e.$$

Substituting this into the second equation and finally using the third relation in (3.2) we have

$$x(\mu) = X(\mu)A^T(AX(\mu)A^T)^{-1}b + \mu e - \mu X(\mu)A^T(AX(\mu)A^T)^{-1}Ae.$$

Now we can apply Lemma 3.1 to obtain

$$\|x(0)\| = \|\lim_{\mu \to 0} x(\mu)\| \leq \chi(A)\|b\|.$$

The lemma is proven. □

Next we shall extend this result to the case when Slater's condition is no longer assumed.

LEMMA 3.4. *Suppose that A has full row rank. Further assume that $\{x \mid Ax = b,\ x \geq 0\} \neq \emptyset$. Then there is a solution $\bar{x}$ in $\mathcal{F} = \{x \mid Ax = b,\ x \geq 0\}$ such that*

$$\|\bar{x}\| \leq \chi(A)\|b\|.$$

Proof. Let $\delta > 0$. Consider a perturbed set

$$\mathcal{F}_\delta = \{x \mid Ax = b + \delta Ae,\ x \geq 0\}.$$

Clearly, $\mathcal{F}_\delta$ contains an interior point; therefore, Lemma 3.3 can be invoked. Let $x^\delta \in \mathcal{F}_\delta$ and

$$\|x^\delta\| \leq \chi(A)\|b + \delta Ae\|.$$

The set $\{x^\delta \mid 0 < \delta < 1\}$ is bounded. Let x^0 be a cluster point of x^δ as $\delta \to 0$. Obviously, $x^0 \in \mathcal{F}$ and

$$\|x^0\| \leq \chi(A)\|b\|. \qquad \square$$

Next we consider the size of a solution restricted to a face.

LEMMA 3.5. *Let I be an index set. Suppose that $\mathcal{F}_I = \{x_I \mid A_I x_I = b,\ x_I \geq 0\} \neq \emptyset$. Then there is a solution $\bar{x}_I \in \mathcal{F}_I$ such that*

$$\|\bar{x}_I\| \leq \chi(A)\|b\|.$$

Proof. Let $\bar{I}$ be the complement index set of I. Consider the linear program

$$\begin{array}{ll} \text{minimize} & e_{\bar{I}}^T x_{\bar{I}} \\ \text{subject to} & Ax = b, \\ & x \geq 0. \end{array}$$

This linear program is feasible and has an optimal value equal to zero. By considering perturbations on the primal and dual sides at the same time if necessary, the lemma follows from similar arguments as in the proofs of Lemma 3.3 and Lemma 3.4. □

Applying Lemma 3.5 to (3.1) we conclude that there is a multiplier μ such that

$$\|\mu\| = \|\mu_J\| \leq \chi(A)\|\bar{x} - z\|. \tag{3.3}$$

Finally we shall give an explicit constant in Hoffman's error bound for linear systems.

THEOREM 3.6. *Suppose that $\mathcal{F} = \{x \mid A^T x \leq b\} \neq \emptyset$ and A has full row rank. It holds that*

$$\operatorname{dist}(z, \mathcal{F}) \leq \chi(A)\|(A^T z - b)_+\|$$

for any $z \in \Re^n$.

Proof. Using (3.1) and (3.3) we have

$$\begin{aligned} \|\bar{x} - z\|^2 &= (\bar{x} - z)^T(-A\mu) \\ &= (A^T z - b)^T \mu \\ &\leq (A^T z - b)_+^T \mu \\ &\leq \|(A^T z - b)_+\|\|\mu\| \\ &\leq \|(A^T z - b)_+\|\chi(A)\|\bar{x} - z\|. \end{aligned}$$

Hence

$$\operatorname{dist}(z, \mathcal{F}) \leq \chi(A)\|(A^T z - b)_+\|. \qquad \square$$

It is interesting to note that the Lipschitz constant established in Theorem 3.6 coincides with the sharp Lipschitz constant of Li [15]. In the absence of equality constraints, which is the case considered here, the sharp Lipschitz constant can be shown to be equal to

$$\lambda(A) = \{\|A_I^{-1}\| \mid |I| = m \text{ and } A_I \text{ nonsingular}\}.$$

(See Klatte and Thiere [13, 14] for a discussion on various Lipschitz constants.)

The proof of Theorem 1 in Todd, Tunçel, and Ye [30] can be used to show that $\chi(A)$ is in fact identical to $\lambda(A)$ (in [30] a bound on $\bar{\chi}(A)$ was considered). For completeness we provide this argument below.

PROPOSITION 3.7. *It holds that*

$$\chi(A) = \lambda(A).$$

Proof. For any I with $|I| = m$ and A_I nonsingular, we let D^ϵ be diagonal and $D^\epsilon_{ii} = 1$ for $i \in I$ and $D^\epsilon_{ii} = \epsilon$ for $i \notin I$. Clearly, $D^\epsilon A^T(AD^\epsilon A^T)^{-1} \to A_I^{-1}$ as $\epsilon \downarrow 0$ and so $\lambda(A) \le \chi(A)$.

To show $\chi(A) \le \lambda(A)$, we apply the equivalent definition of $\chi(A)$ as stipulated in Lemma 3.2. First we choose a fixed $0 \ne c \in \Re^n$ and a fixed positive diagonal matrix D. Consider the unique $y(c, D)$ that minimizes $\|D^{1/2}(A^T y - c)\|$. Obviously the rank of the active constraints at $y(c, D)$ must be equal to m. Let J be such that $|J| = m$, A_J nonsingular, and $A_J^T y(c, D) = c_J$. Hence, $y(c, D) = A_J^{-T} c_J$. This shows that

$$\begin{aligned} \chi(A) &\le \sup\{\|A_J^{-T} c_J\|/\|c\| \mid 0 \ne c \in \Re^n,\ |J| = m, \text{ and } A_J \text{ nonsingular}\} \\ &\le \sup\{\|A_J^{-T}\| \mid\ |J| = m \text{ and } A_J \text{ nonsingular}\} \\ &= \lambda(A). \end{aligned}$$

Combining the two inequalities the proposition follows. □

Theorem 3.6 therefore makes a connection between two previously known quantities for linear systems.

4. Conclusions.

In this paper we discuss error bounds for sets in convex conic form. The notion of constraint violation is extended to this class of problems. For a number of applications the measure of constraint violation is easily computable. We show that under Slater's condition and, additionally, if either the feasible set is bounded or the recession directions satisfy the Slater's condition, then there is a global Lipschitzian-type error bound for general convex conic problems. These results are generalized to the intersection of multiple shifted (noncopointed) convex cones. Under Slater's condition alone it is impossible to have a global Lipschitz error bound. In this case, however, one may still identify a conic region in which a Lipschitzian-type error bound holds. Finally, we discuss the bounds in Hoffman's lemma for linear systems. It is shown that such a bound is nothing but the condition number for linear programming as used in Vavasis and Ye [31].

Acknowledgments. I would like to thank the referees for their valuable comments. One of them gave very detailed comments and pointed out several related papers to me, for which I am especially grateful. I would also like to thank Jos Sturm for pointing out an error in an earlier version of the paper.

REFERENCES

[1] H.H. BAUSCHKE AND J.M. BORWEIN, *On projection algorithms for solving convex feasibility problems*, SIAM Review, 38 (1996), pp. 367–426.

[2] H.H. BAUSCHKE, J.M. BORWEIN, AND W. LI, *Strong conical hull intersection property, bounded linear regularity, Jameson's property (G), and error bounds in convex optimization*, Math. Program., 86 (1999), pp. 135–160.

[3] C. BERGTHALLER AND I. SINGER, *The distance to a polyhedron*, Linear Algebra Appl., 169 (1992), pp. 111–129.

[4] J.V. BURKE AND P. TSENG, *A unified analysis of Hoffman's bound via Fenchel duality*, SIAM J. Optim., 6 (1996), pp. 265–282.

[5] S. Deng and H. Hu, *Computable error bounds for semidefinite programming*, J. Global Optim., 14 (1999), pp. 105–115.

[6] I.I. Dikin, *Iterative solution of problems of linear and quadratic programming*, Soviet Math. Dokl., 8 (1967), pp. 674–675.

[7] O. Güler, A.J. Hoffman, and U.G. Rothblum, *Approximations to solutions to systems of linear inequalities*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 688–696.

[8] A.J. Hoffman, *On approximate solutions of systems of linear inequalities*, J. Research of the National Bureau of Standards, 49 (1952), pp. 263–265.

[9] A.J. Hoffman, *The distance to the intersection of two convex sets by the distances to each of them*, Math. Nachr., 157 (1992), pp. 81–98.

[10] A. Holder, J.F. Sturm, and S. Zhang, *Analytic Central Path, Sensitivity Analysis and Parametric Programming*, Report 9801/A, Econometric Institute, Erasmus University Rotterdam, The Netherlands, 1998.

[11] H. Hu and Q. Wang, *On approximate solutions of infinite systems of linear inequalities*, Linear Algebra Appl., 114/115 (1989), pp. 429–438.

[12] G.J.O. Jameson, *The duality of pairs of wedges*, Proc. London Math. Soc., 24 (1972), pp. 531–547.

[13] D. Klatte and G. Thiere, *Error bounds for solutions of linear equations and inequalities*, ZOR — Math. Methods Oper. Res., 41 (1995), pp. 191–214.

[14] D. Klatte and G. Thiere, *A note of Lipschitz constants for solutions of linear inequalities and equations*, Linear Algebra Appl., 244 (1996), pp. 365–374.

[15] W. Li, *The sharp Lipschitz constants for feasible and optimal solutions of a perturbed linear program*, Linear Algebra Appl., 187 (1993), pp. 15–40.

[16] W. Li and I. Singer, *Global error bounds for convex multifunctions and applications*, Math. Oper. Res., 23 (1998), pp. 443–462.

[17] Z.Q. Luo, J.F. Sturm, and S. Zhang, *Duality Results for Conic Convex Programming*, Report 9719/A, Econometric Institute, Erasmus University Rotterdam, The Netherlands, 1997.

[18] O.L. Mangasarian, *A condition number of linear inequalities and equalities*, in Proceedings of the Sixth Symposium über Operations Research, Universität Ausburg, 1981, G. Bamber and O. Opitz, eds., Methods Oper. Res., 43 (1981), pp. 3–15.

[19] O.L. Mangasarian, *A condition number for differentiable convex inequalities*, Math. Oper. Res., 10 (1985), pp. 474–484.

[20] O.L. Mangasarian and T.-H. Shiau, *Lipschitz continuity of solutions of linear inequalities, programs and complementarity problems*, SIAM J. Control Optim., 25 (1987), pp. 583–595.

[21] Yu. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math., 13, Philadelphia, PA, 1994.

[22] J.-S. Pang, *Error bounds in mathematical programming*, Math. Programming, 79 (1997), pp. 299–332.

[23] J. Renegar, *Some perturbation theory for linear programming*, Math. Programming, 65 (1994), pp. 73–91.

[24] S.M. Robinson, *An application of error bounds for convex programming in a linear space*, SIAM J. Control, 13 (1975), pp. 271–273.

[25] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[26] G.W. Stewart, *On scaled projections and pseudoinverses*, Linear Algebra Appl., 112 (1989), pp. 189–193.

[27] J.F. Sturm, *Error bounds for linear matrix inequalities*, SIAM J. Control Optim., to appear.

[28] J.F. Sturm and S. Zhang, *On Sensitivity of Central Solutions in Semidefinite Programming*, Report 9813/A, Econometric Institute, Erasmus University Rotterdam, The Netherlands, 1998.

[29] M.J. Todd, *A Dantzig-Wolfe-like variant of Karmarkar's interior-point linear programming algorithm*, Oper. Res., 38 (1990), pp. 1006–1018.

[30] M.J. Todd, L. Tunçel, and Y. Ye, *Probabilistic Analysis of Two Complexity Measures for Linear Programming Problems*, Report CORR 98-48, Cornell University, Ithaca, NY, 1998.

[31] S.A. Vavasis and Y. Ye, *A primal-dual interior point method whose running time depends only on the constraint matrix*, Math. Programming, 74 (1996), pp. 79–120.

SIAM J. Optim.
Vol. 10, No. 3, pp. 852–877

A SPECIALIZED INTERIOR-POINT ALGORITHM FOR MULTICOMMODITY NETWORK FLOWS[*]

JORDI CASTRO[†]

Abstract. Despite the efficiency shown by interior-point methods in large-scale linear programming, they usually perform poorly when applied to multicommodity flow problems. The new specialized interior-point algorithm presented here overcomes this drawback. This specialization uses both a preconditioned conjugate gradient solver and a sparse Cholesky factorization to solve a linear system of equations at each iteration of the algorithm. The ad hoc preconditioner developed by exploiting the structure of the problem is instrumental in ensuring the efficiency of the method. An implementation of the algorithm is compared to state-of-the-art packages for multicommodity flows. The computational experiments were carried out using an extensive set of test problems, with sizes of up to 700,000 variables and 150,000 constraints. The results show the effectiveness of the algorithm.

Key words. interior-point methods, linear programming, multicommodity flows, network programming

AMS subject classifications. 90C05, 90C06, 90C35

PII. S1052623498341879

1. Introduction.

Multicommodity problems usually have many variables and constraints, which makes it difficult for them to be solved by general procedures. This has led to the formulation of specialized methods. However, some of the largest and most difficult multicommodity problems are still challenging even for these specializations. The algorithm presented in this paper has three main features. First, it has proven to be computationally efficient and robust in the solution of a wide range of problems, not just for some specific kind of multicommodity instances. Second, it is a specialized primal-dual interior-point algorithm, so it globally converges to the optimum in polynomial time, unlike other methods that provide an ϵ-approximate solution (e.g., [17]). And finally, it has been able to efficiently solve large instances of Patient Distribution System (PDS) problems [7]. This class of problems is commonly used as a de facto standard for testing the performance of multicommodity codes. With our algorithm we solved the PDS90 and PDS100 instances in a relatively reasonable amount of time. (In [17] an approximate solution is provided at most for PDS80.)

Most of the specialized methods attempt to exploit in some way the block structure of the multicommodity problem. Among the earlier approaches, we find primal partitioning and the price and resource directive decompositions (see [2, Chap. 17] and [21] for details). Of these three methods, the first two were regarded as the most successful in [3]. Despite this, no implementation of primal partitioning has been able to solve large problems significantly faster than the state-of-the-art simplex codes. For instance, the recent primal partitioning package PPRN [6] was, on average, no more than an order of magnitude faster than the primal simplex code of MINOS 5.3. In some cases, accurate implementations of the dual simplex—preceded by a warm

[*] Received by the editors July 13, 1998; accepted for publication (in revised form) October 6, 1999; published electronically May 2, 2000. This work was partially supported by Iberdrola S.A. grant 95-005 and by CICYT project TAP96-1044-J02-93.

http://www.siam.org/journals/siopt/10-3/34187.html

[†] Statistics and Operations Research Department, Universitat Politècnica de Catalunya, Campus Sud, Pau Gargallo 5, 08028 Barcelona, Spain (jcastro@eio.upc.es).

start based on solving minimum-cost network problems for each commodity—can even outperform primal partitioning multicommodity specializations (see [13] for a comparison of PPRN and the network+dual solver of CPLEX 3.0). In this paper, we show that our algorithm, in general, outperforms both PPRN and CPLEX 4.0 and, in some cases, by more than an order of magnitude.

The other method regarded as successful in [3], the price directive or Dantzig–Wolfe decomposition, belongs to the class of cost decomposition approaches for multicommodity flows (see [13], [15], and [33] for recent variants based on bundle methods, analytic centers, and smooth penalty functions, respectively). A recent computational study [13] showed that these are promising approaches for solving a wide variety of problems. However, for some classes of instances—typically difficult problems with large networks and not many commodities such as the PDS ones—our interior-point approach seems to give considerably better performance. Furthermore, Frangioni [12] noted that the particular cost decomposition method of [13] might sometimes require the algorithmic parameters to be tuned if performances are to be the best possible, whereas our algorithm works well with default values.

Interior-point methods have also been applied in the past. For the single-commodity case, efficient specializations were developed by Resende and others [26, 28, 29, 30]. These specializations relied on the use of preconditioned conjugate gradient (PCG) solvers. The preconditioners developed, though efficient, were appropriate only for single-commodity problems. The first reported attempt at solving multicommodity problems by an interior-point method was probably that described in [1]. However, the general implementation of Karmarkar's projective algorithm used there was outperformed by a simplex specialized algorithm in the solution of small-size multicommodity instances. Alternative and more efficient approaches were developed in the following years. In fact, the best complexity bound known for multicommodity problems is provided by the two interior-point algorithms described in [19] and [20], though none of these papers provided computational results. In [18], Kamath et al. applied a variant of Karmarkar's projective algorithm using a PCG solver. However, their preconditioner did not take advantage of the multicommodity structure. An attempt to exploit this structure was made in [9] by Choi and Goldfarb. Though the decomposition scheme they presented is similar to the one in this paper, the solution procedure differs substantially. Choi and Goldfarb suggest solving a fairly dense matrix positive definite linear system that appears during the decomposition stage by means of parallel and vector processing, whereas we apply a PCG method, which enables large problems to be solved efficiently using a midsize workstation. A different interior-point approach was developed in [31], using a barrier function to decompose the problem. This strategy provided approximate solutions for some of the large PDS problems (up to PDS70). However, as will be shown, our method gives more accurate solutions. Finally, Portugal et al. introduced in [27] a specialized interior-point algorithm based solely on a PCG, unlike our method, which combines PCG with direct factorizations. The proposed preconditioner was an extension of that developed in [26] by the same authors for single-commodity flows. No computational results were reported in [27] for the solution of multicommodity problems using this preconditioner.

This paper is organized as follows. Section 2 presents the formulation of the problem to be solved. Section 3 outlines the primal-dual algorithm, and in section 4 we develop the specialization for multicommodity problems. Section 5 describes some implementation details of this specialization. Finally, section 6 gives computational results that show the efficiency of the algorithm.

2. Problem formulation. Let $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ be a directed graph, where $\mathcal{N}$ is a set of $m+1$ nodes and $\mathcal{A}$ is a set of n arcs, and let $\mathcal{K}$ be a set of k commodities to be routed through the network represented by $\mathcal{G}$. We shall also consider that the arcs of the network have a capacity for all the commodities, which will be known as the mutual capacity. So, the multicommodity network flow (MCNF) problem can be formulated as follows:

$$\min_{x^{(1)},\ldots,x^{(k)}} \quad \sum_{i=1}^{k} {c^{(i)}}^T x^{(i)} \tag{1}$$

$$\text{subject to } A_N x^{(i)} = b^{(i)}, \qquad i = 1,\ldots,k, \tag{2}$$

$$\sum_{i=1}^{k} x^{(i)} \leq b_{mc}, \tag{3}$$

$$0 \leq x^{(i)} \leq \overline{x}^{(i)}, \qquad i = 1,\ldots,k. \tag{4}$$

Vectors $x^{(i)} \in \mathbb{R}^n$ and $c^{(i)} \in \mathbb{R}^n$ are the flow and cost arrays for each commodity i, $i = 1,\ldots,k$. $A_N \in \mathbb{R}^{m \times n}$ is the node-arc incidence matrix, where each column is related to an arc $a \in \mathcal{A}$, and has only nonzero coefficients in those rows associated with the origin and destination nodes of a (with coefficients 1 and -1, respectively). We shall assume that A_N is a full row-rank matrix. This can always be guaranteed by removing any of the (redundant) node balance constraints. $b^{(i)} \in \mathbb{R}^m$ is the vector of supplies and demands for commodity i at the nodes of the network. Equation (3) represents the mutual capacity constraints, where $b_{mc} \in \mathbb{R}^n$. Constraints (4) are simple bounds on the flows, $\overline{x}^{(i)} \in \mathbb{R}^n$, $i = 1,\ldots,k$, being the upper bounds. These upper bounds represent individual capacities of the arcs for each commodity.

Introducing the slacks s_{mc} for the mutual capacity constraints, (3) can be rewritten as

$$\sum_{i=1}^{k} x^{(i)} + s_{mc} = b_{mc}. \tag{5}$$

We can consider that the slacks s_{mc} are upper bounded by b_{mc}, since all the vectors $x^{(i)}$ in (5) have nonzero components. This gives

$$0 \leq s_{mc} \leq b_{mc}. \tag{6}$$

The MCNF problem can then be recast as

$$\min \text{ (1) subject to (2), (4), (5), and (6).} \tag{7}$$

3. Outline of the primal-dual interior-point algorithm. Let us consider the linear programming problem

$$\begin{array}{rl} \min & c^T x \\ \text{subject to} & Ax = b, \\ & \overline{x} \geq x \geq 0, \end{array} \tag{8}$$

where $x \in \mathbb{R}^{\tilde{n}}$, $\overline{x} \in \mathbb{R}^{\tilde{n}}$ are the upper bounds, $c \in \mathbb{R}^{\tilde{n}}$, $b \in \mathbb{R}^{\tilde{m}}$, and $A \in \mathbb{R}^{\tilde{m} \times \tilde{n}}$ is a full row-rank matrix. The dual of (8) is

$$\begin{array}{rl} \max & b^T y - \overline{x}^T w \\ \text{subject to} & A^T y + z - w = c, \\ & z \geq 0, \quad w \geq 0, \end{array} \tag{9}$$

where $y \in \mathbb{R}^{\tilde{m}}$ are the dual variables and $z \in \mathbb{R}^{\tilde{n}}$ and $w \in \mathbb{R}^{\tilde{n}}$ are the dual slacks. Note that the MCNF problem, as defined in (7), fits the formulation of (8), where $\tilde{n} = (k+1)n$ and $\tilde{m} = km + n$.

Replacing the inequalities in (8) by a logarithmic barrier in the objective function, with parameter μ, and considering the slacks $f = \overline{x} - x$, it can be seen that the KKT first order optimality conditions of (8) and (9) are equivalent to the following system of nonlinear equations (see [32] for a comprehensive description):

$$\begin{aligned} b_{xz} &\equiv \mu e_{\tilde{n}} - XZe_{\tilde{n}} = 0, \\ b_{fw} &\equiv \mu e_{\tilde{n}} - FWe_{\tilde{n}} = 0, \\ b_b &\equiv b - Ax = 0, \\ b_c &\equiv c - (A^Ty + z - w) = 0, \\ &(x, z, w) \geq 0, \quad \overline{x} \geq x, \end{aligned} \tag{10}$$

where $e_{\tilde{n}}$ is the $\tilde{n}$-dimensional vector of 1's; X, Z, F, and W are diagonal matrices defined as $M \in \mathbb{R}^{\tilde{n}\times\tilde{n}} = \text{diag}(m_1, \dots, m_{\tilde{n}})$; and the vectors b_* define the left-hand-side terms of (10). Note that we did not include the slacks equation $x + f = \overline{x}$ in (10). Instead we replaced the slacks f by $\overline{x} - x$, reducing by $\tilde{n}$ the number of equations and variables. This forces the primal variables x of the iterates obtained during the solution of (10) to always be interior in relation to their upper bounds.

The solutions of system (10)—considering inequalities as strict inequalities—for different μ values gives rise to an arc of strictly feasible points known as the central path. As μ tends to 0, the solutions of (10) converge to those of the original primal and dual problems. A path-following algorithm attempts to follow the central path, computing (10)—in long-step methods—through a damped Newton's method together with the reduction of the barrier parameter μ at each iteration of the algorithm. The path-following algorithm considered for the specialization uses the reduction formula $\mu = 0.1(x^Tz + f^Tw)/2\tilde{n}$. It can be seen [32] that obtaining Newton's direction amounts to finding dy and then computing dx, dw, dz, in

$$\begin{aligned} (A\Theta A^T)dy &= b_b + A\Theta r, \\ dx &= \Theta(A^Tdy - r), \\ dw &= F^{-1}(b_{fw} + Wdx), \\ dz &= b_c + dw - A^Tdy, \end{aligned} \tag{11}$$

where

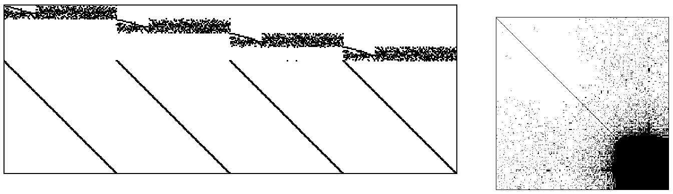
$$r = F^{-1}b_{fw} + b_c - X^{-1}b_{xz}, \quad r \in \mathbb{R}^{\tilde{n}}, \tag{12}$$

$$\Theta = FX(ZF + XW)^{-1}, \quad \Theta \in \mathbb{R}^{\tilde{n}\times\tilde{n}}. \tag{13}$$

Note that Θ is a positive definite diagonal matrix, since it is nothing but a product of positive definite diagonal matrices. Since A is a full row-rank matrix, $A\Theta A^T$ is also positive definite. It is quite clear that the main computational burden of the algorithm is the repeated solution of the linear system

$$(A\Theta A^T)dy = \overline{b}, \tag{14}$$

where $\overline{b}$ denotes $b_b + A\Theta r$ in (11). The performance of any primal-dual multicommodity specialization relies on the efficient solution of (14).



(a) (b)

FIG. 1. (a) *Sparsity pattern of a multicommodity constraint matrix A.* (b) *Sparsity pattern of the factorization of $PA\Theta A^TP^T$.*

4. Primal-dual specialization for the MCNF problem.

4.1. Motivation. General interior-point codes for linear programming attempt to solve (14) through sparse Cholesky factorizations. To reduce the fill-in, they factorize $PA\Theta A^TP^T$, instead of $A\Theta A^T$, where P is a permutation matrix obtained by some heuristic. However, when applied to multicommodity problems, even the best P matrices, such as those provided by good heuristics like the minimum-local-fill-in or minimum-degree orderings, cannot prevent a fairly large dense submatrix from appearing in $LL^T = PA\Theta A^TP^T$. For instance, Figure 1(a) shows the sparsity pattern of the constraint matrix A for a multicommodity problem with 4 commodities, 64 nodes, and 524 arcs (it corresponds to problem M_1 in Table 3 of section 6). Using the state-of-the-art interior-point code BPMPD [23], the sparsity pattern obtained for $L + L^T$ is depicted in Figure 1(b). The dense submatrix created makes the factorization of $PA\Theta A^TP^T$ computationally expensive and, for large problems, its storage in the memory completely prohibitive. Then it is clear that, to be competitive, interior-point methods must exploit the structure of the multicommodity problem to efficiently solve (14).

4.2. Exploiting the multicommodity structure. The constraint matrix A of the MCNF problem defined in (7) has the following structure:

$$A = \begin{array}{|c|c|c|c|c|}\hline A_N & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{0} & A_N & \dots & \mathbf{0} & \mathbf{0} \\ \hline \vdots & \vdots & \ddots & \vdots & \vdots \\ \hline \mathbf{0} & \mathbf{0} & \dots & A_N & \mathbf{0} \\ \hline \mathbb{1}_n & \mathbb{1}_n & \dots & \mathbb{1}_n & \mathbb{1}_n \\ \hline \end{array}\,, \tag{15}$$

where $\mathbb{1}_n$ denotes the $n \times n$ identity matrix and $\mathbf{0}$ is the zero matrix. Moreover, matrix Θ, as defined in (13), can be partitioned as

$$\Theta = \begin{bmatrix} \boxed{\Theta^{(1)}} & & & \\ & \ddots & & \\ & & \boxed{\Theta^{(k)}} & \\ & & & \boxed{\Theta_{mc}} \end{bmatrix}, \tag{16}$$

where $\Theta^{(i)} \in \mathbb{R}^{n \times n}$, $i = 1, \ldots, k$, and $\Theta_{mc} \in \mathbb{R}^{n \times n}$ are associated with the flows $x^{(i)}$ of commodity i and the mutual capacity slacks s_{mc}, respectively. Using (15) and (16), it is straightforward to see that $A\Theta A^T$ in (14) has the following structure:

$$A\Theta A^T = \begin{array}{|c|c|c|c|}\hline A_N\Theta^{(1)}A_N^T & \ldots & \mathbf{0} & A_N\Theta^{(1)} \\ \hline \vdots & \ddots & \vdots & \vdots \\ \hline \mathbf{0} & \ldots & A_N\Theta^{(k)}A_N^T & A_N\Theta^{(k)} \\ \hline \Theta^{(1)}A_N^T & \ldots & \Theta^{(k)}A_N^T & \Theta_{mc} + \sum_{i=1}^k \Theta^{(i)} \\ \hline \end{array} = \begin{array}{|c|c|}\hline B & C \\ \hline C^T & D \\ \hline \end{array}, \tag{17}$$

where $B \in \mathbb{R}^{km \times km}$ is the block diagonal matrix

$$B = \mathrm{diag}(A_N\Theta^{(i)}A_N^T, \quad i = 1, \ldots, k), \tag{18}$$

each block being a square matrix of dimension m, $C \in \mathbb{R}^{km \times n}$ is defined as

$$C = \begin{bmatrix} \Theta^{(1)}A_N^T & \ldots & \Theta^{(k)}A_N^T \end{bmatrix}^T, \tag{19}$$

and $D \in \mathbb{R}^{n \times n}$ corresponds to the lower diagonal submatrix of $A\Theta A^T$:

$$D = \Theta_{mc} + \sum_{i=1}^{k} \Theta^{(i)}. \tag{20}$$

Since Θ is diagonal and positive definite, it holds that D is a positive definite diagonal matrix as well.

The above decomposition of $A\Theta A^T$ can be applied to the solution of (14), partitioning appropriately the dual variables direction dy and the right-hand-side vector $\bar{b}$:

$$\begin{array}{|c|c|}\hline B & C \\ \hline C^T & D \\ \hline \end{array} \; \begin{array}{|c|}\hline dy_1 \\ \hline dy_2 \\ \hline \end{array} = \begin{array}{|c|}\hline \bar{b}_1 \\ \hline \bar{b}_2 \\ \hline \end{array}, \tag{21}$$

where dy_1, $\bar{b}_1 \in \mathbb{R}^{km}$ and dy_2, $\bar{b}_2 \in \mathbb{R}^n$. The solution of (21) can be directly obtained by block multiplication, yielding

$$(D - C^T B^{-1} C) dy_2 = (\bar{b}_2 - C^T B^{-1} \bar{b}_1), \tag{22}$$

$$B dy_1 = (\bar{b}_1 - C dy_2). \tag{23}$$

Matrix $D - C^T B^{-1} C$ is known as the Schur complement, and it will be denoted by S:

$$S = D - C^T B^{-1} C. \tag{24}$$

To efficiently solve (22) and (23)—and obtain the solution to (14)—we only need to deal with systems involving matrices B and S. Systems with matrix B can be considered not too difficult. In fact, exploiting the block structure of B shown in (18), these systems can be decomposed into k smaller ones of dimension m with matrices $A_N\Theta^{(i)}A_N^T$, $i = 1, \ldots, k$. Each of these matrices can be easily obtained. If we denote by $\mathcal{I}_v$ the set of arcs incident to node v and by $a \equiv (v, w)$ the arc of $\mathcal{A}$ that has v and w as origin and destination nodes and consider the structure of the node-arc incidence matrix A_N, it is straightforward to see that $A_N \Theta^{(i)} A_N^T$ can be easily computed as follows:

$$(25)\ (A_N \Theta^{(i)} A_N^T)_{\substack{vw \\ v=1,\dots,m \\ w=1,\dots,m}} = \begin{cases} \sum\limits_{\forall a} -\Theta_a^{(i)} & \text{if } a \equiv (v,w) \in \mathcal{A}, (w,v) \notin \mathcal{A}, \\ \sum\limits_{\forall a,b} \left(-\Theta_a^{(i)} - \Theta_b^{(i)}\right) & \text{if } a \equiv (v,w) \in \mathcal{A}, b \equiv (w,v) \in \mathcal{A}, \\ \sum\limits_{\forall a \in I_v} \Theta_a^{(i)} & \text{if } (v = w), \\ 0 & \text{otherwise,} \end{cases}$$

where $\Theta_a^{(i)}$ is the diagonal term of $\Theta^{(i)}$ associated to arc a. Moreover, since $\Theta^{(i)}$ is symmetric and positive definite and A_N is a full row-rank network matrix, we have that matrices $A_N \Theta^{(i)} A_N^T$ are symmetric and positive definite as well. Therefore, their Cholesky factorizations exist. In practice, to reduce the fill-in, instead of $A_N \Theta^{(i)} A_N^T$, we shall factorize $P_N A_N \Theta^{(i)} A_N^T P_N^T$, where P_N is a permutation matrix of the nodes of the network. Note that P_N will have to be computed only once, since the nonzero pattern of $A_N \Theta^{(i)} A_N^T$ is the same for all the commodities. In general, due to the high sparsity of the network matrix A_N, we can expect that these k Cholesky factorizations—and, hence, the factorization of B—will not be too computationally expensive. Additionally, in a parallel computing environment, these k factorizations—and their respective backward and forward substitutions—can be carried out independently for each commodity.

System (22) still remains to be solved. We could consider computing and factorizing S. However, this would mean solving n systems of equations with matrix B, n being the number of arcs of the network. In addition, S could become fairly dense. In fact, as the proposition below shows, if we perform symbolic computations, matrix S turns out to be completely dense, increasing the solution cost of (22) with a direct method. With no loss of generality and to simplify the notation we will consider a problem with only one commodity and where $P_N = \mathbb{1}$ (no node permutation is required to reduce the fill-in for $A_N \Theta^{(i)} A_N^T$).

Proposition 1. *Let* $L^{(1)} L^{(1)^T} = A_N \Theta^{(1)} A_N^T$ *be the Cholesky factorization of* $B = A_N \Theta^{(1)} A_N^T$*. If we apply this factorization to remove the subdiagonal elements of* B *and submatrix* C *in* (17) *by symbolic Gaussian elimination*

$$\begin{array}{|c|c|}\hline L^{(1)^{-1}} & \mathbf{0} \\ \hline -C^T B^{-1} & \mathbb{1} \\ \hline \end{array} \; \begin{array}{|c|c|}\hline B & C \\ \hline C^T & D \\ \hline \end{array} = \begin{array}{|c|c|}\hline L^{(1)^T} & L^{(1)^{-1}} C \\ \hline \mathbf{0} & D - C^T B^{-1} C \\ \hline \end{array},$$

submatrix $D - C^T B^{-1} C$*—the Schur complement—becomes completely dense.*

Proof. Let $\mathcal{N}_v$ be the set of nodes adjacent to node $v \in \mathcal{N}$, i.e.,

$$\mathcal{N}_v = \{w \in \mathcal{N} \text{ such that } (v,w) \in \mathcal{A} \text{ or } (w,v) \in \mathcal{A}\};$$

this set will be associated to matrix $B = A_N \Theta^{(1)} A_N^T$. Let $\mathcal{I}_v$ be the set of arcs incident to node v, i.e.,

$$\mathcal{I}_v = \{a \in \mathcal{A} \text{ such that } a \equiv (w,v) \in \mathcal{A} \text{ or } a \equiv (v,w) \in \mathcal{A}\};$$

this set will be associated to matrix $C = A_N\Theta^{(1)}$. And let $\mathcal{C}_a$ be the set of nodes connected to arc $a \in \mathcal{A}$ (initially $\mathcal{C}_a = \{v, w\}$, where $a \equiv (v, w)$); this set will be associated to matrix $C^T = \Theta^{(1)}A_N^T$. Moreover we will denote as $M^{j)}/\mathcal{M}^{j)}$ the matrix/set $M/\mathcal{M}$ after j elimination stages—the original sets and matrices correspond to $M^{0)}/\mathcal{M}^{0)}$—and by v_i and a_j the node and arc associated to row i and column j of A_N, respectively.

Let us assume we are starting the Gaussian elimination and we have to remove the subdiagonal terms of the first column of (17). This will be done through the first row of $B^{0)}$ (which corresponds to the first node v_1 of A_N). From (25) it can be seen that we shall have to remove the elements of the rows of $B^{0)}$ related to the nodes in $\mathcal{N}_{v_1}^{0)}$. Two new nonzero elements will then appear, one in the upper and the other in the lower diagonal parts of $B^{1)}$, for each pair of nodes (v_i, v_j) in $\mathcal{N}_{v_1}^{0)}$ not yet connected by any arc. The adjacent node sets are suitably updated as

$$\text{for all } v_i \in \mathcal{N}_{v_1}^{0)} \quad \mathcal{N}_{v_i}^{1)} = \mathcal{N}_{v_i}^{0)} \bigcup \mathcal{N}_{v_1}^{0)} - \{v_1\}.$$

(A comprehensive explanation of this result can be found in [14, Chap. 5].) New nonzero elements will appear in matrix $C^{1)}$ as well. Initially, the only nonzero elements in row i of $C^{0)}$ are found in the columns of the arcs $\mathcal{I}_{v_i}^{0)}$. After the first elimination stage we find that

$$\text{for all } v_i \in \mathcal{N}_{v_1}^{0)} \quad \mathcal{I}_{v_i}^{1)} = \mathcal{I}_{v_i}^{0)} \bigcup \mathcal{I}_{v_1}^{0)}.$$

Similarly, when eliminating the first column of $C^{T0)}$, new nonzero elements will appear in $C^{T1)}$. Unlike $C^{1)}$, these new entries are related to arcs, thus having

$$\text{for all } a_j \in \mathcal{I}_{v_1}^{0)} \quad \mathcal{C}_{a_j}^{1)} = \mathcal{C}_{a_j}^{0)} \bigcup \mathcal{N}_{v_1}^{0)}.$$

Repeating the above procedure, it is not difficult to see that, after $m-1$ elimination stages, all the nodes collapsed into the last one v_m (see [14, Chap. 5] again for a detailed description), yielding

$$\mathcal{N}_{v_m}^{m-1)} = \bigcup_{i=1}^{m} \mathcal{N}_{v_i} = \mathcal{V}.$$

It also holds, for the last row in matrix $C^{m-1)}$, that

$$\mathcal{I}_{v_m}^{m-1)} = \bigcup_{i=1}^{m} \mathcal{I}_{v_i} = \mathcal{A}$$

and, analogously for the last column in matrix $C^{Tm-1)}$, that

$$\text{for all } a_j \in \mathcal{A} \quad v_m \in \mathcal{C}_{a_j}^{m-1)}.$$

Therefore, the last row in $C^{m-1)}$ and the last column in $C^{Tm-1)}$ become dense. It is now clear that, if we attempt to eliminate the last column of $C^{Tm-1)}$ from the last row in the $C^{m-1)}$ matrix, $D - C^TB^{-1}C$ becomes dense. □

In practice, however, if we perform numerical instead of symbolic computations, S will not be completely dense due to cancellations. As shown by the next proposition, the numerical sparsity pattern of S depends on the structure of the network and, in the simplest case, can even be diagonal.

PROPOSITION 2. *If the network is a spanning tree (thus, it is connected and $m = n + 1$), the Schur complement is diagonal.*

Proof. In this case, matrix A_N is square and nonsingular. Using (18), (19), (20), and the nonsingularity of A_N, the Schur complement can be written as

$$\begin{aligned} S = D - C^T B^{-1} &= \Theta_{mc} + \sum_{i=1}^{k} \Theta^{(i)} - \sum_{i=1}^{k} \Theta^{(i)} A_N^T (A_N \Theta^{(i)} A_N^T)^{-1} A_N \Theta^{(i)} \\ &= \Theta_{mc} + \sum_{i=1}^{k} \Theta^{(i)} - \sum_{i=1}^{k} \Theta^{(i)} = \Theta_{mc}. \qquad \square \end{aligned}$$

The density of S increases with the complexity of the network. If each pair of nodes of the network is connected by at least one arc, S can be shown to be numerically completely dense for most Θ matrices. Leaving aside these extreme cases, for general networks the Schur complement will be numerically fairly dense. This fact, together with the cost associated with building matrix S, makes the solution of (22) with a direct method prohibitive. A similar system had to be solved in the approach suggested in [9]. However, no procedure was given there to circumvent this difficulty, and the solution of (22) was addressed through parallel and vector processing. Rather than use a direct method, the specialization we propose attempts to solve (22) through a PCG.

4.3. Solution via a preconditioned conjugate gradient method. Before applying a PCG method to (22) we must guarantee that S is symmetric and positive definite at each iteration of the algorithm.

LEMMA 1. *Let $T \in \mathbb{R}^{t \times t}$ be a square matrix partitioned as follows:*

$$T = \begin{array}{|c|c|} \hline B & C \\ \hline C^T & D \\ \hline \end{array}\,.$$

Then, if T is symmetric and positive definite and B is positive definite, it holds that the Schur complement $S = D - C^T B^{-1} C$ is symmetric and positive definite.

PROPOSITION 3. *The Schur complement matrix $S = D - C^T B^{-1} C$ defined in* (22) *is symmetric and positive definite at each iteration of the primal-dual algorithm.*

Proof. The primal and dual variables—x, and z and w—are interior at each iteration of the primal-dual algorithm. So we have that Θ, as defined in (13), is a positive definite diagonal matrix. Moreover, since the network matrix A_N was assumed to be a full row-rank matrix, the constraint matrix of the MCNF problem defined in (15) is a full row-rank matrix as well. Therefore, matrices $A^T \Theta A$ and B defined in (17) and (18) are both symmetric and positive definite. Applying Lemma 1, with $T = A^T \Theta A$, we get that S is symmetric and positive definite. $\square$

The preconditioner that we propose in this paper, denoted by M, consists of using an approximation of the inverse of S. The development of this preconditioner relies on the following theorem.

THEOREM 1 (P-regular splitting theorem). *If R is symmetric positive definite and $R = P - Q$ is a P-regular splitting—i.e., P is nonsingular and $P + Q$ is positive definite—then $\rho(P^{-1}Q) < 1$ (where $\rho(T)$ denotes the spectral radius of T).*

Proof. See [25, pp. 254–255]. □

PROPOSITION 4. *The inverse of $S = D - C^T B^{-1} C$ can be computed as*

$$S^{-1} = \left(\sum_{i=0}^{\infty} (P^{-1}Q)^i \right) P^{-1}, \tag{26}$$

where

$$P = D, \qquad Q = C^T B^{-1} C. \tag{27}$$

Proof. Premultiplying S by S^{-1} as defined in (26) we get

$$\begin{aligned} S^{-1}S &= \left(\left(\sum_{i=0}^{\infty} (P^{-1}Q)^i \right) P^{-1} \right) (P - Q) \\ &= \sum_{i=0}^{\infty} (P^{-1}Q)^i - \sum_{i=1}^{\infty} (P^{-1}Q)^i. \end{aligned} \tag{28}$$

Since $P = D$ is a diagonal positive definite matrix, it is nonsingular. $P + Q = D + C^T B^{-1} C$ is positive definite as well because both D and B are positive definite. Thus, $P - Q$ is a regular splitting of S. Moreover, S is symmetric and positive definite, as stated by Proposition 3. By Theorem 1, we have that $\rho(P^{-1}Q) < 1$, and then the geometric power series of (28) converge, obtaining the desired result:

$$S^{-1}S = (P^{-1}Q)^0 + \sum_{i=1}^{\infty} (P^{-1}Q)^i - \sum_{i=1}^{\infty} (P^{-1}Q)^i = \mathbb{1}. \qquad \square$$

The preconditioner is then obtained by truncating the infinite geometric power series (26) at some term $\phi \geq 0$, which will be referred to as the order of the preconditioner:

$$M^{-1} = (\mathbb{1} + (P^{-1}Q) + (P^{-1}Q)^2 + \cdots + (P^{-1}Q)^{\phi}) P^{-1}, \tag{29}$$

where P and Q are defined in (27). Note that M is an adequate preconditioner for the PCG, since it is symmetric and positive definite. (This can be easily proved by showing that, from the symmetry and positive definiteness of both P and Q, M^{-1} is symmetric and positive definite as well.) The main drawback of the preconditioner is that matrices P and Q both become ill-conditioned as the iterates approach a solution, which can lead to values $\rho(P^{-1}Q)$ very close to 1; consequently, (29) would be a poor approximation of S^{-1}. Despite this, the preconditioner has shown to be an efficient solution strategy, being able to significantly reduce the number of iterations required by nonpreconditioned conjugate gradient (CG) methods. For instance, Figure 2 shows the evolution of $\rho(P^{-1}Q)$ for M_1 and PDS1, the smallest problems in Tables 3 and 4 in section 6. Both problems required 30 interior-point iterations. It can be seen that, though it tends to decrease for the central iterations, $\rho(P^{-1}Q)$ is close to 1 throughout the execution of the algorithm (especially for problem PDS1). As shown below (next two paragraphs, and Figures 4 and 5), even in these situations the goodness of the preconditioner increases with ϕ, and we obtain a better performance than with a nonpreconditioned CG method.

Clearly, the higher ϕ is, the better the preconditioning and the fewer iterations of the PCG will be required. However, at each iteration of the PCG, we have to solve

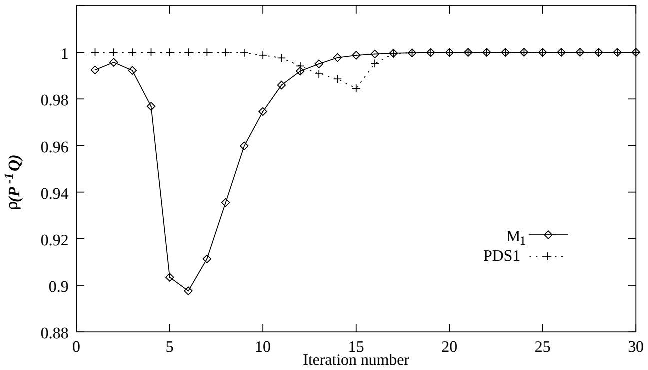


FIG. 2. *Evolution of $\rho(P^{-1}Q)$ for the M_1 and PDS1 problems.*

```
Procedure MZ=r $(P, Q, r, z, \phi)$
$v := P^{-1}r$
$z_0 := v$
for j=1 to $\phi$ do
        $z_j := P^{-1}Qz_{j-1} + v$
end_do
$z := z_\phi$
```

FIG. 3. *Procedure for computing $z = M^{-1}r$.*

the system $Mz = r$, with r being any vector. This system can be easily computed through the procedure presented in Figure 3, which involves solving ϕ systems with matrix B, and thus a total of $k\phi$ systems with matrices $A_N^T\Theta^{(i)}A_N$, $i = 1, \ldots, k$. Then ϕ must be chosen to balance two objectives: to reduce both the number of PCG iterations and the number of systems to be solved. In practice, performances are best for $\phi = 0$ and, in some cases, for $\phi = 1$. For instance, Figure 4 shows the evolution of the CPU time and overall number of PCG iterations required to solve problem M_1 in Table 3 of section 6 for different ϕ values. Clearly, there are fewer PCG iterations when ϕ increases, but the performance tends to be poorer. This is the usual behavior observed in most problems tested. The algorithm uses $\phi = 0$ as the default value, though this parameter can be modified by the user. All the numerical results in section 6 were obtained with this default value. Note that when $\phi = 0$ the preconditioner is nothing but $M = P = D$, the diagonal matrix defined in (20). In this case the computation of $Mz = r$ is reduced to n products.

Despite its simplicity, the diagonal preconditioner obtained for $\phi = 0$ has proven to be very efficient compared to a nonpreconditioned CG method. For instance, Figure 5 shows the number of overall CG iterations required to solve the first 10 PDS problems in Table 4 in section 6, by both the PCG with $\phi = 0$ and a standard CG method. The number of interior-point iterations was almost the same in both types of executions. However, it is clear from the figure than the CG required many more CG iterations to achieve the same accuracy in the solution of (22). For the 10 problems, the code with the PCG was, on average, 3.7 times faster than that with the CG and performed 7.5 times fewer CG iterations.

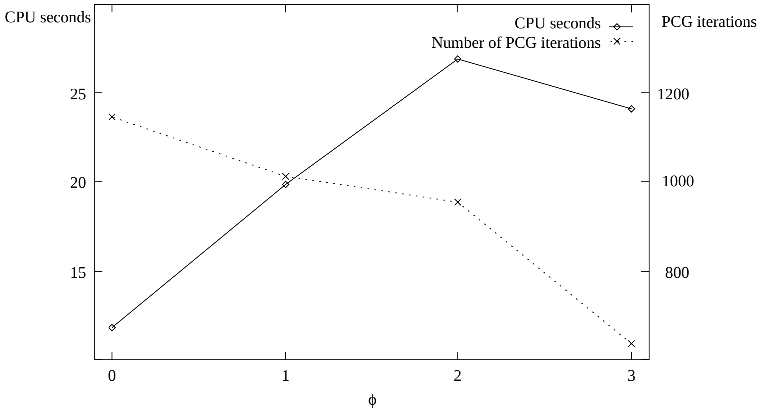


FIG. 4. *CPU time and overall number of PCG iterations for different ϕ values.*

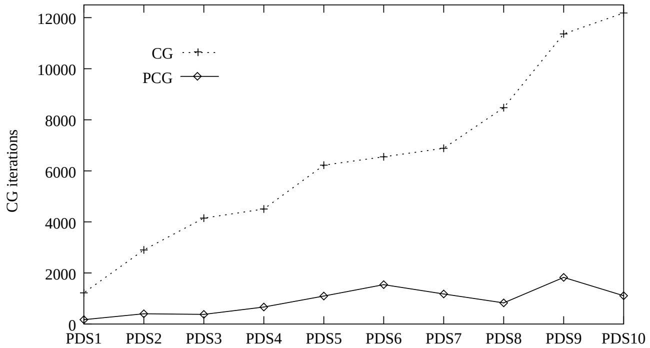


FIG. 5. *Overall number of CG iterations for the PCG with $\phi = 0$ and a CG method.*

Three remarks must be made about this solution strategy.

(i) Though designed for multicommodity instances, it can be applied to other block-structured problems where a similar decomposition to that of (21) is possible. We mention three of them. First, a direct extension of the MCNF problem consisting of replacing the mutual capacity constraints by the more general ones

$$\sum_{i=1}^{k} W^{(i)} x^{(i)} \leq b_{mc},$$

where $W^{(i)}$ is a diagonal matrix of positive weights. Second, the nonoriented multicommodity problem—arcs have no orientation—which commonly appears, for instance, in telecommunication networks [8]. And finally, multicommodity problems with convex separable quadratic objective functions (e.g., $\sum_{i=1}^{k} \sum_{a\in\mathcal{A}} c_a^{(i)} (x_a^{(i)})^2$, $c_a^{(i)} \geq 0$), which only imply a slight modification of the Θ diagonal matrix. Note that simplex-based specializations for multicommodity flows cannot deal with this last class of problems.

(ii) At each iteration of the PCG, $\phi + 1$ systems of equations with matrix B must be solved for computing $Mz = r$ and $q = Sp(z)$, $p(z)$ being a vector that depends on z. Since the k blocks of B have already been factorized, only the forward and backward substitutions must be performed. The solutions to these k systems, however, can

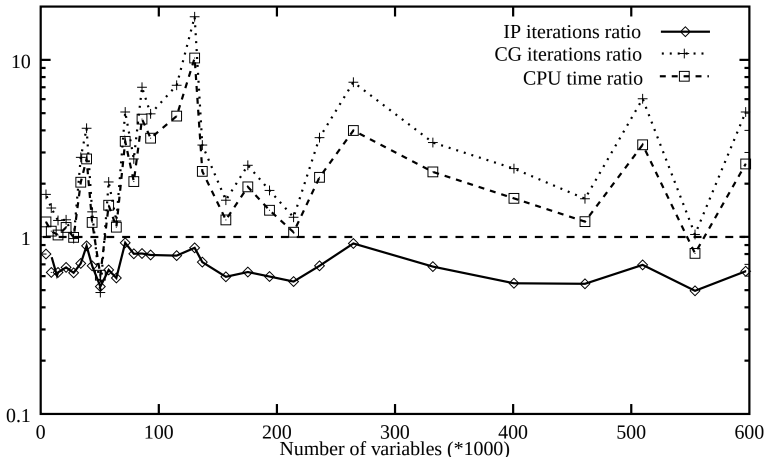


FIG. 6. *Predictor-corrector vs. pure path-following, for the PDS problems.*

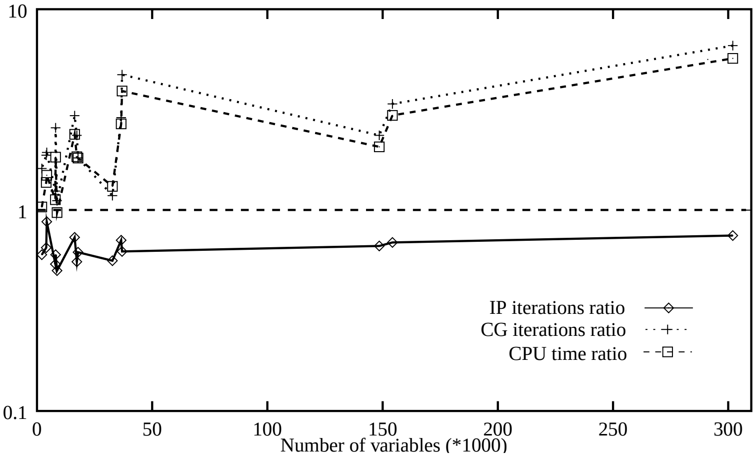


FIG. 7. *Predictor-corrector vs. pure path-following, for the Mnetgen problems.*

be efficiently parallelized since each of them requires the same computational effort (load balancing). We can expect that a coarse-grain parallel implementation of the algorithm would significantly reduce the solution time (at most by a factor of k).

(iii) The solution to (22) using a PCG algorithm forces us to use a pure primal-dual path-following algorithm instead of other more successful approaches, such as Mehrotra's predictor-corrector method. The predictor-corrector method requires two solutions to (21) with different right-hand sides. In our specialization, the benefit obtained by computing this better direction is not worthwhile, since it means applying the PCG method twice. Figures 6 and 7 compare a version of the algorithm using Mehrotra's predictor-corrector with a pure path-following algorithm for the PDS and Mnetgen problems in Tables 3 and 4 in section 6. The predictor-corrector strategy was implemented as described in [22] and [32]. The direction computed by the predictor step was used as the starting point for the PCG of the corrector step in an attempt to reduce the number of CG iterations. Figures 6 and 7 show the ratio of the number of interior-point iterations, CG iterations, and execution time between the predictor-corrector and pure path-following algorithms. The dashed horizontal line separates the executions according to whether the ratio was favorable to the predictor-corrector (region below the line) or the pure path-following algorithm (region above the line). Clearly, the predictor-corrector heuristic reduced the number of iterations (on average, it performed 1.45 and 1.55 times fewer iterations for the PDS and Mnetgen problems, respectively). However, the overall number of CG iterations significantly increased (with average ratios of 3.5 and 2.6 for each class of problems). The execution time ratio, highly correlated with the CG iteration ratio, was favorable to the pure path-following algorithm for most of the problems. Of the larger instances executed, the predictor-corrector strategy was only better for PDS90. The larger Mnetgen instances were not executed with the predictor-corrector method due to their large execution times (e.g., problem M_{18} was stopped after 21 hours of execution, whereas the pure path-following algorithm required only 2.55 hours). On average, the pure path-following algorithm was 2.3 times faster than the predictor-corrector strategy for the PDS problems and 2.2 times faster for the Mnetgen ones.

5. Implementation details.

We have developed an implementation of the algorithm presented in section 4 that will be referred to as IPM. This code is mainly written in C, with only the Cholesky factorization routines coded in Fortran. It can be freely obtained for academic purposes from http://www-eio.upc.es/˜jcastro, at software entry. Below, we discuss some of the implementation aspects of IPM that have proved to be instrumental in the performance of the algorithm.

5.1. Cholesky factorizations.

The factorizations of matrices $A_N^T\Theta^{(i)}A_N$, $i = 1,\dots,k$, and, mainly, their backward and forward substitutions at each iteration of the PCG solver are the most computationally expensive steps of the algorithm. Note that the symbolic factorization must be performed only once, since matrices $A_N^T\Theta^{(i)}A_N$ have the same nonzero pattern for all the commodities. IPM uses the sparse Cholesky package by E. Ng and B. Peyton [24]. For large networks these routines provided significantly better solution times than alternative ones like the Sparspak package [14]—although both share the same minimum-degree-ordering heuristic for computing the permutation of the nodes of the network. Note that, unlike general interior-point methods, the main computational burden is not the Cholesky factorizations but the repeated forward and backward substitutions. Indeed, in practice, and due to the (large) number of PCG iterations, these substitutions represent about 60% of the execution time, whereas the factorizations amount to no more than 20%. (For instance, in problems M_5, M_{11}, PDS10, and PDS30 of Tables 3 and 4 of section 6, these figures were 45.3/0.4, 47.4/4.5, 48.8/2.8, and 65.7/18.7, respectively.) Since the Ng–Peyton package concentrates its effort on the factorization stage, it may be possible to improve the performance of the algorithm by either using or developing a Cholesky solver focused on the solution phase.

5.2. Accuracy of the PCG method.

The tolerance of the stopping criteria of the PCG is the most influential parameter in the overall performance of the algorithm. It determines the accuracy required to solve system (22) and, hence, the number of PCG iterations performed. We followed a similar approach to that used by Resende and Veiga in [29] for single-commodity network problems. At iteration i of the interior-point method, we consider that the jth PCG iterate dy_2^j solves (22) if

$$1 - \cos(Sdy_2^j, \bar{b}_2 - C^T B^{-1}\bar{b}_1) < \epsilon_i, \tag{30}$$

ϵ_i being the PCG tolerance parameter. This tolerance is dynamically updated as

$$\epsilon_i = 0.95\epsilon_{i-1}, \tag{31}$$

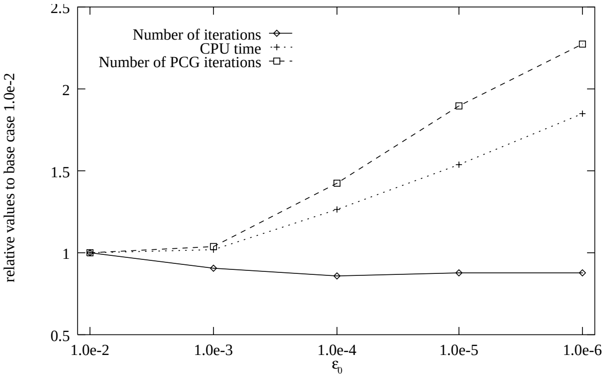


FIG. 8. *CPU time, PCG iterations, and primal-dual iterations for different* ϵ_0 *(problem* M_4*).*

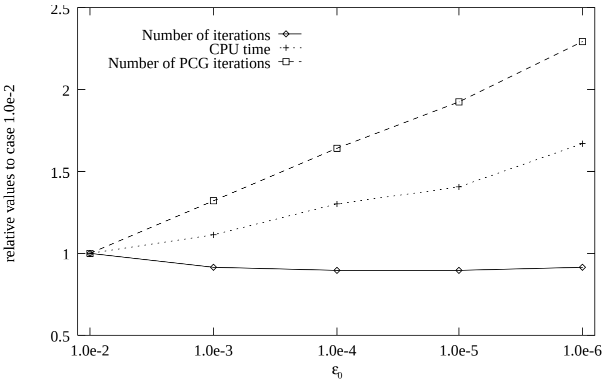


FIG. 9. *CPU time, PCG iterations, and primal-dual iterations for different* ϵ_0 *(problem PDS5).*

which guarantees better dy_2 directions as we get closer to the solution. By default IPM uses an initial tolerance of $\epsilon_0 = 10^{-2}$. Smaller ϵ_0 values provide better movement directions, which reduce the sequence of primal-dual points but considerably increase the number of PCG iterations. On the other hand, if large ϵ_0 are used, the primal-dual algorithm can fail to converge. The default value of 10^{-2} was good enough to solve most of the problems tested (only a few required $\epsilon_0 = 10^{-3}$) and provided the best execution times. For instance, Figures 8 and 9 show the CPU time and the number of PCG and primal-dual iterations required to solve problems M_4 and PDS5 of Tables 3 and 4 of section 6, respectively, for different ϵ_0 values (all data are relative to the base case $\epsilon_0 = 10^{-2}$). Though both problems are very different (M_4 has a much smaller network but three times more commodities), the behavior of IPM was almost the same: for small ϵ_0 values the CPU time and PCG iterations increased significantly whereas the primal-dual iterations hardly decreased.

In our computational experience, we have seen that the ϵ_0 value slightly affects the precision of the optimizer provided. In general, IPM stops with a point where the dual infeasibilities, computed as

$$\frac{\|A^T y + z - w - c\|_2}{1 + \|c\|_2},$$

are about 10^{-6}, regardless of whether ϵ_0 was chosen. In most tests performed, the primal infeasibilities

$$\frac{\|Ax - b\|_2}{1 + \|b\|_2}$$

were about 10^{-5} for $\epsilon_0 = 10^{-2}$ and 10^{-6} for $\epsilon_0 = 10^{-6}$. The gain of this digit in the primal accuracy was at the expense of approximately doubling the solution time. IPM stops when the relative duality gap of the current iterate

$$\frac{|c^T x - (b^T y - \overline{x}^T w)|}{1 + |c^T x|} \tag{32}$$

is less than an optimality tolerance, by default set to 10^{-6}. Unlike general interior-point solvers, and due to the use of a PCG, it is difficult to obtain more accurate solutions. One possible way of overcoming this drawback would be to develop a procedure for detecting the optimal face once we are close to the optimizer. Such a strategy already exists for network linear programs [30], but, to the best of our knowledge, there is not an equivalent result for multicommodity flows. In connection to this, the inclusion of a crossover procedure is also part of further work to be done on IPM.

5.3. Removing inactive mutual capacity constraints. The dimension of the Schur complement S is the number of mutual capacity constraints n. Should we have a procedure for detecting the inactive constraints, these could be removed, thus reducing the computational effort required by (22). IPM implements two kinds of strategy for the detection of inactive constraints. The first one is applied at the beginning, as a preprocessing stage. The second follows the suggestions in [16] and consists of detecting the inactive mutual capacity constraints during the execution of the algorithm, using the complementarity condition $y_j s_{mc_j} = 0$, where y_j and s_{mc_j} denote the dual variable and primal slack of the jth mutual capacity constraint. At iteration i of the primal-dual algorithm, we will remove the mutual capacity constraint of arc a_j if

$$y_j^i \approx 0 \quad \text{and} \quad s_{mc_j}^i \gg 0.$$

IPM implements these conditions as $|y_j^i| < 0.01$ and $s_{mc_j}^i > 0.1 b_{mc_j}$. This removal is only active when the relative duality gap (32) is less than 1.0 (primal and dual functions agree in one figure), in an attempt to guarantee that the current iterate is sufficiently close to the optimizer. Note that, unlike general interior-point solvers, removing mutual capacity constraints does not imply any additional symbolic refactorization.

5.4. Starting point. Following the suggestions in [5], an initial estimate of the primal variables is computed by solving

$$\begin{aligned} \min \quad & c^T x + \frac{\rho}{2}\,(x^T x + (\overline{x} - x)^T (\overline{x} - x)) \\ \text{subject to} \quad & Ax = b, \end{aligned}$$

with $|\rho| = 100$, yielding

$$\lambda = (AA^T)^{-1}\left(\frac{b}{2\rho} + A\left(c - \frac{\overline{x}}{\rho}\right)\right),$$

$$x = \frac{1}{2}\left(\overline{x} + \frac{A^T\lambda - c}{\rho}\right).$$

Thereafter, the components x_i out of bounds are replaced by $\min\{\overline{x}_i/2, 100\}$.

Dual estimators are obtained from the dual feasibility and complementarity slackness conditions

$$\begin{aligned} (A^T y)_i + z_i - w_i &= c_i, \\ x_i z_i &= \mu_0, \\ (\overline{x}_i - x_i) w_i &= \mu_0, \end{aligned} \tag{33}$$

where μ_0 is set to a large value (e.g., 100). Dual variables are initialized as $y = 0$. Dual slacks are computed from (33), yielding

$$z_i = \frac{\mu_0}{\overline{x}_i} + \frac{c_i}{2} + \sqrt{\frac{\mu_0^2}{\overline{x}_i^2} + \frac{c_i^2}{4}},$$

$$w_i = \frac{\mu_0 z_i}{\overline{x}_i z_i - \mu_0}.$$

Note that for $\mu_0 > 0$ the above equations provide strictly positive values for w and z.

6. Computational results.

To test the performance of the algorithm, IPM has been compared with the NetOpt routine of CPLEX 4.0 [10] (which uses the solution to k minimum-cost network problems, one for each commodity, as a warm start of a dual simplex solver), and with PPRN [6], a primal partitioning code for linear and nonlinear multicommodity flows. For all the three codes, we used the default tolerances. All runs were carried out on a Sun/Ultra2 2200 workstation with 200 MHz clock, 256 Mbytes of main memory, ≈68 Mflops Linpack, 14.7 Specfp95, and 7.8 Specint95.

For the comparison, we considered three kinds of problem. The first one was obtained from the meta-generator Dimacs2pprn (see [6]). This meta-generator requires a previous minimum-cost network flow problem that is converted to a multicommodity one. It can be obtained from ftp://ftp-eio.upc.es/pub/onl/codes/pprn/tests (an enhanced version is described in [13]). We used four minimum-cost network generators from the DIMACS suite [11]: Rmfgen (D. Goldfarb and M. Grigoriadis), Grid-on-Torus (A. V. Goldberg), Gridgraph (M. G. C. Resende), and Gridgen (Y. Lee and J. Orlin). They are freely distributed and can be obtained via anonymous ftp from dimacs.rutgers.edu at directory /pub/netflow. We generated two kinds of problem for each generator: with few commodities (small problems) and with many commodities (large problems). The small problems are represented by S_k^i, where $i = 1, \dots, 4$ denotes the DIMACS generator used (1 = Rmfgen, 2 = Grid-on-Torus, 3 = Gridgraph, 4 = Gridgen) and $k \in \{1, 4, 8, 16, 50, 100, 150, 200\}$ is the number of commodities considered. The large problems are called L_k^i, where i and k have the same meaning as

Table 1
Dimensions and results obtained for the small Dimacs2pprn problems.

						CPU time (seconds)				
Pr.	m	n	k	$\tilde{n}$	$\tilde{m}$	IPM	CPLEX	PPRN	f^*	$\frac{f^*-f^*_{\mathrm{IPM}}}{1+f^*}$
S^1_1	2472	9048	1	9048	2472	37.	1.	1.	375675.2	−4.5e−6
S^1_4	2472	9048	4	45240	18936	273.	8.	9.	2027285.0	−5.2e−6
S^1_8	2472	9048	8	81432	28824	1058.	28.	149.	4506263.3	−5.4e−6
S^1_{16}	2472	9048	16	153816	48600	1377.	73.	1166.	9870432.8	−7.8e−6
S^1_{50}	128	496	50	25296	6896	17.	21.	39.	11839382.2	3.9e−5
S^1_{100}	128	496	100	50096	13296	76.	258.	465.	27150952.6	2.9e−5
S^1_{150}	128	496	150	74896	19696	137.	681.	1245.	39835825.1	8.3e−6
S^1_{200}	128	496	200	99696	26096	174.	1204.	2368.	54343948.3	−5.6e−6
S^2_1	1500	9000	1	9000	1500	28.	1.	1.	36896.8	–1.5e–6
S^2_4	1500	9000	4	45000	15000	623.	22.	85.	187962.0	–5.2e–6
S^2_8	1500	9000	8	81000	21000	2499.	647.	814.	1197048.8	4.9e–6
s^2_{16}	1500	9000	16	153000	33000	7550.	12872.	6721.	5876840.3	1.8e–5
S^2_{50}	100	600	50	30600	5600	28.	14.	21.	5207622.7	2.2e–6
S^2_{100}	100	600	100	60600	10600	78.	110.	222.	12922703.9	1.4e–5
S^2_{150}	100	600	150	90600	15600	263.	652.	1137.	22663204.t	–5.9e–6
S^2_{200}	100	600	200	120600	20600	565.	2393.	3495.	36829147.5	1.3e–5
S^3_1	2502	5000	1	5000	2502	2.	1.	1.	94212753.2	–2.5e–6
S^3_4	2502	5000	4	25000	15008	184.	55.	118.	355884986.5	–3.8e–6
S^3_8	2502	5000	8	45000	25016	247.	85.	215.	128743093.7	9.2e–5
S^3_{16}	2502	5000	16	8500	45032	956.	1171.	2666.	253615755.9	8.3e–5
S^3_{50}	227	450	50	22950	11800	60.	21.	56.	27853327.9	6.0e–5
S^3_{100}	227	450	100	45450	23150	173.	290.	670.	65144564.1	6.8e–5
S^3_{150}	227	450	150	67950	34500	104.	144.	745.	27066715.3	4.5e–6
S^3_{200}	227	450	200	90450	45850	247.	550.	1922.	37964963.8	1.6e–5
S^4_1	976	7808	1	7808	976	25.	1.	1.	5541980.3	–5.5e–7
S^4_4	976	7808	4	39040	11712	747.	(a)	41.	23223474.9	–2.6e–6
S^4_8	976	7808	8	70272	15616	4079.	(a)	497.	61792270.7	–4.9e–9
S^4_{16}	976	7808	16	132736	23424	5509.	(a)	6466.	165808232.3	9.1e–5
S^4_{50}	101	606	50	30906	5656	14.	5.	4.	1409470.3	2.2e–5
S^4_{100}	101	606	100	61206	10706	39.	16.	25.	2940217.3	–1.6e–6
S^4_{150}	101	606	150	91506	15756	68.	38.	58.	4614971.4	3.2e–6
S^4_{200}	101	606	200	121806	20806	121.	126.	189.	6440385.6	2.6e–6

(a) Problem reported as infeasible by the solver.

before. (In this case, however, the number of commodities is always greater than 200.) Tables 1 and 2 show the dimensions of these problems. Column Pr. is the name of the problem. Columns m, n, and k show the number of nodes, arcs, and commodities, respectively. Columns $\tilde{n}$ and $\tilde{m}$ give the number of variables and constraints of the linear problem (where $\tilde{n} = (k+1)n$ and $\tilde{m} = km + n$). Columns IPM, CPLEX, and PPRN correspond to the CPU time, in seconds, required by each code to solve the problem. Finally, column f^* gives the optimal objective function value provided by both CPLEX and PPRN, whereas column $\frac{f^*-f^*_{\mathrm{IPM}}}{1+f^*}$ shows the relative error of the solution provided by IPM.

Table 2
Dimensions and results obtained for the large Dimacs2pprn problems.

						CPU time (seconds)				
Pr.	m	n	k	$\tilde{n}$	$\tilde{m}$	IPM	CPLEX	PPRN	f^*	$\frac{f^*-f^*_{\mathrm{IPM}}}{1+f^*}$
L^1_{200}	128	496	200	99400	26096	50.	49.	254.	43496063.1	–2.5e–6
L^1_{400}	128	496	400	198800	51696	140.	319.	2092.	89227358.5	–5.3e–7
L^1_{600}	128	496	600	298200	77296	242.	1328.	6590.	135813634.7	–1.3e–6
L^1_{800}	128	496	800	397600	102896	278.	3001.	13428.	184848693.7	–1.3e–6
L^1_{1000}	128	496	1000	497000	128496	363.	6006.	25813.	235407084.7	–1.2e–6
L^1_{1200}	128	496	1200	596400	154096	546.	11887.	43946.	287243145.4	2.7e–5
L^1_{1400}	128	496	1400	695800	179696	756.	20080.	78800.	339708251.9	7.7e–6
L^2_{200}	80	500	200	100200	16500	71.	92.	278.	1372096.3	3.1e–6
L^2_{400}	80	500	400	200400	32500	522.	2358.	4647.	7004937.6	4.6e–6
L^2_{500}	80	500	500	250500	40500	905.	5395.	10259.	11941741.3	8.1e–6
L^2_{600}	80	500	600	300600	48500	885.	10778.	20478.	17857546.4	1.6e–5
L^3_{200}	242	472	200	94600	48872	59.	71.	387.	8153455.3	9.5e–6
L^3_{400}	242	472	400	189200	97272	202.	685.	3367.	16715597.6	6.2e–6
L^3_{500}	242	472	500	236500	121472	319.	1968.	8025.	21219420.2	1.9e–6
L^3_{600}	242	472	600	283800	145672	384.	3256.	14614.	25646734.6	1.5e–5
L^4_{200}	151	1208	200	241800	31408	310.	104.	231.	1690360.3	–9.7e–7
L^4_{300}	151	1208	300	362700	46508	537.	365.	893.	2614303.6	–4.3e–6
L^4_{400}	151	1208	400	483600	61608	805.	673.	2195.	3389601.0	7.4e–7

The second kind of problems were obtained with A. Frangioni's [13] C version of Ali and Kennington's Mnetgen generator [4]. It can be freely obtained from http://www.di.unipi.it/di/groups/optimize/Data/MMCF.html. We generated 24 problems with different dimensions. They can all be considered difficult problems, since they have a "dense" network (the ratio "number of arcs/number of nodes" is 8), 80% of the arcs have a mutual capacity, 30% of the arcs have a high cost, and 90% of the arcs have individual capacities for each commodity. The parameters used for generating the instances can be found in [13]. The problems obtained with this generator will be denoted as M_i, $i = 1, \dots, 24$. Table 3 shows the dimensions of these tests, where the columns have the same meaning as in Tables 1 and 2.

The last type of problems corresponds to the PDS instances [7]. These problems arise from a model for evacuating patients from a place of military conflict. Each instance depends on a parameter t that denotes the planning horizon under study (in number of days). The size of the network increases with t, whereas the number of commodities is always 11. Problems obtained with this generator are denoted as PDSt, where t is the number of days considered. Their dimensions are shown in Table 4. The meaning of the columns is the same as in previous tables. The largest problems were not solved with CPLEX due to the amount of time required. The PDS problems can be retrieved from http://www.di.unipi.it/di/groups/optimize/Data/MMCF.html.

Tables 1–4 show that IPM and PPRN solved all the problems, whereas CPLEX exited with an infeasibility message in three of the small Dimacs2pprn tests and was not run for the largest PDS. IPM solved most of the problems using the default initial PCG tolerance of $\epsilon_0 = 10^{-2}$. Only in three cases (problems S^3_{16}, L^2_{200}, and L^2_{500}) did this value have to be reduced to 10^{-3} to guarantee the convergence. In general

Table 3
Dimensions and results obtained for the Mnetgen problems.

Pr.	m	n	k	$\tilde{n}$	$\tilde{m}$	CPU time (seconds) IPM	 CPLEX	 PPRN	f^*	$\frac{f^*-f^*_{\text{IPM}}}{1+f^*}$
M_1	64	524	4	2100	780	1.	0.	0.	192400.1	–6.3e–7
M_2	64	532	8	4264	1044	2.	1.	1.	394051.1	4.1e–6
M_3	64	497	16	7968	1521	5.	2.	4.	1071474.9	1.0 e–5
M_4	64	509	32	16320	2557	12.	13.	19.	2146944.1	1.0e–5
M_5	64	511	64	32768	4607	91.	141.	136.	4623138.5	8.1e–6
M_6	128	997	4	3992	1509	2.	0.	1.	919643.2	1.5e–6
M_7	128	1089	8	8720	2113	8.	1.	4.	1924133.9	–6.7e–7
M_8	128	1114	16	17840	3162	25.	13.	34.	4145079.4	6.0e–6
M_9	128	1141	32	36544	5237	155.	214.	478.	9785961.1	6.3e–6
M_{10}	128	1171	64	76115	9363	485.	1647.	3419.	19269824.2	–3.9e–6
M_{11}	128	1204	128	154240	17588	549.	7880.	9334.	40143200.8	9.2e–6
M_{12}	256	2023	4	8096	3047	12.	1.	7.	5026132.3	1.4e–5
M_{13}	256	2165	8	17328	4213	40.	13.	69.	9919483.2	–2.1e–6
M_{14}	256	2308	16	36944	6404	146.	158.	769.	20692883.7	6.9e–6
M_{15}	256	2314	32	74080	10506	465.	1664.	7610.	45671076.1	–1.4e–6
M_{16}	256	2320	64	148544	18704	1040.	9235.	27722.	92249381.1	–1.2e–6
M_{17}	256	2358	128	301952	35126	3742.	45990.	84066.	190137259.9	–7.8e–6
M_{18}	256	2204	256	564480	67740	9187.	181701.	169810.	397882591.3	–1.4e–6
M_{19}	512	4077	4	16312	6125	99.	7.	85.	21324851.2	-7.3e–6
M_{20}	512	4373	8	34992	8469	190.	101.	654.	46339269.9	1.6e–5
M_{21}	512	4620	16	73936	12812	1582.	1457.	7279.	96992237.2	–4.5e–6
M_{22}	512	4646	32	148704	21030	2644.	8302.	73439.	192941834.8	-7.0e–7
M_{23}	512	4768	64	305216	37536	7411.	55028.	178188.	412943158.7	8.9e–8
M_{24}	512	4786	128	612736	70322	21263.	289541.	947790.	828013599.8	–1.3e–6

IPM required no more than 100 iterations to achieve a point with a dual relative gap less than 10^{-6}—the default optimality tolerance. We can also see that the solution provided by IPM can be considered good enough: the relative error in the objective function (last column of Tables 1–4) fluctuates between 10^{-5} and 10^{-7} (the worst case corresponds to problem PDS40, with a relative error of $1.5 \cdot 10^{-4}$). These results are more accurate (two more exact figures in the objective function) than those provided in [17] and [31] for the largest PDS instances. For instance, Table 5 summarizes the results presented by Grigoriadis and Khachiyan in [17] and by Schultz and Meyer in [31] for the largest PDS problems they solved using their ϵ-approximation and barrier decomposition methods, respectively. Columns $f^*_{\epsilon\text{A}}$ and f^*_{BD} give the optimal objective function provided by each method, whereas columns $\frac{f^*-f^*_{\epsilon\text{A}}}{1+f^*}$ and $\frac{f^*-f^*_{\text{BD}}}{1+f^*}$ show the relative errors of the solutions obtained. Looking at Tables 4 and 5, we see that IPM provided two and three more significant figures in all the problems. Indeed, as stated by the authors in [17], ϵ-approximation methods are practical for computing fast approximations to large instances, whereas the solutions provided by IPM can be considered almost optimal.

Figures 10, 11, 12, and 13 show the ratio of the CPU time of CPLEX and PPRN to IPM (i.e., "CPLEX CPU time/IPM CPU time" and "PPRN CPU time/IPM CPU time") for the problems in Tables 1–4. The executions are ordered by the number

Table 4
Dimensions and results obtained for the PDSt problems.

						CPU time (seconds)			
$t^{(a)}$	m	n	$\tilde{n}$	$\tilde{m}$	IPM	CPLEX	PPRN	f^*	$\frac{f^*-f^*_{\mathrm{IPM}}}{1+f^*}$
1	126	372	4464	1758	1.	0.	0.	29083930523.	2.8e–6
2	252	746	8952	3518	4.	1.	2.	28857862010.	–5.9e–6
3	390	1218	14616	5508	8.	2.	5.	28597374145.	–4.9e–6
4	541	1790	21480	7741	16.	4.	10.	28341928581.	–1.5e–6
5	686	2325	27900	9871	29.	8.	19.	28054052607.	–6.6e–6
6	835	2827	33924	12012	47.	13.	35.	27761037600.	1.3e–5
7	971	3241	38892	13922	46.	20.	52.	27510377013.	2.1e–5
8	1104	3629	43548	15773	45.	31.	75.	27239627210.	2.1e–5
9	1253	4205	50460	17988	94.	51.	84.	26974586241.	1.5e–5
10	1399	4792	57504	20181	88.	56.	136.	26727094976.	8.4e–6
11	1541	5342	64101	22293	144.	98.	178.	26418289612.	1.3e–5
12	1692	5965	71580	24577	113.	143.	188.	26103493922.	7.4e–5
13	1837	6571	78852	26778	137.	160.	328.	25825886804.	4.6e–5
14	1981	7151	85812	28942	180.	270.	342.	25529159469.	3.4e–5
15	2125	7756	93072	31131	236.	425.	564.	25177923601.	3.3e–5
18	2558	9589	115068	37727	396.	864.	1227.	24332411902.	6.4e–6
20	2857	10858	130296	42285	386.	1830.	2138.	23821658640.	7.0e–5
21	2996	11401	136812	44357	529.	1912.	2322.	23576150674.	2.6e–5
24	3419	13065	156780	50674	963.	4393.	3411.	22856729593.	5.1e–7
27	3823	14611	175332	56664	1010.	7178.	4810.	22133391961.	1.9e–5
30	4223	16148	193776	62601	1325.	24905.	6827.	21385445736.	–1.7e–6
33	4643	17840	214080	68913	1750.	35397.	9154.	20589962883.	1.4e–5
36	5081	19673	236076	75564	1346.	44144.	12704.	19857712721.	4.4e–5
40	5652	22059	264708	84231	1494.	95064.	16779.	18855198824.	1.5e–4
50	7031	27668	332016	105009	4166.	85840.	46664.	16603525724.	3.5e–5
60	8423	33388	400656	126041	6761.	387577.	75880.	14265904407.	2.4e–6
70	9750	38396	460752	145646	12210.	540606.	112310.	12241162812.	2.0e–5
80	10989	42472	509664	163351	13005.	—	125770.	11469077462.	3.0e–5
90	12186	46161	553932	180207	21781.	—	178248.	11087561635.	1.8e–5
100	13366	49742	596904	196768	17222.	—	214961.	10928229968.	8.8e–5

[a] $k = 11$ for all t.

of variables of the problem. The dashed line of the figures separates the executions according to whether IPM was outperformed or not. For the small Dimacs2pprn problems (Figure 10) both CPLEX and PPRN provided better times than IPM, particularly in the smaller instances. In some cases they were 50 and 33 times faster than IPM, respectively. However, for the large Dimacs2pprn cases (Figure 11), IPM provided the best executions and was up to 26 and 100 times more efficient than CPLEX and PPRN. However, the Dimacs2pprn problems are not very complicated, in spite of the large number of variables. This explains the moderate CPU times required by the three codes in their solution. On the other hand, the Mnetgen and PDS instances (Figures 12 and 13) can be considered to be difficult. It is in these situations that IPM clearly outperforms both CPLEX and PPRN. For the Mnetgen problems it was, on average, 4 times faster than CPLEX (20 in the best case) and

Table 5
Results reported in [17] *and* [31] *for some of the largest PDSt problems.*

t	ϵ-approximation		barrier decomposition	
	$f^*_{\epsilon\mathrm{A}}$	$\frac{f^*-f^*_{\epsilon\mathrm{A}}}{1+f^*}$	f^*_{BD}	$\frac{f^*-f^*_{\mathrm{BD}}}{1+f^*}$
50	$1.66257{\cdot}10^{10}$	-1.3e–3	$1.6625{\cdot}10^{10}$	-1.3e–3
60	$1.42914{\cdot}10^{10}$	-1.8e–3	$1.4462{\cdot}10^{10}$	-1.4e–2
70	$1.22640{\cdot}10^{10}$	-1.9e–3	$1.2311{\cdot}10^{10}$	-5.7e–3
80	$1.15047{\cdot}10^{10}$	-3.1e–3	—	—

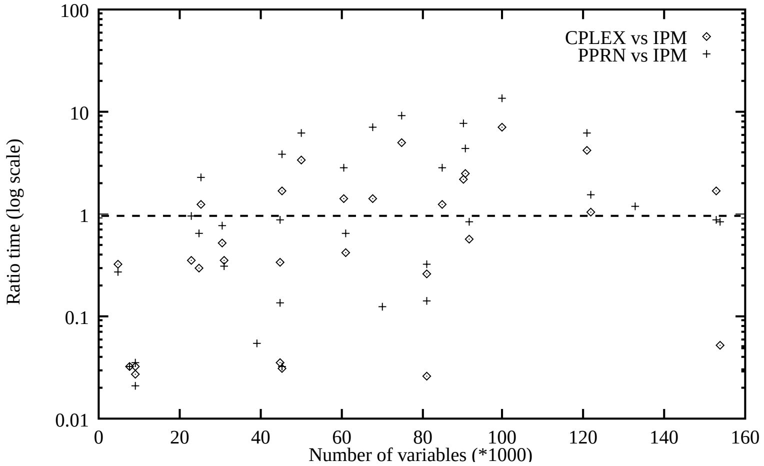


Fig. 10. *Ratio time of CPLEX and PPRN to IPM for the small Dimacs2pprn problems.*

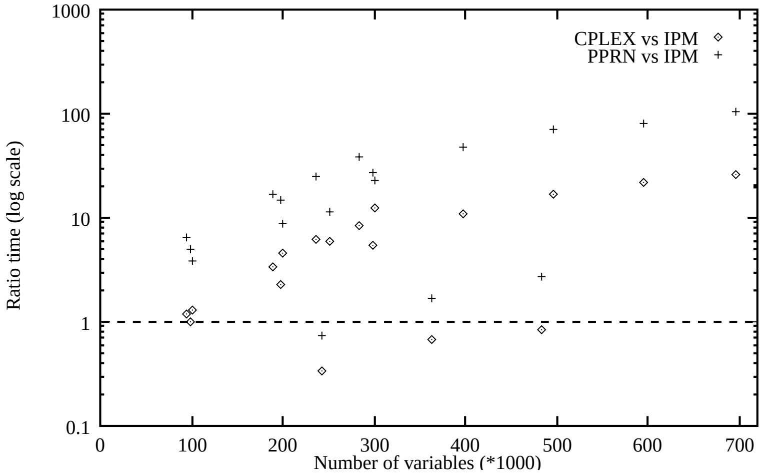


Fig. 11. *Ratio time of CPLEX and PPRN to IPM for the large Dimacs2pprn problems.*

10 times faster than PPRN (45 in the best run). These average figures were 11 for CPLEX and 4 for PPRN when solving the PDS problems (the maximum ratios were of 67 and 12, respectively). It should be pointed out that IPM performed best for the large problems, as indicated by the positive slope of the points in Figures 12 and 13 (note that a log scale is used for the vertical axis). This is especially true for the big

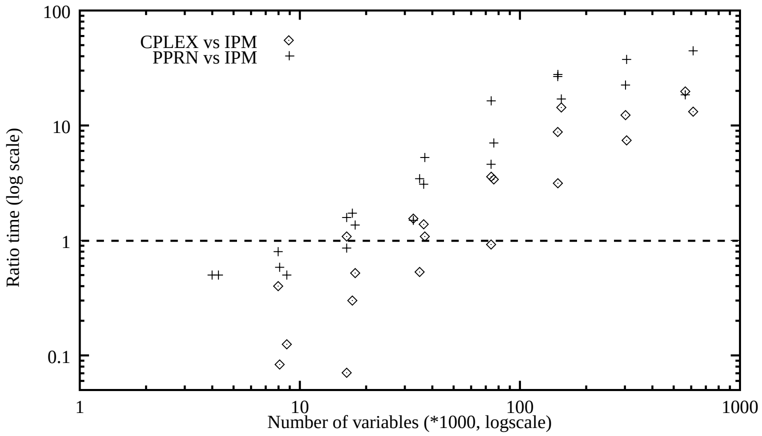


FIG. 12. *Ratio time of CPLEX and PPRN to IPM for the Mnetgen problems.*

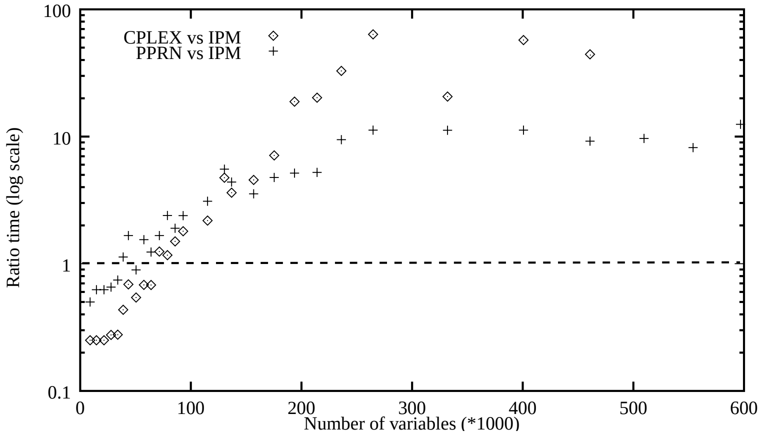


FIG. 13. *Ratio time of CPLEX and PPRN to IPM for the PDS problems.*

Mnetgen problems, where IPM was consistently faster than both CPLEX and PPRN. For the large PDS tests (e.g., $t > 30$) PPRN behaved very well and was only about 11 times slower than IPM, whereas CPLEX provided poorer performances.

Finally, we compared IPM with the CPLEX barrier solver, a state-of-the-art implementation of a general interior-point algorithm. For the comparison, we solved the small Dimacs2pprn problems S_k^i of Table 1. The remaining problems were not executed due to the excessive CPU time the CPLEX barrier solver would take. Table 6 shows the results. The last column gives the ratio time between the CPLEX barrier solver and IPM. The runs not reported correspond to cases where either the system memory was insufficient or the program was stopped because of an excessive execution time. Figure 14 shows the ratio times for the number of variables of the problem. It can be seen that only in some of the smaller problems did the general interior-point code slightly outperform the specialized one. As the size of the problem increases, IPM performs better and is up to 800 times faster in the best case (i.e., S_{100}^4).

7. Conclusions. From the computational experiments reported, it can be stated that the specialized interior-point algorithm is an efficient and promising tool for the

TABLE 6
Performance comparison of IPM and CPLEX (barrier solver).

	CPU time (seconds)		Ratio		CPU time (seconds)		Ratio
Prob.	IPM	CPLEX	time	Prob.	IPM	CPLEX	time
S^1_1	36.6	35.6	0.97	S^3_1	2.0	2.6	1.28
S^1_4	273.3	2865.7	10.49	S^3_4	183.6	64.7	0.35
S^1_8	1057.8	29445.6	27.84	S^3_8	246.7	548.6	2.22
S^1_{16}	1377.1	—	—	S^3_{16}	956.1	16290.0	17.04
S^1_{50}	17.2	995.5	57.91	S^3_{50}	59.8	213.0	3.56
S^1_{100}	76.3	3147.7	41.27	S^3_{100}	172.7	462.9	2.68
S^1_{150}	136.8	6684.8	48.86	S^3_{150}	103.6	623.0	6.01
S^1_{200}	173.9	13777.1	79.22	S^3_{200}	246.9	1254.3	5.08
S^2_1	28.1	28.7	1.02	S^4_1	24.8	23.3	0.94
S^2_4	622.6	2467.0	3.96	S^4_4	747.3	1697.4	2.27
S^2_8	2498.7	23289.5	9.32	S^4_8	4078.8	17400.7	4.27
S^2_{16}	7550.3	—	—	S^4_{16}	5508.8	—	—
S^2_{50}	27.6	1195.0	43.22	S^4_{50}	14.1	5409.8	383.67
S^2_{100}	77.8	4263.7	54.78	S^4_{100}	39.0	32760.9	839.81
S^2_{150}	262.6	7584.8	28.89	S^4_{150}	68.3	—	—
S^2_{200}	565.2	6095.8	10.79	S^4_{200}	120.7	—	—

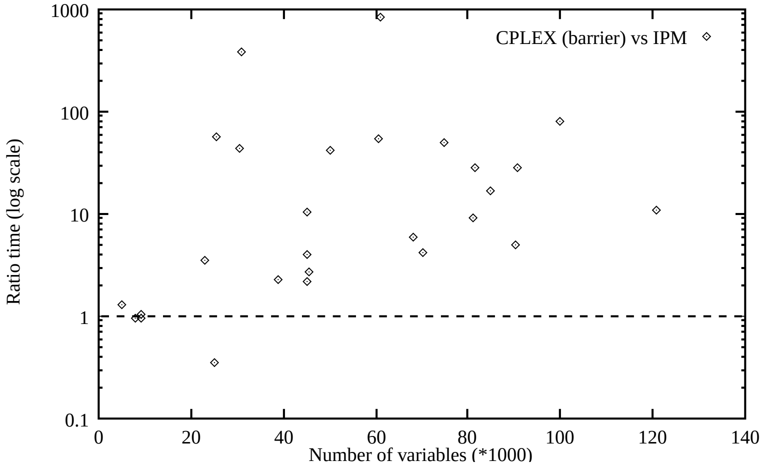


FIG. 14. *Ratio time of CPLEX (barrier solver) to IPM for the small Dimacs2pprn problems.*

solution of large and difficult multicommodity problems. However, the algorithm can still be improved with additional refinements. These include optimal face detection and crossover procedures, improvement in the accuracy of the solution provided by the PCG, and a more appropriate Cholesky factorization solver to reduce the time spent by the forward and backward substitutions. A coarse-grain parallel implementation of the algorithm should also be developed in the future.

Acknowledgments. The author is indebted to N. Nabona, A. Frangioni, and one of the referees for helpful comments and suggestions. The author also thanks Cs. Mészaros for providing a version of BPMPD that reports sparsity pattern information, E. Ng for delivering the Ng–Peyton sparse Cholesky package, and, again, A. Frangioni for the PDS and Mnetgen test instances.

REFERENCES

[1] I. Adler, M.G.C. Resende, and G. Veiga, *An implementation of Karmarkar's algorithm for linear programming*, Math. Programming, 44 (1989), pp. 297–335.

[2] R.K. Ahuja, T.L. Magnanti, and J.B. Orlin, *Network Flows*, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[3] A. Ali, R.V. Helgason, J.L. Kennington, and H. Lall, *Computational comparison among three multicommodity network flow algorithms*, Oper. Res., 28 (1980), pp. 995–1000.

[4] A. Ali and J.L. Kennington, *Mnetgen Program Documentation*, Technical Report 77003, Department of Industrial Engineering and Operations Research, Southern Methodist University, Dallas, TX, 1977.

[5] E.D. Andersen, J. Gondzio, C. Mészáros, and X. Xu, *Implementation of interior point methods for large scale linear programming*, in Interior Point Methods in Mathematical Programming, T. Terlaky, ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996, pp. 189–252.

[6] J. Castro and N. Nabona, *An implementation of linear and nonlinear multicommodity network flows*, European J. Oper. Res., 92 (1996), pp. 37–53.

[7] W.J. Carolan, J.E. Hill, J.L. Kennington, S. Niemi, and S.J. Wichmann, *An empirical evaluation of the KORBX algorithms for military airlift applications*, Oper. Res., 38 (1990), pp. 240–248.

[8] P. Chardaire and A. Lisser, *Simplex and interior point specialized algorithms for solving non-oriented multicommodity flow problems*, Oper. Res., accepted subject to revision.

[9] I.C. Choi and D. Goldfarb, *Solving multicommodity network flow problems by an interior point method*, in Large-Scale Numerical Optimization, T.F. Coleman and Y. Li, eds., SIAM, Philadelphia, PA, 1990, pp. 58–69.

[10] CPLEX Optimization Inc., *Using the CPLEX Callable Library*, Incline Village, NV, 1995.

[11] DIMACS, *The First DIMACS International Algorithm Implementation Challenge: The Benchmark Experiments*, Technical Report, DIMACS, New Brunswick, NJ, 1991.

[12] A. Frangioni, *personal communication*, Department of Computer Science, Universitá di Pisa, Pisa, Italy, 1998.

[13] A. Frangioni and G. Gallo, *A bundle type dual-ascent approach to linear multicommodity min cost flow problems*, INFORMS J. Comput., 11 (1999), pp. 370–393.

[14] J.A. George and J.W. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[15] J.-L. Goffin, J. Gondzio, R. Sarkissian, and J.-P. Vial, *Solving nonlinear multicommodity flow problems by the analytic center cutting plane method*, Math. Programming, 76 (1996), pp. 131–154.

[16] J. Gondzio and M. Makowski, *Solving a class of LP problems with a primal-dual logarithmic barrier method*, European J. Oper. Res., 80 (1995), pp. 184–192.

[17] M.D. Grigoriadis and L.G. Khachiyan, *An exponential-function reduction method for block-angular convex programs*, Networks, 26 (1995), pp. 59–68.

[18] A.P. Kamath, N.K. Karmarkar, and K.G. Ramakrishnan, *Computational and Complexity Results for an Interior Point Algorithm on Multicommodity Flow Problems*, Technical Report TR-21/93, Dip. di Informatica, Univ. di Pisa, Italy, 1993, pp. 116–122.

[19] A.P. Kamath and O. Palmon, *Improved Interior Point Algorithms for Exact and Approximate Solutions of Multicommodity Flow Problems*, Technical Report, Dept. of Computer Sciences, Stanford University, Stanford, CA, 1994.

[20] S. Kapoor and P.M. Vaidya, *Speeding up Karmarkar's algorithm for multicommodity flows*, Math. Programming, 73 (1996), pp. 111–127.

[21] J.L. Kennington and R.V. Helgason, *Algorithms for Network Programming*, Wiley, New York, 1980.

[22] I.J. Lustig, R.E. Marsten, and D.F. Shanno, *On implementing Mehrotra's predictor-corrector interior-point method for linear programming*, SIAM J. Optim., 2 (1992), pp. 435–449.

[23] Cs. Mészáros, *The Efficient Implementation of Interior Point Methods for Linear Programming and Their Applications*, Ph.D. Thesis, Eötvös Loránd University of Sciences, Budapest, Hungary, 1996.

[24] E. Ng and B.W. Peyton, *Block sparse Cholesky algorithms on advanced uniprocessor computers*, SIAM J. Sci. Comput., 14 (1993), pp. 1034–1056.

[25] J.M. Ortega, *Introduction to Parallel and Vector Solution of Linear Systems*, Plenum Press, New York, 1988.

[26] L. Portugal, M.G.C. Resende, G. Veiga, G., and J. Júdice, *A truncated primal-infeasible*

dual-feasible network interior point method, Networks, 35 (2000), pp. 91–108.

[27] L. Portugal, M.G.C. Resende, G. Veiga, G., and J. Júdice, *A truncated interior-point method for the solution of minimum cost flow problems on an undirected multicommodity flow network*, in Proceedings of the First Portuguese National Telecommunications Conference, Aveiro, Portugal, 1997, pp. 381–384 (in Portuguese).

[28] M.G.C. Resende and P. Pardalos, *Interior point algorithms for network flow problems*, in Advances in Linear and Integer Programming, J.E. Beasley, ed., Oxford University Press, New York, 1996, pp. 149–189.

[29] M.G.C. Resende and G. Veiga, *An implementation of the dual affine scaling algorithm for minimum-cost flow on bipartite uncapacitated networks*, SIAM J. Optim., 3 (1993), pp. 516–537.

[30] M.G.C. Resende, T. Tsuchiya, and G. Veiga, *Identifying the optimal face of a network linear program with a globally convergent interior point method*, in Large Scale Optimization: State of the Art, W. Hager, D. Hearn, and P. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 362–387.

[31] G.L. Schultz and R.R. Meyer, *An interior point method for block angular optimization*, SIAM J. Optim., 1 (1991), pp. 583–602.

[32] S.J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, PA, 1996.

[33] S. Zenios, *A smooth penalty function algorithm for network-structured problems*, European J. Oper. Res., 83 (1995), pp. 220–236.

SIAM J. Optim.
Vol. 10, No. 3, pp. 878–895

REFORMULATION OF VARIATIONAL INEQUALITIES ON A SIMPLEX AND COMPACTIFICATION OF COMPLEMENTARITY PROBLEMS[*]

ROBERTO ANDREANI[†] AND JOSÉ MARIO MARTÍNEZ[‡]

Abstract. Many variational inequality problems (VIPs) can be reduced, by a compactification procedure, to a VIP on the canonical simplex. Reformulations of this problem are studied, including smooth reformulations with simple constraints and unconstrained reformulations based on the penalized Fischer–Burmeister function. It is proved that bounded level set results hold for these reformulations under quite general assumptions on the operator. Therefore, it can be guaranteed that minimization algorithms generate bounded sequences and, under monotonicity conditions, these algorithms necessarily find solutions of the original problem. Some numerical experiments are presented.

Key words. variational inequalities, complementarity, minimization algorithms, reformulation

AMS subject classifications. 90C33, 90C30

PII. S1052623499352826

1. Introduction.

We are interested in reformulations of variational inequality problems (VIPs) where the domain is a simplex. The main motivation is that variational inequalities on generalized (perhaps unbounded) boxes can be reduced to the simplex case if one knows appropriate lower bounds for each variable and a bound for the sum of the variables. The reformulations of the VIP on a simplex do not have, in principle, bounded variables. However, we will be able to show, for some reformulations, that the objective function has bounded level sets. It is worth mentioning that reformulations of complementarity problems do not have, in general, bounded level sets, unless suitable restrictions are imposed on the problem. Therefore, when one applies a general solver to such a reformulation, the risk of divergence exists, even when one knows that stationary points are solutions of the VIP.

The following example will clarify the compactification strategy. Suppose that we want to solve the nonlinear system of equations

$$F(x) = 0, \tag{1.1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ has continuous first derivatives. Usually, globally convergent algorithms for solving (1.1) rely on the unconstrained minimization problem

$$\text{Minimize } \|F(x)\|_2^2. \tag{1.2}$$

See [12, 38, 41]. (Obviously, (1.1) is a variational inequality problem where the domain is $\mathbb{R}^n$.) Most algorithms (for example, globalizations of Newton's method) have the property that every limit point of the iterates is stationary, that is,

$$\nabla \|F(x)\|_2^2 \equiv 2F'(x)^T F(x) = 0. \tag{1.3}$$

[*] Received by the editors March 3, 1999; accepted for publication (in revised form) October 8, 1999; published electronically May 2, 2000.

http://www.siam.org/journals/siopt/10-3/35282.html

[†] Department of Computer Science and Statistics, University of the State of S. Paulo (UNESP), C.P. 136, CEP 15054-000, São José do Rio Preto-SP, Brazil (andreani@nimitz.dcce.ibilce.unesp.br). This author was supported by FAPESP (grant 99/08381-4) and CNPq (grant 301115/96-6).

[‡] Department of Applied Mathematics, IMECC-UNICAMP, University of Campinas, CP 6065, 13081-970 Campinas SP, Brazil (martinez@ime.unicamp.br). This author was supported by PRONEX, FAPESP (grant 90-3724-6), CNPq, and FAEP-UNICAMP.

The most obvious drawbacks of this approach are

(1) The algorithm might converge to a stationary point that is not a solution ($F'(x)$ being singular in this case).

(2) Limit points of the generated sequence might not exist at all.

With the aim of guaranteeing the existence of limit points (and, in fact, avoiding possible overflows in the computer calculation), artificial bounds are frequently added to (1.2). In this case, the globalization procedure must use techniques of bound-constrained minimization [7, 10, 18, 19, 31, 36] and the limit points will be stationary points of the *bound-constrained* problem. Unfortunately, if an artificial constraint is active at the limit point, the limit point might not be a stationary point of (1.2) and, hence, also not a solution of (1.1). This usually happens when the sequence generated by the unconstrained algorithm applied to (1.2) tends to infinity.

In [5] it has been suggested that a better way in which the domain of (1.1) can be compactified is to consider the *variational inequality problem* defined by $F(x)$ on the domain defined by the artificial bounds. In that paper it was proved that, under suitable conditions, any *stationary* point of a smooth reformulation (with bounded level sets) of the variational inequality problem must be a *solution* of (1.1). Therefore, the results in [5] represent sufficient conditions under which neither of the two objections just exposed is problematic.

The discussion above can be repeated, with minor modifications, if the original problem is a complementarity problem instead of a nonlinear system. See [2, 3, 4, 5, 13, 24, 34, 35].

The main drawback of the approach of [5] is that the reformulation of the original problem requires $2n$ additional variables. In [4, 14] a different reformulation with the same triplicating property can be found. In the present research, we introduce reformulations with $n+3$ additional variables having similar properties as those proved in [5]. The idea, as we mentioned above, is to consider first the variational inequality problem on a smaller simplicial region.

This paper is organized as follows. In section 2 we explain how a variational inequality problem on a generalized box can be reduced to a VIP in which the domain is a simplex. In section 3 we define *smooth* reformulations of the VIP on the simplex, for which the level sets are bounded and, under suitable conditions, stationary points coincide with solutions of the variational inequality problem. In section 4 we repeat the work of section 3 with respect to an unconstrained reformulation that uses the penalized Fischer–Burmeister [8] function. See, also, [11, 14, 16, 17, 23, 27, 29, 30]. In section 5 we present numerical experiments. Conclusions are given in section 6.

2. Reduction to the simplex form.

The fact that, under certain conditions, the solution of a restricted variational inequality problem is a solution of the original one seems to be known by many researchers, although the result is not easily found in the literature. The argument is as follows.

Consider the variational inequality problem $VIP(F, \Omega)$, which consists of finding $x \in \Omega$ such that

$$\langle F(x), z - x\rangle \geq 0 \quad \forall z \in \Omega, \tag{2.1}$$

where $F : \mathbb{R}^n \to \mathbb{R}^n$ and Ω is closed and convex. Let $B \subset \mathbb{R}^n$ be closed and convex too. Denote by B' the set of interior points of B. Define $\Omega_{small} = \Omega \cap B$ and consider the variational inequality problem defined by

$$\langle F(x), z - x\rangle \geq 0 \quad \forall z \in \Omega_{small}. \tag{2.2}$$

Clearly, any solution of (2.1) that belongs to Ω_{small} will be a solution of (2.2). Let us show that, under certain conditions, every solution of (2.2) is a solution of (2.1). Denote by S_{small} the set of solutions of (2.2). Essentially, the proof of the following theorem is that given (under slightly stronger hypotheses) in [45].

THEOREM 2.1. *Assume that the set of solutions of* (2.1) *is closed. Assume, moreover, that for every solution x of* (2.2) *there exists a sequence $\{x^k\} \subset B' \cap S_{small}$ such that $\lim x^k = x$. Then, every solution of* (2.2) *solves* (2.1).

Proof. Let x be a solution of (2.2). Let $\{x^k\} \subset B' \cap S_{small}$ be the sequence (convergent to x) that is mentioned in the hypothesis. Let $z \in \Omega$. Since $x^k \in B'$ and Ω is convex, there exists $t > 0$ such that $x^k + t(z - x^k) \in \Omega_{small}$. Therefore, since x^k solves (2.2), $\langle F(x^k), t(z - x^k)\rangle \geq 0$. So, $\langle F(x^k), z - x^k\rangle \geq 0$. Since z and k are arbitrary, this means that x^k solves (2.1) for all k. But the set of solutions of (2.1) is closed, so x also solves (2.1). □

Now, we consider the problem $VIP(F, \Omega_1)$, where

$$\Omega_1 = \{x \in \mathbb{R}^n \mid x_i \geq 0 \quad \forall\, i \in I\} \tag{2.3}$$

and $I \subset \{1, \dots, n\}$. Nonlinear complementarity problems (NCPs) and nonlinear systems are particular cases of $VIP(F, \Omega_1)$, where $I = \{1, \dots, n\}$ and $I = \emptyset$, respectively.

Define

$$\Omega_2 = \left\{ x \in \mathbb{R}^n \mid x \geq \ell \;\text{ and }\; \sum_{i=1}^{n} x_i \leq M \right\},$$

where $\ell \in \mathbb{R}^n$, $\ell_i = 0$ for all $i \in I$, and $\sum_{i=1}^n \ell_i < M$. Clearly, $\Omega_2 \subset \Omega_1$. We denote by S_2 the set of solutions of $VIP(F, \Omega_2)$. The application of Theorem 2.1 to $VIP(F, \Omega_1)$ is given in the following theorem.

THEOREM 2.2. *Suppose that F is continuous on Ω_1, S_2 is convex, and there exists $\bar{x} \in S_2$ such that $\sum_{i=1}^n \bar{x}_i < M$ and $\bar{x}_i > \ell_i$ for all $i \notin I$. Then, any solution of $VIP(F, \Omega_2)$ solves $VIP(F, \Omega_1)$.*

Proof. Since F is continuous, the set of solutions of $VIP(F, \Omega_1)$ is closed. Let $x \in S_2$. By the convexity of S_2, we have that $[\bar{x}, x) \subset S_2$. Moreover, for all $y \in [\bar{x}, x)$, we have that $\sum_{i=1}^n y_i < M$ and $y_i > \ell_i$, $i \notin I$. Therefore, the hypothesis of Theorem 2.1 holds for the sequence defined by $x^k = x + \frac{1}{k}(\bar{x} - x)$. This implies the desired result. □

Defining $G_1 : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$ by

$$G_1(y, x_{n+1}) = (F(y), 0) \quad \forall\, y \in \mathbb{R}^n, x_{n+1} \in \mathbb{R}, \tag{2.4}$$

$$\Omega_3 = \left\{ x \in \mathbb{R}^{n+1} \mid \sum_{i=1}^{n+1} x_i = M \text{ and } \; x_i \geq \ell_i, i = 1, \dots, n+1 \right\},$$

and $\ell_{n+1} = 0$, it is easy to see that solving $VIP(G_1, \Omega_3)$ is equivalent to solving $VIP(F, \Omega_2)$. Finally, after a suitable change of variables, we can consider that $M = 1$ and $\ell_i = 0$ for all $i = 1, \dots, n+1$, so that the original problem is reduced to a variational inequality problem on the canonical simplex.

3. Bounded smooth reformulations.

The discussion in section 2 justifies the study of the problem $VIP(G, \mathcal{S})$, where $G : \mathbb{R}^m \to \mathbb{R}^m$ and

$$\mathcal{S} = \left\{ x \in \mathbb{R}^m \mid x \geq 0 \text{ and } \sum_{i=1}^{m} x_i = 1 \right\}.$$

According to [1, 22] (see, also, [2, 3, 4, 15, 20, 21, 32]) we define the following reformulation for $VIP(G, \mathcal{S})$:

$$\text{Minimize } \Phi_1(x, v, \lambda) \quad \text{subject to } x \geq 0, v \geq 0, \tag{3.1}$$

where

$$\Phi_1(x, v, \lambda) = \rho_0 \|G(x) + \mathbf{1}\lambda - v\|_2^2 + \rho_1 \left(\sum_{i=1}^m x_i - 1 \right)^2 + \langle x, v \rangle^2,$$

$x, v \in \mathbb{R}^m$, $\lambda \in \mathbb{R}$, $\mathbf{1} = (1, \dots, 1) \in \mathbb{R}^m$, $\rho_0, \rho_1 > 0$.

Let us prove that $\Phi_1(x, v, \lambda)$ has bounded level sets on the set $x \geq 0, v \geq 0$. From now on, we denote $\mathbb{R}_+^n = \{x \in \mathbb{R}^n \mid x \geq 0\}$.

THEOREM 3.1. *Assume that G is continuous on $\mathbb{R}_+^m$. Then, for all $\theta \in (0, \rho_1)$, the set*

$$L_1 = \{(x, v, \lambda) \in \mathbb{R}^{2m+1} \mid x \geq 0, v \geq 0, \ \Phi_1(x, v, \lambda) \leq \theta\}$$

is bounded.

Proof. Assume that

$$(x^k, v^k, \lambda_k) \in L_1 \quad \forall \ k = 0, 1, 2, \dots.$$

Since $(\sum_{i=1}^m x_i^k - 1)^2 \leq \theta$ and $x^k \geq 0$ for all $k = 0, 1, 2, \dots$, we have that the sequence $\{x^k\}$ is bounded. Therefore, by the continuity of G, the sequence $\{G(x^k)\}$ is also bounded.

Since $\rho_0 \|G(x^k) + \mathbf{1}\lambda_k - v^k\|_2^2 \leq \theta$ for all $k = 0, 1, 2, \dots$ and $\{G(x^k)\}$ is bounded, we have that $\{\lambda_k - v_i^k\}$ is bounded for all $i = 1, \dots, m$. Therefore, if $\{v_i^k\}$ is bounded for some $i \in \{1, \dots, m\}$, λ_k is also bounded, implying that v_i^k is bounded for all $i = 1, \dots, m$.

Now, if there exists $j \in \{1, \dots, m\}$ such that $\{v_j^k\}$ is unbounded, we can extract a subsequence such that $v_j^k \to \infty$. For the same subsequence, $\lambda_k \to \infty$ and, so, $v_i^k \to \infty$ for all $i = 1, \dots, m$. Let us call, for this subsequence, $v^k = (v_1^k, \dots, v_m^k)$.

But $\langle x^k, v^k \rangle^2 \leq \theta$ for all $k = 0, 1, 2, \dots$. Therefore, since $x^k \geq 0$ for all k, we have that

$$\lim_{k \to \infty} x^k = 0.$$

Hence,

$$\lim_{k \to \infty} \rho_1 \left(\sum_{i=1}^m x_i^k - 1 \right)^2 = \rho_1.$$

Thus, for k large enough, $\Phi_1(x^k, v^k, \lambda_k) > \theta$ and, so, $(x^k, v^k, \lambda_k) \notin L_1$. This means that the assumption on the unboundedness of v_i^k is not possible. This completes the proof. □

It is easy to find (x^0, v^0, λ_0) such that $\Phi_1(x^0, v^0, \lambda_0) < \rho_1$. For example, take $x^0 \geq 0$ such that $\sum_{i=1}^m x_i^0 = 1$, arbitrarily choosing $\lambda_0 \in \mathbb{R}$ and $v^0 \geq 0$. Therefore,

$$\Phi_1(x^0, v^0, \lambda_0) = \rho_0 \|G(x^0) + \mathbf{1}\lambda_0 - v^0\|_2^2 + \langle x^0, v^0 \rangle^2$$

and, so, the condition $\Phi_1(x^0, v^0, \lambda_0) < \rho_1$ holds if we choose

$$\rho_1 > \rho_0 \|G(x^0) + \mathbf{1}\lambda_0 - v^0\|_2^2 + \langle x^0, v^0 \rangle^2.$$

Theorem 3.1 implies that, with the proper choice of x^0, ρ_0, and ρ_1, any reasonable iterative minimization algorithm for solving (3.1) necessarily produces a sequence that has limit points. In fact, the sequence generated by such an algorithm will satisfy

$$\Phi_1(x^k, v^k, \lambda^k) \leq \Phi_1(x^0, v^0, \lambda^0) \quad \forall\ k = 0, 1, 2, \dots;$$

so, by Theorem 3.1, (x^k, v^k, λ^k) will be bounded. Moreover, for most iterative minimization algorithms, limit points are stationary (KKT) points of the minimization problem. This guarantees that stationary points of problem (3.1) will be necessarily found. It remains to relate the stationary points of (3.1) to the solutions of $VIP(G, \mathcal{S})$. This is done in the following theorem.

THEOREM 3.2. *If G is monotone and has continuous first derivatives, all the stationary points of* (3.1) *are solutions of $VIP(G, \mathcal{S})$.*

Proof. Since $\mathcal{S}$ is bounded, this result follows from Theorem 4 of [22]. See also [1, 20]. □

It is easy to see that G_1, defined by (2.4), is monotone if and only if F is monotone. Therefore, stationary points of (3.1) define (after changing variables) solutions of $VIP(F, \Omega_2)$. Under the interiority hypothesis of Theorem 2.2, these are also solutions of $VIP(F, \Omega_1)$.

An interesting consequence of the results of this section comes from analyzing the nonlinear system $F(x) = 0$, where F is monotone (see [43]) but $\|F(x)\|_2^2$ has stationary points or even local minimizers that are not solutions of the system. Essentially, in this section it has been proved that if one selects adequate artificial bounds ℓ and M and the reformulation (3.1) is applied, there is no risk of convergence to spurious stationary points of the squared norm of F.

We finish this section considering a different smooth reformulation of $VIP(G, \mathcal{S})$. See [37]. Consider the minimization problem

$$\text{Minimize } \Phi_2(x, v, \lambda) \quad \text{subject to} \quad x \geq 0, v \geq 0, \tag{3.2}$$

where

$$\Phi_2(x, v, \lambda) = \rho_0 \|G(x) + \mathbf{1}\lambda - v\|_2^2 + \rho_1 \left(\sum_{i=1}^m x_i - 1 \right)^2 + \sum_{i=1}^m (x_i v_i)^2.$$

As in the case of (3.1) it is easy to see that solutions of $VIP(G, \mathcal{S})$ correspond to global solutions of (3.2) for which the objective function vanishes. Moreover, the following results can be proved using the same techniques of Theorem 3.1 and Theorem 3.2. Finally, an initial bounded level set can be obtained choosing ρ_1 similarly to above.

THEOREM 3.3. *Assume that G is continuous on $\mathbb{R}^m_+$. Then, for all $\theta \in (0, \rho_1)$, the set*

$$L_2 \equiv \{(x, v, \lambda) \in \mathbb{R}^{2m+1} \mid x \geq 0, v \geq 0,\ \Phi_2(x, v, \lambda) \leq \theta\}$$

is bounded.

THEOREM 3.4. *If G is monotone and has continuous first derivatives, all the stationary points of* (3.2) *are solutions of $VIP(G, \mathcal{S})$.*

Remark. The compactification procedure is essential to guarantee that stationary points of smooth reformulations are solutions of the associated monotone NCPs. For example, consider the NCP defined by $F : \mathbb{R}^1 \to \mathbb{R}^1$, where

$$F(x) = \begin{cases} -1 & \text{if } x \le 1, \\ -1 + \frac{2}{3}(x-1)^2 & \text{if } 1 \le x \le 2, \\ 1 - \frac{4}{3}e^{-x+2} & \text{if } x \ge 2. \end{cases} \tag{3.3}$$

Clearly, F is monotone and the unique solution of the associated NCP is $2 + \ln \frac{4}{3}$.

The optimization problem associated with the smooth reformulations (without compactification) is to minimize $\Phi(x,z) = (F(x) - z)^2 + (xz)^2$ subject to $x, z \ge 0$. This problem has, besides the global solution, the stationary point $(0,0)$, which is not a solution of the NCP. Moreover, since $\Phi(k, \frac{1}{k^2}) \le 1$ for all $k = 3, 4, \ldots$, the level set

$$\{(x,z) \mid \Phi(x,z) \le 1,\ x \ge 0, z \ge 0\}$$

is not bounded.

4. Penalized Fischer–Burmeister reformulation.

The Fischer–Burmeister function, defined by

$$\varphi(a,b) = a + b - \sqrt{a^2 + b^2} \quad \forall\, a, b \in \mathbb{R}, \tag{4.1}$$

has been used in many reformulations of complementarity and variational inequality problems [11, 14, 16, 17, 23, 27, 29, 30, 42]. Its main property is that $\varphi(a,b) = 0$ if and only if $a \ge 0, b \ge 0$, and $ab = 0$.

The penalized Fischer–Burmeister (PFB) function has been introduced recently in [8]. It is defined by

$$\psi_\mu(a,b) = \varphi(a,b) + \mu a_+ b_+, \tag{4.2}$$

where $\mu \ge 0$, $c_+ = \max\{c, 0\}$, and φ is the Fischer–Burmeister function (4.1). Related functions have been proposed in [30, 33].

Based on this function, Chen, Chen, and Kanzow [8] introduced a new method for solving NCPs for which an excellent practical performance has been reported. These authors proved a bounded level set result (if $\mu > 0$) under the condition that F is a monotone function with a strictly feasible point or that F is an R_0-function (see [9]).

Similarly to (3.1), we define the following reformulation of $VIP(G, \mathcal{S})$:

$$\text{Minimize } \Phi_3(x, v, \lambda), \tag{4.3}$$

where

$$\Phi_3(x,v,\lambda) = \rho_0 \|G(x) + \mathbf{1}\lambda - v\|_2^2 + \rho_1 \left(\sum_{i=1}^m x_i - 1 \right)^2 + \sum_{i=1}^m \psi_\mu(x_i, v_i)^2$$

and ρ_0, ρ_1, and $\mathbf{1}$ are as in (3.1). As in the previously defined reformulations, the objective function of (4.3) vanishes if and only if x is a solution of $VIP(G, \mathcal{S})$.

If G is differentiable, the objective function Φ_3 is once (but not twice) continuously differentiable. Boundedness of the level sets associated with Fischer–Burmeister ($\mu = 0$) reformulations of complementarity problems has been proved in [29] under restrictive conditions on F. Here we will prove bounded level set results that hold assuming only continuity of G.

THEOREM 4.1. *Assume that G is continuous on $\mathbb{R}^m$, $\mu = 0$, and $\theta \in (0, 1/m)$ is such that*

$$\rho_1(\sqrt{\theta m} - 1)^2 > \theta. \tag{4.4}$$

Then, the set

$$L_3 = \{(x, v, \lambda) \in \mathbb{R}^{2m+1} \mid \Phi_3(x, v, \lambda) \le \theta\} \tag{4.5}$$

is bounded.

Proof. Suppose that $\{(x^k, v^k, \lambda_k)\} \in L_3$ for all $k = 0, 1, 2, \ldots$. Let us suppose, by contradiction, that this sequence is not bounded.

Since $\Phi_3(x^k, v^k, \lambda_k) \le \theta$, we have that

$$\varphi(x_i^k, v_i^k)^2 \le \theta \quad \forall\ k = 0, 1, 2, \ldots.$$

So,

$$-\varphi(x_i^k, v_i^k) \le \sqrt{\theta} \quad \forall\ k = 0, 1, 2, \ldots.$$

By an elementary property of the Fischer–Burmeister function, this implies that

$$x_i^k \ge -\sqrt{\theta} \quad \text{and} \quad v_i^k \ge -\sqrt{\theta} \quad \forall\ k = 0, 1, 2, \ldots.$$

So, $x^k \ge (-\sqrt{\theta}, \ldots, -\sqrt{\theta})$ and $(\sum_{i=1}^m x_i^k - 1)^2 \le \theta/\rho_1$ for all $k = 0, 1, 2, \ldots$. This implies that $\{x^k\}$ is bounded.

By the continuity of G, $\{G(x^k)\}$ is also bounded. Therefore, since $\{\|G(x^k) + \mathbf{1}\lambda_k - v^k\|_2^2\}$ is obviously bounded and $v_i^k \ge -\sqrt{\theta}\ \ \forall\ k = 0, 1, 2, \ldots$, the unboundedness of $\{(x^k, v^k, \lambda_k)\}$ implies that there exists a subsequence such that (after relabeling)

$$\lim_{k \to \infty} v_i^k = \infty \quad \forall\ i = 1, \ldots, m. \tag{4.6}$$

But the sequence $\{x^k\}$ is bounded, so it has a convergent subsequence. Therefore, we can ensure that for a suitable subsequence (4.6) holds. So, after a new relabeling,

$$\lim_{k \to \infty} x_i^k = a_i \quad \forall\ i = 1, \ldots, m. \tag{4.7}$$

By an elementary property of (4.1), (4.6) and (4.7) imply that

$$\lim_{k \to \infty} \varphi(x_i^k, v_i^k) = a_i$$

for some $a_i \ge -\sqrt{\theta}$, $i = 1, \ldots, m$. Thus

$$\lim_{k \to \infty} \sum_{i=1}^m \varphi(x_i^k, v_i^k)^2 = \sum_{i=1}^m a_i^2.$$

Since $\sum_{i=1}^m \varphi(x_i^k, v_i^k)^2 \le \theta$ for all k, this implies that

$$\sum_{i=1}^m a_i^2 \le \theta.$$

Hence,

$$\sum_{i=1}^{m} a_i \leq \sqrt{\theta m}.$$

But, by (4.4), $\sqrt{\theta m} < 1$, so

$$\rho_1 \left(\sum_{i=1}^{m} a_i - 1 \right)^2 \geq \rho_1 (\sqrt{\theta m} - 1)^2.$$

Therefore, by (4.4),

$$\rho_1 \left(\sum_{i=1}^{m} a_i - 1 \right)^2 > \theta.$$

This implies that, for k large enough,

$$\rho_1 \left(\sum_{i=1}^{m} x_i^k - 1 \right)^2 > \theta,$$

and, so,

$$\Phi_3(x^k, v^k, \lambda_k) > \theta.$$

This contradicts the fact that $(x^k, v^k, \lambda_k) \in L_3$. □

As in the case of Φ_1 and Φ_2, with a suitable choice of ρ_0, we can ensure that

$$\Phi_3(x^0, v^0, \lambda_0) < \theta, \tag{4.8}$$

where θ satisfies (4.4). In fact, we take $x^0 \geq 0$ such that $\sum_{i=1}^{m} x_i^0 = 1$ and $v^0 = 0$. Then, $\Phi_3(x^0, v^0, \lambda_0) = \rho_0 \|G(x^0) + \mathbf{1}\lambda_0 - v^0\|_2^2$ and condition (4.8) holds if ρ_0 is chosen to be sufficiently small.

Remark. The classical Fischer–Burmeister reformulation of the NCP defined by (3.3) consists of minimizing $\Phi(x) \equiv (\sqrt{x^2 + F(x)^2} - x - F(x))^2$. Since $\Phi(k) \leq 1$ for all $k = 3, 4, 5, \ldots$, this function fails to have bounded level sets. Of course, the level sets $F(x) \leq \alpha$ are bounded if $\alpha > 0$ is small enough, but it is not possible to predict for which point x^0 the level set $\{x \in \mathbb{R} \mid \Phi(x) \leq \Phi(x^0)\}$ is bounded. Of course, a rather trivial way to obtain examples where the level sets of all classical reformulations of the monotone NCP are not bounded is to consider problems with an unbounded solution set. Finally, in the absence of monotonicity, examples of unbounded level sets are easy to obtain for all the classical reformulations.

THEOREM 4.2. *Assume that G is continuous on $\mathbb{R}^m$. If $\mu > 0$, for all $\theta \in (0, \rho_1)$, the set L_3, defined in* (4.5)*, is bounded.*

Proof. Suppose that $(x^k, v^k, \lambda_k) \in L_3$ for all $k = 0, 1, 2, \ldots$. Therefore,

$$\psi_\mu(x_i^k, v_i^k)^2 \leq \theta \quad \forall\, i = 1, \ldots, m, \ k = 0, 1, 2, \ldots.$$

So, by (4.2),

$$-\varphi(x_i^k, v_i^k) \leq \sqrt{\theta} \quad \forall \ i = 1, \ldots, m, \ k = 0, 1, 2, \ldots.$$

This implies, as in Theorem 4.1, that

$$x_i^k \geq -\sqrt{\theta} \quad \forall\ i = 1, \ldots, m,\ k = 0, 1, 2, \ldots.$$

So, since $\rho_1(\sum_{i=1}^m x_i^k - 1)^2 \leq \theta$ for all $k = 0, 1, 2, \ldots$, we have that $\{x^k\}$ is bounded. By the continuity of G, $\{G(x^k)\}$ is bounded and, so, $\{\lambda_k - v_i^k\}$ is bounded for all $i = 1, \ldots, m$. As in previous boundedness theorems, we only need to prove that the assumption $v_i^k \to \infty$ for all $i = 1, \ldots, m$ leads to a contradiction. In fact, if $v_i^k \to \infty$ for all $i = 1, \ldots, m$ we have, as in Theorem 4.1, that, for a suitable subsequence, $\{x_i^k\}$ is convergent and $\{\varphi(x_i^k, v_i^k)\}$ is bounded. Assume, for a moment, that there exists $i \in \{1, \ldots, m\}$ and $\varepsilon > 0$ such that

$$x_i^k \geq \varepsilon$$

for an infinite set of indices. This implies that $x_i^k v_i^k \to \infty$ and, so, the sequence is not contained in L_3. Therefore,

$$\lim_{k\to\infty} x^k \leq 0.$$

This implies that

$$\lim_{k\to\infty} \rho_1 \left(\sum_{i=1}^m x_i^k - 1 \right)^2 \geq \rho_1.$$

This is impossible, since $(x^k, v^k, \lambda_k) \in L_3$. So, the proof is complete. □

Clearly, an initial estimate that belongs to a bounded level set can be chosen as we did in the smooth reformulations studied in section 3.

The following theorem is a sufficiency result for the reformulation (4.3) that corresponds to Theorem 3.2 and Theorem 3.4 of section 3.

THEOREM 4.3. *If G is monotone and has continuous first derivatives, all the stationary points of* (4.3) *are solutions of $VIP(G, \mathcal{S})$.*

Proof. The case $\mu = 0$ follows from a straightforward generalization of Theorem 2.4 of [28]. In the case $\mu > 0$, use Proposition 3.3 of [8] to generalize Theorem 2.4 of [28]. Then, generalize this result as in the case $\mu = 0$. □

5. Preliminary numerical experience. We solved some VIPs on the simplex $\mathcal{S}$ using the reformulations studied in this paper. Our objective here is to get a preliminary idea of the comparative behavior of different reformulations. The first problem considered was

$$\langle G(x), z - x \rangle \geq 0 \quad \forall\ z \in \mathcal{S},$$

where $G : \mathbb{R}^m \to \mathbb{R}^m$ was given by $G(x) = Ax - c$, A was the 10×10 Hilbert matrix $([A]_{i,j} = \frac{1}{i+j-1})$, and the entries of c_i were chosen randomly in $[0, 2]$. In Table 5.1 we recall the different reformulations studied in this paper.

To solve the optimization problems associated with different reformulations, we used the general purpose algorithm SPG given in [6]. This is a very simple algorithm that generally outperforms conjugate gradient methods in the unconstrained case (see [40]) and is comparable to good large-scale bound-constrained solvers when simple constraints are present. Of course, this algorithm does not take into account the structure of the problems at all and, so, can be very inefficient in many cases, but it

Table 5.1
Reformulations and optimization problems.

Reformulation	Objective function	Complementarity term	Feasible region
Smooth 1	Φ_1	$\langle x, v\rangle^2$	$\mathbb{R}_+^m \times \mathbb{R}_+^m \times \mathbb{R}$
Smooth 2	$\Phi2$	$\sum_{i=1}^m (x_i v_i)^2$	$\mathbb{R}_+^m \times \mathbb{R}_+^m \times \mathbb{R}$
PFB	Φ_3	$\sum_{i=1}^m \psi_\mu(x_i, v_i)^2$	$\mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}$

Table 5.2
Comparison of reformulations.

Reformulation	Successful executions	Best in
Smooth 1 (3.1)	4	1 problem
Smooth 2 (3.2)	3	0 problems
PFB (4.3), $\mu = 0$	7	3 problems
PFB (4.3), $\mu = 0.1$	6	3 problems
PFB (4.3), $\mu = 1.0$	8	1 problem
PFB (4.3), $\mu = 10.$	4	0 problems

is useful when the goal is to compare reformulations as we do in this case. In this first set of experiments we used a modest computer environment (Pentium with 90 MHZ) and the code was written in (double precision) Fortran 77.

The convergence criterion used to terminate the execution of SPG was

$$\|P(z - \nabla\Phi_i(z)) - z\| \leq 10^{-6},$$

where $z = (x, v, \lambda)$ and P is the projection on the feasible region. As an initial approximation we took x_i = uniformly random between 0 and 1 and then divided each coordinate by $\sum_{i=1}^m x_i$. We also took $v = 0$ and $\lambda = 0$. In order to ensure bounded level sets we chose $\rho_0 = 1$ and $\rho_1 = \max\{1, 1.1\|G(x^0)\|_2^2\}$.

We solved 10 problems with different random generations of c and the initial x. We considered that the execution was successful if the solution was obtained in less than 25 seconds. In general, successful executions used less than 5 seconds for all the formulations. In Table 5.2, we show the number of successful executions and the number of times each reformulation was the best, in terms of execution time. In all the successful cases, the solutions were obtained with the same precision. In two problems, all the reformulations failed. We considered that there was not a "best reformulation" in these two cases.

Both in the condensed Table 5.2, and looking in detail at the experiments, the behavior of "Smooth 2" appears to be similar to PFB with $\mu = 10$. This is not surprising, since a large μ in $\psi_\mu(a, b)$ gives more weight to the multiplicative term ab in the positive orthant and "Smooth 2" only uses this term.

The penalty parameter μ in the function $\psi_\mu(a, b)$ affects the measure of "lack of complementarity" in the positive orthant ($a \geq 0, b \geq 0$) in the following way: If $\mu \approx 0$, then $\psi_\mu(a, b) \approx \varphi(a, b)$ and, so, $\psi_\mu(\varepsilon, M)$ is "approximately independent" of M if $\varepsilon > 0$ is small and M is large. This comes from $\lim_{M\to\infty} \varphi(\varepsilon, M) = \varepsilon$. In other words, $\varphi(\varepsilon, M) \approx \min\{\varepsilon, M\}$. On the other hand, if μ is large or if we are using "Smooth 2," the measure of lack of complementarity tends quickly to ∞ if one of the variables tends to infinity and the other is kept fixed. Whether it is better to consider that (ε, M) is almost complementary or not is a problem-dependent question. However, at the beginning of iterative processes, it is dubious that the variable that corresponds to the smaller complementary variable will be zero at the solution and, so, it seems convenient to try to reduce both. This decision corresponds to "μ large" in the PFB reformulation.

The "Smooth 1" reformulation is "more global" in the sense that the influence of the lack of complementarity of the pair (x_j, v_j) depends on the lack of complementarity of the other pairs. In fact, since

$$\frac{\partial}{\partial v_j}\left(\sum_{i=1}^m x_i v_i\right)^2 = 2\left(\sum_{i=1}^m x_i v_i\right) x_j \quad \text{and} \quad \frac{\partial}{\partial x_j}\left(\sum_{i=1}^m x_i v_i\right)^2 = 2\left(\sum_{i=1}^m x_i v_i\right) v_j,$$

the contribution of the jth lack of complementarity to the gradient of the objective function grows with the deviation from complementarity of the remaining pairs. In other words, "Smooth 1" will try a large step toward zero on the variable v_j not only when x_j and $x_j v_j$ are large but also when some of the products $x_i v_i$ (for $i \neq j$) are large.

The small number of experiments described above encouraged us to define a Newton-type algorithm that uses Φ_3 as the objective function, with the aim of comparing more systematically different choices of μ. Observe that finding a zero value of Φ_3 is equivalent to solving the $(2m+1) \times (2m+1)$ nonlinear system

$$H(z) = 0,$$

where $z = (x, v, \lambda)$ and

$$H(z) = \left(G(x) - v + \lambda \mathbf{1},\ \sqrt{\rho_1}\left(\sum_{i=1}^m x_i - 1\right),\ \psi_\mu(x_1, v_1), \dots, \psi_\mu(x_m, v_m)\right).$$

If G is smooth, H is smooth except when $x_i v_i = 0$ for some i. (However, Φ_3 is smooth for all z.) Therefore, the Newtonian direction

$$d(z) = -B(z)^{-1} H(z), \tag{5.1}$$

where $B(z) \in \partial_B G(z)$, is well defined whenever a nonsingular element of $\partial_B G(z)$ can be found. See [39]. This allows us to define a nonmonotone safeguarded Newton-gradient algorithm along the lines of [11, 25]. From now on, we write $\Phi(z) = \Phi_3(x, v, \lambda)$ for the sake of simplicity.

Assume that $\gamma \in (0,1), \beta_1, \beta_2 > 0, \alpha \in (0, 1/2), \nu \in \{0, 1, 2, \dots\}$ are given independently of k. Suppose that the iterate z^k has been computed for some $k \geq 0$. Then, if $\nabla\Phi(z^k) \neq 0$, the iterate z^{k+1} is computed as follows.

ALGORITHM 5.1 (nonmonotone safeguarded Newton-gradient).

Step 1. *If $d(z^k)$ (given by* (5.1)*) exists and, in addition,*

$$\langle d(z^k), \nabla\Phi(z^k)\rangle \leq -\gamma \|d(z^k)\|_2 \|\nabla\Phi(z^k)\|_2 \tag{5.2}$$

and

$$\beta_1 \|\nabla\Phi(z^k)\|_2 \leq \|d(z^k)\|_2 \leq \beta_2 \|\nabla\Phi(z^k)\|_2, \tag{5.3}$$

define $d^k = d(z^k)$. Otherwise, define $d^k = -\nabla\Phi(z^k)$.

Step 2. *Starting with $t = 1$ and using classical safeguarded backtracking (see* [11, 12]*), compute $t_k > 0$ such that*

$$\Phi(z^k + t_k d^k) \leq \tilde{\Phi}_k + \alpha t_k \langle d^k, \nabla\Phi(z^k)\rangle, \tag{5.4}$$

where

$$\tilde{\Phi}_k = \max\{\Phi(z^k), \ldots, \Phi(z^\tau)\}$$

and $\tau = \max\{0, k - \nu\}$ *(see* [25]*).*

Define $z^{k+1} = z^k + t_k d^k$. Using slight modifications of the results of [11] and [25] we can prove that every limit point of a sequence generated by this algorithm is stationary. Since we have proved that, choosing the appropriate initial point and ρ_1, the generated sequences are bounded, it turns out that stationary points are necessarily found, in the limit, by Algorithm 5.1.

We wrote a double precision Fortran code implementing this algorithm for the unconstrained minimization of Φ_3. We chose $\gamma = \beta_1 = 1/\beta_2 = \alpha = 10^{-4}$ and $\nu = 9$. As initial point we took $x^0 = (1, 1/2, \ldots, 1/m)/\sum_{i=1}^m (1/i)$, $v^0 = 0$, $\lambda_0 = 0$. We ran the algorithm for different choices of μ using problems defined by operators $G(x)$ taken from the nonlinear-system literature. Namely, we define the following problems.

Problem 1 (Hilbert). $m = 100$.

$G(x) = Ax - c$, where A is defined as the Hilbert matrix and $c = (1, 1/2, \ldots, 1/m)$.

Problem 2 (Broyden). $m = 100$.

$$[G(x)]_1 = (3 - 2x_1)x_1 - 2x_2 + 1,$$

$$[G(x)]_i = (3 - 2x_i)x_i - x_{i-1} - 2x_{i+1} + 1, \quad i = 2, \ldots, m-1,$$

$$[G(x)]_m = (3 - 2x_m)x_m - x_{m-1}.$$

Problem 3 (Rosenbrock). $m = 20$.

$$[G(x)]_i = 10(x_{i+1} - x_i^2) \quad \text{if } i \text{ is odd},$$

$$[G(x)]_i = 1 - x_{i-1} \quad \text{if } i \text{ is even}.$$

Problem 4 (Helical valley). $m = 99$.

For $i = 1, \ldots, m/3$,

$$[G(x)]_{3i} = 10x_{3i+2} - \frac{50}{\pi} \operatorname{atan}\,(x_{3i+1}/3i) - 50 \quad \text{if } x_{3i} < 0,$$

$$[G(x)]_{3i} = 10x_{3i+2} - \frac{50}{\pi} \operatorname{atan}\,(x_{3i+1}/3i) \quad \text{if } x_{3i} > 0,$$

$$[G(x)]_{3i+1} = \sqrt{x_{3i}^2 + x_{3i+1}^2},$$

and

$$[G(x)]_{3i+2} = x_{3i+2}.$$

Problem 5 (Watson). $m = 31$.

For $i = 1, \ldots, 29$,

Table 5.3
Comparison of different penalty parameters in PFB.

Problem	m	μ	$\Vert H(z)\Vert_\infty$	It	FE	Changes	Time
Hilbert	100	10^{-6}	$0.95E-10$	10	64	0	2.16
		0.10	$0.22E-10$	13	74	0	2.81
		1.00	$0.14E-09$	12	84	0	2.56
		10.00	$0.17E-11$	11	69	0	2.32
		100.0	$0.34E-10$	13	92	0	2.78
Broyden	100	10^{-6}	$0.42E-09$	4	5	0	0.82
		0.10	$0.42E-09$	4	5	0	0.82
		1.00	$0.42E-09$	4	5	0	0.82
		10.00	$0.42E-09$	4	5	0	0.82
		100.0	$0.42E-09$	4	5	0	0.82
Rosenbrock	20	10^{-6}	$0.48E-09$	3	4	0	< 0.1
		0.10	$0.48E-09$	3	4	0	< 0.1
		1.00	$0.48E-09$	3	4	0	< 0.1
		10.00	$0.48E-09$	3	4	0	< 0.1
		100.0	$0.48E-09$	3	4	0	< 0.1
Helical	99	10^{-6}	$0.36E-14$	122	2021	104	25.7
		0.10	$0.36E-14$	122	2025	105	26.3
		1.00	$0.13E-11$	122	1995	104	26.2
		10.00	$0.28E-11$	62	830	45	13.0
		100.0	$0.40E-09$	153	2594	134	32.6
Watson	31	10^{-6}	$0.10E-09$	35	197	1	0.1
		0.10	$0.10E-09$	35	197	1	0.1
		1.00	$0.10E-09$	35	197	1	0.1
		10.00	$0.10E-09$	35	197	1	0.1
		100.0	$0.10E-09$	35	197	1	0.1
Murty	100	10^{-6}	$0.44E-15$	176	561	1	25.6
		0.10	$0.77E-10$	150	397	1	31.3
		1.00	$0.26E-09$	173	491	1	35.9
		10.00	$0.80E-12$	166	511	1	34.4
		100.0	$0.16E-11$	165	493	1	34.2

$$[G(x)]_i = \sum_{j=1}^{m}(j-1)x_j(i/29)^{j-2} - \left[\sum_{j=1}^{m} x_j(i/29)(i/29)^{j-2}\right]^2 - 1,$$

$$[G(x)]_{30} = x_1,$$

$$[G(x)]_{31} = x_2 - x_1^2 - 1.$$

Problem 6 (Murty). $m = 100$.

$G(x) = Ax - c$, where A is upper-triangular, $[A]_{ij} = 2$ if $i < j$, $[A]_{ii} = 1$ for all $i = 1, \ldots, m$, and $c = (1, \ldots, 1)$.

The experiments that we report below were run in a SPARCstation Sun Ultra 1, with an UltraSPARC 64 bits processor, 167-MHz clock, and 128-MBytes of RAM memory. The stopping criterion was $\Vert H(z)\Vert_\infty \leq 10^{-8}$. Besides number of iterations (It), number of function evaluations (FE), and CPU time (in seconds), we report in Table 5.3 the number of times the Newton direction needed to be replaced by the gradient direction. In this preliminary implementation, the linear systems were solved by Gaussian elimination, without taking advantage of their sparsity. Obviously, the computer time must decrease dramatically if a sparse implementation is developed, but the other indicators would not change.

We observe that in three problems (Broyden, Rosenbrock, and Watson) the behavior of the five penalty parameters is the same. In Hilbert and Murty the smallest

TABLE 5.4
Algorithm 5.1 *with Problem Hilbert,* $\mu = 0.1$.

Iteration k	Evaluations	$\|H(z^k)\|_2$
0	1	1.1659909612460
1	7	1.1506050628261
2	19	1.1559111113737
3	31	1.1586923058790
4	39	1.1556372319450
5	46	1.1414773334813
6	51	1.1299367786945
7	56	1.1386723439549
8	62	1.1617009941393
9	66	1.0730683927287
10	70	0.98505951078780
11	72	0.67329707514733
12	73	$4.2472036158979E-03$
13	74	$2.6507263515338E-11$

μ was, marginally, the best. However, in Helical, "$\mu = 10$" clearly outperformed the other alternatives. Probably, very large values of μ should be discarded from practical implementations (at least in well-scaled problems), but the best choice among "small" values of μ seems to depend strongly on the problem characteristics.

The number of functional evaluations per iteration appears to be large in the problems Hilbert, Watson, and Murty. With the aim of understanding this phenomenon, we ran some problems choosing the gradient direction in the first (5 or 10) iterations. Running Hilbert with $\mu = 0.1$ and 5 (first) gradient iterations, the computer time decreased to 1.6 seconds and, with 10 (first) gradient iterations, to 1.1 seconds. We found it instructive to show the detailed behavior of the algorithm in the ordinary case and in the two modified cases. See Tables 5.4, 5.5, and 5.6. We observe that, in fact, the first Newton iterations are not worthwhile in terms of the progress they provide, whereas, of course, they are much more expensive than gradient iterations. The quadratic convergence of Newton is quite evident in the last two iterations. We also ran the algorithm using only gradient iterations, and we observed, as expected, an extremely slow convergence behavior. In fact, convergence did not occur after 1000 iterations in this case.

The qualitative behavior described for Hilbert is essentially the same in the Watson problem. In this case, with 10 initial gradient iterations, the computer time reduced to 0.18 seconds and even the number of iterations decreased. On the other hand, the modification of the algorithm in the Murty problem did not cause meaningful improvements. In this problem, the number of iterations increased moderately and the computer time remained more or less the same.

Problem Helical is instructive in a different sense. In this case, the Newton direction was rejected at most iterations and the algorithm behaves, essentially, as a steepest descent method. We decided to modify the algorithmic parameters in order to weaken the criterion of acceptance of the Newton direction at Step 1 of Algorithm 5.1. Consequently, we chose $\gamma = \beta_1 = 1/\beta_2 = 10^{-25}$ and ran the problem with these new parameters. The results were quite impressive, showing how sensitive this type of algorithm can be with respect to safeguarding constants. For $\mu = 0.1$ the Newton direction was never rejected, convergence occurred in 10 iterations with 18 function evaluations and 1.9 seconds of CPU time. Similar improvements were obtained for the other values of μ.

Table 5.5
Algorithm 5.1 *with Problem Hilbert,* $\mu = 0.1$ *First* 5 *are gradient iterations.*

Iteration k	Evaluations	$\|H(z^k)\|_2$
0	1	1.1659909612460
1	9	1.0818877741731
2	17	1.0192301426184
3	24	1.0804586556667
4	32	0.98801773901278
5	40	0.91336945465545
6	41	0.46240603597402
7	45	0.78454888864694
8	46	0.86704125514515
9	47	0.93591242869472
10	54	1.0832676503144
11	55	$1.2486098348917E-06$
12	56	$9.6170809859694E-14$

Table 5.6
Algorithm 5.1 *with Problem Hilbert,* $\mu = 0.1$. *First* 10 *are gradient iterations.*

Iteration k	Evaluations	$\|H(z^k)\|_2$
0	1	1.1659909612460
1	9	1.0818877741731
2	17	1.0192301426184
3	24	1.0804586556667
4	32	0.98801773901278
5	40	0.91336945465545
6	48	0.85190911080106
7	55	1.1099440799153
8	63	0.97638264851556
9	71	0.87222456838420
10	79	0.78914420687610
11	80	0.29364275115164
12	81	0.95227479134610
13	82	$1.0932851184041E-02$
14	83	$1.9355846720355E-09$

6. Final remarks. We believe that the results presented in this paper have a reasonably wide scope of applications. Consider the general variational inequality problem defined by F_1 on Ω, where F_1 is smooth,

$$\Omega = \{x \in \mathbb{R}^q \mid g(x) \leq 0\},$$

$g = (g_1, \ldots, g_p)$, and g_i smooth and convex for all $i = 1, \ldots, p$. Under a suitable constraint qualification [26], this problem is equivalent to

$$F_1(x) + \sum_{i=1}^{p} w_i \nabla g_i(x) = 0,$$

$$w \geq 0, \qquad g(x) \leq 0,$$

$$\sum_{i=1}^{p} g_i(x) w_i = 0.$$

Defining $n = p+q$, $z = (x, w)$, $F(z) = (F_1(x) + \sum_{i=1}^{p} w_i \nabla g_i(x), -g(x))$, and $I = \{p+1, \ldots, p+q\}$ we obtain a problem of type $VIP(F, \Omega_1)$ (2.3). So, after compactification, we obtain the VIP on the simplex.

In this research we proved that, using several potentially useful reformulations, the boundedness of the sequences generated by standard algorithms can be guaranteed, so that limit points exist and sufficiency results can be applied.

Sufficiency results of the type "stationarity implies solution" usually depend on "monotonicity-like" assumptions. However, one should not interpret that the reformulations must be tried *only when* the monotonicity assumption is guaranteed to hold. Optimization algorithms usually guarantee stationary points, but their practical efficiency is linked to their ability to find global minimizers in a substantial number of cases. This means that we can try to solve the reformulation in any situation, with the hope that using good global strategies we will probably find solutions of the original problem.

In [43, 44], Solodov and Svaiter presented Newton-like methods for solving monotone nonlinear systems and monotone NCPs, respectively. Their convergence results are very strong but, on the other hand, the monotonicity assumption seems to be more essential for their algorithms than it is for the different reformulations presented here. The conditions under which specific algorithms for reformulations enjoy the "true" convergence properties of [43, 44] should be investigated.

Acknowledgments. We are indebted to two anonymous referees, whose comments helped us to improve the paper.

REFERENCES

[1] R. Andreani, A. Friedlander, and J. M. Martínez, *On the solution of finite-dimensional variational inequalities using smooth optimization with simple bounds*, J. Optim. Theory Appl., 94 (1997), pp. 635–657.

[2] R. Andreani and J. M. Martínez, *Solving complementarity problems by means of a new smooth constrained nonlinear solver*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, pp. 1–24.

[3] R. Andreani and J. M. Martínez, *On the solution of the extended linear complementarity problem*, Linear Algebra Appl., 281 (1998), pp. 247–257.

[4] R. Andreani and J. M. Martínez, *On the reformulation of nonlinear complementarity problems using the Fischer–Burmeister function*, Appl. Math. Lett., 12 (1999), pp. 7–12.

[5] R. Andreani and J. M. Martínez, *On the solution of bounded and unbounded mixed complementarity problems*, Optimization, to appear.

[6] E. G. Birgin, J. M. Martínez, and M. Raydan, *Nonmonotone spectral projected gradient methods on convex sets*, SIAM J. Optim., to appear.

[7] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, *A limited memory algorithm for bound constrained optimization*, SIAM J. Sci. Comput., 16 (1995), pp. 1190–1208.

[8] B. Chen, X. Chen, and C. Kanzow, *A penalized Fischer–Burmeister NCP-function: Theoretical investigation and numerical results*, Math. Program., to appear.

[9] B. Chen and P. T. Harker, *Smooth approximations to nonlinear complementarity problems*, SIAM J. Optim., 7 (1997), pp. 403–420.

[10] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 25 (1988), pp. 433–460. See also SIAM J. Numer. Anal., 26 (1989) pp. 764–767.

[11] T. De Luca, F. Facchinei, and C. Kanzow, *A semismooth equation approach to the solution of nonlinear complementarity problem*, Math. Programming, 75 (1996), pp. 407–439.

[12] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall Ser. Comput. Math., Prentice-Hall, Englewood Cliffs, NJ, 1983.

[13] S. P. Dirkse and M. C. Ferris, *A collection of nonlinear mixed complementarity problems*, Optim. Methods Softw., 5 (1995), pp. 319–345.

[14] F. Facchinei, A. Fischer, and C. Kanzow, *A semismooth Newton method for variational inequalities: The case of box constraints*, in Complementarity and Variational Problems:

State of the Art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 76–90.

[15] L. Fernandes, A. Friedlander, M. Guedese, and J. Júdice, *Solution of generalized linear complementarity problems using smooth optimization and its application to bilinear programming and LCP*, Appl. Math. Optim., to appear.

[16] M. C. Ferris and C. Kanzow, *Complementarity and Related Problems: A Survey*, Tech. Report, Comput. Sci. Dept., University of Wisconsin, Madison, WI, 1998.

[17] M. C. Ferris, C. Kanzow, and T. S. Munson, *Feasible descent algorithms for mixed complementarity problems*, Math. Program., 86 (1999), pp. 475–497.

[18] A. Friedlander and J. M. Martínez, *On the numerical solution of bound constrained optimization problems*, RAIRO Rech. Opér., 23 (1989), pp. 319–341.

[19] A. Friedlander, J. M. Martínez, and S. A. Santos, *A new trust region algorithm for bound constrained minimization*, Appl. Math. Optim., 30 (1994), pp. 235–266.

[20] A. Friedlander, J. M. Martínez, and S. A. Santos, *On the resolution of linearly constrained convex minimization problems*, SIAM J. Optim., 4 (1994), pp. 331–339.

[21] A. Friedlander, J. M. Martínez, and S. A. Santos, *Solution of linear complementarity problems using minimization with simple bounds*, J. Global Optim., 6 (1995), pp. 253–267.

[22] A. Friedlander, J. M. Martínez, and S. A. Santos, *A new strategy for solving variational inequalities on bounded polytopes*, Numer. Funct. Anal. Optim., 16 (1995), pp. 653–668.

[23] C. Geiger and C. Kanzow, *On the resolution of monotone complementarity problems*, Comput. Optim. Appl., 5 (1996), pp. 155–173.

[24] M. S. Gowda, *On the extended linear complementarity problem*, Math. Programming, 72 (1996), pp. 33–50.

[25] L. Grippo, F. Lampariello, and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23 (1986), pp. 707–716.

[26] P. T. Harker and J. S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problem: A survey of theory, algorithms and applications*, Math. Programming, 48 (1990), pp. 161–220.

[27] H. Jiang and L. Qi, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Control Optim., 35 (1997), pp. 178–193.

[28] C. Kanzow, *An unconstrained optimization technique for large-scale linearly constrained convex minimization problems*, Computing, 53 (1994), pp. 101–117.

[29] C. Kanzow, *A new approach to continuation methods for complementarity problems with uniform P-functions*, Oper. Res. Lett., 20 (1997), pp. 85–92.

[30] C. Kanzow, N. Yamashita, and M. Fukushima, *New NCP-functions and properties*, J. Optim. Theory Appl., 94 (1997), pp. 115–135.

[31] D. N. Kozakevich, J. M. Martínez, and S. A. Santos, *Solving nonlinear systems of equations with simple constraints*, Comput. Appl. Math., 16 (1997), pp. 215–235.

[32] J. J. Júdice, *Algorithms for linear complementarity problems*, in Algorithms for Continuous Optimization, E. Spedicato, ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994, pp. 435–474.

[33] Z.-Q. Luo and P. Tseng, *A new class of merit functions for the nonlinear complementarity problem*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 204–225.

[34] O. L. Mangasarian and J. S. Pang, *The extended linear complementarity problem*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 359–368.

[35] O. L. Mangasarian and M. V. Solodov, *Nonlinear complementarity problem as unconstrained and constrained minimization*, Math. Programming, 62 (1993), pp. 277–297.

[36] J. M. Martínez, *Quasi-inexact-Newton methods with global convergence for solving constrained nonlinear systems*, Nonlinear Anal., 30 (1997), pp. 1–8.

[37] J. J. Moré, *Global methods for nonlinear complementarity problems*, Math. Oper. Res., 21 (1996), pp. 589–614.

[38] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*, Academic Press, New York, 1970.

[39] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–368.

[40] M. Raydan, *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7 (1997), pp. 26–33.

[41] H. Schwetlick, *Numerische Lōsung nichtlinearer Gleinchungen*, Deutscher Verlag der Wissenschaften, Berlin, 1978.

[42] M. V. Solodov, *Some optimization reformulations of the extended linear complementarity problem*, Comput. Optim. Appl., 13 (1999), pp. 187–200.

[43] M. V. Solodov and B. F. Svaiter, *A globally convergent inexact Newton method for systems of monotone equations*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, pp. 355–369.

[44] M. V. Solodov and B. F. Svaiter, *A truly globally convergent Newton-type method for the monotone nonlinear complementarity problem*, SIAM J. Optim., 10 (2000), pp. 605–625.

[45] M. V. Solodov and P. Tseng, *Two methods based on the D-gap function for solving monotone variational inequalities*, Comput. Optim. Appl., to appear.

SIAM J. Optim.
Vol. 10, No. 3, pp. 896–916

NEWTON'S METHOD FOR A CLASS OF WEAKLY SINGULAR OPTIMAL CONTROL PROBLEMS[*]

KAZUFUMI ITO[†] AND KARL KUNISCH[‡]

Abstract. Newton's method for optimal control of highly nonlinear partial differential equations is analyzed using a 2-norm technique. We consider the case where neither the linearization of the equality constraint e characterizing the differential equation is surjective nor a second order sufficient optimality condition holds for the topology on which e is well defined. Such problems occur, for instance, in optimal control of semilinear elliptic equations or for parameter estimation problems. Despite the above mentioned difficulties, sufficient conditions for second order convergence are obtained.

Key words. Newton's method, optimal control, singular systems, 2-norm technique

AMS subject classifications. 49J20, 49M29, 65K10

PII. S1052623497320840

1. Introduction.

The purpose of this paper is the derivation of optimality systems and the analysis of Newton's method for a class of optimal control problems which are weakly singular in a sense to be described shortly. We consider

$$\text{(P)} \qquad \begin{array}{l} \min J(y,u) \\ \text{subject to } e(y,u)=0, \end{array}$$

where e represents a partial differential equation in the state variable y and with control variable u. Both y and u will be elements of infinite dimensional (function) spaces. In the situation that we have in mind e is defined on a Banach space $Y_1 \times U$ with values in the dual Z^* of a Hilbert space Z. To derive an optimality system one introduces the Lagrangian L associated with (P) by

$$L(y,u,\lambda) = J(y,u) + (\lambda, e(y,u))_{Z,Z^*}.$$

Proceeding formally, assume that (y^*,u^*) is a local solution to (P). Then a necessary optimality condition for (P) is given by

$$\text{(1.1)} \qquad \left\{ \begin{array}{l} L'(y^*,u^*,\lambda^*) = 0, \\ e(y^*,u^*) = 0 \end{array} \right.$$

for an appropriate choice of $\lambda^* \in Z$. Here and throughout primes denote the (partial) derivatives with respect to (y,u). Derivatives with respect to other variables are denoted by subscripts. A sufficient condition for the existence of λ^* such that (1.1) holds consists of smoothness assumptions and surjectivity of the linearization $e'(y^*,u^*)$ of e at (y^*,u^*). In the situations that we have in mind $e'(y^*,u^*)$ is not surjective. It may, however, be surjective if $e'(y^*,u^*)$ is considered as a mapping with domain in a

[*] Received by the editors May 5, 1997; accepted for publication (in revised form) September 12, 1998; published electronically May 2, 2000.
http://www.siam.org/journals/siopt/10-3/32084.html

[†] Department of Mathematics, North Carolina State University, Raleigh, NC 27695 (kito@ncsu.edu).

[‡] Institut für Mathematik, Karl-Franzens-Universität Graz, A-8010 Graz, Austria (karl.kunisch@kfunigraz.ac.at).

larger space $Y \times U (\supset Y_1 \times U)$. On $Y \times U$ the nonlinear mapping e is not necessarily well defined. Consider, for example, the case where e is a function of y only and $e(y) = -\Delta y + \exp(y)$, with $Y = H_0^1(\Omega)$ and $Y_1 = H_0^1(\Omega) \cap L^\infty(\Omega)$.

Thus weakly singular refers to the situation in which e is Fréchet differentiable from $Y_1 \times U$ to Z^* but the Fréchet derivative e' is not surjective from $Y_1 \times U$ to Z^*. In order to overcome this difficulty a second space $Y \times U$ is utilized. Considered as an operator with domain in $Y \times U$, the operator e' is assumed to be closable, but it is not necessarily defined on all of $Y \times U$ or continuous. The derivation of the necessary optimality conditions and the second order convergence analysis of Newton's method will depend on using a 2-norm framework.

Let us next consider Newton's method for solving (P). It is well known that this method has good convergence properties if, e.g., $e'(y^*, u^*)$, considered as an operator on $Y_1 \times U$, is surjective and the Jacobian of (1.1) is uniformly monotone or, alternatively, the Hessian of $L(y, u, \lambda)$ is uniformly positive definite on the kernel of $e'(y^*, u^*)$. Here we only assume such properties with respect to the $Y \times U$ topology. Utilizing the smoothing properties enjoyed by the solutions to the primal and adjoint equations it will be shown that quadratic convergence nevertheless can be achieved with respect to the $Y_1 \times U$ norm.

It has been pointed out before that the analysis of optimal control problems with respect to optimality conditions, sensitivity analysis, or justification of numerical algorithms may require us to simultaneously consider (P) with respect to two spaces. In our work e is well defined and smooth on the smaller space with the finer topology. The second larger space is chosen such that $e'(y^*, u^*)$ has closed range and that a second order sufficient optimality condition holds. In the literature the necessity of utilizing two different norms to guarantee amenable properties of the optimization problem is referred to as 2-norm discrepancy. Differing from our contribution, previous authors used two norms for the control space rather than for the state space. In [I] necessary and sufficient optimality conditions for optimization problems in Banach spaces are given. In that work the Fréchet derivative of the equality constraint is assumed to be an operator with closed range from the smaller space with the finer topology (denoted by $Y_1 \times U$ in our work) to Z. In the applications that motivate our research such a property does not hold. To obtain the optimality system we require only that the linearization of e, defined as an operator with domain in $Y \times U$, is closable. In [M] as well optimization problems with equality and inequality constraints are treated. Transformed to our setup, the assumptions in [M] require the Fréchet derivative to be surjective from $Y_1 \times U$ to Z, which is not true for our applications. In applying 2-norm techniques to optimal control problems the authors of [CTU, GT1, GT2, M] utilize two norms for the control space, typically L^∞ and $L^r, r < \infty$, whereas only one norm is used for the state space. We, on the contrary, use two norms for the variables of the state space and only one for the control space, which is typically L^2. This difference is motivated by the different applications that we have in mind. If the partial differential equations are sufficiently regularizing with respect to the state variable, then it is advantageous to avoid the use of L^∞-norms for the control space since neither the cost functional nor the equality constraint which characterizes the partial differential equation provides radial unboundedness of the optimization problem with respect to the control variable in the L^∞-norm. In [CTU, GT1, GT2] existence of solutions to the optimal control problem studied there is guaranteed by imposing L^∞-bounds on the set of admissible controls. Such constraints are not needed to guarantee existence of optimal controls in our approach. The above references are mainly focused on optimal control in abstract spaces with applications to partial differential equations in mind. The use of 2-norm techniques can also be necessary for the analysis of optimal control of ordinary differential equations. We refer to, e.g., [AM] in this respect.

The paper is organized in the following manner. In section 2 we present a framework that allows us to obtain an optimality system without a surjectivity assumption for $e'(y^*,u^*) : Y_1 \times U \to Z^*$. Examples are given to a control-in-the-coefficient problem that arises from parameter estimation of the convection coefficient and to elliptic boundary value problems with nonlinearities of exponential type. Section 3 is dedicated to the analysis of Newton's method for weakly singular problems. It will allow us to retain the second order convergence rate despite the fact that $e'(y^*,u^*)$ is not closed on $Y_1 \times U$ and that a second order optimality condition is satisfied only for a topology that is coarser than that for which e is well defined. Applications of the general results are given to nonlinear elliptic equations. In the appendix we provide $L^\infty(\Omega)$ a priori estimates for elliptic boundary value problems with exponential nonlinearities.

2. The optimality system.

We consider the equality constrained optimization problem

$$\text{(P)} \qquad \begin{aligned} &\min J(y,u) \\ &e(y,u) = 0, \end{aligned}$$

with $J : Y \times U \to \mathbb{R}$, $e : Y_1 \times U \to Z^*$, where Y, U, Z are Hilbert spaces and Y_1 is a Banach space that is densely embedded in Y. Further, Z^* denotes the dual of Z. Let (y^*,u^*) denote a local solution to (P) and let $V(y^*) \times V(u^*) \subset Y_1 \times U$ denote a neighborhood such that $J(y^*,u^*) \le J(y,u)$ for all $(y,u) \in V(y^*) \times V(u^*)$ satisfying $e(y,u) = 0$. It is assumed throughout that J is Fréchet differentiable in a neighborhood in the $Y \times U$ topology of (y^*,u^*) and that the Fréchet derivative is locally Lipschitz continuous. Further, e is assumed to be Fréchet differentiable at (y^*,u^*) with Fréchet derivative

$$e'(y^*,u^*)(\delta y, \delta u) = e_y(y^*,u^*)\delta y + e_u(y^*,u^*)\delta u.$$

Thus $e_y(y^*,u^*) \in \mathcal{L}(Y_1, Z^*)$. Since Y_1 is dense in Y, one may consider $e_y(y^*,u^*)$ as a densely defined linear operator with domain in Y. To distinguish this operator from $e_y(y^*,u^*)$ defined on Y we shall denote it in this section by

$$G : D(G) \subset Y \to Z^*$$

and we assume that

(H1) $G^* : D(G^*) \subset Z \to Y^*$ is densely defined.

Then necessarily G is closable [K, p. 168]. Its closure will be denoted by the same symbol. In addition the following assumptions will be required:

(H2) $J_y(y^*,u^*) \in \operatorname{Rg} G^*$.

Condition (H2) is a regularity assumption. It implies the existence of a solution $\lambda^* \in D(G^*)$ to the adjoint equation

$$G^*\lambda + J_y(y^*,u^*) = 0.$$

We shall refer to λ^* as the Lagrange multiplier.

(H3) There exists a dense linear subspace $D \subset U$ with the following property: For every $v \in D$ there exists $t_v > 0$ such that for all $t \in [0, t_v]$ there exists $y(t) \in Y_1$ satisfying $e(y(t), u^* + tv) = 0$ and

$$\lim_{t \to 0^+} \frac{1}{t} |y(t) - y^*|_Y^2 = 0. \tag{2.1}$$

(H4) For every $v \in D$ and $y(\cdot)$ as in (H3), e is directionally differentiable at every element of $\{(y(t), u^* + tv) : t \in [0, t_v]\}$ in directions $(y, u) \in Y_1 \times U$ and

$$\lim_{t \to 0^+} \frac{1}{t} \left(\int_0^1 [e'(y^* + s(y(t) - y^*), u^* + stv) - e'(y^*, u^*)](y(t) - y^*, tv) ds, \lambda^* \right)_{Z^*, Z} = 0.$$

Note that (H4) is satisfied if (H3) holds with Y replaced by Y_1 and if $e : Y_1 \to Z^*$ is Fréchet differentiable with locally Lipschitzian derivative.

Our assumptions do not require surjectivity of $e'(y^*, u^*) : Y_1 \times U \mapsto Z^*$ (which is a typical assumption for the derivation of an optimality system [MZ]) nor that $e'(y^*, u^*)$ is well defined on all of $Y \times U$.

We next give several examples which demonstrate the applicability of hypotheses (H1)–(H4) and the necessity to allow for two spaces Y_1 and Y with $Y_1 \subsetneq Y$. The typical situation that we have in mind is $Y_1 = Y \cap L^\infty(\Omega)$ with Y a Hilbertian function space over Ω.

Example 2.1. Consider first the finite dimensional equality constrained optimization problem

$$\begin{cases} \min y_1^2 + u^2, \\ y_1 - y_2^2 = u, \\ y_2^3 = u^2, \end{cases} \tag{2.2}$$

for which $(y^*, u^*) = (0, 0, 0)$ is the solution. Here $Y = \mathbb{R}^2$, $U = \mathbb{R}^1$, $Z = \mathbb{R}^1$, and

$$e(y, u) = \begin{pmatrix} y_1 - y_2^2 - u \\ y_2^3 - u^2 \end{pmatrix}.$$

Note that $e'(y, u) = \begin{pmatrix} 1 & -2y_2 & -1 \\ 0 & 3y_2^2 & -2u \end{pmatrix}$ and that $e'(y^*, u^*)$ is not surjective. Moreover, $G^* = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$ and the adjoint equation

$$G^* \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

has (infinitely many) solutions. Thus (H1), (H2) are satisfied. As for (H3) note that $\begin{pmatrix} y_1(u) \\ y_2(u) \end{pmatrix} = \begin{pmatrix} u + u^{\frac{4}{3}} \\ u^{\frac{2}{3}} \end{pmatrix}$ defines a solution branch to $e(y, u) = 0$ for which (H3) is satisfied. It is simple to verify (H4). Hence (2.2) is an example for an optimization where all hypotheses (H1)–(H4) are satisfied (and Theorem 2.3 will be applicable) but $e'(y^*, u^*)$ is not surjective.

Example 2.2. We consider the optimal control problems with distributed control

$$\min \frac{1}{2} |y - z|_{L^2(\Omega)}^2 + \frac{\beta}{2} |u|_{L^2(\Omega)}^2 \tag{2.3}$$

subject to

$$\text{(2.4)} \qquad \begin{cases} -\Delta y + \exp y = u \text{ in } \Omega, \\ \frac{\partial y}{\partial n} = 0 \text{ on } \Gamma, \\ y = 0 \text{ on } \partial\Omega \setminus \Gamma, \end{cases}$$

where $\beta > 0, z \in L^2(\Omega)$, Ω is a bounded domain in $\mathbb{R}^n$, $n \leq 3$, with smooth boundary $\partial\Omega$ and Γ is a nonempty connected subset of $\partial\Omega$. In Example 2.3 below we shall reconsider this problem with distributed control replaced by Neumann boundary control. To consider this problem in the general setting the control variable u is chosen in $U = L^2(\Gamma)$. We set $Y = Z = H^1_\Gamma(\Omega) = \{y \in H^1(\Omega) : y = 0 \text{ on } \partial\Omega \setminus \Gamma\}$, $Y_1 = H^1_\Gamma(\Omega) \cap L^\infty(\Omega)$, and $Z^* = (H^1_\Gamma(\Omega))^*$. Moreover, let

$$J(y,u) = \frac{1}{2}|y - z|^2_{L^2(\Omega)} + \frac{\beta}{2}|u|^2_{L^2(\Omega)}$$

and let $e : Y_1 \times U \to Z^*$ be defined by assigning to $(y,u) \in Y_1 \times U$ the functional on Z given by

$$v \to (\nabla y, \nabla v) + (\exp y, v) - (u, v).$$

The following lemma is essential. Its proof will be given in the appendix.

LEMMA 2.1. *For every $u \in L^2(\Omega)$ the variational problem*

$$(\nabla y, \nabla v) + (\exp y, v) = (u, v) \text{ for } v \in H^1_\Gamma(\Omega)$$

has a unique solution $y = y(u) \in H^1_\Gamma(\Omega)$ and there exists a constant C such that

$$\text{(2.5)} \qquad |y|_{H^1_\Gamma \cap L^\infty} \leq C(|u|_{L^2(\Omega)} + C) \text{ for all } u \in L^2(\Omega)$$

and

$$\text{(2.6)} \qquad |y(u_1) - y(u_2)|_{H^1_\Gamma \cap L^\infty} \leq C|u_1 - u_2|_{L^2(\Omega)} \text{ for all } u_i \in L^2(\Omega), \quad i = 1, 2.$$

The concept of a solution of (2.4) *is the variational one of Lemma* 2.1. *The control component of any minimizing sequences $\{(y_n, u_n)\}$ for J is clearly bounded. Together with Lemma* 2.1, *it is therefore simple to argue the existence of a solution $(y^*, u^*) \in Y_1$ of* (2.3), (2.4). *For $(y,u) \in Y_1 \times L^2(\Omega)$ let $e_y(y,u)\delta y$ denote the Fréchet derivative of e with respect to y in direction $\delta y \in H^1_\Gamma(\Omega)$. It is given by the functional $v \to (\nabla \delta y, \nabla v) + ((\exp y)\delta y, v)$. Clearly $e_y(y,u) : Y \to Z^*$ is symmetric and* (H1), (H2) *are satisfied. Assumption* (H3) *is a direct consequence of Lemma* 2.1. *To verify* (H4) *note that $e'(y,u)(\delta y, \delta u)$ is the functional defined by*

$$v \to (\nabla \delta y, \nabla v) + (\exp(y)\delta y, v) - (\delta u, v)_\Omega, \quad v \in H^1_\Gamma(\Omega).$$

If $(y,u) \in Y_1 \times U$, then $e'(y,u)$ is well defined on $Y \times U$. Assumption (H4) *requires us to consider*

$$\text{(2.7)} \qquad \lim_{t \to 0^+} \frac{1}{t} \int_0^1 \int_\Omega (\exp(y^* + s(y(t) - y^*)) - \exp y^*)(y(t) - y^*)\lambda^* dx ds,$$

where $y(t)$ is the solution of (2.4) *with $u = u^* + tv$, $v \in U$. Note that $\lambda^* \in L^\infty$, and that $\{|y(t)|_{L^\infty} : t \in [0,1]\}$ is bounded by Lemma* 2.1. *Moreover, $|y(t) - y^*|_{Y_1} \leq Ct|v|_{L^2}$*

and thus the pointwise local Lipschitz property of the exponential function implies that the limit in (2.7) *is zero. Assumption* (H4) *now easily follows.*

The considerations of this example remain correct for cost functionals J that are much more general than the one in (2.3)*. In fact, it suffices that J is weakly lower semicontinuous from $Y \times U$ to $\mathbb{R}$ and radially unbounded with respect to u, i.e.,* $\lim_{n\to\infty} |u_n|_{L^2} = \infty$ *implies that* $\limsup_{n\to\infty} \inf_{y\in Y} J(y,u_n) = \infty$*. This guarantees existence of a solution (y^*, u^*). The general regularity assumptions and* (H1)–(H4) *are satisfied if $J : Y \times U \to \mathbb{R}$ is continuous and Fréchet differentiable in a neighborhood of (x^*, u^*) with locally Lipschitz continuous derivative.*

Example 2.3. Consider the nonlinear optimal boundary control problem

$$\min \frac{1}{2}|y-z|^2_{L^2(\Omega)} + \frac{\beta}{2}|u|^2_{H^s(\Gamma)} \tag{2.8}$$

subject to

$$\begin{cases} -\Delta y + \exp y = f \text{ in } \Omega, \\ \frac{\partial y}{\partial n} = u \text{ on } \Gamma, \\ y = 0 \text{ on } \partial\Omega \setminus \Gamma, \end{cases} \tag{2.9}$$

where $\beta > 0$, $z \in L^2(\Omega)$, $f \in L^\infty$ are fixed, Ω is a bounded domain in $\mathbb{R}^n$, with smooth boundary $\partial\Omega$, and Γ is a nonempty connected subset of $\partial\Omega$. Further assume that s is a real number strictly larger than $\frac{n-3}{2}$ if $n \geq 3$ and that $s = 0$ if $n < 3$. Unlike in Example 2.2 the dimension n of Ω is now allowed to be arbitrary. This example can be treated within the general framework of this paper by setting Y, Y_1, Z^* as in Example 2.2 and

$$J(y,u) = \frac{1}{2}|y-z|^2_{L^2(\Omega)} + \frac{\beta}{2}|u|^2_{H^s(\Gamma)}.$$

The control space U is chosen to be $H^s(\Gamma)$. To verify (H1)–(H4) one proceeds as in Example 2.2 by utilizing the following lemma, the proof of which is given in the appendix.

Lemma 2.2. *The variational problem*

$$(\nabla y, \nabla v) + (\exp y, v) = (f, v) + (u, v)_\Gamma, \quad v \in H^1_\Gamma(\Omega),$$

has a unique solution $y = y(u) \in H^1_\Gamma(\Omega) \cap L^\infty(\Omega)$ for every $u \in H^s(\Gamma)$, and there exists a constant C such that

$$|y|_{H^1_\Gamma(\Omega)\cap L^\infty} \leq C(|u|_{H^s(\Gamma)} + C) \text{ for all } u \in H^s(\Gamma). \tag{2.10}$$

Moreover, C can be chosen such that

$$|y(u_1) - y(u_2)|_{H^1_\Gamma(\Omega)\cap L^\infty} \leq C|u_1 - u_2|_{H^s} \text{ for all } u_i \in H^s(\Gamma), \qquad i = 1, 2. \tag{2.11}$$

Example 2.4. We consider the least squares problem for the estimation of the convection coefficient u in

$$-\Delta y + u \cdot \nabla y = f \text{ in } \Omega,$$
$$y = 0 \text{ on } \partial\Omega, \quad \text{div } u = 0,$$

from data $z \in L^2(\Omega)$. Here Ω is a bounded domain in $\mathbb{R}^n$ with smooth boundary $\partial\Omega$ and $f \in L^2(\Omega)$. To cast this problem in abstract form we choose the space

$U = \{u \in L^2_n(\Omega) \colon \text{div } u = 0\}$ and define $e \colon (H^1_0(\Omega) \cap L^\infty(\Omega)) \times U \to H^1_0(\Omega)^*$ by assigning to each $(y, u) \in (H^1_0(\Omega) \cap L^\infty(\Omega)) \times U$ the functional on $H^1_0(\Omega)$ given by

$$v \to (\nabla y, \nabla v) + (u\, y, \nabla v) - (f, v) \text{ for } v \in H^1_0(\Omega).$$

Note that $e(y, u)$ is not well defined for $(y, u) \in H^1_0(\Omega) \times U$ since $u \cdot \nabla y \in L^1(\Omega)$ only. The regularized least squares problem is given by

$$\text{(2.12)} \qquad \begin{cases} \min \frac{1}{2}|y - z|^2_{L^2} + \frac{\beta}{2}|u|^2_{L^2_n} \\ \text{subject to } e(y, u) = 0, \quad \text{div } u = 0, \end{cases}$$

where $e(y, u) = -\Delta y + u\nabla y - f$. This problem was considered with a direct analysis in [IK] and we shall now argue that it is a special case of problem (P) satisfying (H1)–(H4). For this purpose we set $Y = H^1_0(\Omega)$, $Y_1 = H^1_0(\Omega) \cap L^\infty(\Omega)$, and $Z = H^1_0(\Omega)$. Note that $(u \cdot \nabla y, y) = (\nabla \cdot (uy), y) = 0$ for $(y, u) \in Y_1 \times U$. Using this fact and techniques similar to those of the proof of Lemma 2.1 (compare [T, section 2.3] and [IK]) it can be shown that for every $u \in U$ there exists a unique solution $y = y(u) \in Y_1$ and for every bounded subset B of U there exists a constant $k(B)$ such that

$$\text{(2.13)} \qquad |y(u)|_{Y_1} \le k(B)|u|_{L^2_n} \text{ for all } u \in B.$$

Extracting a subsequence of a minimizing sequence to (2.12), it is simple to argue the existence of a solution $(y^*, u^*) \in Y_1 \times U$ of (2.12). Clearly e is Fréchet differentiable at (y^*, u^*) and $e'(y^*, u^*) \in Y^*$ is the functional given by

$$(e'(y^*, u^*)(\delta y, \delta u), v)_{Y^*, Y} = (\nabla \delta y, \nabla v) + (u^* \cdot \nabla \delta y, v) + (\delta u \cdot \nabla y^*, v).$$

Note that $G = e_y(y^*, u^*)$ is well defined on Y_1 and $G \in \mathcal{L}(Y_1, Z^*)$. But as a consequence of the quadratic term $u^* \cdot \nabla \delta y$, which is only in $L^1(\Omega)$, G is not defined on all of $Y = H^1_0(\Omega)$. As an operator from Y_1 to Z^* the operator G is not surjective. Considered as an operator with domain in Y its adjoint is given by

$$G^* w = -\Delta w - \nabla \cdot (u^* w).$$

The domain of G^* contains Y_1 and hence G^* is densely defined. Moreover, its range contains $L^2(\Omega)$ and thus (H1) as well as (H2) are satisfied. Let $U(u^*) \subset U$ be a bounded neighborhood of u^*. Since for every $u \in B$

$$(\nabla(y(u) - y^*), \nabla v) - (u(y(u) - y^*), \nabla v) = ((u - u^*)y^*, \nabla v) \text{ for all } v \in H^1_0(\Omega),$$

it follows that there exists a constant $k > 0$ such that

$$\text{(2.14)} \qquad |y(u) - y^*|_{H^1} \le k|u - u^*|_{L^2_n} \text{ for all } u \in U(u^*),$$

and (H3) follows. The validity of (H4) is a consequence of (2.14) and the fact that λ^* is the unique variational solution in $H^1_0(\Omega)$ of

$$-\Delta \lambda^* - \nabla \cdot (u^* \lambda^*) = -(y^* - z)$$

and hence an element of $L^\infty(\Omega)$.

Remark 2.1. Comparing Examples 2.2 and 2.3 with Example 2.4 we observe that the linearization $e'(y, u)$, with $(y, u) \in Y_1 \times U$, is well defined on $Y \times U$ for

Examples 2.2 and 2.3 but it is only defined with domain strictly contained in $Y \times U$ for Example 2.4. For none of these examples is e defined on all of $Y \times U$.

Example 2.5. Here we consider the nonlinear optimal control problem with nonlinearity of blowup type,

$$\left\{ \begin{array}{l} \min \frac{1}{2}|\nabla(y-z)|^2_{L^2_2(\Omega)} + \frac{\beta}{2}|u|^2_{L^2(\Gamma)} \text{ subject to} \\ -\Delta y - \exp y = f \text{ in } \Omega, \\ \frac{\partial y}{\partial n} = u \text{ on } \Gamma, \\ y = 0 \text{ on } \partial\Omega \setminus \Gamma, \end{array} \right. \tag{2.15}$$

where $\beta > 0$, $z \in H^1_\Gamma(\Omega)$, $f \in L^2(\Omega)$, Ω is a smooth bounded domain in $\mathbb{R}^2$ and $\Gamma \subset \partial\Omega$ is a connected strict subset of $\partial\Omega$. Since Ω is assumed to be a two-dimensional domain we have the following property of the exponential function: For every $p \in [1, \infty)$,

$$\{|\exp y|_{L^p} : y \in B\} \text{ is bounded,} \tag{2.16}$$

provided that B is a bounded subset of $H^1_0(\Omega)$, [GT, p. 155]. The variational form of the boundary value problem in (2.15) is given by

$$(\nabla y, \nabla v) - (\exp\, y, v) = (f, v) + (u, v)_\Gamma \quad \text{for all } v \in H^1_\Gamma(\Omega), \tag{2.17}$$

where $H^1_\Gamma(\Omega)$ is defined in Example 2.2. To argue existence of a solution of (2.15) let $\{(y_n, u_n)\}$ be a minimizing sequence with weak limit $(y^*, u^*) \in H^1_0(\Omega) \times L^2(\Omega)$. Due to (2.16) and the radial unboundedness of the cost functional with respect to the $H^1_\Gamma(\Omega) \times L^2(\Gamma)$-norm the set $\{|\exp y_n|_{L^p} : n \in \mathbb{N}\}$ is bounded for every $p \in [1, \infty)$ and $\{|\exp y_n|_{W^{1,p}} : n \in \mathbb{N}\}$ is bounded for every $p \in [1, 2)$. Since $W^{1,p}(\Omega)$ is compactly embedded in $L^2(\Omega)$ for every $p \in (1, 2)$ it follows that for a subsequence of $\{y_n\}$, denoted by the same symbol, $\lim \exp(y_n) = \exp y^*$ in $L^2(\Omega)$. It is now simple to argue that (y^*, u^*) is a solution of (2.15). Let us discuss then the validity of (H1)–(H4) with $Y = Y_1 = H^1_\Gamma(\Omega)$, $Z^* = (H^1_\Gamma(\Omega))^*$, with the obvious choice for J, and with $e : Y \times U \to Z^*$ the mapping assigning to $(y, u) \in Y \times U$ the functional $v \to (\nabla y, \nabla v) - (\exp y, v) - (f, v) - (u, v)_\Gamma$ for $v \in H^1_\Gamma(\Omega)$. From (2.16) it follows that e is well defined and its Fréchet derivative at (y, u) in direction $(\delta y, \delta u)$ is characterized by

$$(e'(y, u)(\delta y, \delta u), v) = (\nabla \delta y, \nabla v) - (\exp(y)\delta y, v) - (\delta u, v)_\Gamma \quad \text{for } v \in H^1_\Gamma(\Omega).$$

The operator $G = e_y(y^*, u^*)$ can be expressed as

$$G(\delta y) = -\Delta \delta y - \exp(y^*)\delta y.$$

Note that $G \in \mathcal{L}(Y, Z)$ and G is self-adjoint with compact resolvent. In particular, (H1) is satisfied. The spectrum of G consists of eigenvalues only. It will be assumed that

$$0 \text{ is not an eigenvalue of } G. \tag{2.18}$$

Due to the regularity assumption for z (note that it would suffice to have $\Delta z \in (H^1_\Gamma)^*$), (2.18) implies that (H2) holds. To argue the validity of (H3) and (H4) one can rely on the implicit function theorem. Let B be a bounded open neighborhood of y^* in $H^1_\Gamma(\Omega)$. Using (2.16) one argues the existence of a constant $\kappa > 0$ such that

$$|\exp y - \exp \bar{y}|_{L^4} \leq \kappa |y - \bar{y}|_{H^1} \quad \text{for all } y, \bar{y} \in B.$$

It follows that e is continuous on $B \times U$, and its partial derivative $e_y(y,u)$ is Lipschitz continuous with respect to $(y,u) \in B \times U$. The implicit function theorem implies the existence of a neighborhood $U(u^*)$ of u^* such that for every $u \in U(u^*)$ there exists a solution $y(u^*)$ of (2.17) depending continuously on u. Since $e(y,u)$ is Lipschitz continuous with respect to u it follows, moreover, that there exists $L > 0$ such that

$$|y(u) - y^*|_{H^1} \le L|u - u^*|_{L^2(\Gamma)} \qquad \text{for all } u \in U(u^*).$$

Assumptions (H3) and (H4) are a direct consequence of this estimate.

The methodology utilized to consider this example can also be applied to Examples 2.2 and 2.3 provided that Ω is restricted to be two-dimensional. This is essential for (2.16) to hold and we are not aware of generalizations of (2.16) to dimensions higher than 2. For Example 2.5 it is essential for the cost functional to be radially unbounded with respect to the $H^1_\Gamma(\Omega)$-norm for the y-component to guarantee that minimizing sequences are bounded. For Examples 2.2 and 2.3 the a priori bound on the y-component of minimizing sequences can be obtained through the state equation.

The following result provides a general technique to obtain a system of first order optimality conditions for examples of the type discussed above.

Theorem 2.3. *Let (y^*, u^*) be a local solution of* (P) *and assume that* (H1)–(H4) *hold. Then*

$$\left\{ \begin{array}{lll} e(y^*,u^*) = 0 & \text{in } Z^* & \textit{(primal equation)}, \\ G^*\lambda^* + J_y(y^*,u^*) = 0 & \text{in } Y^* & \textit{(adjoint equation)}, \\ C^*\lambda^* + J_u(y^*,u^*) = 0 & \text{in } U^* & \textit{(optimality)}, \end{array} \right. \tag{2.19}$$

where $C = e_u(y^*,u^*)$.

The system of equations (2.19) is referred to as the optimality system.

Proof. Let $v \in D$, choose t_v according to (H3), and assume that $t \in (0, t_v]$. Due to (H3) and (H4)

$$\begin{aligned} 0 &= e(y(t), u^* + tv) - e(y^*,u^*) = G(y(t) - y^*) + tCv \\ &\quad + \int_0^1 [e'(y^* + s(y(t) - y^*), u^* + stv) - e'(y^*,u^*)](y(t) - y^*, tv)ds. \end{aligned} \tag{2.20}$$

Assumption (H2) implies the existence of a solution λ^* to the adjoint equation. Observe that by (2.20) and the fact that u^* is a local solution to (P)

$$\begin{aligned} 0 \le J(y(t), u^* + tv) - J(y^*,u^*) &= J'(y^*,u^*)(y(t) - y^*, tv) \\ &+ \int_0^1 [J'(y^* + s(y(t) - y^*), u^* + stv) - J'(y^*,u^*)]\,(y(t) - y^*, tv)ds \\ &+ (G^*\lambda^*, y(t) - y^*)_{Y^*,Y} + t(Cv, \lambda^*)_{Z^*,Z} \\ &+ \int_0^1 ([e'(y^* + s(y(t) - y^*), u^* + stv) - e'(y^*,u^*)]\,(y(t) - y^*, tv), \lambda^*)_{Z^*,Z} ds. \end{aligned}$$

By the second equation in (2.19), local Lipschitz continuous differentiability of J, (H3), (H4), and the fact that $J'(y^*,u^*)(y(t) - y^*, tv) = J_y(y^*,u^*)(y(t) - y^*) + tJ_u(y^*,u^*)v$ we obtain

$$0 \le \lim_{t \to 0^+} \frac{1}{t} J(y(t), u^* + tv) - J(y^*,u^*) = (J_u(y^*,u^*)(u^*) + C^*\lambda^*, v)_{U^*,U}.$$

Since v is arbitrary in D it follows that

$$J_u(y^*, u^*)(u^*) + C^*\lambda^* = 0 \text{ in } U^*.$$

This ends the proof.

Consider the optimization problem which contains additional control constraints which are not necessarily described by equalities:

$$\text{(P}'\text{)} \qquad \begin{aligned} &\min J(y,u),\\ &e(y,u) = 0,\\ &u \in K, \end{aligned}$$

where K is a closed convex subset of U. The following corollary to the proof of Theorem 2.3 can easily be obtained.

COROLLARY 2.4. *Let* (y^*, u^*) *be a solution to* (P$'$). *Then under the assumptions of Theorem* 2.3

$$\text{(2.21)} \qquad \begin{cases} e(y^*, u^*) = 0 & \text{in } Z^*,\\ G^*\lambda^* + J_y(y^*, u^*) = 0 & \text{in } Y^*,\\ \Big(C^*\lambda^* + J_u(y^*, u^*), u - u^*\Big)_{U^*,U} \ge 0 & \text{for all } u \in K. \end{cases}$$

3. Newton's algorithm for weakly singular problems. In this section we describe and analyze Newton's method as applied to the weakly singular problem (P). It will be convenient to start with the description of a Newton step applied to the reduced form of (P). Thus we consider

$$\min \hat{J}(u) = J(y(u), u),$$

where $y(u)$ denotes a solution to $e(y,u) = 0$. In the following formal computation we use y to denote $y(u)$. The computation can be made rigorous under the assumptions that will be specified below. These assumptions will be imposed in a neighborhood $V(u^*) \subset U$ of u^*. The first derivative of $\hat{J}(u)$ is given by

$$\text{(3.1)} \qquad \hat{J}_u = e_u(y,u)^*\lambda + J_u(y,u),$$

where $u \in V(u^*)$ and $\lambda = \lambda(u)$ satisfies

$$\text{(3.2)} \qquad e_y(y,u)^*\lambda = -J_y(y,u).$$

Here and below $e_y(y,u)$ is considered as an operator from its domain $D(e_y(y,u))$ in Y to Z^* and $e_y(y,u)^*$ denotes its conjugate. To justify (3.2) we compute

$$\langle \hat{J}_u(u), \delta u\rangle_U = \langle J_y(y,u), \delta y\rangle_Y + \langle J_u, \delta u\rangle_U,$$

where $e_y\delta y + e_u\delta u = 0$ and thus

$$\begin{aligned} \langle \hat{J}_u(u), \delta u\rangle_U &= -\langle (e_y(y,u)^*)^{-1} J_y(y,u), e_u(y,u)\delta u\rangle_Y + \langle J_u(y,u), \delta u\rangle_U\\ &= \langle e_u(y,u)^*\lambda + J_u(y,u), \delta u\rangle_U, \end{aligned}$$

as desired. As is common in optimal control problems we expressed the first derivative of the reduced cost functional $\hat{J}$ by means of the adjoint equation (3.2). The primal variable $y(u)$ and the adjoint variable $\lambda(u)$ satisfy

$$\text{(3.3)} \qquad L_y(y(u), u, \lambda(u)) = 0 \quad \text{for } u \in V(u^*).$$

Using this equation a short computation shows that the second derivative of $\hat{J}(u)$ can be expressed as

$$\hat{J}_{uu} = W(y,u)^* L''(y,u,\lambda) W(y,u),$$

where

$$W(y,u) = \begin{pmatrix} -e_y(y,u)^{-1}\, e_u(y,u) \\ I \end{pmatrix}.$$

Setting $\delta y = -e_y(y,u)^{-1}\, e_u(y,u)\delta u$, the Newton equation

$$\hat{J}_{uu}(u)\delta u = -\hat{J}_u(u)$$

can be expressed as

$$\begin{cases} W(y,u)^* L''(y,u,\lambda) \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} & = W(y,u)^* \begin{pmatrix} 0 \\ -(e_u(y,u)^*\lambda + J_u(y,u)) \end{pmatrix}, \\ e_y(y,u)\delta y + e_u(y,u)\delta u & = 0 \end{cases}$$

or as

$$\begin{cases} L''(y,u,\lambda) \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} - \begin{pmatrix} 0 \\ (e_u(y,u)^*\lambda + J_u(y,u)) \end{pmatrix} \in \mathcal{N}(W(y,u)^*), \\ \\ e_y(y,u)\delta y + e_u(y,u)\delta u = 0, \end{cases}$$

where $\mathcal{N}(W(y,u)^*)$ denotes the nullspace of $W(y,u)^*$. By the definition of $W(y,u)$ we have

$$\mathcal{R}(W(y,u)) = \mathcal{N}(e'(y,u)) = \mathcal{N}((e_y(y,u), e_u(y,u))).$$

Moreover, if $e'(y,u)$ has closed range, then

$$\mathcal{N}(W(y,u)^*) = \mathcal{R}(W(y,u))^\perp = \mathcal{N}(e'(y,u))^\perp = \mathcal{R}(e'(y,u)^*).$$

As a consequence the Newton update can be expressed as

$$\text{(3.4)} \qquad \begin{pmatrix} L''(y,u,\lambda) & e'(y,u)^* \\ & \\ e'(y,u) & 0 \end{pmatrix} \begin{pmatrix} \delta y \\ \delta u \\ \delta\lambda \end{pmatrix} = - \begin{pmatrix} 0 \\ e_u(y,u)^*\lambda + J_u(y,u) \\ 0 \end{pmatrix},$$

where the equality is understood in $Y^* \times U^* \times Z^*$. The question arises in which space to solve (3.4) for $(\delta y, \delta u, \delta\lambda)$. Since it is not assumed that e' is surjective from $Y_1 \times U$ to Z^* we aim for solutions in $Y \times U \times Z$. Therefore the updates $(y+\delta y, u+\delta u, \lambda+\delta\lambda)$ shall not necessarily remain in $Y_1 \times U \times D(e_y(y,u)^*)$. However, our assumptions will guarantee that the feasibility steps consisting in solving the primal equation

$$\text{(3.5)} \qquad e(y, u_c) = 0$$

for y_c and the adjoint equation

$$\text{(3.6)} \qquad e_y(y_c, u_c)^*\lambda + J_y(y_c, u_c) = 0$$

for the dual variable λ_c are such that $(y_c, u_c) \in Y_1 \times Z_1$ holds. Here $u_c = u + \delta u$ and $Z_1 \subset D(e_y(y^*, u^*))$ denotes a Banach space densely embedded into Z, with $\lambda^* \in Z_1$, and as above $e_y(y^*, u^*)^*$ denotes the conjugate of $e_y(y^*, u^*)\colon D(e_y(y^*, u^*)) \subset Y \to Z^*$. Such a Banach space always exists. For instance, one can take $Z_1 = D(e_y(y^*, u^*))$ endowed with the graph norm. Since Y_1 and Z_1 are contained in Y and Z the feasibility step can also be considered as a smoothing step.

We are now prepared to specify the Newton iteration for weakly singular problems.

Algorithm.

(i) Initialization: Choose $u_0 \in V(u^*)$, solve

$$e(y, u_0) = 0, \quad e_y(y, u_0)^* \lambda + J_y(y_0, u_0) = 0 \quad \text{for } (y_0, \lambda_0) \in Y_1 \times Z_1,$$

and set $k = 0$.

(ii) Newton step: Solve for $(\delta y, \delta u, \delta \lambda) \in Y \times U \times Z$

$$\begin{pmatrix} L''(y_k, u_k, \lambda_k) & e'(y_k, u_k)^* \\ & \\ e'(y_k, u_k) & 0 \end{pmatrix} \begin{pmatrix} \delta y \\ \delta u \\ \delta \lambda \end{pmatrix} = - \begin{pmatrix} 0 \\ e_u(y_k, u_k)^* \lambda_k + J_y(y_k, u_k) \\ 0 \end{pmatrix}.$$

(iii) Update $u_{k+1} = u_k + \delta u$.

(iv) Feasibility step: Solve for $(y_{k+1}, \lambda_{k+1}) \in Y_1 \times Z_1$:

$$e(y, u_{k+1}) = 0, \quad e_y(y, u_{k+1})^* \lambda + J_y(y_{k+1}, u_{k+1}) = 0.$$

(v) Stop, or set $k = k + 1$ and goto (ii).

Remark 3.1. Let us briefly interpret the above algorithm from the point of view of the SQP-method applied to (P). In this method both y and u are considered as independent variables related by the equality constraint $e(y, u) = 0$ which is realized by a Lagrangian term. The SQP-method is essentially Newton's method applied to (2.16) to iteratively solve for (y^*, u^*, λ^*). This results in determining updates from the linear system

$$\begin{pmatrix} L''(y_k, u_k, \lambda_k) & e'(y_k, u_k)^* \\ & \\ e'(y_k, u_k) & 0 \end{pmatrix} \begin{pmatrix} \delta y \\ \delta u \\ \delta \lambda \end{pmatrix} = - \begin{pmatrix} e_y(y_k, u_k)^* \lambda_k + J_y(y_k, u_k) \\ e_u(y_k, u_k)^* \lambda_k + J_u(y_k, u_k) \\ e(y_k, u_k) \end{pmatrix}. \tag{3.7}$$

If the values for y_k and λ_k are obtained by means of a feasibility step as in (iv) of the algorithm, then the first and the last components on the right-hand side of (3.7) are 0 and the linear system (3.7) coincides with that of the algorithm. Let us point out that under the conditions of this paper the SQP-iteration is not well defined since the matrix appearing on the left-hand side of (3.7) is not well defined at $(y_{k+1}, u_{k+1}, \lambda_{k+1}) = (y_k, u_k, \lambda_k) + (\delta y, \delta u, \delta \lambda)$. If the feasibility step is also used in the SQP-method, then Newton's method and the SQP-method coincide.

We next specify additional assumptions which justify the above computations and under which well-posedness and convergence of the algorithm can be argued. These assumptions are imposed on J and e in a convex bounded neighborhood of (y^*, u^*, λ^*) which for convenience we again denote by the symbol

$$V(y^*) \times V(u^*) \times V(\lambda^*) \subset Y_1 \times U \times Z_1.$$

(H5) (a) For every $u \in V(u^*)$ there exists a solution $y = y(u) \in V(y^*)$ of $e(y, u) = 0$. Moreover, there exists $\ell > 0$ such that $|y(u) - y^*|_{Y_1} \le \ell |u - u^*|_U$.
(b) For every $(y, u) \in V(y^*) \times V(u^*)$ there exists a solution $\lambda = \lambda(y, u) \in V(\lambda^*)$ of $e_y^*(y, u)\lambda + J_y(y, u) = 0$ and $|\lambda(y, u) - \lambda^*|_{Z_1} \le \ell |(y, u) - (y^*, u^*)|_{Y_1 \times U}$.

(H6) J is twice continuously Fréchet differentiable on $Y \times U$ with the second derivative locally Lipschitz continuous.

(H7) The operator $e : V(y^*) \times V(u^*) \subset Y_1 \times U \to Z^*$ is Fréchet differentiable with Lipschitz continuous Fréchet derivative $e'(y, u) \in \mathcal{L}(Y_1 \times U, Z^*)$. Moreover, for each $(y, u) \in V(y^*) \times V(u^*)$ the operator $e'(y, u)$ with domain $\underline{\text{in}}$ $Y \times U$ has closed range.

(H8) For every $\lambda \in V(\lambda^*)$ the mapping $(y,u) \to (\lambda, e(y,u))_{Z,Z^*}$ from $V(y^*) \times V(u^*) \to \mathbb{R}$ is twice Fréchet differentiable and the mapping $(y,u,\lambda) \to (\lambda, e''(y,u)(\cdot,\cdot))_{Z,Z^*}$ from $V(y^*) \times V(u^*) \times V(\lambda^*) \subset Y_1 \times U \times Z_1 \to \mathcal{L}(Y_1 \times U, Y^* \times U^*)$ is Lipschitz continuous. Moreover, for each $(y,u,\lambda) \in Y_1 \times U \times Z_1$, the bilinear form $(\lambda, e''(y,u)(\cdot,\cdot))_{Z,Z^*}$ can be extended as a continuous bilinear form on $(Y \times U)^2$.

Condition (H5) requires well-posedness of the primal and the adjoint equation in Y_1, respectively, Z_1. The adjoint equations arise from linearization of e at elements of $Y_1 \times U$. Condition (H6) requires smoothness of J. Finally, in (H7) and (H8) the necessary regularity requirements for e as mapping on $Y_1 \times U$ and in $Y \times U$ are specified. From (H5) it follows that the initialization as well as the feasibility step are well defined provided that $u_{k+1} \in V(u^*)$. As a consequence the derivatives of J and e that are required for defining the Newton step are taken at elements $(y_k, u_k, \lambda_k) \in Y_1 \times U \times Z_1$. In Theorem 3.2 below sufficient conditions are given which imply that $(y_k, u_k, \lambda_k) \in V(y^*) \times V(u^*) \times V(\lambda^*)$. Conditions (H6)–(H8) guarantee that the operator appearing on the left-hand side of the Newton step is well defined as an operator on $Y_1 \times U \times Z_1$ and in $Y \times U \times Z$ with range in $Y^* \times U^* \times Z^*$.

We proceed by addressing the solvability of the linear system arising in the Newton step for $(\delta y, \delta u, \delta\lambda) \in Y \times U \times Z$. Let $\mathcal{J}$ denote the canonical isomorphism from $Y \times U$ to $Y^* \times U^*$ and let $(y,u,\lambda) \in V(y^*) \times V(u^*) \times V(\lambda^*)$. Then due to the assumption that $e'(y,u)$, considered as an operator with domain in $Y \times U$, has closed range in Z^* we have

$$Y^* \times U^* = \text{ Rg } e'(y,u)^* \oplus \mathcal{J} \text{ ker } e'(y,u);$$

see [Y, p. 205]. Let $P\colon Y^* \times U^* \to \mathcal{J}$ ker $e'(y,u)$ denote the Hilbert space projection. Then solving

$$\begin{pmatrix} L''(y,u) & e'(y,u)^* \\ & \\ e'(y,u) & 0 \end{pmatrix} \begin{pmatrix} \delta y \\ \delta u \\ \delta\lambda \end{pmatrix} = -\begin{pmatrix} F(y,u,\lambda) \\ 0 \end{pmatrix},$$

where

$$F(y,u,\lambda) = \begin{pmatrix} 0 \\ e_u(y,u)^*\lambda + J_u(y,u) \end{pmatrix},$$

for $(\delta y, \delta u, \delta\lambda) \in Y \times U \times Z$ is equivalent to solving

$$PL'' \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} + (I-P)L'' \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} + (e')^*\delta\lambda = -PF - (I-P)F \tag{3.8}$$

for $\begin{pmatrix} \delta y \\ \delta u \end{pmatrix} \in$ ker $e'(y,u) \subset Y \times U$ and $\delta\lambda \in Z$. In (3.8) all operators are evaluated at $(y,u,\lambda) \in V(y^*) \times V(u^*) \times V(\lambda^*)$. Let us assume that

(H9) $PL''(y,u,\lambda)(\text{ker } e'(y,u)) \supset \mathcal{J} \text{ ker } e'(y,u)$

for all $(y,u,\lambda) \in V(y^*) \times V(u^*) \times V(\lambda^*)$. Note that (H9) holds, for example, if there exist $\kappa > 0$ such that

$$\langle L''(y,u)v, v\rangle_{Y^* \times U^*, Y \times U} \geq \kappa |v|^2_{Y \times U} \text{ for all } v \in \ker e'(y,u)$$

for all $(y,u,\lambda) \in V(y^*) \times V(u^*) \times V(\lambda^*)$. With (H9) holding one solves

$$PL''(y,u) \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} = -PF(y,u,\lambda) \tag{3.9}$$

for $\begin{pmatrix} \delta y \\ \delta u \end{pmatrix} \in \ker e'(y,u)$, and with $\begin{pmatrix} \delta y \\ \delta u \end{pmatrix}$ thus determined, $\delta\lambda$ is chosen to satisfy

$$e'(y,u)^* \delta\lambda = (P - I) \left(F(y,u,\lambda) + L''(y,u) \begin{pmatrix} \delta y \\ \delta u \end{pmatrix} \right). \tag{3.10}$$

We summarize these steps in the following proposition.

Proposition 3.1. *If* (H6)–(H9) *hold, then the Newton step has a solution* $(\delta y, \delta u, \delta\lambda) \in Y \times U \times Z$*, whenever* $(y_k, u_k, \lambda_k) \in V(y^*) \times V(u^*) \times V(\lambda^*)$*.*

With (H5)–(H9) holding, the algorithm is well defined provided that the iterates $\{u_k\}$ remain in $V(u^*)$. The following theorem will guarantee this property (in a possibly smaller neighborhood) and establishes the convergence rate of the algorithm. For $x = (y,u,\lambda) \in V(y^*) \times V(u^*) \times V(\lambda^*)$ let

$$\mathcal{A}(x) : D(\mathcal{A}(x)) \subset Y \times U \times Z \to Y^* \times U^* \times Z^*$$

denote the operator

$$\mathcal{A}(x) = \begin{pmatrix} L''(x) & e'(y,u)^* \\ e'(y,u) & 0 \end{pmatrix}.$$

We shall require the following assumption:

$$\text{(H10)} \begin{cases} \text{There exists a neighborhood } V(x^*) \text{ of radius } \rho > 0 \text{ of } x^* = (y^*, u^*, \lambda^*) \\ \text{in } V(y^*) \times V(u^*) \times V(\lambda^*) \subset Y_1 \times U \times Z_1 \text{ and } \kappa > 0 \text{ such that for every} \\ x \in V(x^*) \text{ and } \delta w \in Y^* \times U^* \times Z^* \\ \qquad \mathcal{A}(x)\delta x = \delta w \\ \text{admits a unique solution } \delta x \text{ satisfying} \\ \qquad |\delta x|_{Y \times U \times Z} \le \kappa |\delta w|_{Y^* \times U^* \times Z^*} \end{cases}$$

Remark 3.2. Let us give sufficient conditions for (H10) to hold:

$$\text{(H11)} \begin{cases} L''(x^*) \in \mathcal{L}(Y \times U, Y^* \times U^*) \text{ and there exists } \kappa_1 > 0 \text{ such that} \\ (L''(x^*)v, v)_{Y^* \times U^*, Y \times U} \ge \kappa_1 |v|^2_{Y \times U} \text{ for all } v \in \ker e'(y^*, u^*), \end{cases}$$

(H12) $e'(y^*, u^*) \in \mathcal{L}(Y \times U, Z^*)$ is surjective.

Replacing the term $|\bar{x}|^2_{Y \times U}$ in (H11) by $|\bar{x}|^2_{Y_1 \times U}$ would result in a classical second order sufficient optimality condition; see [MZ], for example. Despite the fact that positivity of $L''(x^*)$ (or more generally $\mathcal{A}$ as in (H10)) is demanded only with respect to the $Y \times U$ (respectively, $Y \times U \times Z$) norm, we nevertheless obtain convergence of the algorithm in $Y_1 \times U \times Z_1$ due to the feasibility step (iv) and the smoothing properties of the partial differential equation. With (H11) and (H12) holding, $\mathcal{A}(x^*)$

allows a continuous inverse from $Y^* \times U^* \times Z^*$ to $Y \times U \times Z$. If in addition

$$\text{(H13)} \quad \begin{cases} \text{for every } (y,u) \in V(y^*) \times V(u^*) \text{ the operator } e'(y,u) \text{ can be extended} \\ \text{as continuous linear operator from } Y \times U \text{ to } Z^*, \text{ and the mapping} \\ (y,u) \to e'(y,u) \text{ from } V(y^*) \times V(u^*) \subset Y_1 \times U \to \mathcal{L}(Y \times U, Z^*) \text{ is} \\ \text{continuous, and} \\ \ (y,u,\lambda) \to (\lambda, e''(y,u)(\cdot,\cdot)) \text{ from } V(y^*) \times V(u^*) \times V(\lambda^*) \subset Y_1 \times U \times Z_1 \\ \qquad\qquad\qquad\qquad\qquad \to \mathcal{L}(Y \times U, Y^* \times U^*) \text{ is continuous,} \end{cases}$$

then, together with (H6), $x \to \mathcal{A}(x)$ from $V(y^*) \times V(u^*) \times V(\lambda^*)$ to $\mathcal{L}(Y \times U \times Z, Y^* \times U^* \times Z^*)$ is continuous and (H10) holds.

THEOREM 3.2. *If* (H5)–(H8) *and* (H10) *hold and* $|u_0 - u^*|_U$ *is sufficiently small, then the iterates of the algorithm are well defined and they satisfy*

$$|(y_{k+1}, u_{k+1}, \lambda_{k+1}) - (y^*, u^*, \lambda^*)|_{Y_1 \times U \times Z_1} \le K |u_k - u^*|_U^2 \tag{3.11}$$

for a constant K *independent of* k.

Proof. We argued above that the algorithm is well defined. To prove convergence let us denote by x^* the triple (y^*, u^*, λ^*), similarly $\delta x = (\delta y, \delta u, \delta\lambda)$ and $x_k = (y_k, u_k, \lambda_k)$. The Newton step of the algorithm can be expressed as

$$\mathcal{A}(x_k)\delta x = -\mathcal{F}(x_k), \tag{3.12}$$

with $\mathcal{F} : Y_1 \times U \times Z_1 \to Y^* \times U^* \times Z^*$ defined by

$$\mathcal{F}(y,u,\lambda) = -\begin{pmatrix} e_y(y,u)^*\lambda + J_y(y,u) \\ e_u(y,u)^*\lambda + J_u(y,u) \\ e(y,u) \end{pmatrix}.$$

Note that due to the smoothing step the first and third coordinates of $\mathcal{F}$ are 0. Due to (H5) there exists $\ell \ge 1$ such that

$$|y_k - y^*|_{Y_1} \le \ell |u_k - u^*|_U \text{ if } u_k \in V(u^*) \tag{3.13}$$

and

$$|\lambda_k - \lambda^*|_{Z_1} \le \ell |(y_k, u_k) - (y^*, u^*)|_{Y_1 \times U} \text{ if } (y_k, u_k) \in V(y^*) \times V(u^*). \tag{3.14}$$

Here y_k, λ_k are determined by the feasibility step (iv). For any $x \in V(y^*) \times V(u^*) \times V(\lambda^*)$ we find from (H6)–(H8) that ℓ in (3.13), (3.14) can also be chosen such that

$$\begin{aligned} &|\mathcal{F}(x^*) - \mathcal{F}(x) - \mathcal{F}'(x)(x^* - x)|_{Y^* \times U^* \times Z^*} \\ &= \int_0^1 |(\mathcal{F}'(x + s(x^* - x)) - \mathcal{F}'(x))(x^* - x)|_{Y^* \times U^* \times Z^*} ds \\ &= \int_0^1 |(\mathcal{A}(x + s(x^* - x)) - \mathcal{A}(x))(x^* - x)|_{Y^* \times U^* \times Z^*} ds \\ &\le \frac{\ell}{2} |x^* - x|_{Y_1 \times U \times Z_1}^2 . \end{aligned} \tag{3.15}$$

Let us assume that

$$x_k \in V(x^*),$$

with $V(x^*)$ as in (H10). We estimate, using (3.12), (3.15), and the fact that $\mathcal{F}(x^*) = 0$

$$\begin{aligned}|\mathcal{A}(x_k)(\delta x + x_k - x^*)|_{Y^*\times U^*\times Z^*} &= |\mathcal{F}(x^*) - \mathcal{F}(x_k) - \mathcal{F}'(x_k)(x^* - x_k)|\\ &\le \frac{\ell}{2}|x^* - x_k|^2_{Y_1\times U\times Z_1}\end{aligned}$$

and by (H10)

$$|x_k + \delta x - x^*|_{Y\times U\times Z} \le \frac{\ell\kappa}{2}|x_k - x^*|^2_{Y_1\times U\times Z_1}. \tag{3.16}$$

Consequently, we obtain for $u_{k+1} = u_k + \delta u$

$$|u_{k+1} - u^*|_U \le \frac{\ell\kappa}{2}|x_k - x^*|^2_{Y_1\times U\times Z_1}. \tag{3.17}$$

The proof will be completed by an induction argument with respect to k. Let $r := |u_0 - u^*|$ be chosen such that

$$2\ell^5\kappa r \le 1 \text{ and } 2\ell^2 r < \rho. \tag{3.18}$$

Then $|y_0 - y^*|_{Y_1} \le \ell r$ by (3.13), and $|\lambda_0 - \lambda^*|_{Z_1} \le \sqrt{2}\ell^2 r$ by (3.14). It follows that $|x_0 - x^*|_{Y_1\times U\times Z_1} \le 2\ell^2 r < \rho$ and hence $x_0 \in V(x^*)$. Estimate (3.11) for $k = 0$ follows in an identical manner as the general case and hence we go directly to the induction step. Let $|x_k - x^*|_{Y_1\times U\times Z_1} < 2\ell^2 r$. From (3.17) it follows that

$$|u_{k+1} - u^*|_U \le 2\ell^5\kappa r^2 \le r < \rho. \tag{3.19}$$

Consequently, (3.13) and (3.14) are applicable and imply, combined with (3.17),

$$|x_{k+1} - x^*|_{Y_1\times U\times Z_1} \le 2\ell^2|u_{k+1} - u^*|_U \le \ell^3\kappa|x_k - x^*|^2_{Y_1\times U\times Z_1}.$$

Applying (3.13) and (3.14) for $x_k - x^*$ we obtain

$$|x_{k+1} - x^*|_{Y_1\times U\times Z_1} \le 4\ell^7\kappa|u_k - u^*|^2_U,$$

which gives (3.11) with $K = 4\ell^7\kappa$. From (3.18) it follows that $|x_{k+1} - x^*|_{Y_1\times U\times Z_1} \le 2\ell^2 r < \rho$. This ends the induction step.

Let us return now to some of the examples of section 2 and discuss the applicability of conditions (H5)–(H9).

Example 2.2 *revisited.* Condition (H5)(a) is a direct consequence of (2.5) in Lemma 2.1. Condition (H5)(b) requires us to consider the variational form of the linear equation

$$\left\{\begin{array}{l} -\Delta\lambda + e^y\lambda = -(y - z),\\ \lambda = 0 \text{ on } \partial\Omega\setminus\Gamma, \quad \frac{\partial\lambda}{\partial n} = 0 \text{ on } \Gamma,\end{array}\right. \tag{3.20}$$

with $y \in V(y^*) \subset Y_1 = Z_1 = H^1_\Gamma \times L^\infty(\Omega)$. From [T, Chapter 2.3] it follows that there exists a unique solution $\lambda = \lambda(y) \in Y_1$ to (3.20). Moreover, if $y \in V(y^*)$ and $w = \lambda(y) - \lambda(y^*)$, then w satisfies

$$\left\{\begin{array}{l} -\Delta w + e^{y^*}w = (e^{y^*} - e^y)\lambda(y) + y^* - y,\\ w = 0 \text{ on } \partial\Omega\setminus\Gamma, \qquad \frac{\partial w}{\partial n} = 0 \text{ on } \Gamma.\end{array}\right.$$

It follows that there exists $L > 0$ such that

$$|\lambda(y) - \lambda(y^*)|_{Y_1} \leq L|y - y^*|_{Y_1} \quad \text{for all } y \in V(y^*),$$

and thus (H5)(b) is satisfied. It is simple to argue the validity of (H6)–(H8) as well as of (H12) and (H13). Note that $e'(y^*, u^*)$ is not surjective from $Y_1 \times U$ to Z^*. Condition (H9) is subsumed by the stronger assumption (H10). According to Remark 3.2, the latter holds provided that (H11) can be verified. As for (H11), this condition is equivalent to the requirement that

$$|\delta y|^2_{H^1_\Gamma} + (\lambda^*, e^{y^*}(\delta y)^2) + \beta|\delta u| \geq \kappa(|\delta y|^2_{H^1_\Gamma} + |\delta u|^2),\ \kappa > 0, \tag{3.21}$$

for all $(\delta y, \delta u)$ satisfying

$$\begin{cases} -\Delta\delta y + e^{y^*}\delta y = \delta u, \\ \delta y = 0 \text{ on } \partial\Omega \setminus \Gamma, \quad \frac{\partial \delta y}{\partial n} = 0 \text{ on } \Gamma, \end{cases} \tag{3.22}$$

where λ^* is the solution of (3.20) with $y = y^*$. There exists $\bar{k}$ such that

$$|\delta y|_{H^1_\Gamma} \leq \bar{k}|\delta u|_{L^2} \text{ for all } (\delta y, \delta u) \text{ satisfying (3.22).}$$

Consequently, (3.21) is satisfied provided there exists $\kappa > 0$ such that

$$(\lambda^*, e^{y^*}(\delta y)^2) + \frac{\beta}{2\bar{k}}|\delta y|^2_{H^1_\Gamma} \geq \kappa|\delta y|^2_{H^1_\Gamma},\ \delta y \in H^1_\Gamma(\Omega),$$

which is the case, for example, if $\lambda^* \geq 0$ or (see (3.20)) if $|y^* - z|_{L^2}$ is sufficiently small (small residue problem). If $z \geq y^*$, then the weak maximum principle [T] applied to (3.20) gives $\lambda^* \geq 0$.

Example 2.3 *revisited.* The verification of conditions (H5)–(H8) as well as of (H12) and (H13) with Y, Y_1, Z, and Z_1 as in Example 2.2 is almost identical to the one for Example 2.2 revisited. Note that the adjoint equations coincide for both examples. The second order sufficient optimality condition (H11) has the form

$$|\delta y|^2_{L^2} + (\lambda^*, e^{y^*}(\delta y)^2) + \beta|\delta u|^2_{H^s(\Gamma)} \geq \kappa(|\delta y|^2_{H^1_\Gamma} + |\delta u|^2_{H^s\Gamma})$$

for some $\kappa > 0$ independent of $(\delta y, \delta u)$ satisfying

$$\begin{aligned} &-\Delta\delta y + e^{y^*}\delta y = 0, \\ &\delta y = 0 \text{ on } \partial\Omega \setminus \Gamma, \quad \tfrac{\partial \delta y}{\partial n} = \delta u \text{ on } \Gamma, \end{aligned} \tag{3.23}$$

and λ^* is a solution of (3.20) with $y = y^*$. There exists $\bar{k}$ such that

$$|\delta y|_{Y_1} \leq \bar{k}|\delta u|_{H^s(\Gamma)} \quad \text{for all } (\delta y, \delta u) \text{ satisfying (3.23).}$$

Thus, as for Example 2.2, sufficient conditions for (H11) are given by positivity of λ^* or smallness of $|y^* - z|_{L^\infty}$.

Example 2.4 *revisited.* For convenience we recall that the state equation that appears as a constraint in (2.5) is given by

$$\begin{cases} -\Delta y + u \cdot \nabla y = f \text{ in } \Omega, \\ y = 0 \text{ on } \partial\Omega, \end{cases} \tag{3.24}$$

where div $u = 0$. Using (2.6) and triangle inequality arguments it is simple to verify (H5) provided that

$$(y^*, \lambda^*) \in W^{1,\infty}(\Omega) \times W^{1,\infty}(\Omega). \tag{3.25}$$

Conditions (H6)–(H8) are easily verified except for the closed range property of $e'(y,u)$, with $(y,u) \in V(y^*) \times V(u^*) \in Y_1 \times U$. To verify the latter, one uses the Lax–Milgram lemma to first argue surjectivity of $e_y(y,u)$ for $u \in C_n^\infty$ with div $u = 0$. A density argument can then be used to assert surjectivity of $e_y(y,u)$ for every $u \in U$. Hence the range of $e'(y,u)$ considered as mapping with domain in $Y \times U$ equals Z^*. In [IK] it was shown that (H10) holds, provided that

$$0 < \beta - 2|\lambda^*|_{L^\infty} |y^*|_{L^\infty}. \tag{3.26}$$

To interpret condition (3.26), we note that if u^* were smooth, then $\mathcal{A}(y^*, u^*, \lambda^*)$ would be well defined on $Y \times U \times Z$ and (3.26) implies (H11). However, we cannot rely on Remark 3.2 to verify (H10) for this example, since (H13) is not satisfied. In conclusion, Theorem 3.2 is applicable if (3.25) and (3.26) hold.

For this example numerical results based on the algorithm given above are contained in [IK, Ku].

Appendix. For the proof of Lemmas 2.1 and 2.2 we require the following lemma which we take from [T].

LEMMA A.1. *Let $\varphi : (k_1, h_1) \to \mathbb{R}$ be a nonnegative, nonincreasing function and suppose that there are positive constants r, K and β with $\beta > 1$ such that*

$$\varphi(h) \le K(h-k)^{-r} \varphi(k)^\beta \quad \textit{for } k_1 < k < h < h_1.$$

If $\hat{k} := K^{\frac{1}{r}} 2^{\frac{\beta}{\beta-1}} \varphi(k_1)^{\frac{\beta-1}{r}}$ satisfies $k_1 + \hat{k} < h_1$, then $\varphi(k_1 + \hat{k}) = 0$.

Proof of Lemma 2.1. Let us first argue the existence of a solution $y = y(u) \in H^1_\Gamma(\Omega)$ of

$$(\nabla y, \nabla v) + (e^y, v)_{H^1_\Gamma(\Omega)^*, H^1_\Gamma(\Omega)} = (u, v) \text{ for all } v \in H^1_\Gamma(\Omega), \tag{A.1}$$

where $H^1_\Gamma(\Omega) = \{y \in H^1(\Omega) : y = 0 \text{ on } \partial\Omega \setminus \Gamma\}$, $u \in L^2(\Gamma)$, and $(\cdot,\cdot)_{H^1_\Gamma(\Omega)^*, H^1_\Gamma(\Omega)}$ denotes the duality pairing from $H^1_\Gamma(\Omega)$ to its dual. The argument is based on the theory of maximal monotone operators. Let A stand for $-\Delta$ considered as an operator from $H^1_\Gamma(\Omega)$ to $H^1_\Gamma(\Omega)^*$ and let $E : D(E) \subset H^1_\Gamma(\Omega) \to H^1_\Gamma(\Omega)^*$ denote the operator $E(y) = \exp(y)$. Clearly E is monotone and as such it has a maximal monotone extension which we again denote by E [B, Theorem 1.4]. Since Γ is nonempty the Poincaré inequality implies the existence of a positive scalar γ, such that $A - \gamma I$ is maximal monotone. Here I denotes the duality mapping from $H^1_\Gamma(\Omega)$ to $H^1_\Gamma(\Omega)^*$. Moreover, $D(A - \gamma I) \cap D(E) = D(E)$ and hence $A - \gamma I + E$ is maximal monotone [B, Theorem 1.7]. It follows that $(A - \gamma I + E) + \gamma I = A + E$ is surjective [B, Theorem 1.2].

Throughout the remainder of the proof, C will denote a generic constant, independent of $u \in L^2(\Omega)$. Using a monotonicity argument it is simple to argue that the solution $y = y(u) \in H^1_\Gamma(\Omega)$ to (A.1) is unique and that

$$|y(u_1) - y(u_2)|_{H^1} \le C|u_1 - u_2|_{L^2} \tag{A.2}$$

for every pair $u_i \in L^2(\Omega)$. It follows that

$$|y(u)|_{H^1} \le C(|u|_{L^2} + C)$$

for every $u \in L^2(\Omega)$. To verify (2.5) it remains to obtain an $L^\infty(\Omega)$ bound for $y = y(u)$. The proof is based on a generalization of well-known $L^\infty(\Omega)$-estimates due to Stampacchia and Miranda [T] for linear variational problems to the nonlinear problem (A.1). Let us aim first for a pointwise (a.e.) upper bound for y. For $k \in (0, \infty)$ we set $y_k = (y - k)^+$ and $\Omega_k = \{x \in \Omega : y_k > 0\}$. Note that $y_k \in H^1_\Gamma(\Omega)$ and $y_k \ge 0$. Using (A.1) we find

$$(\nabla y_k, \nabla y_k) = (\nabla y, \nabla y_k) = (u, y_k) - (e^y, y_k) \le (u, y_k),$$

and hence

$$|\nabla y_k|^2_{L^2} \le |u|_{L^2} |y_k|_{L^2}. \tag{A.3}$$

By Hölder's inequality and a well-known embedding result

$$|y_k|_{L^2} = \left(\int_{\Omega_k} y_k^2\right)^{\frac{1}{2}} \le |y_k|_{L^6} |\Omega_k|^{\frac{1}{3}} \le C |\nabla y_k|_{L^2} |\Omega_k|^{\frac{1}{3}}.$$

Here we used the assumption that $n \le 3$. Employing this estimate in (A.3) implies that

$$|\nabla y_k|_{L^2} \le C |\Omega_k|^{\frac{1}{3}} |u|_{L^2}. \tag{A.4}$$

We denote by h and k arbitrary real numbers satisfying $0 < k < h < \infty$ and we find

$$|y_k|^4_{L^4} = \int_{\Omega_k} (y - k)^4 > \int_{\Omega_h} (y - k)^4 \ge |\Omega_h| (h - k)^4,$$

which, combined with (A.4), gives

$$|\Omega_h| \le \hat{C} (h - k)^{-4} |\Omega_k|^{\frac{4}{3}} |u|^4_{L^2}, \tag{A.5}$$

where the constant $\hat{C}$ is independent of h, k, and u. It will be shown that Lemma A.1 is applicable to (A.5) with $\varphi(k) = |\Omega_k|$, $\beta = \frac{4}{3}$, and $K = \hat{C} |u|^4_{L^2}$. The conditions on k_1 and h_1 can easily be satisfied. In fact, in our case $k_1 = 0$, $h_1 = \infty$, and $\hat{k} = \hat{C}^{\frac{1}{4}} |u|_{L^2} 2^{\frac{\beta}{\beta-1}} |\Omega_0|^{\frac{\beta-1}{4}}$. The condition $k_1 + \hat{k} < h_1$ is satisfied since

$$\hat{k} = \hat{C}^{\frac{1}{4}} |u|_{L^2} 2^{\frac{\beta}{\beta-1}} |\Omega_0|^{\frac{\beta-1}{4}} < \hat{C}^{\frac{1}{4}} |u|_{L^2} 2^{\frac{\beta}{\beta-1}} |\Omega|^{\frac{\beta-1}{4}} < \infty.$$

We conclude that $|\Omega_{\hat{k}}| = 0$ and hence $y \le \hat{k}$ a.e. in Ω. A uniform lower bound on y can be obtained in an analogous manner by considering $y_k = (-(k + y))^+$. We leave the details to the reader. This concludes the proof of (2.5). To verify (2.6) the H^1 estimate for $y(u_1) - y(u_2)$ is already clear from (A.2) and it remains to verify the $L^\infty(\Omega)$ estimate. Let us set $y_i = y(u_i)$, $z = y_1 - y_2$, $z_k = (z - k)^+$, and $\Omega_k = \{x \in \Omega : z_k > 0\}$ for $k \in (0, \infty)$. We obtain

$$|\nabla z_k|^2_{L^2} = (\nabla z, \nabla z_k) = (u_1 - u_2, z_k) - (e^{y_1} - e^{y_2}, z_k) \le (u_1 - u_2, z_k).$$

Proceeding as above with y and y_k replaced by z and z_k the desired pointwise upper bound for $y_1 - y_2$ is obtained. For the lower bound we define $z_k = (-(k+z))^+$ for $k \in (0,\infty)$ and $\Omega_k = \{x \in \Omega : z_k > 0\} = \{x : k + y_1(x) < y_2(x)\}$. It follows that

$$|\nabla z_k|^2_{L^2} = -(\nabla(y_1 - y_2), \nabla z_k) = (e^{y_1} - e^{y_2}, z_k) - (u_1 - u_2, z_k) \leq -(u_1 - u_2, z_k).$$

From this inequality we obtain the desired uniform pointwise lower bound on $y_1 - y_2$.

Proof of Lemma 2.2. The variational form of (2.9) is given by

$$(\nabla y, \nabla v) + (e^y, y)_{H^1_\Gamma(\Omega)^*, H^1_\Gamma} = (f, v) + (u, v)_\Gamma \text{ for all } v \in H^1_\Gamma(\Omega), \tag{A.6}$$

where u varies in $H^s(\Gamma)$ and $f \in L^\infty(\Omega)$ is fixed. As in the proof of Lemma 2.1 one argues the existence of a solution $y = y(u)$ to (A.6). A monotonicity argument implies the existence of $C > 0$ such that

$$|y(u_1) - y(u_2)|_{H^1} \leq C|u_1 - u_2|_{H^s(\Gamma)} \tag{A.7}$$

for every pair $u_i \in L^2(\Gamma),\ i = 1, 2$. Further, C can be chosen such that

$$|y(u)|_{H^1} \leq C\left(|u|_{H^s(\Gamma)} + C\right).$$

To complete the proof it remains to verify the appropriate L^∞-bounds. We proceed as in the proof of Lemma 2.1 and give only the necessary changes. Since $s > \frac{n-3}{2}$ if $n \geq 3$, there exists $p > n-1$ such that $H^s(\Gamma)$ embeds continuously into $L^p(\Gamma)$. If $n = 2$ we may choose $s = 0$. To verify (2.10) we define $y_k = (y-k)^+$ and obtain from (A.6)

$$|\nabla y_k|^2 \leq |f|_{L^\infty}|y|_{L^1} + |u|_{L^p(\Gamma)}|\tau_\Gamma y_k|_{L^{p'}(\Gamma)}, \tag{A.8}$$

where τ_Γ denotes the trace operator on $H^1_\Gamma(\Omega)$ and p' is conjugate to p, i.e., $\frac{1}{p} + \frac{1}{p'} = 1$ if $p \geq 3$ and $p' = 2$ if $n < 3$. Since $p > n-1$ we have $p' < \frac{n-1}{n-2}$. We next use a well-known embedding result for trace operators [T, p. 70] and obtain from (A.8) that

$$|\nabla y_k|^2 \leq |f|_{L^\infty}|y_k|_{L^1} + C|u|_{H^s(\Gamma)} \cdot |y_k|_{W^{1,\frac{p'\cdot n}{n-1+p'}}}.$$

It follows that there exists a constant C independent of k such that

$$|\nabla y_k|^2 \leq C(|f|_{L^\infty} + |u|_{H^s(\Gamma)})|\nabla y_k|_{L^{\frac{p'\cdot n}{n-1+p'}}}. \tag{A.9}$$

Utilizing Hölder's inequality we find

$$|\nabla y_k| \leq C(|f|_{L^\infty} + |u|_{H^s(\Gamma)})|\Omega_k|^r, \tag{A.10}$$

where $r = \frac{2(n-1+p')-p'n}{2p'n}$. Note that as a consequence of $p' < \frac{n-1}{n-2}$ we have $r > \frac{n-2}{2n}$ if $n \geq 3$, $r = \frac{1}{4}$ if $n = 2$, and $r = \frac{1}{2}$ if $n = 1$. Let $0 < k < h < \infty$. Then as in the proof of Lemma 2.1

$$|y|^t_{L^t} \geq |\Omega_h|(h-k)^t,$$

where $t = \frac{2n}{n-2}$ if $n \geq 3$ and $t > \frac{1}{r}$ if $n < 3$. Since $H^1(\Omega)$ embeds continuously into $L^{\frac{2n}{n-2}}$ for $n \geq 3$ and into L^q for every $q < \infty$ if $n = 2$, estimate (A.10) implies that

$$|\Omega_h| \leq \hat{C}(h-k)^{-t} \left(|f|_{L^\infty} + |u|_{H^s(\Gamma)}\right)^t |\Omega_k|^{tr},$$

where $\hat{C}$ is independent of h, k, u (and also f). Note that $tr > 1$. Utilizing Lemma A.1 with $\beta = tr$ the proof can be completed in the same manner as the one of Lemma 2.1.

Acknowledgment. The authors would like to thank an anonymous referee for a careful reading of the first version of the paper and for many helpful comments.

REFERENCES

[AM] W. Alt and K. Malanowski, *The Lagrange-Newton method for nonlinear optimal control problems*, Comput. Optim. Appl., 2 (1993), pp. 77–100.

[B] V. Barbu, *Nonlinear Semigroups and Differential Equations in Banach Spaces*, P. Noordhoff, Groningen, The Netherlands, 1976.

[CTU] E. Casas, F. Tröltzsch, and A. Unger, *Second order sufficient optimality conditions for a nonlinear elliptic boundary control problem*, Z. Anal. Anwendungen, 15 (1996), pp. 687–707.

[GT] D. Gilbarg and N.S. Trudinger, *Elliptic Partial Differential Equations of Second Order*, Springer-Verlag, Berlin, 1977.

[GT1] H. Goldberg and F. Tröltzsch, *Second order sufficient optimality conditions for a class of nonlinear parabolic boundary control problems*, SIAM J. Control Optim., 31 (1993), pp. 1007–1025.

[GT2] H. Goldberg and F. Tröltzsch, *On a Lagrange–Newton Method for a Nonlinear Parabolic Boundary Control Problem*, preprint, Technischen Universität Chemnitz, Zwickau, 1996.

[I] A. D. Ioffe, *Necessary and sufficient conditions for a local minimum* III. *Second order conditions and augmented duality*, SIAM J. Control Optim., 17 (1979), pp. 266–288.

[IK] K. Ito and K. Kunisch, *Estimation of the convection coefficient in elliptic systems*, Inverse Problems, 13 (1997), pp. 995–1013.

[K] K. Kato, *Perturbation Theory for Linear Operators*, Springer-Verlag, Berlin, 1980.

[Ku] R. Küpfer, *Geschachtelte Iterationsverfahren für die Parameteridentifikation bei Konvektions–Diffusions–Gleichungen*, Master's thesis, Technischen Universität Berlin, 1997.

[L] J.-L. Lions, *Control of Distributed Singular Systems*, Gauthier-Villars, Kent, 1985.

[M] H. Maurer, *First and second order sufficient optimality conditions in mathematical programming and optimal control*, Math. Programming Stud., 14 (1981), pp. 163–177.

[MZ] H. Maurer and J. Zowe, *First and second order necessary and sufficient optimality conditions for infinite-dimensional programming problems*, Math. Programming, 25 (1987), pp. 1542–1556.

[T] G.M. Troianiello, *Elliptic Differential Equations and Obstacle Problems*, Plenum Press, New York, 1987.

[Y] K. Yosida, *Functional Analysis*, Springer-Verlag, Berlin, 1974.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 917–941

PATTERN SEARCH METHODS FOR LINEARLY CONSTRAINED MINIMIZATION*

ROBERT MICHAEL LEWIS† AND VIRGINIA TORCZON‡

Abstract. We extend pattern search methods to linearly constrained minimization. We develop a general class of feasible point pattern search algorithms and prove global convergence to a Karush–Kuhn–Tucker point. As in the case of unconstrained minimization, pattern search methods for linearly constrained problems accomplish this without explicit recourse to the gradient or the directional derivative of the objective. Key to the analysis of the algorithms is the way in which the local search patterns conform to the geometry of the boundary of the feasible region.

Key words. pattern search, linearly constrained minimization

AMS subject classifications. 49M30, 65K05

PII. S1052623497331373

1. Introduction.

This paper continues the line of development in [8, 9, 15] and extends pattern search algorithms to optimization problems with linear constraints:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & \ell \leq Ax \leq u, \end{array} \tag{1.1}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $x \in \mathbf{R}^n$, $A \in \mathbf{Q}^{m \times n}$, $\ell, u \in \mathbf{R}^m$, and $\ell \leq u$. We allow the possibility that some of the variables are unbounded either above or below by permitting $\ell_i, u_i = \pm\infty$, $i \in \{1, \ldots, m\}$. We also admit equality constraints by allowing $\ell_i = u_i$.

We can guarantee that if the objective f is continuously differentiable, then a subsequence of the iterates produced by a pattern search method for linearly constrained minimization converges to a Karush–Kuhn–Tucker (KKT) point of problem (1.1). As in the case of unconstrained minimization, pattern search methods for linearly constrained problems accomplish this without explicit recourse to the gradient or the directional derivative of the objective. We also do not attempt to estimate Lagrange multipliers.

As with pattern search methods for bound constrained minimization [8], when we are close to the boundary of the feasible region the pattern of points over which we search must conform to the geometry of the boundary. The general idea, which also applies to unconstrained minimization [9], is that the pattern must contain search directions that comprise a set of generators for the cone of feasible directions. We must be a bit more careful than this; we must also take into account the constraints

*Received by the editors December 11, 1997; accepted for publication (in revised form) October 11, 1999; published electronically May 15, 2000.
http://www.siam.org/journals/siopt/10-3/33137.html

†Institute for Computer Applications in Science and Engineering, Mail Stop 132C, NASA Langley Research Center, Hampton, VA 23681–2199 (buckaroo@icase.edu). The research of this author was supported by the National Aeronautics and Space Administration under NASA contracts NAS1-19480 and NAS1-97046 while the author was in residence at the Institute for Computer Applications in Science and Engineering (ICASE), NASA Langley Research Center, Hampton, VA.

‡Department of Computer Science, College of William & Mary, P.O. Box 8795, Williamsburg, VA 23187–8795 (va@cs.wm.edu). The research of this author was supported by the Air Force Office of Scientific Research under grant F49620-95-1-0210, by the National Science Foundation under grant CCR-9734044, and by the National Aeronautics and Space Administration under NASA contracts NAS1-19480 and NAS1-97046 while the author was in residence at ICASE, NASA Langley Research Center, Hampton, VA.

that are almost binding in order to be able to take sufficiently long steps. In the bound constrained case this turns out to be simple to ensure (though in section 8.3 we will sharpen the results in [8]). In the case of general linear constraints the situation is more complicated.

Practically, we imagine pattern search methods being most applicable in the case where there are relatively few linear constraints besides simple bounds on the variables. This is true for the applications that motivated our investigation. Our analysis does not assume nondegeneracy, but the class of algorithms we propose will be most practical when the problem is nondegenerate.

2. Background. After we presented this work at the 16th International Symposium on Mathematical Programming in Lausanne, Robert Mifflin brought to our attention the work of Jerrold May in [11], which extended the derivative-free algorithm for unconstrained minimization in [12] to linearly constrained problems. May proves both global convergence and superlinear local convergence for his method. To the best of our knowledge, this is the only other provably convergent derivative-free method for linearly constrained minimization.

Both May's approach and the methods described here use only values of the objective at feasible points to conduct their searches. Moreover, the idea of using as search directions the generators of cones that are polar to cones generated by the normals of faces near the current iterate appears already in [11]. This is unavoidable if one wishes to be assured of not overlooking any possible feasible descent in f using only values of f at feasible points.

On the other hand, there are significant differences between May's work and the approach we discuss here. May's algorithm is more obviously akin to a finite-difference quasi-Newton method. Most significantly, May enforces a sufficient decrease condition; pattern search methods do not. Avoiding a sufficient decrease condition is useful in certain situations where the objective is prone to numerical error. The absence of a quantitative decrease condition also allows pattern search methods to be used in situations where only comparison (ranking) of objective values is possible.

May also assumes that the active constraints are never linearly dependent—i.e., nondegeneracy. Our analysis, which is based on the intrinsic geometry of the feasible region rather than its algebraic description, handles degeneracy (though from a practical perspective, degeneracy can make the calculation of the pattern expensive). On the other hand, we must place additional algebraic restrictions on the search directions since pattern search methods require their iterates to lie on a rational lattice. To do so, we require that the matrix of constraints A in (1.1) be rational. This mild restriction is a price paid for not enforcing a sufficient decrease condition.

May's algorithm also has a more elaborate way of sampling f than the general pattern search algorithm we discuss here. This, and the sufficient decrease condition he uses, enables May to prove local superlinear convergence, which is stronger than the purely global results we prove here.

In section 3 we outline the general definition of pattern search methods for linearly constrained minimization. In section 4 we present the convergence results. In section 5 we review those results from the analysis for the unconstrained case upon which we rely for the analysis in the presence of linear constraints. In section 6 we prove our main results. In section 7 we discuss stopping criteria and the questions of identifying active constraints and estimating Lagrange multipliers. In section 8 we outline practical implementations of pattern search methods for linearly constrained minimization. Section 9 contains some concluding remarks, while section 10 contains essential, but rather technical, results concerning the geometry of polyhedra that are required for the proofs in section 6.

Notation. We denote by $\mathbf{R}$, $\mathbf{Q}$, $\mathbf{Z}$, and $\mathbf{N}$ the sets of real, rational, integer, and natural numbers, respectively. The ith standard basis vector will be denoted by e_i. Unless otherwise noted, norms and inner products are assumed to be the Euclidean norm and inner product. We will denote the gradient of the objective by $g(x)$.

We will use Ω to denote the feasible region for problem (1.1):

$$\Omega = \{ \, x \in \mathbf{R}^n \;\mid\; \ell \leq Ax \leq u \, \} .$$

Given a convex cone $K \subset \mathbf{R}^n$ we denote its polar cone by K°; K° is the set of $v \in \mathbf{R}^n$ such that $(v, w) \leq 0$ for all $w \in K$, where (v, w) denotes the Euclidean inner product.

If Y is a matrix, $y \in Y$ means that the vector y is a column of Y.

3. Pattern search methods.

We begin our discussion with a simple instance of a pattern search algorithm for unconstrained minimization: minimize $f(x)$. At iteration k, we have an iterate $x_k \in \mathbf{R}^n$ and a step-length parameter $\Delta_k > 0$. We successively look at the points $x_+ = x_k \pm \Delta_k e_i$, $i \in \{1, \ldots, n\}$, until we find x_+ for which $f(x_+) < f(x_k)$. Figure 3.1 illustrates the set of points among which we search for x_+ for $n = 2$. This set of points is an instance of what we call a pattern, from which pattern search takes its name. If we find no x_+ such that $f(x_+) < f(x_k)$, then we reduce Δ_k by half and continue; otherwise, we leave the step-length parameter alone, setting $\Delta_{k+1} = \Delta_k$ and $x_{k+1} = x_+$. In the latter case we can also increase the step-length parameter, say, by a factor of 2, if we feel a longer step might be justified. We repeat the iteration just described until Δ_k is deemed sufficiently small.

One important feature of pattern search that plays a significant role in the global convergence analysis is that we do not need to have an estimate of the derivative of f at x_k so long as included in the search is a sufficient set of directions to form a positive spanning set for the cone of feasible directions, which in the unconstrained case is all of $\mathbf{R}^n$. In the unconstrained case the set $\{ \, \pm e_i \;\mid\; i = 1, \ldots, n \, \}$ satisfies this condition, the purpose of which is to ensure that if the current iterate is not a stationary point of the problem, then we have at least one descent direction.

For the linearly constrained case we expand the notion of what constitutes a sufficient set of search directions. Now we must take into account explicit information about the problem: to wit, the geometry of the *nearby* linear constraints. We need to ensure that if we are not at a constrained stationary point, we have at least one feasible direction of descent. Moreover, we need a feasible direction of descent along which we will remain feasible for a sufficiently long distance to avoid taking too short a step. This is a crucial point since, just as in the unconstrained case, we will not enforce any notion of sufficient decrease. Practically, we must ensure that we have directions that allow us to move parallel to the constraints.

We modify the example given above by adding linear constraints near the current iterate x_k and show in Figure 3.2 the effect this has on the choice of pattern. We add one further qualification to the essential logic of pattern search for the unconstrained case by noting that we are considering a feasible point method, so the initial iterate x_0 and all subsequent iterates must be feasible. To enforce this, we can introduce the simple rule of assigning an arbitrarily high function value (say $+\infty$) to any step that takes the search outside the feasible region defined by Ω. Otherwise, the logic of pattern search remains unchanged.

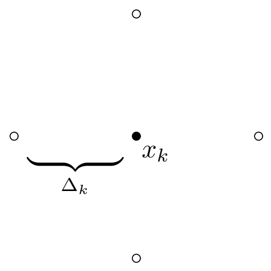


FIG. 3.1. *An illustration of pattern search for unconstrained minimization.*

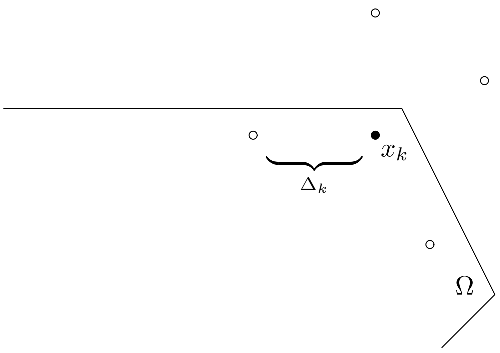


FIG. 3.2. *An illustration of pattern search for linearly constrained minimization.*

We now turn to the technical components of the general pattern search method for the linearly constrained problem (1.1). We borrow much of the machinery from the unconstrained case [15], modified in view of more recent developments in [8, 9]. We begin by describing how the pattern is specified and then used to generate subsequent iterates.

3.1. The pattern. The pattern for linearly constrained minimization is defined in a way that is a little less flexible than for patterns in the unconstrained case. In [15], at each iteration the pattern P_k is specified as the product $P_k = BC_k$ of two components, a fixed *basis matrix* B and a *generating matrix* C_k that can vary from iteration to iteration. This description of the pattern was introduced in the unconstrained case in order to unify the features of such disparate algorithms as the method of Hooke and Jeeves [7] and multidirectional search (MDS) [14]. In the case of bound constrained problems [8], we introduced restrictions on the pattern P_k itself rather than on B and C_k independently but maintained the pretense of the independence of the choice of the basis and generating matrices.

For linearly constrained problems, we will ignore the basis—i.e., we will take $B = I$—and work directly in terms of the pattern P_k (for many of the classical pattern search methods for unconstrained minimization, $B = I$). We do this because, as with bound constrained problems, we need to place restrictions on P_k itself and it is simplest just to ignore B.

A *pattern* P_k is a matrix $P_k \in \mathbf{Z}^{n \times p_k}$. We will place a lower bound on p_k, but it has no upper bound. To obtain the lower bound, we begin by partitioning the pattern

(1) $s_k \in \Delta_k P_k = \Delta_k\,[\Gamma_k\ L_k]$.
(2) $(x_k + s_k) \in \Omega$.
(3) If $\min\{\, f(x_k + y) \mid y \in \Delta_k \Gamma_k$ and $(x_k + y) \in \Omega \,\} < f(x_k)$,
then $f(x_k + s_k) < f(x_k)$.

FIG. 3.3. *Hypotheses on the result of the linearly constrained exploratory moves.*

matrix into components

$$P_k \quad = \quad [\quad \Gamma_k \quad L_k \quad].$$

In section 3.5 we will describe certain geometrical restrictions that $\Gamma_k \in \mathbf{Z}^{n \times r_k}$ must satisfy. For now we simply observe that $r_k \geq n + 1$. We will have more to say on r_k in section 8, in particular how we may reasonably expect to arrange $r_k \leq 2n$. We also will have occasion to refer to Γ_k as the *core* pattern since it represents the set of sufficient directions required for the analysis. We require that $L_k \in \mathbf{Z}^{n \times (p_k - r_k)}$ contains at least one column, a column of zeroes; this is purely a convenience we will explain shortly. Additional columns of L_k may be present to allow algorithmic refinements but play little active role in the analysis. Given these definitions of the components Γ_k and L_k of P_k, it should be clear that $p_k \geq r_k + 1 > n + 1$.

We define a *trial step* s_k^i to be any vector of the form $s_k^i = \Delta_k c_k^i$, where $\Delta_k \in \mathbf{R}$, $\Delta_k > 0$, and c_k^i denotes a column of $P_k = [c_k^1 \cdots c_k^{p_k}]$. We call a trial step s_k^i *feasible* if $(x_k + s_k^i) \in \Omega$. At iteration k, a *trial point* is any point of the form $x_k^i = x_k + s_k^i$, where x_k is the current iterate. We will accept a step s_k from among the trial steps s_k^i that have been considered to form the next iterate $x_{k+1} = x_k + s_k$. The inclusion of a column of zeroes in L_k allows for a zero step, i.e., $x_{k+1} = x_k$. Among other things, this ensures that if x_k is feasible, then the pattern P_k always contains at least one step—the zero step—that makes it possible to produce a feasible x_{k+1}.

3.2. The linearly constrained exploratory moves.

Pattern search methods proceed by conducting a series of *exploratory moves* about the current iterate x_k to choose a new iterate $x_{k+1} = x_k + s_k$ for some feasible step s_k determined during the course of the exploratory moves. The hypotheses on the result of the linearly constrained exploratory moves, given in Figure 3.3, allow a broad choice of exploratory moves while ensuring the properties required to prove convergence. In the analysis of pattern search methods, these hypotheses assume the role played by sufficient decrease conditions in quasi-Newton methods.

We also observe that the last of the hypotheses is not as restrictive as may first appear. Another way to state the condition is to say that the exploratory moves are allowed to return the zero step only if there is no feasible step $s_k \in \Delta_k \Gamma_k$ that yields improvement over $f(x_k)$. Otherwise, we may accept *any* feasible $s_k \in \Delta_k P_k$ for which $f(x_k + s_k) < f(x_k)$. Thus, in the unconstrained example depicted in Figure 3.1, while we look successively in each of the directions defined by the unit basis vectors for x_+ for which $f(x_+) < f(x_k)$, we are free to abandon the search the moment we find such an x_+. This means that if we are lucky, we can get by with as few as one evaluation of $f(x)$ in an iteration. The same holds for the example with linear constraints. This economy is possible because we do not enforce a sufficient decrease condition on the improvement realized in the objective.

```
Let $x_0 \in \Omega$ and $\Delta_0 > 0$ be given.
For $k = 0, 1, \ldots,$
    (a) Compute $f(x_k)$.
    (b) Determine a step $s_k$ using a linearly constrained exploratory moves algorithm.
    (c) If $f(x_k + s_k) < f(x_k)$, then $x_{k+1} = x_k + s_k$. Otherwise $x_{k+1} = x_k$.
    (d) Update $P_k$ and $\Delta_k$.
```

FIG. 3.4. *The generalized pattern search method for linearly constrained problems.*

There are two possibilities:

(a) If $f(x_k + s_k) \geq f(x_k)$ (i.e., the iteration is unsuccessful), then $\Delta_{k+1} = \theta_k \Delta_k$, where $\theta_k \in (0, 1)$.

(b) If $f(x_k + s_k) < f(x_k)$ (i.e., the iteration is successful), then $\Delta_{k+1} = \lambda_k \Delta_k$, where $\lambda_k \in [1, +\infty)$.

The parameters θ_k and λ_k are not allowed to be arbitrary but must be of the following particular form. Let $\tau \in \mathbf{Q}$, $\tau > 1$, and $\{w_0, \ldots, w_L\} \subset \mathbf{Z}$, $w_0 < 0$, $w_L \geq 0$, and $w_0 < w_1 < \cdots < w_L$, where $L > 1$ is independent of k. Then θ_k must be of the form τ^{w_i} for some $w_i \in \{w_0, \ldots, w_L\}$ such that $w_i < 0$, while λ_k must be of the form τ^{w_j} for some $w_j \in \{w_0, \ldots, w_L\}$ such that $w_j \geq 0$.

FIG. 3.5. *Updating* Δ_k.

3.3. The generalized pattern search method for linearly constrained problems. Figure 3.4 states the general pattern search method for minimization with linear constraints. To define a particular pattern search method, we must specify the pattern P_k, the linearly constrained exploratory moves to be used to produce a feasible step s_k, and the algorithms for updating P_k and Δ_k. We defer a discussion of stopping criteria to section 7.

3.4. The updates. Figure 3.5 specifies the rules for updating Δ_k. The aim of the update of Δ_k is to force decrease in $f(x)$. An iteration with $f(x_k + s_k) < f(x_k)$ is *successful;* otherwise, the iteration is *unsuccessful.* As is characteristic of pattern search methods, a step need only yield *simple* decrease, as opposed to *sufficient* decrease, in order to be acceptable.

We will sometimes refer to outcome (a) in Figure 3.5, a reduction of Δ_k, as backtracking, in a loose analogy to backtracking in line-search methods. Note that part (3) in Figure 3.3 prevents backtracking, and thus shorter steps, unless we first sample $f(x)$ in a suitably large set of directions from x_k and find no improvement. This is at the heart of the global convergence analysis.

3.5. Geometrical restrictions on the pattern. In the case of linearly constrained minimization, the core pattern Γ_k must reflect the geometry of the feasible region when the iterates are near the boundary. Pattern search methods do not approximate the gradient of the objective but instead rely on a sufficient sampling of $f(x)$ to ensure that feasible descent will not be overlooked if the pattern is sufficiently small. We now discuss the geometrical restrictions on the pattern that make this possible in the presence of linear constraints.

3.5.1. The geometry of the nearby boundary. We begin with the relevant features of the boundary of the feasible region near an iterate. Let a_i^T be the ith row of the constraint matrix A in (1.1), and define

$$\begin{aligned} A_{\ell_i} &= \{ \ x \ \mid \ a_i^T x = \ell_i \ \} , \\ A_{u_i} &= \{ \ x \ \mid \ a_i^T x = u_i \ \} . \end{aligned}$$

These are the boundaries of the half-spaces whose intersection defines Ω. Set

$$\begin{aligned} \partial\Omega_{\ell_i}(\varepsilon) &= \{ \ x \in \Omega \ \mid \ \mathrm{dist}(x, A_{\ell_i}) \le \varepsilon \ \} , \\ \partial\Omega_{u_i}(\varepsilon) &= \{ \ x \in \Omega \ \mid \ \mathrm{dist}(x, A_{u_i}) \le \varepsilon \ \} , \end{aligned}$$

and

$$\partial\Omega(\varepsilon) = \bigcup_{i=1}^{m} \left(\partial\Omega_{\ell_i}(\varepsilon) \cup \partial\Omega_{u_i}(\varepsilon) \right) .$$

Given $x \in \Omega$ and $\varepsilon \ge 0$ we define the index sets

$$I_\ell(x, \varepsilon) = \{ \ i \ \mid \ x \in \partial\Omega_{\ell_i}(\varepsilon) \ \} , \tag{3.1}$$

$$I_u(x, \varepsilon) = \{ \ i \ \mid \ x \in \partial\Omega_{u_i}(\varepsilon) \ \} . \tag{3.2}$$

For $i \in I_\ell(x, \varepsilon)$ we define

$$\nu_{\ell_i}(x, \varepsilon) = -a_i, \tag{3.3}$$

and for $i \in I_u(x, \varepsilon)$ we define

$$\nu_{u_i}(x, \varepsilon) = a_i. \tag{3.4}$$

These are the outward pointing normals to the corresponding faces of Ω.

Given $x \in \Omega$ we will define the cone $K(x, \varepsilon)$ to be the cone generated by the vectors $\nu_{\ell_i}(x, \varepsilon)$ for $i \in I_\ell(x, \varepsilon)$ and $\nu_{u_i}(x, \varepsilon)$ for $i \in I_u(x, \varepsilon)$. Recall that a convex cone K is called finitely generated if there exists a finite set of vectors $\{v_1, \dots, v_r\}$ (the generators of K) such that

$$K = \left\{ v \ \mid \ v = \sum_{i=1}^{r} \lambda_i v_i, \ \lambda_i \ge 0, \ i = 1, \dots, r \right\} .$$

Finally, let $P_{K(x,\varepsilon)}$ and $P_{K^\circ(x,\varepsilon)}$ be the projections (in the Euclidean norm) onto $K(x, \varepsilon)$ and $K^\circ(x, \varepsilon)$, respectively. By convention, if $K(x, \varepsilon) = \emptyset$, then $K^\circ(x, \varepsilon) = \mathbf{R}^n$. Observe that $K(x, 0)$ is the cone of normals to Ω at x, while $K^\circ(x, 0)$ is the cone of tangents to Ω at x.

The cone $K(x, \varepsilon)$, illustrated in Figure 3.6, is the cone generated by the normals to the faces of the boundary within distance ε of x. Its polar $K^\circ(x, \varepsilon)$ is important because if $\varepsilon > 0$ is sufficiently small we can proceed from x along all directions in $K^\circ(x, \varepsilon)$ for a distance $\delta > 0$, depending only on ε, and still remain inside the feasible region. This is not the case for directions in the tangent cone of the feasible region at x, since the latter cone does not reflect the proximity of the boundary for points close to, but not on, the boundary.

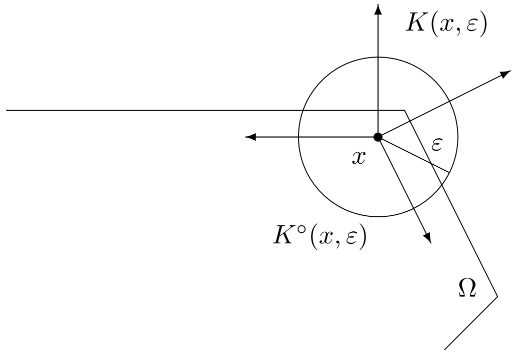


FIG. 3.6. *The situation near the boundary.*

3.5.2. Specifying the pattern. We now state the geometrical restriction on the pattern P_k. We require the core pattern Γ_k of P_k to include generators for all of the cones $K^\circ(x_k, \varepsilon)$, $0 \leq \varepsilon \leq \varepsilon^*$, for some $\varepsilon^* > 0$ that is independent of k.

We also require that the collection $\mathbf{\Gamma} = \cup_{k=0}^{\infty} \Gamma_k$ be finite. Thus (and this is the real point), $\mathbf{\Gamma}$ will contain a finite set of generators for all of the cones $K^\circ(x_k, \varepsilon)$, $0 \leq \varepsilon \leq \varepsilon^*$. Note that as ε varies from 0 to ε^* there is only a finite number of distinct cones $K(x_k, \varepsilon)$ since there is only a finite number of faces of Ω. This means that the finite cardinality of $\mathbf{\Gamma}$ is not an issue. There remains the question of constructing sets of generators that are also integral; we address the issue of constructing suitable patterns, by implicitly estimating ε^*, in section 8. However, we will see that the construction is computationally tractable and, in many cases, is not particularly difficult. We close by noting that the condition that Γ_k contains generators of $K^\circ(x_k, \varepsilon)$ implies that Γ_k contains generators for all tangent cones to Ω at all feasible points near x_k.

If x_k is "far" from the boundary in the sense that $K(x_k, \varepsilon) = \emptyset$, then $K^\circ(x_k, \varepsilon) = \mathbf{R}^n$ and a set of generators for $K^\circ(x_k, \varepsilon)$ is simply a positive spanning set for $\mathbf{R}^n$ [5, 9]. (A positive spanning set is a set of generators for a cone in the case that the cone is a vector space.) If the iterate is suitably in the interior of Ω, the algorithm will look like a pattern search algorithm for unconstrained minimization [9], as it ought. On the other hand, if x_k is near the boundary, $K(x_k, \varepsilon) \neq \emptyset$ and the pattern must conform to the local geometry of the boundary, as depicted in Figures 3.2 and 3.6.

The design of the pattern reflects the fundamental challenge in the development of constrained pattern search methods. We do not have an estimate of the gradient of the objective and, consequently, we have no idea which constraints locally limit feasible improvement in $f(x)$. In a projected gradient method one has the gradient and can detect the local interaction of the descent direction with the boundary by conducting a line-search along the projected gradient path. In derivative-free methods such as pattern search we must have a sufficiently rich set of directions in the pattern since any subset of the nearby faces may be the ones that limit the feasibility of the steepest descent direction, which is itself unavailable for use in the detection of the important nearby constraints. Nonetheless, in section 4 we are able to outline the conditions for global convergence, and in section 8 we outline practical implementations of pattern search methods for linearly constrained minimization.

4. Convergence analysis. In this section we state the convergence results for pattern search methods for linearly constrained minimization. We defer the proofs of these results to section 6, after reviewing existing results for pattern search methods in section 5.

We first summarize features of the algorithm whose statements are scattered throughout section 3.

Hypothesis 0.

(1) *The pattern* $P_k = [\,\Gamma_k \; L_k\,] \in \mathbf{Z}^{n\times p_k}$, $p_k > n+1$, *so that all search directions are integral vectors scaled by* $\Delta_k \in \mathbf{R}^n$. *All steps* s_k *are then required to be of the form* $\Delta_k c_k^i$, *where* c_k^i *denotes a column of* $P_k = [c_k^1 \cdots c_k^{p_k}]$.

(2) *The core pattern* $\Gamma_k \in \mathbf{Z}^{n\times r_k}$, $r_k \geq n+1$, *belongs to* $\mathbf{\Gamma}$, *where* $\mathbf{\Gamma}$ *is a finite set of integral matrices, the columns of which include generators for all of the cones* $K^\circ(x_k, \varepsilon)$, $0 \leq \varepsilon \leq \varepsilon^*$, *for some* $\varepsilon^* > 0$ *that is independent of* k.

(3) *The matrix* $L_k \in \mathbf{Z}^{n\times(p_k - r_k)}$ *contains at least one column, a column of zeroes.*

(4) *The rules for updating* Δ_k *are as given in Figure* 3.5.

(5) *The exploratory moves algorithm returns steps that satisfy the conditions given in Figure* 3.3.

We now add some additional hypotheses on the problem (1.1).

Hypothesis 1. *The constraint matrix* A *is rational.*

Hypothesis 1 is a simple way of ensuring that we can find a rational lattice that fits inside the feasible region in a suitable way. In particular, the rationality of A ensures that we can construct Γ_k satisfying part (2) of Hypothesis 0, as discussed further in section 8.

Hypothesis 2. *The set* $L_\Omega(x_0) = \{\, x \in \Omega \;\mid\; f(x) \leq f(x_0)\,\}$ *is compact.*

Hypothesis 3. *The objective* $f(x)$ *is continuously differentiable on an open neighborhood* D *of* $L_\Omega(x_0)$.

We next remind the reader that unless otherwise noted norms are assumed to be the Euclidean norm and that we denote by $g(x)$ the gradient of the objective f at x. Let P_Ω be the projection onto Ω. For feasible x, let

$$q(x) = P_\Omega(x - g(x)) - x.$$

Note that because the projection P_Ω is nonexpansive $q(x)$ is continuous on Ω. The following proposition summarizes properties of q that we need, particularly the fact that x is a constrained stationary point for (1.1) if and only if $q(x) = 0$. The results are classical; see section 2 of [6], for instance.

Proposition 4.1. *Let* $x \in \Omega$. *Then*

$$\|\, q(x)\,\| \leq \|\, g(x)\,\|$$

and x *is a stationary point for problem* (1.1) *if and only if* $q(x) = 0$.

We can now state the first convergence result for the general pattern search method for linearly constrained minimization.

Theorem 4.2. *Assume Hypotheses* 0–3 *hold. Let* $\{x_k\}$ *be the sequence of iterates produced by the generalized pattern search method for linearly constrained minimization (Figure* 3.4*). Then*

$$\liminf_{k\to+\infty} \|\, q(x_k)\,\| = 0\,.$$

As an immediate corollary, we have the following result.

Corollary 4.3. *There exists a limit point of* $\{x_k\}$ *that is a constrained stationary point for* (1.1).

Note that Hypothesis 2 guarantees the existence of one such limit point.

(1) $s_k \in \Delta_k P_k = \Delta_k \, [\Gamma_k \;\, L_k]$.
(2) $(x_k + s_k) \in \Omega$.
(3) If $\min \{ \, f(x_k + y) \;\mid\; y \in \Delta_k \Gamma_k$ and $(x_k + y) \in \Omega \, \} < f(x_k)$,
then $f(x_k + s_k) \le \min \{ \, f(x_k + y) \;\mid\; y \in \Delta_k \Gamma_k$ and $(x_k + y) \in \Omega \, \}$.

FIG. 4.1. *Strong hypotheses on the result of the linearly constrained exploratory moves.*

We can strengthen Theorem 4.2, in the same way that we do in the unconstrained and bound constrained cases [8, 15], by adding the following hypotheses.

HYPOTHESIS 4. *The columns of the pattern matrix P_k remain bounded in norm; i.e., there exists $c_4 > 0$ such that for all k, $c_4 > \| \, c_k^i \, \|$, for all $i = 1, \ldots, p_k$.*

HYPOTHESIS 5. *The original hypotheses on the result of the linearly constrained exploratory moves are replaced with the stronger version given in Figure* 4.1.

The third condition is stronger than the hypotheses on the result of the linearly constrained exploratory moves given in Figure 3.3. Now we tie the amount of decrease in $f(x)$ that must be realized by the step s_k to the amount of decrease that could be realized were we to rely on the local behavior of the linearly constrained problem, as defined by the columns of Γ_k.

HYPOTHESIS 6. *We have* $\lim_{k \to +\infty} \Delta_k = 0$.

Note that we do not require Δ_k to be monotone nonincreasing.

Then we obtain the following stronger results.

THEOREM 4.4. *Assume Hypotheses* 0–6 *hold. Then for the sequence of iterates $\{x_k\}$ produced by the generalized pattern search method for linearly constrained minimization (Figure* 3.4*),*

$$\lim_{k \to +\infty} \| \, q(x_k) \, \| = 0 \, .$$

COROLLARY 4.5. *Every limit point of $\{x_k\}$ is a constrained stationary point for* (1.1).

Again, Hypothesis 2 guarantees the existence of at least one such limit point.

5. Results from the standard theory.

We need the following results from the analysis of pattern search methods in the unconstrained case. For the proofs, see [15]; these results generalize to the linearly constrained case without change. Theorem 5.1 is central to the convergence analysis for pattern search methods; it allows us to prove convergence for these methods in the absence of any sufficient decrease condition.

THEOREM 5.1. *Any iterate x_N produced by a generalized pattern search method for linearly constrained problems (Figure* 3.4*) can be expressed in the form*

$$x_N = x_0 + \left(\beta^{r_{LB}} \alpha^{-r_{UB}} \right) \Delta_0 B \sum_{k=0}^{N-1} z_k, \tag{5.1}$$

where

- *x_0 is the initial guess;*
- *$\beta / \alpha \equiv \tau$, with $\alpha, \beta \in \mathbf{N}$ and relatively prime and τ is as defined in the rules for updating Δ_k (Figure* 3.5*);*
- *r_{LB} and r_{UB} are integers depending on N, where $r_{LB} \le 0$ and $r_{UB} \ge 0$;*
- *Δ_0 is the initial choice for the step-length control parameter;*

- *B is the basis matrix; and*
- *$z_k \in \mathbf{Z}^n$, $k = 0, \dots, N-1$.*

Recall that in the case of linearly constrained minimization we set $B = I$.

The quantity Δ_k regulates step-length as indicated by the following.

LEMMA 5.2. (i) *There exists a constant $\zeta_* > 0$, independent of k, such that for any trial step $s_k^i \neq 0$ produced by a generalized pattern search method for linearly constrained problems we have $\| s_k^i \| \geq \zeta_* \Delta_k$.*

(ii) *Under Hypothesis* 4, *there exists a constant $\psi_* > 0$, independent of k, such that for any trial step s_k^i produced by a generalized pattern search method for linearly constrained problems we have $\Delta_k \geq \psi_* \| s_k^i \|$.*

In the case of pattern search for linearly constrained problems, P_k is integral. Since $s_k^i \in \Delta_k P_k$, we may take $\zeta_* = 1$.

6. Proof of Theorems 4.2 and 4.4.

We now proceed with the proofs of the two main results stated in section 4. Essential to our arguments are some results concerning the geometry of polyhedra. We defer the treatment of these technical details to section 10.

Given an iterate x_k, let $g_k = g(x_k)$ and $q_k = P_\Omega(x_k - g_k) - x_k$. Let $B(x, \delta)$ be the ball with center x and radius δ, and let ω denote the following modulus of continuity of g. Given $x \in L_\Omega(x_0)$ and $\varepsilon > 0$,

$$\omega(x, \varepsilon) = \sup \{ \, \delta > 0 \mid B(x, \delta) \subset D \text{ and } \| g(y) - g(x) \| < \varepsilon \text{ for all } y \in B(x, \delta) \, \}.$$

Then we have this elementary proposition concerning descent directions, whose proof we omit (see [8]).

PROPOSITION 6.1. *Let $s \in \mathbf{R}^n$ and $x \in L_\Omega(x_0)$. Assume that $g(x) \neq 0$ and $g(x)^T s \leq -\varepsilon \| s \|$ for some $\varepsilon > 0$. Then, if $\| s \| < \omega(x, \varepsilon/2)$,*

$$f(x+s) - f(x) \leq -\frac{\varepsilon}{2} \| s \|.$$

The next result is the crux of the convergence analysis. Using the results in section 10, we show that if we are not at a constrained stationary point, then the pattern always contains a descent direction along which we remain feasible for a sufficiently long distance.

Let Γ^* be the maximum norm of any column of the matrices in the set $\mathbf{\Gamma}$, where $\mathbf{\Gamma}$ is as in section 3.1 and section 3.5. If $\Delta_k \leq \delta / \Gamma^*$, then $\| s_k^i \| \leq \delta$ for all $s_k^i \in \Delta_k \Gamma_k$. Also define

$$h = \min_{\substack{1 \leq i \leq m \\ \ell_i \neq u_i}} \frac{u_i - \ell_i}{\| a_i \|}. \tag{6.1}$$

This is the minimum distance between the faces of Ω associated with the constraints that are not equality constraints. Finally, $\| g_k \|$ is bounded on $L_\Omega(x_0)$ by hypothesis; let g^* be an upper bound for $\| g_k \|$.

PROPOSITION 6.2. *There exist $r_{6.2} > 0$ and $c_{6.2} > 0$ such that if $\eta > 0$, $\| q_k \| \geq \eta$, and $\Delta_k \leq r_{6.2} \eta^2$, then there is a trial step s_k^i defined by a column of $\Delta_k \Gamma_k$ for which, given $x_k \in \Omega$, $(x_k + s_k^i) \in \Omega$ and*

$$-g_k^T s_k^i \geq c_{6.2} \| q_k \| \| s_k^i \|.$$

Proof. Let

$$r = \min(\varepsilon^* / (g^*)^2, \; h / (2 (g^*)^2), \; r_{10.7}),$$

where ε^* is the constant introduced in section 3.5.2, h is given by (6.1), and $r_{10.7}$ is the constant that appears in Proposition 10.7.

Now consider $\varepsilon = r\eta^2$. From Proposition 4.1, $\| q_k \| \leq \| g_k \| \leq g^*$, so our choice of r ensures that ε is sufficiently small that

(1) $\varepsilon \leq \varepsilon^*$,
(2) $\varepsilon \leq h/2$, and
(3) $\varepsilon \leq r_{10.7}\eta^2$.

Because of this last fact, (3), we may apply Proposition 10.7 to $w = -g_k$ with $x = x_k$ and $\gamma = g^*$ to obtain

$$\| P_{K^\circ(x_k,\varepsilon)}(-g_k) \| \geq c_{10.7}\| q_k \|. \tag{6.2}$$

Meanwhile, since we require the core pattern Γ_k of P_k to include generators for all of the cones $K^\circ(x_k, \delta)$, $\delta \leq \varepsilon^*$, then, because $\varepsilon \leq \varepsilon^*$, some subset of the core pattern steps s_k^i forms a set of generators for $K^\circ(x_k, \varepsilon)$. Consequently, by virtue of (6.2) we may invoke Corollary 10.4: for some $s_k^i \in \Delta_k\Gamma_k$ we have

$$-g_k^T s_k^i \geq c_{10.4} \; \| P_{K^\circ(x_k,\varepsilon)}(-g_k) \| \; \| s_k^i \|. \tag{6.3}$$

From (6.3) we then obtain

$$-g_k^T s_k^i \geq c_{10.4} \; c_{10.7} \; \| q_k \| \; \| s_k^i \| = c_{6.2} \; \| q_k \| \; \| s_k^i \|,$$

where $c_{6.2} = c_{10.4} \; c_{10.7}$. Thus, we are assured of a descent direction inside the pattern.

Now we must show that we can take a sufficiently long step along this descent direction and remain feasible. Define

$$r_{6.2} = r/(2\Gamma^*)$$

and consider what happens when $\Delta_k \leq r_{6.2}\eta^2$. We have $\Delta_k \leq \varepsilon/(2\Gamma^*)$; and since $s_k^i \in \Delta_k\Gamma_k$, we have $\| s_k^i \| \leq \varepsilon/2$. Since $s_k^i \in K^\circ(x_k, \varepsilon)$, and $\varepsilon \leq h/2$ by (2) above, we can apply Proposition 10.8 to $w = s_k^i$ to conclude that $(x_k + s_k^i) \in \Omega$. □

We now show that if we are not at a constrained stationary point, we can always find a step in the pattern that both is feasible and yields improvement in the objective.

PROPOSITION 6.3. *Given any $\eta > 0$, there exists $r_{6.3} > 0$, independent of k, such that if $\Delta_k \leq r_{6.3}\eta^2$ and $\| q_k \| \geq \eta$, the pattern search method for linearly constrained minimization will find an acceptable step s_k; i.e., $f(x_{k+1}) < f(x_k)$ and $x_{k+1} = (x_k + s_k) \in \Omega$.*

If, in addition, the columns of the generating matrix remain bounded in norm and we enforce the strong hypotheses on the results of the linearly constrained exploratory moves (Hypotheses 4 and 5), then, given any $\eta > 0$, there exists $\sigma > 0$, independent of k, such that if $\Delta_k < r_{6.3}\eta^2$ and $\| q_k \| \geq \eta$, then

$$f(x_{k+1}) \leq f(x_k) - \sigma \| q_k \| \| s_k \|.$$

Proof. Proposition 6.2 assures us of the existence of $r_{6.2}$ and a step s_k^i defined by a column of $\Delta_k\Gamma_k$ such that $(x_k + s_k^i) \in \Omega$ and

$$g_k^T s_k^i \leq -c_{6.2}\| q_k \| \| s_k^i \|,$$

provided $\Delta_k \leq r_{6.2}\eta^2$. Also, since $g(x)$ is uniformly continuous on $L_\Omega(x_0)$ and $L_\Omega(x_0)$ is a compact subset of the open set D on which $f(x)$ is continuously differentiable, there exists $\omega_* > 0$ such that

$$\omega\left(x_k, \frac{c_{6.2}}{2}\eta\right) \geq \omega_*$$

for all k for which $\| q_k \| \geq \eta$.

Now define

$$r_{6.3} = \min\left(r_{6.2}, \omega_*/(\Gamma^*(g^*)^2)\right)$$

and suppose $\| q_k \| \geq \eta$ and $\Delta_k \leq r_{6.3}\eta^2$. We have

$$\| s_k^i \| \leq \Delta_k \Gamma^* \leq \omega_* \leq \omega\left(x_k, \frac{c_{6.2}}{2}\| q_k \|\right).$$

Hence, by Proposition 6.1,

$$f(x_k + s_k^i) - f(x_k) \leq -\frac{c_{6.2}}{2}\| q_k \|\| s_k^i \|.$$

Thus, when $\Delta_k \leq r_{6.3}\eta^2$, $f(x_k + s_k^i) < f(x_k)$ for at least one feasible $s_k^i \in \Delta_k \Gamma_k$. The hypotheses on linearly constrained exploratory moves guarantee that if

$$\min\{\, f(x_k + y) \;\mid\; y \in \Delta_k \Gamma_k,\; (x_k + y) \in \Omega \,\} < f(x_k),$$

then $f(x_k+s_k) < f(x_k)$ and $(x_k+s_k) \in \Omega$. This proves the first part of the proposition.

If, in addition, we enforce the strong hypotheses on the result of the linearly constrained exploratory moves, then we actually have

$$f(x_{k+1}) - f(x_k) \leq -\frac{c_{6.2}}{2}\| q_k \|\| s_k^i \|.$$

Part (i) of Lemma 5.2 then ensures that

$$f(x_{k+1}) \leq f(x_k) - \frac{c_{6.2}}{2}\zeta_* \Delta_k \| q_k \|.$$

Applying part (ii) of Lemma 5.2, we arrive at

$$f(x_{k+1}) \leq f(x_k) - \frac{c_{6.2}}{2}\zeta_* \psi_* \| q_k \|\| s_k \|.$$

This yields the second part of the proposition with $\sigma = (c_{6.2}/2)\zeta_*\psi_*$. □

COROLLARY 6.4. *If* $\liminf_{k\to+\infty} \| q_k \| \neq 0$, *then there exists a constant* $\Delta_* > 0$ *such that for all* k, $\Delta_k \geq \Delta_*$.

Proof. By hypothesis, there exists N and $\eta > 0$ such that for all $k > N$, $\| q_k \| \geq \eta$. By Proposition 6.3, we can find $\delta = r_{6.3}\eta^2$ such that if $k > N$ and $\Delta_k < \delta$, then we will find an acceptable step. In view of the update of Δ_k given in Figure 3.5, we are assured that for all $k > N$, $\Delta_k \geq \min(\Delta_N, \tau^{w_0}\delta)$. We may then take $\Delta_* = \min\{\Delta_0, \ldots, \Delta_N, \tau^{w_0}\delta\}$. □

The next theorem combines the strict algebraic structure of the iterates with the simple decrease condition of the generalized pattern search algorithm for linearly constrained problems, along with the rules for updating Δ_k, to tell us the limiting behavior of Δ_k.

THEOREM 6.5. *Under Hypotheses* 0–3, $\liminf_{k\to+\infty} \Delta_k = 0$.

Proof. The proof is like that of Theorem 3.3 in [15]. Suppose $0 < \Delta_{LB} \leq \Delta_k$ for all k. Using the rules for updating Δ_k, found in Figure 3.5, it is possible to write Δ_k as $\Delta_k = \tau^{r_k}\Delta_0$, where $r_k \in \mathbf{Z}$.

The hypothesis that $\Delta_{LB} \leq \Delta_k$ for all k means that the sequence $\{\tau^{r_k}\}$ is bounded away from zero. Meanwhile, we also know that the sequence $\{\Delta_k\}$ is bounded above because all the iterates x_k must lie inside the set $L_\Omega(x_0) = \{\ x \in \Omega \ \mid\ f(x) \le f(x_0)\ \}$ and the latter set is compact; part (i) of Lemma 5.2 (which is a consequence of the rules for updating Δ_k) then guarantees an upper bound Δ_{UB} for $\{\Delta_k\}$. This, in turn, means that the sequence $\{\tau^{r_k}\}$ is bounded above. Consequently, the sequence $\{\tau^{r_k}\}$ is a finite set. Equivalently, the sequence $\{r_k\}$ is bounded above and below.

Next we recall the exact identity of the quantities r_{LB} and r_{UB} in Theorem 5.1; the details are found in the proof of Theorem 3.3 in [15]. In the context of Theorem 5.1,

$$r_{LB} = \min_{0 \le k < N} \{r_k\}, \qquad r_{UB} = \max_{0 \le k < N} \{r_k\}.$$

If, in the matter at hand, we let

$$r_{LB} = \min_{0 \le k < +\infty} \{r_k\}, \qquad r_{UB} = \max_{0 \le k < +\infty} \{r_k\}, \tag{6.4}$$

then (5.1) holds for the bounds given in (6.4) and we see that for all k, x_k lies in the translated integer lattice G generated by x_0 and the columns of $\beta^{r_{LB}} \alpha^{-r_{UB}} \Delta_0 I$.

The intersection of the compact set $L_\Omega(x_0)$ with the lattice G is finite. Thus, there must exist at least one point x_* in the lattice for which $x_k = x_*$ for infinitely many k.

We now appeal to the simple decrease condition in part (c) of Figure 3.4, which guarantees that a lattice point cannot be revisited infinitely many times since we accept a new step s_k if and only if $f(x_k) > f(x_k + s_k)$ and $(x_k + s_k) \in \Omega$. Thus there exists an N such that for all $k \ge N$, $x_k = x_*$, which implies $f(x_k) = f(x_k + s_k)$.

We now appeal to the algorithm for updating Δ_k (part (a) in Figure 3.5) to see that $\Delta_k \to 0$, thus leading to a contradiction. □

6.1. The proof of Theorem 4.2. The proof is like that of Theorem 3.5 in [15]. Suppose that $\liminf_{k \to +\infty} \|\ q_k\ \| \ne 0$. Then Corollary 6.4 tells us that there exists $\Delta_* > 0$ such that for all k, $\Delta_k \ge \Delta_*$. But this contradicts Theorem 6.5.

6.2. The proof of Theorem 4.4. The proof, also by contradiction, follows that of Theorem 3.7 in [15]. Suppose $\limsup_{k \to +\infty} \|\ q_k\ \| \ne 0$. Let $\varepsilon > 0$ be such that there exists a subsequence $\|\ q(x_{m_i})\ \| \ge \varepsilon$. Since

$$\liminf_{k \to +\infty} \|\ q_k\ \| = 0,$$

given any $0 < \eta < \varepsilon$, there exists an associated subsequence l_i such that

$$\|\ q_k\ \| \quad \ge \quad \eta \qquad \text{for} \qquad m_i \le k < l_i, \qquad \|\ q(x_{l_i})\ \| \quad < \quad \eta.$$

Since $\Delta_k \to 0$, we can appeal to Proposition 6.3 to obtain for $m_i \le k < l_i$, i sufficiently large,

$$f(x_k) - f(x_{k+1}) \quad \ge \quad \sigma \|\ q_k\ \| \|\ s_k\ \| \quad \ge \quad \sigma\eta \|\ s_k\ \|,$$

where $\sigma > 0$. Summation then yields

$$f(x_{m_i}) - f(x_{l_i}) \quad \ge \quad \textstyle\sum_{k=m_i}^{l_i} \sigma\eta \|\ s_k\ \| \quad \ge \quad \sigma\eta \|\ x_{m_i} - x_{l_i}\ \|.$$

Since f is bounded below on the set $L_\Omega(x_0)$, we know that $f(x_{m_i}) - f(x_{l_i}) \to 0$ as $i \to +\infty$, so $\|\ x_{m_i} - x_{l_i}\ \| \to 0$ as $i \to +\infty$. Then, because q is uniformly continuous, $\|\ q(x_{m_i}) - q(x_{l_i})\ \| < \eta$ for i sufficiently large. However,

$$\|\ q(x_{m_i})\ \| \quad \le \quad \|\ q(x_{m_i}) - q(x_{l_i})\ \| + \|\ q(x_{l_i})\ \| \quad \le \quad 2\eta. \tag{6.5}$$

Since (6.5) must hold for any η, $0 < \eta < \varepsilon$, we have a contradiction (e.g., try $\eta = \frac{\varepsilon}{4}$).

7. Comments on the algorithm.

We next discuss some practical aspects of pattern search algorithms for linearly constrained problems. In this section we propose some stopping criteria for these algorithms as well as examine the questions of estimating Lagrange multipliers and identifying the constraints active at a solution.

7.1. Stopping criteria.

The stopping criterion that seems most natural to us is to halt the algorithm once Δ_k falls below some prescribed tolerance Δ_*. Equivalently, one can halt once the absolute length of the steps in the core pattern falls below some prescribed tolerance δ_*.

The following proposition concerning the correlation of stationarity and the size of Δ_k lends support to this choice of a stopping criterion. The result relates $\| q_k \|$ and the Δ_k at those steps where Δ_k is reduced (i.e., where backtracking occurs); if we terminate the algorithm at such an iterate, then, if Δ_k is sufficiently small, $\| q_k \|$ will also be small. In the case of bound constraints, a similar result allows one to establish convergence for a pattern search algorithm for general nonlinearly constrained problems via inexact bound constrained minimization of the augmented Lagrangian [10]. For convenience, we assume that $\nabla f(x)$ is Lipschitz continuous. However, if we assume only that $\nabla f(x)$ is uniformly continuous on $L_\Omega(x_0)$, we can still establish a correlation between stationarity and the size of Δ_k.

PROPOSITION 7.1. *Suppose $\nabla f(x)$ is Lipschitz continuous on $L_\Omega(x_0)$ with Lipschitz constant C. There exists $c_{7.1} > 0$ for which the following holds. If x_k is an iterate at which there is an unsuccessful iteration, then*

$$\| q_k \|^2 \leq c_{7.1} \Delta_k. \tag{7.1}$$

Proof. We need only consider the situation where $\eta = \| q_k \| > 0$. There are two cases to consider. First suppose $r_{6.2} \| q_k \|^2 \leq \Delta_k$, where $r_{6.2}$ is the constant of the same name in Proposition 6.2. Then we immediately have

$$\| q_k \|^2 \leq \Delta_k / r_{6.2}. \tag{7.2}$$

On the other hand, suppose $r_{6.2} \| q_k \|^2 > \Delta_k$. By Proposition 6.2, there exists $s_k^i \in \Delta_k \Gamma_k$ such that $(x_k + s_k^i) \in \Omega$ and

$$-g_k^T s_k^i \geq c_{6.2} \| q_k \| \| s_k^i \|. \tag{7.3}$$

Since iteration k is unsuccessful, it follows from Figure 3.3 that $f(x_k + s_k^i) - f(x_k) \geq 0$ for all feasible $s_k^i \in \Delta_k \Gamma_k$. By the mean-value theorem, for some ξ in the line segment connecting x_k and $x_k + s_k^i$ we have

$$\begin{aligned} 0 &\leq f(x_k + s_k^i) - f(x_k) \\ &= \nabla f(x_k)^T s_k^i + (\nabla f(\xi) - \nabla f(x_k))^T s_k^i \\ &\leq -c_{6.2} \| q_k \| \| s_k^i \| + \| \nabla f(\xi) - \nabla f(x_k) \| \| s_k^i \|, \end{aligned}$$

where s_k^i is the step for which (7.3) holds. Thus

$$c_{6.2} \| q_k \| \leq \| \nabla f(\xi) - \nabla f(x_k) \|.$$

Using the Lipschitz constant C for $\nabla f(x)$, we obtain

$$c_{6.2} \| q_k \| \leq C \| \xi - x_k \| \leq C \Gamma^* \Delta_k,$$

where Γ^* is the maximum norm of any column of the matrices in the set $\mathbf{\Gamma}$. Thus

$$c_{6.2} \| \, q_k \, \|^2 \leq g^* C \Gamma^* \Delta_k, \tag{7.4}$$

where g^* is the upper bound on $\nabla f(x)$. The proposition then follows from (7.2) and (7.4). □

Remark. We conjecture that one can establish the estimate $\| \, q_k \, \| \leq c\Delta_k$ at unsuccessful steps. The appearance of $\| \, q_k \, \|^2$ rather than $\| \, q_k \, \|$ in (7.1) is a consequence of the appearance of η^2 in the hypotheses of Proposition 10.7, which in turn derives from the limitations of the way in which the latter proposition is proved.

May's algorithm [11], which is based on a difference approximation of feasible directions of descent, uses a difference approximation of local feasible descent in its stopping criterion. In connection with pattern search one could also attempt to do something similar, estimating $\nabla f(x)$ by either a difference approximation or a regression fit and using this information in a stopping test. However, depending on the application, the simpler stopping criterion $\Delta_k < \Delta_*$ may be preferable—for instance, if the objective is believed to be untrustworthy in its accuracy, or if $f(x)$ is not available as a numerical value and only comparison of objective values is possible.

7.2. Identifying active constraints. Another practical issue is that of identifying active constraints, as in [2, 3, 4]. A desirable feature of an algorithm for linearly constrained minimization is the identification of active constraints in a finite number of iterations; that is, if the sequence $\{x_k\}$ converges to a stationary point x_*, then in a finite number of iterations the iterates x_k land on the constraints active at x_* and remain thereafter on those constraints.

As discussed in [8] for the case of bound constraints, there are several impediments to proving such results for pattern search algorithms and showing that ultimately the iterates will land on the active constraints and remain there. For algorithms such as those considered in [2, 3, 4], this is not a problem because the explicit use of the gradient impels the iterates to do so in the neighborhood of a constrained stationary point. However, pattern search methods do not have this information, and at this point it is not clear how to avoid the possibility that these algorithms take a purely interior approach to a point on the boundary. On the other hand, the kinship of pattern search methods and gradient projection methods makes us hopeful that we may be able to devise a suitable mechanism to ensure pattern search methods also identify the active constraints in a finite number of iterations.

7.3. Estimating Lagrange multipliers. Similar limitations pertain to estimating Lagrange multipliers as do to identifying active constraints. Pattern search methods do not use an explicit estimate of $\nabla f(x)$, and one does not obtain an estimate of the Lagrange multipliers for (1.1) from the usual workings of the algorithm. Some manner of postoptimality sensitivity analysis would be required to obtain estimates of the multipliers, again, either through difference estimates or regression estimates of $\nabla f(x)$.

The authors are indebted to one of the referees for pointing out that May's algorithm [11] obtains multiplier estimates using only values of the objective at feasible points and for suggesting that the same idea could be used in our algorithm. As already mentioned in section 7.1, May's algorithm explicitly employs difference estimates of the directional derivatives of the objective in feasible directions (taking into account constraints that are nearby but perhaps not active). These directional derivatives are computed in a coordinate system corresponding to the generators of the cone of feasible directions, and one thereby obtains multiplier estimates. We sketch the idea here; for the details, see section 2.3 and Chapter 4 in [11].

Let N denote a matrix whose columns are a set of inward pointing normals to the active constraints at x. We assume that N has full column rank r. Let N^+ denote the pseudo-inverse of N, and let $n_1^T, \ldots, n_r^T$ denote the rows of N^+. One can show that the n_i are part of a set of generators for the cone tangent to the feasible region at x (see Lemma 2.1 in [11]; Proposition 8.2 below is a version of this result). If x is a constrained stationary point, then

$$\nabla f(x) - N\lambda = 0, \qquad \lambda \geq 0. \tag{7.5}$$

(The condition (7.5) is also sufficient for x to be a constrained stationary point; see Lemma 2.2 in [11].) The multipliers λ are then given by $\lambda = N^+\nabla f(x)$. Thus λ_i, the ith component of $N^+\nabla f(x)$, is simply the directional derivative $n_i^T\nabla f(x)$. Since n_i is a feasible direction, the directional derivative $n_i^T\nabla f(x)$ can be estimated using values of f only at feasible points. In May's algorithm, N^+ is computed via the QR decomposition as part of the calculation of the feasible search directions. In the case of pattern search, a construction of the requisite feasible directions is given in section 8.2.

For yet another way in which one can obtain information about multipliers from pattern search methods, see the augmented Lagrangian approach in [10].

8. Constructing patterns for problems with linear constraints.

In this section we outline practical implementations of pattern search methods for linearly constrained minimization. The details will be the subject of future work. In the process we also show that under the assumption that A is rational, one can actually construct patterns with both the algebraic properties required in section 3.1 and the geometric properties required in section 3.5.

8.1. Remarks on the general case.

We begin by showing that in general it is possible to find rational generators for the cones $K^\circ(x, \varepsilon)$. By clearing denominators we then obtain the integral vectors for $\mathbf{\Gamma}$ as required in section 3.1. The construction is an elaboration of the proof that polyhedral cones are finitely generated (see [16], for instance). The proof outlines an algorithm for the construction of generators of cones. Given a cone K we will use V to denote a matrix whose columns are generators of K

$$K = \{\ x \ \mid \ x = V\lambda,\ \lambda \geq 0\ \}.$$

PROPOSITION 8.1. *Suppose K is a cone with rational generators V. Then there exists a set of rational generators for K°.*

Proof. Suppose $w \in K^\circ$; then $(w, v) \leq 0$ for all $v \in K$. Let $v = V\lambda$, $\lambda \geq 0$. Then

$$(w\ ,\ v) = \left(P_{\mathcal{N}(V^T)}w + P_{\mathcal{N}(V^T)^\perp}w\ ,\ V\lambda\right) \leq 0,$$

where $P_{\mathcal{N}(V^T)}$ and $P_{(\mathcal{N}(V^T))^\perp}$ are the projections onto the nullspace $\mathcal{N}(V^T)$ of V^T and its orthogonal complement $\mathcal{N}(V^T)^\perp$, respectively. Since $\mathcal{N}(V^T)^\perp$ is the same as the range $\mathcal{R}(V)$ of V, we have

$$(w\ ,\ v) = \left(P_{\mathcal{R}(V)}w\ ,\ V\lambda\right) \leq 0.$$

Let N and R be rational bases for $\mathcal{N}(V^T)$ and $\mathcal{R}(V)$, respectively; these can be constructed, for instance, via reduction to row echelon form since V is rational.

Let $\{p_1, \dots, p_t\}$ be a rational positive basis for $\mathcal{N}(V^T)$. Such a positive basis can be constructed as follows. If N is $n \times r$, then if Π is a rational positive basis (with t elements) for $\mathbf{R}^r$ (e.g., $\Pi = [I \ -I]$), then $N\Pi$ is a rational positive basis for $\mathcal{N}(V^T)$.

Meanwhile, if R is a rational basis for $\mathcal{R}(V)$, then for some z we have

$$P_{\mathcal{R}(V)} w = Rz,$$

whence

$$(w\ ,\ v) = (Rz\ ,\ V\lambda) \le 0.$$

Since $\lambda^T V^T R z \le 0$ for all $\lambda \ge 0$, it follows that $V^T R z \le 0$. Let $e = (1, \dots, 1)^T$ and consider

$$C = \left\{\ z \ \mid \ V^T R z \le 0,\ e^T V^T R z \ge -1\ \right\}.$$

Since C is convex and compact, it is the convex hull of its extreme points $\{c_1, \dots, c_s\}$. Furthermore, note that the extreme points of C will define a set of generators for the cone $\left\{\ z \ \mid \ V^T R z \le 0\ \right\}$. The extreme points of C are also rational since $V^T R$ is rational; the extreme points will be solutions to systems of equations with rational coefficients. These extreme points, which are the vertices of the polyhedron C, can be computed by any number of vertex enumeration techniques (e.g., see [1] and the references cited therein).

Returning to $w \in K^\circ$, we see that we can express w as a positive linear combination of the vectors $\{p_1, \dots, p_t, c_1, \dots, c_s\}$. Moreover, by construction the latter vectors are rational. □

8.2. The nondegenerate case. As we have seen, the construction of sets of generators for cones is nontrivial and is related to the enumeration of vertices of polyhedra. However, in the case of nondegeneracy—the absence of any point on the boundary at which the set of binding constraints is linearly dependent—we can compute the required generators in a straightforward way. This case is handled in [11] by using the QR factorization to derive the search directions. Because we require rational search directions, we use the LU factorization (reduction to row echelon form, to be more precise) since the latter can be done in rational arithmetic.

The following proposition shows that once we have identified a cone $K(x_k, \delta)$ with a linearly independent set of generators, we can construct generators for all the cones $K(x_k, \varepsilon)$, $0 \le \varepsilon \le \delta$.

PROPOSITION 8.2. *Suppose that for some δ, $K(x, \delta)$ has a linearly independent set of rational generators V. Let N be a rational positive basis for the nullspace of V^T.*

Then for any ε, $0 \le \varepsilon \le \delta$, a set of rational generators for $K^\circ(x, \varepsilon)$ can be found among the columns of N, $V(V^T V)^{-1}$, and $-V(V^T V)^{-1}$.

Proof. Given $x \in \Omega$ and $\delta > 0$, let $K = K(x, \delta)$. Suppose $w \in K^\circ$; then $(w, v) \le 0$ for all $v \in K$. Let $v = V\lambda$, $\lambda \ge 0$. Since V has full column rank, we have

$$(w, v) = ((I - V(V^T V)^{-1} V^T) w + V(V^T V)^{-1} V^T w, V\lambda) \le 0$$

or $(V^T w, \lambda) \le 0$ for all $\lambda \ge 0$. Let $\xi = V^T w$; then we have $(\xi, \lambda) \le 0$ for all $\lambda \ge 0$, so $\xi \le 0$.

The matrix N is a positive basis for the range of $I - V(V^T V)^{-1} V^T$, since the latter subspace is the same as the nullspace of V^T. Then any $w \in K^\circ$ can be written in the form

$$w = N\zeta - V(V^TV)^{-1}\xi,$$

where $\zeta \geq 0$ and $\xi \geq 0$. Thus the columns of N and $-V(V^TV)^{-1}$ are a set of generators for K°.

Moreover, for $\varepsilon < \delta$ we obtain $\tilde{K} = K(x, \varepsilon)$ by dropping generators from V. Without loss of generality we will assume that we drop the first r columns of V, where V has p columns. Then consider $w \in \tilde{K}^\circ$. Proceeding as before, we obtain $(V^Tw, \lambda) \leq 0$ for all $\lambda \geq 0$, $\lambda_1, \ldots, \lambda_r = 0$. If we once again define $\xi = V^Tw$, then we see that $\xi_{r+1}, \ldots, \xi_p \leq 0$, while $\xi_1, \ldots, \xi_r$ are unrestricted in sign. Hence we obtain a set of generators for $\tilde{K}^\circ$ from the columns of N, the first r columns of $V(V^TV)^{-1}$ and their negatives, and the last $p - r$ columns of $-V(V^TV)^{-1}$. □

Proposition 8.2 leads to the following construction of patterns for linearly constrained minimization. Under the assumption of nondegeneracy, we know there exists ε^* such that if $0 \leq \varepsilon \leq \varepsilon^*$, then $K(x, \varepsilon)$ has a linearly independent set of generators. If we knew this ε^*, it would be a convenient choice for the ε^* required in section 3.5. The following algorithm implicitly estimates ε^*: it conducts what amounts to a safeguarded backtracking on ε at each iteration to find a value of ε_k for which $K(x_k, \varepsilon_k)$ has a linearly independent set of generators.

Given ε_* independent of k, choose $\varepsilon_k \geq \varepsilon_*$. Then

(1) Define the cone $K(x_k, \varepsilon_k)$ as in section 3.5.

(2) Let V represent the matrix whose columns are the generators $\nu_i^\ell(x_k, \varepsilon_k)$ and $\nu_i^u(x_k, \varepsilon_k)$ of $K(x_k, \varepsilon_k)$ (defined in (3.3)–(3.4)). Determine whether or not V has full column rank. If so, go to step 3. Otherwise, reduce ε_k just until $|I_\ell(x_k, \varepsilon_k)| + |I_u(x_k, \varepsilon_k)|$ is decreased. Return to step 1.

(3) Construct a rational positive basis N for the range of $I - V(V^TV)^{-1}V^T$. This can be done via reduction to row echelon form, or simply by taking the columns of the matrices $\pm\left(I - V(V^TV)^{-1}V^T\right)$.

(4) Form the matrix $\Gamma_k = [\,N \;\; V(V^TV)^{-1} \;\; -V(V^TV)^{-1}\,]$.

Under the assumption of nondegeneracy, ε_k will remain bounded away from 0 as a function of k, implicitly giving us the ε^* introduced in section 3.5.2.

This construction also shows that we may reasonably expect to arrange that r_k, the number of columns of Γ_k defined in section 3.1, to be at most $2n$. Suppose V has rank r. Then the nullspace of V has dimension $n - r$, so we can find a positive basis N for the nullspace with as few as $n - r + 1$ elements (or 0 elements, if $n = r$). At the same time, $V(V^TV)^{-1}$ has r columns, so we can arrange Γ_k to have as few as $(n - r + 1) + 2r = n + r - 1$ columns, if $r < n$, or $2r$ elements, if $r = n$. In either case Γ_k has at most $2n$ columns.

8.3. The case of bound constraints.

Matters simplify enormously in the case of bound constraints, previously considered in [8]. We will briefly discuss the specialization to bound constrained minimization and in the process sharpen the results in [8].

In the case of bound constraints we have

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & l \leq x \leq u. \end{array}$$

Again, we allow the possibility that some of the variables are unbounded either above or below by permitting $\ell_j, u_j = \pm\infty$, $j \in \{1, \ldots, n\}$.

In the case of bound constraints we know a priori the possible generators of the cones $K(x, \varepsilon)$ and $K^\circ(x, \varepsilon)$. For any $x \in \Omega$ and any $\varepsilon > 0$ the cone $K(x, \varepsilon)$ is generated by some subset of the coordinate vectors $\pm e_i$. If $K(x, \varepsilon)$ is generated by $\nu_{i_1}, \ldots, \nu_{i_r}$, where $\nu_{i_j} \in \{e_{i_j}, -e_{i_j}\}$, then $K^\circ(x, \varepsilon)$ is generated by the set $-\nu_{i_1}, \ldots, -\nu_{i_r}$ together with a positive basis for the orthogonal complement of the space spanned by $\nu_{i_1}, \ldots, \nu_{i_r}$. This orthogonal complement simply corresponds to the remaining coordinate directions.

This simplicity allows us to prescribe in advance patterns that work for all $K(x, \varepsilon)$. In [8] we gave the prescription $\Gamma_k = [I - I]$. This choice, independent of k, includes generators for all possible $K^\circ(x, \varepsilon)$. However, if not all the variables are bounded, then one can make a choice of Γ_k that is independent of k but more parsimonious in the number of directions. Let $x_{i_1}, \ldots, x_{i_r}$ be the variables with either a lower or upper bound; then Γ_k should include the coordinate vectors $\pm e_{i_1}, \ldots, \pm e_{i_r}$ together with a positive basis for the orthogonal complement of the linear span of $e_{i_1}, \ldots, e_{i_r}$; a positive basis for the orthogonal complement can have as few as $(n-r)+1$ elements.

The choice of $\Gamma_k = [I - I]$ in [8] requires, in the worst case, $2n$ objective evaluations per iteration. The more detailed analysis given here leads to a reduction in this cost if not all the variables are bounded. If only $r < n$ variables are bounded, then we can find an acceptable pattern containing as few as $2r + ((n - r) + 1) = n + r + 1$ points.

Finally, note that if general linear constraints are present but A has full row rank (i.e., there are no more than n constraints and they are all linearly independent), then one can carry out a construction similar to that for bound constraints.

9. Conclusions.

We have introduced pattern search algorithms for solving problems with general linear constraints. We have shown that under mild assumptions we can guarantee global convergence of pattern search methods for linearly constrained problems to a KKT point. As in the case of unconstrained minimization, pattern search methods for linearly constrained problems accomplish this without explicit recourse to the gradient or the directional derivative. In addition, we have outlined particular instances of such algorithms and shown how the general approach can be greatly simplified when the only constraints are bounds on the variables. The effectiveness of these techniques will be the subject of future work.

10. Appendix: Results concerning the geometry of polyhedra.

We need a number of results concerning the geometry of polyhedra for the proofs of section 6. We begin with a classical result on the structure of finitely generated cones.

THEOREM 10.1. *Let C be a finitely generated convex cone in $\mathbf{R}^n$. Then C is the union of finitely many finitely generated convex cones each having a linearly independent set of generators chosen from the generators of C.*

Proof. See Theorem 4.17 in [16]. □

COROLLARY 10.2. *Let C be a finitely generated convex cone in $\mathbf{R}^n$ with generators $\{v_1, \ldots, v_r\}$. Then there exists $c_{10.2} > 0$, depending only on $\{v_1, \ldots, v_r\}$, such that any $z \in C$ can be written in the form $z = \sum_{i=1}^r \lambda_i v_i$ with $\lambda \geq 0$ and $\| \lambda \| \leq c_{10.2} \| z \|$.*

Proof. Theorem 10.1 says that we can write z in the form $z = \sum_{j=1}^{r_z} \lambda_{i_j} v_{i_j}$, where $r_z \leq r$, $\lambda_{i_j} \geq 0$, and the matrix $V_z = [v_{i_1} \cdots v_{i_{r_z}}]$ has full column rank. The full column rank of V_z means that the induced linear transformation is one-to-one; so if V_z^+ is the pseudoinverse of V_z, then $(\lambda_{i_1}, \ldots, \lambda_{i_{r_z}})^T = V_z^+ z$. If we define λ via

$$\lambda_i = \begin{cases} \lambda_{i_j} & \text{if } i = i_j, \\ 0 & \text{otherwise,} \end{cases}$$

then $\lambda \geq 0$, $z = V\lambda$, and $\| \lambda \| \leq \| V_z^+ \| \| z \|$. Since the matrix V_z is drawn from a finite set of possibilities (e.g., the set of all subsets of $\{v_1, \ldots, v_r\}$), we can find the desired constant $c_{10.2}$, independent of z. □

Let C be a closed convex cone in $\mathbf{R}^n$ with vertex at the origin and let C° be its polar. Given any vector z, we will denote by z_C and z_{C° the projections of z onto the cones C and C°, respectively. The classical polar decomposition [13, 17] allows us to express z as

$$z = z_C + z_{C^\circ},$$

where $(z_C, z_{C^\circ}) = 0$.

PROPOSITION 10.3. *Suppose the cone C is generated by $\{v_1, \ldots, v_r\}$. Then there exists $c_{10.3} > 0$, depending only on $\{v_1, \ldots, v_r\}$, such that for any z for which $z_C \neq 0$*

$$\max_{1 \leq i \leq r} \frac{z^T v_i}{\| v_i \|} \geq c_{10.3} \| z_C \|.$$

Proof. By Corollary 10.2, we have $c_{10.2} > 0$, depending only on $\{v_1, \ldots, v_r\}$, such that we can write z_C as $z_C = \sum_{i=1}^r \lambda_i v_i$, with $\| \lambda \| \leq c_{10.2} \| z_C \|$ and $\lambda \geq 0$. Then

$$z^T z_C = \sum_{i=1}^r \lambda_i z^T v_i,$$

so for some i we must have

$$\lambda_i z^T v_i \geq \frac{1}{r} z^T z_C = \frac{1}{r} \| z_C \|^2.$$

Since $\| \lambda \| \leq c_{10.2} \| z_C \|$ and $\| z_C \| \neq 0$, we obtain

$$z^T v_i \geq \frac{1}{r} \frac{1}{c_{10.2}} \| z_C \|.$$

If we let

$$v^* = \max_{1 \leq i \leq r} \| v_i \|$$

we obtain

$$z^T v_i \geq \frac{1}{r} \frac{1}{c_{10.2}} \frac{1}{v^*} \| v_i \| \| z_C \|$$

and the desired result, with $c_{10.3} = (r c_{10.2} v^*)^{-1}$. □

For the polyhedron defining the feasible region of (1.1), we have the following.

COROLLARY 10.4. *There exists $c_{10.4} > 0$, depending only on A, for which the following holds. For any $x \in \Omega$ and $\varepsilon \geq 0$, let $K = K(x, \varepsilon)$. Then for any z for which $z_{K^\circ} \neq 0$,*

$$\max_{1 \leq i \leq r} \frac{z^T v_i}{\| v_i \|} \geq c_{10.4} \| z_{K^\circ} \|,$$

where $\{v_1, \ldots, v_r\}$ are the generators of $K^\circ(x, \varepsilon)$ required in section 3.5.2 to be in $\mathbf{\Gamma}$.

Proof. The corollary follows from the observation that since $K(x, \varepsilon)$ is generated by subsets of the rows of A, $K(x, \varepsilon)$ can be one of only a finite number of possible cones. Consequently $K^\circ(x, \varepsilon)$ will also be one of only a finite number of possible cones. Applying Proposition 10.3 to each of these latter cones in turn (with the generators in $\mathbf{\Gamma}$ for $K^\circ(x, \varepsilon)$) and taking the minimum yields the corollary. □

Let

$$a^* = \max_{1 \le i \le m} \{\| a_i \|\}.$$

Then we have the following straightforward proposition.

PROPOSITION 10.5. *For any $x \in \Omega$ and $\varepsilon \ge 0$, we have*

$$\ell_i \quad \le \quad a_i^T x \quad \le \quad \ell_i + \varepsilon \| a_i \| \quad \le \quad \ell_i + \varepsilon a^* \quad \textit{for } i \in I_\ell(x, \varepsilon), \tag{10.1}$$

$$u_i - \varepsilon a^* \quad \le \quad u_i - \varepsilon \| a_i \| \quad \le \quad a_i^T x \quad \le \quad u_i \quad \textit{for } i \in I_u(x, \varepsilon), \tag{10.2}$$

where $I_\ell(x, \varepsilon)$ and $I_u(x, \varepsilon)$ are the index sets defined in (3.1)–(3.2).

Proof. A simple calculation shows that the distance from any point x to the affine subspace defined by $a_i^T z = b$ is $\left| b - a_i^T x \right| / \| a_i \|$. Thus, if the distance from x to $a_i^T z = b$ is no more than ε, then

$$b - \varepsilon \| a_i \| \le a_i^T x \le b + \varepsilon \| a_i \|.$$

Then (10.1) and (10.2) follow from the fact that $x \in \Omega$ and the definitions of $I_\ell(x, \varepsilon)$ and $I_u(x, \varepsilon)$. □

Despite its unpromising appearance, the following result is extremely useful, as it relates the local geometry of Ω (as manifest in $K(x, \varepsilon)$) to the global geometry of Ω (as manifest in the projection P_Ω).

PROPOSITION 10.6. *There exists $c_{10.6} > 0$ such that for any $x \in \Omega$, $\varepsilon \ge 0$, and $w \in \mathbf{R}^n$,*

$$\| (x + w) - P_\Omega(x + w) \|^2 \ge \| P_{K(x,\varepsilon)} w \|^2 - c_{10.6} \; \varepsilon \; \| P_{K(x,\varepsilon)} w \|.$$

Proof. $P_\Omega(x + w)$ is the solution y of the convex quadratic program

$$\begin{array}{ll} \text{minimize} & \frac{1}{2} \| y - (x + w) \|^2 \\ \text{subject to} & \ell \le Ay \le u. \end{array} \tag{10.3}$$

The dual of (10.3) is the following program in (z, μ_1, μ_2):

$$\begin{array}{ll} \text{maximize} & \frac{1}{2} \| z - (x + w) \|^2 - \mu_1^T (u - Az) - \mu_2^T (Az - \ell) \\ \text{subject to} & z - (x + w) + A^T \mu_1 - A^T \mu_2 = 0 \\ & \mu_1, \mu_2 \ge 0. \end{array} \tag{10.4}$$

The proposition will follow from a felicitous choice of (z, μ_1, μ_2) for the dual.

Given $x \in \Omega$ and $\varepsilon \ge 0$, let $K = K(x, \varepsilon)$ and consider the polar decomposition $w = w_K + w_{K^\circ}$. We can write

$$w_K = A^T \mu_1 - A^T \mu_2,$$

where $\mu_1, \mu_2 \ge 0$ and the only nonzero components of μ_1, μ_2 correspond to the generators of $K(x, \varepsilon)$, which are the outward pointing normals to the constraints within distance ε of x. More precisely,

$$\mu_1^i \ne 0 \quad \text{only if } i \in I_u(x, \varepsilon), \qquad \mu_2^i \ne 0 \quad \text{only if } i \in I_\ell(x, \varepsilon). \tag{10.5}$$

Furthermore, by Corollary 10.2 we can choose μ_1, μ_2 in such a way that there exists $c_{10.2} > 0$, depending only on A, such that

$$\| \mu_1 \| + \| \mu_2 \| \leq c_{10.2} \| w_K \|. \tag{10.6}$$

Meanwhile, let $z = x + w_{K^\circ}$. Then

$$w = w_K + w_{K^\circ} = z - x + A^T\mu_1 - A^T\mu_2,$$

so (z, μ_1, μ_2) is feasible for the dual (10.4). Since $y = P_\Omega(x + w)$ is feasible for the primal (10.3), by duality we have

$$\begin{aligned}&\tfrac{1}{2}\| (x+w) - P_\Omega(x+w) \|^2 \\ &\geq \tfrac{1}{2}\| (x+w) - z \|^2 - \mu_1^T(u - Az) - \mu_2^T(Az - \ell) \\ &= \tfrac{1}{2}\| w_K \|^2 - \mu_1^T(u - Ax) - \mu_2^T(Ax - \ell) + (A^T\mu_1 - A^T\mu_2)^T w_{K^\circ}.\end{aligned}$$

Since $w_K = A^T\mu_1 - A^T\mu_2$ and $(w_K, w_{K^\circ}) = 0$, the latter expression reduces to

$$\tfrac{1}{2}\| (x+w) - P_\Omega(x+w) \|^2 \geq \tfrac{1}{2}\| w_K \|^2 - \mu_1^T(u - Ax) - \mu_2^T(Ax - \ell). \tag{10.7}$$

Now, in light of (10.5) and Proposition 10.5 we have

$$\mu_1^T(u - Ax) + \mu_2^T(Ax - \ell) \leq a^* \ \varepsilon \ \| \mu_1 \|_1 + a^* \ \varepsilon \ \| \mu_2 \|_1 \leq a^*\sqrt{n} \ \varepsilon \ (\| \mu_1 \| + \| \mu_2 \|).$$

Applying (10.6) we obtain

$$\mu_1^T(u - Ax) + \mu_2^T(Ax - \ell) \leq c_{10.2} \ a^*\sqrt{n} \ \varepsilon \ \| w_K \|.$$

Substituting this into (10.7) yields

$$\tfrac{1}{2}\| (x+w) - P_\Omega(x+w) \|^2 \geq \tfrac{1}{2}\| w_K \|^2 - c_{10.2} \ a^*\sqrt{n} \ \varepsilon \ \| w_K \|,$$

which is the desired result, with $c_{10.6} = 2c_{10.2}a^*\sqrt{n}$. □

The consequence of Proposition 10.6 of utility to us is the following. It says that if $x \in \Omega$ is close to $\partial\Omega$ and the step from x to $P_\Omega(x + w)$ is sufficiently long, then w cannot be "too normal" to $\partial\Omega$ near x.

PROPOSITION 10.7. *Given $\gamma > 0$, there exist $r_{10.7} > 0$ and $c_{10.7} > 0$, depending only on A and γ, such that if $\eta > 0$, $x \in \Omega$, $0 \leq \varepsilon \leq r_{10.7}\eta^2$, $\| w \| \leq \gamma$, and $\| P_\Omega(x + w) - x \| \geq \eta$, then*

$$\| P_{K^\circ(x,\varepsilon)} w \| \geq c_{10.7} \ \| P_\Omega(x + w) - x \|.$$

Proof. Given $x \in \Omega$ and $\varepsilon \geq 0$, let $K = K(x, \varepsilon)$ and consider the polar decomposition $w = w_K + w_{K^\circ}$. Let $q = P_\Omega(x + w) - x$. We have

$$\| w \|^2 = \| w_K \|^2 + \| w_{K^\circ} \|^2 = \| (w - q) + q \|^2 = \| w - q \|^2 + 2\,(w - q \ , \ q) + \| q \|^2.$$

We know that $(z - P_\Omega(z) \ , \ P_\Omega(z) - y) \geq 0$ for all $y \in \Omega$ from the properties of the projection P_Ω [17]. Choosing $z = x + w$ and $y = x$ we obtain $(w - q \ , \ q) \geq 0$, so

$$\| w_K \|^2 + \| w_{K^\circ} \|^2 \geq \| w - q \|^2 + \| q \|^2.$$

From Proposition 10.6 we obtain

$$\| w_K \|^2 + \| w_{K^\circ} \|^2 \geq \| w_K \|^2 - c_{10.6} \ \varepsilon \ \| w_K \| + \| q \|^2.$$

Using the hypothesis that $\| w \| \leq \gamma$, we obtain

$$\| w_{K^\circ} \|^2 \geq -c_{10.6}\ \varepsilon\gamma + \| q \|^2.$$

Let

$$r_{10.7} = \frac{3}{4}\frac{1}{\gamma}\frac{1}{c_{10.6}}.$$

Then, if $\varepsilon \geq 0$ satisfies $\varepsilon \leq r_{10.7}\eta^2$, we have

$$\| w_{K^\circ} \|^2 \geq \| q \|^2/4.$$

Taking square roots yields the proposition, with $c_{10.7} = 1/2$. □

As we noted at the introduction of $K^\circ(x, \varepsilon)$, we can proceed from x along all directions in $K^\circ(x, \varepsilon)$ for a distance $\delta > 0$, depending only on ε, and still remain inside the feasible region. The following proposition is the formal statement of this observation.

PROPOSITION 10.8. *Suppose $\varepsilon > 0$ satisfies $\varepsilon \leq h/2$, where h is defined by* (6.1). *Then for any $x \in \Omega$, if $w \in K^\circ(x, \varepsilon)$ and $\| w \| \leq \varepsilon/2$, then $(x + w) \in \Omega$.*

Proof. Consider any index $i \in \{1, \ldots, m\}$. We will show that $x + w$ is feasible with respect to the ith constraint.

If $x \notin \partial\Omega_{\ell_i}(\varepsilon) \cup \partial\Omega_{u_i}(\varepsilon)$, then $\ell_i + \varepsilon\| a_i \| < a_i^T x < u_i - \varepsilon\| a_i \|$, so

$$a_i^T x + a_i^T w \geq \ell_i + \varepsilon\| a_i \| - \| a_i \|\| w \| \geq \ell_i + (\varepsilon/2)\| a_i \| \geq \ell_i$$

and

$$a_i^T x + a_i^T w \leq u_i - \varepsilon\| a_i \| + \| a_i \|\| w \| \leq u_i - (\varepsilon/2)\| a_i \| \leq u_i.$$

On the other hand, suppose $x \in \partial\Omega_{\ell_i}(\varepsilon) \cup \partial\Omega_{u_i}(\varepsilon)$. There are three cases to consider. First suppose $x \in \Omega_{\ell_i}(\varepsilon)$ and $x \in \partial\Omega_{u_i}(\varepsilon)$. Since $\varepsilon < h/2$, this means that $\ell_i = u_i$ (i.e., the constraint is an equality constraint). Then, if $w \in K^\circ(x, \varepsilon)$, we have both $(w\ ,\ -a_i) \leq 0$ and $(w\ ,\ a_i) \leq 0$, so $(w\ ,\ a_i) = 0$. Thus

$$\ell_i = a_i^T x + a_i^T w = u_i.$$

Next suppose $x \in \partial\Omega_{\ell_i}(\varepsilon)$ but $x \notin \partial\Omega_{u_i}(\varepsilon)$. If $w \in K^\circ(x, \varepsilon)$, we have $(-a_i, w) \leq 0$. Applying Proposition 10.5 we obtain

$$\ell_i \leq a_i^T x + a_i^T w \leq \ell_i + \varepsilon\| a_i \| + \| a_i \|\| w \| \leq \ell_i + (3\varepsilon/2)\| a_i \| \leq u_i.$$

Finally, if $x \in \partial\Omega_{u_i}(\varepsilon)$ but $x \notin \partial\Omega_{\ell_i}(\varepsilon)$, then, if $w \in K^\circ(x, \varepsilon)$, $(a_i, w) \leq 0$, so

$$u_i \geq a_i^T x + a_i^T w \geq u_i - \varepsilon\| a_i \| - \| a_i \|\| w \| \geq u_i - (3\varepsilon/2)\| a_i \| \geq \ell_i.$$

Thus $(x + w)$ satisfies the constraints for all $i \in \{1, \ldots, m\}$, so $(x + w) \in \Omega$. □

Acknowledgments. We thank Robert Mifflin for bringing to our attention the work cited in section 2 and for providing us with a copy of Jerrold May's thesis. We also thank Bob Bixby and Bill Cook for directing us to the literature on vertex enumeration, to which we refer in section 8, and David Avis and David Bremner for explaining to us the practical issues in vertex enumeration.

Finally, we thank the referees both for their careful reading of earlier drafts of this paper and for their many helpful comments. The results are stronger and the presentation is both clearer and more accurate as a consequence of their questions and suggestions.

REFERENCES

[1] D. M. Avis and K. Fukuda, *A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra*, Discrete Comput. Geom., 8 (1992), pp. 295–313.

[2] J. V. Burke and J. J. Moré, *On the identification of active constraints*, SIAM J. Numer. Anal., 25 (1988), pp. 1197–1211.

[3] J. V. Burke and J. J. Moré, *Exposing constraints*, SIAM J. Optim., 4 (1994), pp. 573–595.

[4] P. H. Calamai and J. J. Moré, *Projected gradient methods for linearly constrained problems*, Math. Programming, 39 (1987), pp. 93–116.

[5] C. Davis, *Theory of positive linear dependence*, Amer. J. Math., 76 (1954), pp. 733–746.

[6] J. C. Dunn, *Global and asymptotic convergence rate estimates for a class of projected gradient processes*, SIAM J. Control Optim., 19 (1981), pp. 368–400.

[7] R. Hooke and T. A. Jeeves, *Direct search solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–229.

[8] R. M. Lewis and V. J. Torczon, *Pattern search algorithms for bound constrained minimization*, SIAM J. Optim., 9 (1999), pp. 1082–1099.

[9] R. M. Lewis and V. J. Torczon, *Rank Ordering and Positive Bases in Pattern Search Algorithms*, Tech. Rep. 96–71, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA, 1996.

[10] R. M. Lewis and V. J. Torczon, *A Globally Convergent Augmented Lagrangian Pattern Search Algorithm for Optimization with General Constraints and Simple Bounds*, Tech. Rep. 98–31, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton, VA, 1998; also available online from http://www.icase.edu/library/reports/rdp/1998.html#98-31; SIAM J. Optim., submitted.

[11] J. H. May, *Linearly Constrained Nonlinear Programming: A Solution Method That Does Not Require Analytic Derivatives*, Ph.D. thesis, Yale University, New Haven, CT, 1974.

[12] R. Mifflin, *A superlinearly convergent algorithm for minimization without evaluating derivatives*, Math. Programming, 9 (1975), pp. 100–117.

[13] J. J. Moreau, *Décomposition orthgonale d'un espace hilbertien selon deux cônes mutuellement polaires*, C. R. Acad. Sci. Paris, 255 (1962), pp. 238–240.

[14] V. Torczon, *On the convergence of the multidirectional search algorithm*, SIAM J. Optim., 1 (1991), pp. 123–145.

[15] V. Torczon, *On the convergence of pattern search algorithms*, SIAM J. Optim., 7 (1997), pp. 1–25.

[16] J. van Tiel, *Convex Analysis: An Introductory Text*, John Wiley & Sons, New York, 1984.

[17] E. H. Zarantonello, *Projections on convex sets in Hilbert space and spectral theory*, in Contributions to Nonlinear Functional Analysis, E. H. Zarantonello, ed., Academic Press, New York, 1971, pp. 237–424.

SIAM J. OPTIM.
Vol. 10, No. 3, pp. 942–942

ERRATUM: WEAK SHARP SOLUTIONS OF VARIATIONAL INEQUALITIES*

PATRICE MARCOTTE† AND DAOLI ZHU‡

Abstract. Xuexiang Huang pointed out that the statement on line 5 of page 184 of [P. Marcotte and D. Zhu, *SIAM J. Optim.*, 9 (1999), pp. 179–189] is invalid, and supplied an alternative proof of the inequality that appears on line 10. Here is a simple proof of this result due to an anonymous referee. The substitution starts at line 4 and ends at line 11.

PII. S1052623499360616

For any y^* in X^* we have

$$\langle d, y^* - x^* \rangle \leq 0 \qquad \text{since} \quad d \in N_{X^*}(x^*).$$

This implies that X^* can be separated from $x^* + d$ by a hyperplane H_d passing through x^* and orthogonal to d. Let $\{d^k\}$ be a sequence converging to d, such that $x^* + t_k d^k \in X$ for some sequence of positive numbers $\{t_k\}$. (Such a sequence exists because $d \in T_X(x^*)$.) Since d is not on H_d, we can assume without loss of generality that H_d separates X^* from $x^* + t_k d^k$. Thus,

$$\begin{aligned} \operatorname{dist}(x^* + t_k d^k, X^*) &\geq \operatorname{dist}(x^* + t_k d^k, H_d) \\ &= \frac{t_k \langle d, d^k \rangle}{\|d\|}. \end{aligned}$$

REFERENCE

[1] P. Marcotte and D. Zhu, *Weak sharp solutions of variational inequalities*, SIAM J. Optim., 9 (1999), pp. 179–189.

*Received by the editors November 16, 1999; accepted for publication November 16, 1999; published electronically May 15, 2000.

http://www.siam.org/journals/siopt/10-3/36061.html

†DIRO, Université de Montréal, C. P. 6128, succursale Centre-Ville, Montréal, Québec, H3C 3J7 Canada (marcotte@iro.umontreal.ca).

‡CRT, Université de Montréal, C. P. 6128, succursale Centre-Ville, Montréal, Québec, H3C 3J7 Canada (daoli@crt.umontreal.ca).

SIAM J. Optim.
Vol. 10, No. 4, pp. 943–962

CONSTRAINT QUALIFICATIONS AND NECESSARY OPTIMALITY CONDITIONS FOR OPTIMIZATION PROBLEMS WITH VARIATIONAL INEQUALITY CONSTRAINTS[*]

J. J. YE[†]

Abstract. A very general optimization problem with a variational inequality constraint, inequality constraints, and an abstract constraint are studied. Fritz John type and Kuhn–Tucker type necessary optimality conditions involving Mordukhovich coderivatives are derived. Several constraint qualifications for the Kuhn–Tucker type necessary optimality conditions involving Mordukhovich coderivatives are introduced and their relationships are studied. Applications to bilevel programming problems are also given.

Key words. optimization problems, variational inequality constraints, necessary optimality conditions, constraint qualifications, coderivatives of set-valued maps, nonsmooth analysis

AMS subject classifications. 49K99, 90C, 90D65

PII. S105262349834847X

1. Introduction.

An optimization problem with variational inequality constraints (OPVIC) is a special class of an optimization problem over variables x and y in which some or all of its constraints are defined by a parametric variational inequality with y as its primary variable and x as the parameter. In this paper we consider a very general optimization problem with variational inequality constraints in finite dimensional spaces defined as follows:

$$\begin{aligned} \text{(OPVIC)} \quad & \text{minimize } f(x,y) \\ & \text{subject to (s.t.) } \psi(x,y) \le 0, (x,y) \in C, \\ & \qquad y \in \Omega, \langle F(x,y), y-z\rangle \le 0 \quad \forall z \in \Omega, \end{aligned}$$

where $f : R^{n+m} \to R$, $\psi : R^{n+m} \to R^d$, $F : R^{n+m} \to R^m$ are Lipschitz near all optimal solutions of (OPVIC), C is a nonempty closed subset of R^{n+m}, and Ω is a closed convex subset of R^m. The above problem is also called a generalized bilevel programming problem (see, e.g., Ye, Zhu, and Zhu [27]) or a mathematical program with equilibrium constraints (see, e.g., Luo, Pang, and Ralph [10]). The reader is referred to [10] for recent developments on the subject and references for other types of optimality conditions.

Although under certain constraint qualifications one can reduce (OPVIC) to an ordinary nonlinear programming problem, it is known that the usual constraint qualification such as the Mangasarian–Fromovitz constraint qualification cannot in general be satisfied for the equivalent nonlinear programming problem (see [27, Proposition 1.1]). In Ye and Ye [26], under the pseudoupper-Lipschitz continuity, a Kuhn–Tucker type necessary optimality condition involving Mordukhovich coderivatives was derived for (OPVIC). In Ye [25], it was shown that a Kuhn–Tucker type necessary optimality condition involving the proximal coderivatives (which are in general smaller than

[*] Received by the editors December 3, 1998; accepted for publication (in revised form) November 23, 1999; published electronically June 6, 2000. This work was supported by NSERC and a University of Victoria internal research grant.

http://www.siam.org/journals/siopt/10-4/34847.html

[†] Department of Mathematics and Statistics, University of Victoria, Victoria, BC V8W 3P4, Canada (janeye@math.uvic.ca).

Mordukhovich coderivatives) holds under a stronger constraint qualification in the case where the variational inequality is a complementarity system, i.e., $\Omega = R^a \times R^b_+$ with $a + b = m$.

The purpose of this paper is to study (OPVIC) under much weaker assumptions and derive more powerful results than those in [26]. In particular, we incorporate inequality constraints and an abstract constraint in our problems and we do not assume the smoothness of the mapping F as in [26].

As in [26], we formulate (OPVIC) as the following optimization problem with a generalized equation constraint:

$$\begin{aligned} \text{(GP)} \quad & \min f(x,y) \\ & \text{s.t. } \psi(x,y) \le 0, (x,y) \in C, \\ & \quad 0 \in F(x,y) + N(y,\Omega), \end{aligned}$$

where

$$N(y,\Omega) := \begin{cases} \text{the normal cone of } \Omega \text{ at } y & \text{if } y \in \Omega, \\ \emptyset & \text{if } y \notin \Omega \end{cases}$$

is the normal cone operator.

We show that if $(\bar{x},\bar{y})$ is a local solution of (OPVIC), then there exist $\lambda \ge 0, \eta \in R^m$, and $\gamma \in R^d_+$ not all zero such that

$$\begin{aligned} & 0 \in \lambda \partial f(\bar{x},\bar{y}) + \partial \langle \psi, \gamma \rangle (\bar{x},\bar{y}) + \partial \langle F, \eta \rangle (\bar{x},\bar{y}) \\ & \quad + \{0\} \times D^* N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta) + N((\bar{x},\bar{y}), C), \\ & \langle \psi(\bar{x},\bar{y}), \gamma \rangle = 0, \end{aligned}$$

where ∂ denotes the limiting subgradient (see Definition 2.2), N_Ω denotes the set-valued map $y \Rightarrow N(y,\Omega)$, and D^* denotes the coderivative of a set-valued map (see Definition 2.4). Moreover we introduce the concept of calmness for (OPVIC) and show that under the calmness condition λ can be taken as 1. Several constraint qualifications that are stronger than the calmness condition but easier to verify are introduced and their relationships are indicated.

Note that in the case where $\Omega = R^m$, (OPVIC) is reduced to an ordinary nonlinear programming problem with equality, inequality, and abstract constraints. Hence our results are applicable even for an ordinary nonlinear programming problem.

We organize the paper as follows. Section 2 contains background material on nonsmooth analysis. In section 3, we derive the Fritz John type necessary optimality condition involving Mordukhovich coderivatives and the Kuhn–Tucker type necessary optimality conditions involving Mordukhovich coderivatives under the calmness condition. In section 4 we introduce several constraint qualifications for the Kuhn–Tucker necessary optimality conditions involving the Mordukhovich coderivatives and study the relationships between these constraint qualifications. Applications to bilevel programming problems are discussed in section 5.

The following notations are used throughout the paper. For an m-by-n matrix A and index sets $I \subseteq \{1, 2, \ldots, m\}$, $J \subseteq \{1, 2, \ldots, n\}$, A_I and $A_{I,J}$ denote the submatrix of A with rows specified by I and the submatrix of A with rows and columns specified by I and J, respectively. For a mapping $\psi : R^n \to R^d$ and a vector $\gamma \in R^d$, $\langle \psi, \gamma \rangle (x)$ is the function defined by $\langle \psi, \gamma \rangle (x) := \langle \psi(x), \gamma \rangle$. For a vector $d \in R^n$, d_i is the ith component of d and d_I is the subvector composed from the components $d_i, i \in I$.

$\langle a, b\rangle$ is the inner product of vectors a and b. $gph\Phi$ is the graph of a set-valued map Φ and $epif$ is the epigraph of a function f. intΩ, clΩ, and coΩ denote the interior, the closure, and the convex hull of a set Ω. We denote by $B_\delta(x_0)$ and B the open ball centered at x_0 with radius $\delta > 0$ and the open unit ball, respectively.

2. Preliminaries.

This section contains some background material on nonsmooth analysis which will be used later. We give only concise definitions that will be needed in the paper. For more detailed information on the subject, our references are Clarke [3], Clarke et al. [4], Rockafellar and Wets [19], Loewen [9], and Mordukhovich [13, 14].

First we give some concepts for various normal cones.

DEFINITION 2.1. *Let Ω be a nonempty subset of R^n. Given $\bar{z} \in \text{cl}\Omega$, the convex cone*

$$N^\pi(\bar{z}, \Omega) := \{\xi \in R^n : \exists M > 0 \ s.t. \ \langle \xi, z - \bar{z}\rangle \le M\|z - \bar{z}\|^2 \ \forall z \in \Omega\}$$

is called the proximal normal cone to set Ω at point $\bar{z}$, the closed cone

$$N(\bar{z}, \Omega) := \left\{\lim_{k\to\infty} \xi^k : \xi^k \in N^\pi(z^k, \Omega), z^k \to \bar{z}\right\}$$

is called the limiting normal cone to Ω at point $\bar{z}$, and the closed convex hull of the limiting normal cone

$$N_C(\bar{z}, \Omega) := \text{clco}N(\bar{z}, \Omega)$$

is called the Clarke normal cone to set Ω at point $\bar{z}$.

Using the definitions for normal cones, we now give definitions for subgradients of a single-valued map.

DEFINITION 2.2. *Let $f : R^n \to R \cup \{+\infty\}$ be lower semicontinuous and finite at $\bar{z} \in R^n$. The proximal subgradient of f at $\bar{z}$ is defined by*

$$\partial^\pi f(\bar{z}) := \{\xi : (\xi, -1) \in N^\pi((\bar{z}, f(\bar{z})), epif)\},$$

the limiting subgradient of f at $\bar{z}$ is defined by

$$\partial f(\bar{z}) := \{\xi : (\xi, -1) \in N((\bar{z}, f(\bar{z})), epif)\},$$

and the Clarke generalized gradient of f at $\bar{z}$ is defined by

$$\partial_C f(\bar{z}) := \{\xi : (\xi, -1) \in N_C((\bar{z}, f(\bar{z})), epif)\},$$

where $epif := \{(x, r) \in R^n \times R : f(x) \le r\}$ is the epigraph of f.

The following calculus rules for subgradients are well known and can be found in the references given in the beginning of this section (see, e.g., [9, Proposition 5A.4, Theorem 5A.8], proof of [5, Lemma 2.2]).

PROPOSITION 2.3. *Let functions $f : R^n \to R \cup \{+\infty\}$ be lower semicontinuous and finite at $\bar{z} \in R^n$, $g : R^n \to R$ be Lipschitz near $\bar{z}$, and $h : R^n \to R$ is C^{1+} at $\bar{z}$ (i.e., the gradient of h is Lipschitz near $\bar{z}$). Then the nonnegative scalar multiplication rule is*

$$\partial(\lambda f)(\bar{z}) = \lambda\partial f(\bar{z}) \qquad \forall\lambda \ge 0$$

and the sum rules are

$$\partial(f+g)(\bar{z}) \subseteq \partial f(\bar{z}) + \partial g(\bar{z}),$$

$$\partial^{\pi}(f+h)(\bar{z}) = \partial^{\pi} f(\bar{z}) + \nabla h(\bar{z}).$$

Let $\varphi(x) := f(F(x))$*, where* $F : R^m \to R^n$ *is Lipschitz near* $\bar{x}$ *and* $f : R^n \to R$ *is Lipschitz near* $F(\bar{x})$*. Then the chain rule is*

$$\partial\varphi(\bar{x}) \subseteq \bigcup\{\partial\langle\eta, F\rangle(\bar{x}) : \eta \in \partial f(F(\bar{x}))\}.$$

For set-valued maps, the definition for a limiting normal cone leads to the definition of coderivative of a set-valued map introduced by Mordukhovich (see, e.g., [14]).

DEFINITION 2.4. *Let* $\Phi : R^n \rightrightarrows R^q$ *be an arbitrary set-valued map (assigning to each* $z \in R^n$ *a set* $\Phi(z) \subset R^q$ *which may be empty) and* $(\bar{x}, \bar{y}) \in \mathrm{cl}\,gph\Phi$*, where* $gph\Phi := \{(z,v) : v \in \Phi(z)\}$ *denotes the graph of the set-valued map* Φ*. The set-valued map* $D^*\Phi(\bar{z}, \bar{v})$ *from* R^q *into* R^n *defined by*

$$D^*\Phi(\bar{z},\bar{v})(\eta) = \{\xi \in R^n : (\xi, -\eta) \in N((\bar{z},\bar{v}), gph\Phi)\}$$

is called the coderivative of Φ *at the point* $(\bar{z},\bar{v})$*. By convention for* $(\bar{z},\bar{v}) \notin \mathrm{cl}\,gph\Phi$ *we define* $D^*\Phi(\bar{z},\bar{v})(\eta) = \emptyset$*. The symbol* $D^*\Phi(\bar{z})$ *is used when* Φ *is single-valued at* $\bar{z}$ *and* $\bar{v} = \Phi(\bar{z})$*.*

In the special case when a set-valued map is single-valued, the coderivative is related to the limiting subgradient in the following way.

PROPOSITION 2.5 (see [14, Proposition 2.11]). *Let* $\Phi : R^n \to R^q$ *be single-valued and Lipschitz near* $\bar{z}$*. Then*

$$D^*\Phi(\bar{z})(\eta) = \partial\langle\Phi, \eta\rangle(\bar{z}) \quad \forall \eta \in R^q.$$

We now give some concepts for Lipschitz behavior of a set-valued map. The following concept for Lipschitz behavior was introduced by Aubin [1].

DEFINITION 2.6. *A set-valued map* $\Phi : R^n \rightrightarrows R^q$ *is said to be pseudo-Lipschitz continuous around* $(\bar{z},\bar{v}) \in gph\Phi$ *if there exist a neighborhood* U *of* $\bar{z}$*, a neighborhood* V *of* $\bar{v}$*, and* $\mu \geq 0$ *such that*

$$\Phi(z) \cap V \subset \Phi(z') + \mu\|z' - z\|\mathrm{cl}B \quad \forall z', z \in U.$$

On the other hand, the following upper-Lipschitz behavior was studied by Robinson [21].

DEFINITION 2.7. *A set-valued map* $\Phi : R^n \rightrightarrows R^q$ *is said to be upper-Lipschitz continuous at* $\bar{z} \in R^n$ *if there exist a neighborhood* U *of* $\bar{z}$ *and* $\mu \geq 0$ *such that*

$$\Phi(z) \subset \Phi(\bar{z}) + \mu\|z - \bar{z}\|\mathrm{cl}B \quad \forall z \in U.$$

The following proposition is a sum rule for coderivatives.

PROPOSITION 2.8 (see [14, Corollary 4.2]). *Let* Φ_1 *and* Φ_2 *be closed-graph set-valued maps from* R^n *into* R^q *and let* $\bar{v} \in \Phi_1(\bar{z}) + \Phi_2(\bar{z})$*. Assume that the multifunction* $S : R^{n+q} \rightrightarrows R^{2q}$ *defined by*

$$S(z,v) := \{(v_1, v_2) \in R^{2q} | v_1 \in \Phi_1(z), v_2 \in \Phi_2(z), v_1 + v_2 = v\}$$

is locally bounded around $(\bar{z}, \bar{v})$ *and either* Φ_1 *is pseudo-Lipschitz around* $(\bar{z}, v_1)$ *or* Φ_2 *is pseudo-Lipschitz around* $(\bar{z}, v_2)$ *for each* $(v_1, v_2) \in S(\bar{z}, \bar{v})$*. Then for any* $\eta \in R^q$

$$D^*(\Phi_1 + \Phi_2)(\bar{z}, \bar{v})(\eta) \subseteq \cup_{(v_1,v_2)\in S(\bar{z},\bar{v})}[D^*\Phi_1(\bar{z}, v_1)(\eta) + D^*\Phi_2(\bar{z}, v_2)(\eta)].$$

The following sum rule for the case where one of the set-valued maps is single-valued follows from Propositions 2.5 and 2.8.

Corollary 2.9. *Let* $\Phi_1 : R^n \to R^q$ *be single-valued and Lipschitz near* $\bar{z}$ *and* $\Phi_2 : R^n \rightrightarrows R^q$ *be an arbitrary closed set-valued map. Then for any* $\bar{v} \in \Phi_1(\bar{z}) + \Phi_2(\bar{z})$ *and* $\eta \in R^q$

$$D^*(\Phi_1 + \Phi_2)(\bar{z}, \bar{v})(\eta) \subseteq \partial\langle\Phi_1, \eta\rangle(\bar{z}) + D^*\Phi_2(\bar{z}, \bar{v} - \Phi_1(\bar{z}))(\eta).$$

3. Necessary optimality conditions. The purpose of this section is to derive the necessary optimality conditions involving Mordukhovich coderivatives for (OPVIC).

The following fundamental results obtained by Mordukhovich will be useful in proving the Fritz John type necessary optimality condition involving Mordukhovich coderivatives.

Lemma 3.1 (extremal principle). *Let* $\Omega_1, \dots, \Omega_n$ *be closed sets in* R^m *and let* $\bar{x} \in \cap_{i=1}^n \Omega_i$*. Suppose that there exist a neighborhood* U *of* $\bar{x}$ *and sequences* $\{a_{ik}\} \subseteq R^m$*,* $i = 1, 2, \dots, n$ *such that* $a_{ik} \to 0$ *as* $k \to \infty$ *for* $i = 1, 2, \dots, n$ *and*

$$\cap_{i=1}^n (\Omega_i - a_{ik}) \cap U = \emptyset \quad \forall k = 1, 2, \dots.$$

Then there exists $\xi_i \in N(\bar{x}, \Omega_i), i = 1, \dots, n$ *such that*

$$\xi_1 + \xi_2 + \cdots + \xi_n = 0, \|\xi_1\| + \|\xi_2\| + \cdots + \|\xi_n\| = 1.$$

Although the terminology of the extremal principle was first given by Mordukhovich [14], the essence of the results can be traced back to Mordukhovich [11]. We may usefully view it as an extension of the Hahn–Banach separation theorem to nonconvex sets. The proof for the case when $n = 2$ can be found in [14, Theorem 3.2]. For the case when $n > 2$, the result can be proved in exactly the same way as the proof of [14, Theorem 3.2] or mathematical induction on n can be used as in the proof of Mordukhovich and Shao [17, Theorem 3.2].

The extremal principle turns out to be very useful in deriving the Fritz John type necessary optimality condition as shown in the following theorem.

Theorem 3.2. *Let* $(\bar{x}, \bar{y})$ *be a local solution of* (OPVIC)*. Then there exist* $\lambda \geq 0$*,* $\eta \in R^m, \gamma \in R^d_+$ *not all zero such that*

$$\begin{aligned} &0 \in \lambda\partial f(\bar{x}, \bar{y}) + \partial\langle\psi, \gamma\rangle(\bar{x}, \bar{y}) + \partial\langle F, \eta\rangle(\bar{x}, \bar{y}) \\ &\qquad +\{0\} \times D^*N_\Omega(\bar{y}, -F(\bar{x}, \bar{y}))(\eta) + N((\bar{x}, \bar{y}), C), \\ &\langle\psi(\bar{x}, \bar{y}), \gamma\rangle = 0. \end{aligned}$$

Proof. Define

$$\begin{aligned} \Omega_1 &:= \{(x, y, u_0, u, v) : f(x, y) \leq u_0\}, \\ \Omega_2 &:= \{(x, y, f(\bar{x}, \bar{y}), u, 0) : \psi(x, y) \leq u\}, \\ \Omega_3 &:= \{(x, y, f(\bar{x}, \bar{y}), 0, 0) : (x, y) \in C\}, \\ \Omega_4 &:= \{(x, y, f(\bar{x}, \bar{y}), 0, v) : v \in F(x, y) + N(y, \Omega)\}. \end{aligned}$$

Then $(\bar{x}, \bar{y}, f(\bar{x}, \bar{y}), 0, 0) \in \cap_{i=1}^{4} \Omega_i$. By taking $a_{1k} = (0, 0, \nu_k, 0, 0)$ with $\nu_k < 0, \nu_k \to 0$, $a_{ik} = 0 \;\; \forall i = 2, 3, 4$, and $U = V \times R^{1+d+m}$, where V is a neighborhood of the local minimizer $(\bar{x}, \bar{y})$, it is easy to verify that

$$\cap_{i=1}^{4} (\Omega_i - a_{ik}) \cap U = \emptyset \quad \forall k = 1, 2, \ldots.$$

By Lemma 3.1, there exist ξ_i, not all zero such that $\xi_i \in N((\bar{x}, \bar{y}, f(\bar{x}, \bar{y}), 0, 0), \Omega_i)$, $i = 1, 2, 3, 4$, and

$$0 = \xi_1 + \xi_2 + \xi_3 + \xi_4.$$

That is, there exist $(a, -\lambda) \in R^{n+m+1}, (b, -\gamma) \in R^{n+m+d}, c \in R^{n+m}, (d, -\eta) \in R^{n+m+m}$ not all zero such that

$$(a, -\lambda) \in N((\bar{x}, \bar{y}, f(\bar{x}, \bar{y})), epif), \tag{3.1}$$

$$(b, -\gamma) \in N((\bar{x}, \bar{y}, 0), epi\psi), \tag{3.2}$$

$$c \in N((\bar{x}, \bar{y}), C), \tag{3.3}$$

$$(d, -\eta) \in N((\bar{x}, \bar{y}, 0), gph\varphi) \text{ where } \varphi(x, y) := F(x, y) + N(y, \Omega), \tag{3.4}$$

and

$$0 = a + b + c + d. \tag{3.5}$$

By the definition of epigraph, inclusion (3.1) implies that $\lambda \geq 0$. Since f is Lipschitz near $(\bar{x}, \bar{y})$, either $a = 0, \lambda = 0$, or $\lambda > 0$ and $(\frac{a}{\lambda}, -1) \in N((\bar{x}, \bar{y}, f(\bar{x}, \bar{y})), epif)$, which by definition implies that $\frac{a}{\lambda} \in \partial f(\bar{x}, \bar{y})$. Hence (3.1) implies that

$$\lambda \geq 0, a \in \lambda \partial f(\bar{x}, \bar{y}). \tag{3.6}$$

Similarly, inclusion (3.2) implies that $\gamma \geq 0$. Let $M := \{i : \psi_i(\bar{x}, \bar{y}) = 0\}$ be the index set of the binding constraints. Inclusion (3.2) implies that $(b, -\gamma_M) \in N((\bar{x}, \bar{y}, 0), gph\psi_M)$, which is equivalent to $b \in D^*\psi_M(\bar{x}, \bar{y})(\gamma_M)$ by definition of coderivatives. By virtue of Proposition 2.5, we have $D^*\psi_M(\bar{x}, \bar{y})(\gamma_M) = \partial\langle\psi_M, \gamma_M\rangle(\bar{x}, \bar{y})$. Therefore we have

$$\gamma \geq 0, \langle\psi(\bar{x}, \bar{y}), \gamma\rangle = 0, b \in \partial\langle\psi, \gamma\rangle(\bar{x}, \bar{y}). \tag{3.7}$$

By definition of coderivatives, (3.4) implies that $d \in D^*\varphi(\bar{x}, \bar{y}, 0)(\eta)$. By Corollary 2.9, we have

$$\begin{aligned} d &\in D^*\varphi(\bar{x}, \bar{y}, 0)(\eta) \\ &\subseteq \partial\langle F, \eta\rangle(\bar{x}, \bar{y}) + \{0\} \times D^*N_\Omega(\bar{y}, -F(\bar{x}, \bar{y}))(\eta). \end{aligned} \tag{3.8}$$

The conclusion of the theorem follows from inclusions (3.6), (3.7), (3.3), (3.8), and (3.5). □

Remark. In the case of ordinary mathematical programming problems, $\Omega = R^m$, Theorem 3.2 is a limiting subgradient version of the generalized Lagrange multiplier rules in Clarke [3, Theorem 6.1.1] and was obtained by Mordukhovich in [12, Theorem 1(b)].

The following constraint qualification called *no nonzero abnormal multiplier constraint qualification* (NNAMCQ) follows from the Fritz John type necessary condition.

Corollary 3.3. *Let $(\bar{x},\bar{y})$ be a local solution of* (OPVIC). *Assume that condition* (NNAMCQ) *is satisfied, i.e., there is no nonzero vector $(\gamma,\eta)\in R^d_+\times R^m$ such that*

$$\begin{aligned}&0\in\partial\langle\psi,\gamma\rangle(\bar{x},\bar{y})+\partial\langle F,\eta\rangle(\bar{x},\bar{y})\\&\qquad+\{0\}\times D^*N_\Omega(\bar{y},-F(\bar{x},\bar{y}))(\eta)+N((\bar{x},\bar{y}),C),\\&\langle\psi(\bar{x},\bar{y}),\gamma\rangle=0\end{aligned}\tag{3.9}$$

is satisfied at $(\bar{x},\bar{y})$. Then $\lambda>0$ in the conclusion of Theorem 3.2.

Proof. By Theorem 3.2, there exists $\lambda\geq 0$, $\eta\in R^m$, $\gamma\in R^d_+$ not all zero such that

$$\begin{aligned}&0\in\lambda\partial f(\bar{x},\bar{y})+\partial\langle\psi,\gamma\rangle(\bar{x},\bar{y})+\partial\langle F,\eta\rangle(\bar{x},\bar{y})\\&\qquad+\{0\}\times D^*N_\Omega(\bar{y},-F(\bar{x},\bar{y}))(\eta)+N((\bar{x},\bar{y}),C),\\&\langle\psi(\bar{x},\bar{y}),\gamma\rangle=0.\end{aligned}\tag{3.10}$$

The case $\lambda=0$ is impossible under condition (NNAMCQ). Indeed, if $\lambda=0$ in the above condition, then the inclusion (3.10) coincides with inclusion (3.9). But this is impossible since (γ,η) is nonzero. □

It is well known that the calmness condition (see, e.g., Clarke [3]) is the weakest constraint qualification for nonlinear programming problems with Lipschitz problem data. We now extend the concept to the setting of (OPVIC).

Definition 3.4. *Let $(\bar{x},\bar{y})$ be a local solution to* (OPVIC). (GP) *is said to be calm at $(\bar{x},\bar{y})$ provided that there exist $\epsilon>0$ and $\mu>0$ such that $\forall\ (p,q)\in\epsilon B$ $\forall\ (x,y)\in B_\epsilon(\bar{x},\bar{y})$ satisfying*

$$\begin{aligned}&\psi(x,y)+p\leq 0,(x,y)\in C,\\&q\in F(x,y)+N(y,\Omega)\end{aligned}$$

it follows that

$$f(\bar{x},\bar{y})\leq f(x,y)+\mu\|(p,q)\|.$$

Lemma 3.5. *Let $(\bar{x},\bar{y})$ be a local solution to* (GP), *where* (GP) *is calm at $(\bar{x},\bar{y})$. Then $(\bar{x},\bar{y},0)$ is a local solution to the following problem:*

$$\begin{aligned}\min\quad&f(x,y)+d\mu\max\{\psi_i(x,y),0,i=1,\ldots,d\}+\mu\|q\|\\ \textit{s.t.}\quad&(x,y,q)\in gph\Phi\cap C\times R^m,\end{aligned}$$

where Φ is a set-valued map defined by $\Phi(x,y):=F(x,y)+N(y,\Omega)$.

Proof. By definition of the calmness,

$$\begin{aligned}&f(\bar{x},\bar{y})\leq f(x,y)+\mu(\|p\|+\|q\|)\quad\forall(x,y,p,q)\\&\ \text{s.t. }\psi(x,y)+p\leq 0,(x,y,q)\in gph\Phi\cap C\times R^m,(x,y)\in B_\epsilon(\bar{x},\bar{y}),(p,q)\in\epsilon B.\end{aligned}$$

Since

$$\psi_i(x,y)-\psi_i^+(x,y)\leq 0,i=1,\ldots,d$$

taking $p_i=-\psi_i^+(x,y)$, we have for (x,y) in a neighborhood of $(\bar{x},\bar{y})$ and q near 0,

$$\begin{aligned}f(\bar{x},\bar{y})&\leq f(x,y)+\mu\left(\sum_{i=1}^d\psi_i^+(x,y)+\|q\|\right)\\&\leq f(x,y)+d\mu\max\{\psi_i(x,y),0,i=1,\ldots,d\}+\mu\|q\|.\end{aligned}$$

Notice that $\max\{\psi_i(\bar{x},\bar{y}), 0, i=1,\dots,d\} = 0$. The proof is complete. □

THEOREM 3.6. *Let* $(\bar{x},\bar{y})$ *be a local solution of* (OPVIC). *Suppose that* (GP) *is calm at* $(\bar{x},\bar{y})$. *Then* λ *can be taken as* 1 *in the conclusion of Theorem* 3.2.

Proof. By Lemma 3.5, $(\bar{x},\bar{y},0)$ is a local solution to the new (OPVIC):

$$\begin{aligned} \min \quad & \tilde{f}(x,y,q) \\ \text{s.t.} \quad & 0 \in \tilde{F}(x,y,q) + N(y,\Omega), \end{aligned}$$

where $\tilde{f}(x,y,q) := f(x,y) + d\mu \max\{\psi_i(x,y), 0, i=1,\dots,d\} + \mu\|q\|$ and $\tilde{F}(x,y,q) := -q + F(x,y)$.

We now prove that condition (NNAMCQ) is satisfied. Indeed, it is easy to see that the inclusion (3.9) for the new (OPVIC) is

$$0 \in \partial\langle F,\eta\rangle(\bar{x},\bar{y}) \times \{-\eta\} + \{0\} \times D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta) \times \{0\} + N((\bar{x},\bar{y}),C) \times \{0\},$$

which is only satisfied by the zero vector $\eta = 0$.

Applying Corollary 3.3, there exists $\eta \in R^m$ such that

$$\begin{aligned} & 0 \in \partial\tilde{f}(\bar{x},\bar{y},0) + \partial\langle F,\eta\rangle(\bar{x},\bar{y}) \times \{-\eta\} \\ & +\{0\} \times D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta) \times \{0\} + N((\bar{x},\bar{y}),C) \times \{0\}. \end{aligned}$$

Note that

$$\tilde{f}(x,y,q) = f(x,y) + d\mu h(\psi(x,y)) + \mu\|q\|,$$

where $h : R^d \to R$ is defined by $h(u) := \max\{u_1,\dots,u_d,0\}$. By the sum rule and the chain rule in Proposition 2.3,

$$\partial\tilde{f}(\bar{x},\bar{y},0) \subseteq \partial f(\bar{x},\bar{y}) \times \{0\} + d\mu \cup \{\partial\langle\eta,\psi\rangle(\bar{x},\bar{y}) : \eta \in \partial h(\psi(\bar{x},\bar{y}))\} + \mu(\{0\} \times \{0\} \times B).$$

The proof of the theorem is completed after calculating the subgradient of the convex function h at $\psi(\bar{x},\bar{y})$, i.e.,

$$\partial h(\psi(\bar{x},\bar{y})) = \left\{ \gamma \in R^d : \gamma_i \geq 0, \gamma_i\psi_i(\bar{x},\bar{y}) = 0, i=1,\dots,d \text{ and } \sum_{i=1}^d \gamma_i = 1 \right\}. \qquad □$$

Remark. In the case of ordinary mathematical programming problems, $\Omega = R^m$, Theorem 3.6 can be considered as a limiting subgradient version of the generalized Lagrange multiplier rules in Clarke [3, Proposition 6.4.4].

Note that Theorems 3.2 and 3.6 involve the coderivative $D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta)$. By the definition of coderivatives,

$$\xi \in D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta) \Longleftrightarrow (\xi, -\eta) \in N((\bar{y}, -F(\bar{x},\bar{y})), gphN_\Omega).$$

Hence calculation of the coderivative $D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta)$ depends on calculation of the limiting normal cone $N((\bar{y}, -F(\bar{x},\bar{y})), gphN_\Omega)$. In the case $\Omega = R^m_+$, the limiting normal cone $N((\bar{y}, -F(\bar{x},\bar{y})), gphN_\Omega)$ can be calculated explicitly by using the following proposition.

PROPOSITION 3.7. *For any* $(\bar{y},\bar{z}) \in gphN_{R^m_+}$, *define*

$$\begin{aligned} L := L(\bar{y},\bar{z}) &:= \{i \in \{1,2,\cdots,m\} : \bar{y}_i > 0, \bar{z}_i = 0\}, \\ I_+ := I_+(\bar{y},\bar{z}) &:= \{i \in \{1,2,\cdots,m\} : \bar{y}_i = 0, \bar{z}_i < 0\}, \\ I_0 := I_0(\bar{y},\bar{z}) &:= \{i \in \{1,2,\cdots,m\} : \bar{y}_i = 0, \bar{z}_i = 0\}. \end{aligned}$$

Then

$$N((\bar{y},\bar{z}), gphN_{R^m_+}) = \{(\alpha,-\beta) \in R^{2m} : \alpha_L = 0, \beta_{I_+} = 0$$
$$\forall i \in I_0,\ \textit{either } \alpha_i\beta_i = 0 \textit{ or } \alpha_i < 0 \textit{ and } \beta_i < 0\}.$$

Proof. The proof of the above proposition follows from [25, Proposition 2.7] and the definition of the limiting normal cones. □

In many applications, Ω can be chosen as $\Omega = R^a \times R^b_+$ for some nonnegative integers a, b with $a + b = m$. Let $y = (z,u), F(x,y) = (G(x,y), H(x,y))$. Since $N_{R^a}(z) = \{0\}$ is a constant map, we have

$$D^*N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\alpha,\beta) = \{0\} \times D^*N_{R^b_+}(\bar{u}, -H(\bar{x},\bar{z},\bar{u}))(\beta).$$

Again the limiting normal cone $N((\bar{u}, -H(\bar{x},\bar{z},\bar{u})), gphN_{R^b_+})$ can be calculated by using Proposition 3.7.

In the case where Ω is a polyhedral convex set, a calculation of the limiting normal cone to the graph of the normal cone to the set Ω was first given in the proof of [6, Theorem 2] and stated in [20, Proposition 4.4].

4. Constraint qualifications.

In this section we study sufficient conditions for the calmness, introduce some constraint qualifications, and discuss the relationships between them.

DEFINITION 4.1. *We say that the constraint system* (CS) *for* (OPVIC)

$$\text{(CS)} \qquad \psi(x,y) \le 0, (x,y) \in C,$$
$$0 \in F(x,y) + N(y,\Omega)$$

has a local error bound at a point $(\bar{x},\bar{y})$ *if there exist positive constants* μ, δ, *and* ϵ *such that*

$$d((x,y), \Sigma(0,0)) \le \mu\|(p,q)\| \quad \forall\ (p,q) \in \epsilon B,$$
$$(x,y) \in \Sigma(p,q) \cap B_\delta(\bar{x},\bar{y}), \tag{4.1}$$

where

$$\Sigma(p,q) := \{(x,y) \in C : \psi(x,y) + p \le 0, q \in F(x,y) + N(y,\Omega)\} \tag{4.2}$$

is the set of solutions to the perturbed generalized equation.

Note that (CS) has a local error bound at a point $(\bar{x},\bar{y})$ if and only if $\Sigma(p,q)$ is pseudoupper-Lipschitz continuous around $(0,0,\bar{x},\bar{y})$ in the terminology of [26, Definition 2.8]. $\Sigma(p,q)$ being either pseudo-Lipschitz continuous around $(0,0,\bar{x},\bar{y})$ (see Definition 2.6) or upper-Lipschitz continuous (see Definition 2.7) at $(\bar{x},\bar{y})$ implies that (CS) has a local error bound at $(\bar{x},\bar{y})$.

We now prove that the existence of a local error bound for the constraint system of (OPVIC) at a solution $(\bar{x},\bar{y})$ implies that (OPVIC) is calm at $(\bar{x},\bar{y})$.

PROPOSITION 4.2. *Suppose that* (CS) *has a local error bound at* $(\bar{x},\bar{y})$, *a local solution to* (OPVIC). *Then* (GP) *is calm at* $(\bar{x},\bar{y})$.

Proof. Since (CS) has a local error bound at $(\bar{x},\bar{y})$, there exist positive numbers μ, δ, ϵ such that (4.1) is satisfied. Let $(p,q) \in \epsilon B$, $(x,y) \in \Sigma(p,q) \cap B_\delta(\bar{x},\bar{y})$ and $(x^*,y^*) \in \Sigma(0,0)$ be the projection of the vector (x,y). Then

$$\begin{aligned} f(\bar{x},\bar{y}) &\le f(x^*,y^*) \quad \text{since } (\bar{x},\bar{y}) \text{ solves (OPVIC)} \\ &= f(x,y) + (f(x^*,y^*) - f(x,y)) \\ &\le f(x,y) + L_f\|(x^*,y^*) - (x,y)\| \quad \text{where } L_f \text{ is the Lipschitz constant of } f \\ &= f(x,y) + L_f d((x,y),\Sigma(0,0)) \\ &\le f(x,y) + L_f\mu\|(p,q)\| \text{ by virtue of (4.1).} \end{aligned}$$

The proof is complete. □

We now study sufficient conditions for existence of a local error bound that are easier to verify. Recall that a set-valued map is called a polyhedral multifunction if its graph is a union of finitely many polyhedral convex sets. This class of set-valued maps is closed under (finite) addition, scalar multiplication, and (finite) composition. By Robinson [23, Proposition 1], a polyhedral multifunction is upper-Lipschitz. Hence the following result provides a sufficient condition for existence of a local error bound.

THEOREM 4.3. *Suppose that the mappings ψ, F are affine, C is polyhedral, and Ω is a polyhedral convex set. Then the solution map for the perturbed generalized equation* (4.2) *is upper-Lipschitz at any feasible solution of* (OPVIC) *and hence* (CS) *has a local error bound at any feasible solution of* (OPVIC).

Proof. Since the graph of N_Ω is a finite union of polyhedral convex sets, N_Ω is polyhedral. Hence $(\psi, F)+R^d_+\times N_\Omega$ (as the sum of polyhedral maps (ψ, F), $R^d_+\times N_\Omega$) is polyhedral, and so therefore is its inverse map

$$S(p,q) := \{(x,y) : \psi(x,y) + p \le 0, q \in F(x,y) + N(y,\Omega)\}.$$

That is, the graph

$$gphS := \{(x,y,p,q) : \psi(x,y) + p \le 0, q \in F(x,y) + N(y,\Omega)\}$$

is a union of polyhedral convex sets. Since

$$\begin{aligned} gph\Sigma &= \{(x,y,p,q) \in C \times R^d \times R^m : \psi(x,y) + p \le 0, q \in F(x,y) + N(y,\Omega)\} \\ &= (C \times R^d \times R^m) \cap gphS, \end{aligned}$$

which is also a union of polyhedral convex sets, Σ is also a polyhedral multifunction. By [23, Proposition 1], Σ is upper-Lipschitz. Hence (CS) has a local error bound at any feasible point. □

Remark. The result in the case $\Omega = R^m$ is actually the well-known error bound result for linear systems due to Hoffman [7]. In this case, the above result recovers the well-known result in nonlinear programming that no other constraint qualification is needed when the constraint system is linear.

We now prove that condition (NNAMCQ) defined in Corollary 3.3 is a sufficient condition for existence of a local error bound.

THEOREM 4.4. *Assume that condition* (NNAMCQ) *is satisfied at* $(\bar{x},\bar{y})$. *Then the solution map for the perturbed generalized equation* (4.2) *is pseudo-Lipschitz continuous around* $(0,0,\bar{x},\bar{y})$ *and hence* (CS) *has a local error bound at* $(\bar{x},\bar{y})$.

Proof. By virtue of [16, Proposition 3.5], it suffices to prove that

$$D^*\Sigma(0,0,\bar{x},\bar{y})(0,0) = \{(0,0)\}.$$

Suppose that $(\gamma,-\eta) \in D^*\Sigma(0,0,\bar{x},\bar{y})(0,0)$, which means by the definition of coderivatives that

$$(\gamma,-\eta,0,0) \in N((0,0,\bar{x},\bar{y}), gph\Sigma).$$

By the definition of limiting normal cones, there are sequences $(p^k, q^k, x^k, y^k) \to (0, 0, \bar{x}, \bar{y})$ and $(\gamma^k, -\eta^k, \alpha^k, \beta^k) \to (\gamma, -\eta, 0, 0)$ with

$$(\gamma^k, -\eta^k, \alpha^k, \beta^k) \in N^\pi((p^k, q^k, x^k, y^k), gph\Sigma).$$

For each k by the definition of proximal normal cones, there are $M > 0$ such that $\forall\ (p, q, x, y) \in gph\Sigma$,

$$\langle(\gamma^k, -\eta^k, \alpha^k, \beta^k), (p, q, x, y) - (p^k, q^k, x^k, y^k)\rangle \le M\|(p, q, x, y) - (p^k, q^k, x^k, y^k)\|^2.$$

That is, (p^k, q^k, x^k, y^k) is a solution to the optimization problem

$$\begin{aligned}&\min\langle-(\gamma^k, -\eta^k, \alpha^k, \beta^k), (p, q, x, y)\rangle + M\|(p, q, x, y) - (p^k, q^k, x^k, y^k)\|^2\\ &\text{s.t. } \psi(x, y) + p \le 0, (x, y) \in C,\\ &\qquad q \in F(x, y) + N(y, \Omega).\end{aligned}$$

Inclusion (3.9) for the above problem is

$$\begin{aligned}&0 \in \{(\gamma, 0)\} \times \partial\langle\psi, \gamma\rangle(x^k, y^k) + \{(0, -\eta)\} \times \partial\langle F, \eta\rangle(x^k, y^k)\\ &\qquad +\{(0, 0, 0)\} \times D^*N_\Omega(y^k, q^k - F(x^k, y^k))(\eta) + \{(0, 0)\} \times N((x, y), C),\\ &\langle\psi(x^k, y^k) + p^k, \gamma\rangle = 0,\end{aligned}$$

which is only satisfied by $\gamma = 0, \eta = 0$ and hence (NNAMCQ) is satisfied at (p^k, q^k, x^k, y^k). Applying Corollary 3.3, there exist $\tilde{\gamma}^k \in R^d, \tilde{\eta}^k \in R^m$ such that

$$\begin{aligned}&0 \in -(\gamma^k, -\eta^k, \alpha^k, \beta^k) + \{(\tilde{\gamma}^k, 0)\} \times \partial\langle\psi, \tilde{\gamma}^k\rangle(x^k, y^k) + \{(0, -\tilde{\eta}^k)\} \times \partial\langle F, \tilde{\eta}^k\rangle(x^k, y^k)\\ &\qquad +\{(0, 0, 0)\} \times D^*N_\Omega(y^k, q^k - F(x^k, y^k))(\tilde{\eta}^k) + \{(0, 0)\} \times N((x^k, y^k), C),\\ &\langle\psi(x^k, y^k) + p^k, \tilde{\gamma}^k\rangle = 0.\end{aligned}$$

That is,

$$\begin{aligned}&(\alpha^k, \beta^k) \in \partial\langle\psi, \gamma^k\rangle(x^k, y^k) + \partial\langle F, \eta^k\rangle(x^k, y^k)\\ &\qquad +\{0\} \times D^*N_\Omega(y^k, q^k - F(x^k, y^k))(\eta^k) + N((x^k, y^k), C),\\ &\langle\psi(x^k, y^k) + p^k, \gamma^k\rangle = 0.\end{aligned}$$

Taking limits as $k \to \infty$ by virtue of Lipschitz continuity of ψ and F, we have

$$\begin{aligned}&0 \in \partial\langle\psi, \gamma\rangle(\bar{x}, \bar{y}) + \partial\langle F, \eta\rangle(\bar{x}, \bar{y}) + \{0\} \times D^*N_\Omega(\bar{y}, -F(\bar{x}, \bar{y}))(\eta) + N((\bar{x}, \bar{y}), C),\\ &\langle\psi(\bar{x}, \bar{y}), \gamma\rangle = 0.\end{aligned}$$

Consequently, by condition (NNAMCQ), $(\gamma, \eta) = (0, 0)$ and hence Σ is pseudo-Lipschitz continuous around $(0, 0, \bar{x}, \bar{y})$. □

In the case of the nonlinear programming problem (i.e, when $\Omega = R^m$), condition (NNAMCQ), with the limiting subgradient replaced by the Clarke generalized gradient, is equivalent to the generalized Mangasarian–Fromovitz constraint qualification (see, e.g., [24, Proposition 3.1] and [8]). We now extend the equivalence to the case where $\Omega = R^a \times R^b_+$. The result was proved by Outrata [18, Proposition 3.3] for the case where $\Omega = R^m_+$, ψ is independent of y and there are no abstract constraints. Note that our result improves the one in [18] in that no extra assumption such as (A) in [18] is needed for the inequality constraints. However, the proof technique is the same as that in [18]. Hence we only sketch the proof.

PROPOSITION 4.5. *Assume that* $\Omega = R^a \times R^b_+$ *with* a, b *nonnegative integers and* $a + b = m$, $C = D \times R^b$, *where* D *is a closed subset of* R^{n+a}. *Let* $y = (z, u)$ *and* $F(x, y) = (G(x, y), H(x, y))$ *and suppose all mappings* ψ, G, H *are* C^1. *We say that the generalized Mangasarian–Fromovitz constraint qualification* (GMFCQ) *is satisfied at* $(\bar{x}, \bar{y})$ *if*

(i) *for every partition of* I_0 *into sets* P, Q, R *with* $R \neq \emptyset$, *there exist vectors* $k \in intT_C((\bar{x}, \bar{z}), D), h \in R^b$ *such that* $h_{I_+} = 0, h_Q = 0, h_R \geq 0$,

$$\begin{aligned}
&\nabla_{x,z}\psi_M(\bar{x}, \bar{z}, \bar{u})k + \nabla_u\psi_M(\bar{x}, \bar{z}, \bar{u})h \leq 0,\\
&\nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u})k + \nabla_u G(\bar{x}, \bar{z}, \bar{u})h = 0,\\
&\nabla_{x,z}H_{L\cup P}(\bar{x}, \bar{z}, \bar{u})k + \nabla_u H_{L\cup P}(\bar{x}, \bar{z}, \bar{u})h = 0,\\
&\nabla_{x,z}H_R(\bar{x}, \bar{z}, \bar{u})k + \nabla_u H_R(\bar{x}, \bar{z}, \bar{u})h \geq 0,
\end{aligned}$$

and either $h_i > 0$ *or*

$$\nabla_{x,z}H_i(\bar{x}, \bar{z}, \bar{u})k + \nabla_u H_i(\bar{x}, \bar{z}, \bar{u})h > 0 \textit{ for some } i \in R;$$

(ii) *for every partition of* I_0 *into the sets* P, Q, *the matrix*

$$\begin{bmatrix} \nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u}) & \nabla_u G_{L\cup P}(\bar{x}, \bar{z}, \bar{u}) \\ \nabla_{x,z}H_{L\cup P}(\bar{x}, \bar{z}, \bar{u}) & \nabla_u H_{L\cup P, L\cup P}(\bar{x}, \bar{z}, \bar{u}) \end{bmatrix}$$

has full row rank and there exist vectors $k \in intT_C((\bar{x}, \bar{z}), D), h \in R^b$ *such that*

$$\begin{aligned}
&h_{I_+} = 0, h_Q = 0,\\
&\nabla_{x,z}\psi_M(\bar{x}, \bar{z}, \bar{u})k + \nabla_u\psi_M(\bar{x}, \bar{z}, \bar{u})h < 0,\\
&\nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u})k + \nabla_u G(\bar{x}, \bar{z}, \bar{u})h = 0,\\
&\nabla_{x,z}H_{L\cup P}(\bar{x}, \bar{z}, \bar{u})k + \nabla_u H_{L\cup P}(\bar{x}, \bar{z}, \bar{u})h = 0,
\end{aligned}$$

where $T_C((\bar{x}, \bar{z}), D)$ *denotes the Clarke tangent cone of* D *at* $(\bar{x}, \bar{z})$, $M := \{i : \psi_i(\bar{x}, \bar{z}, \bar{u}) = 0\}$ *is the index set of binding inequality constraints, and*

$$\begin{aligned}
L := L(\bar{x}, \bar{z}, \bar{u}) &:= \{i \in \{1, 2, \cdots, b\} : \bar{u}_i > 0, H_i(\bar{x}, \bar{z}, \bar{u}) = 0\},\\
I_+ := I_+(\bar{x}, \bar{z}, \bar{u}) &:= \{i \in \{1, 2, \cdots, b\} : \bar{u}_i = 0, H_i(\bar{x}, \bar{z}, \bar{u}) > 0\},\\
I_0 := I_0(\bar{x}, \bar{z}, \bar{u}) &:= \{i \in \{1, 2, \cdots, b\} : \bar{u}_i = 0, H_i(\bar{x}, \bar{z}, \bar{u}) = 0\}.
\end{aligned}$$

Then (GMFCQ) *implies* (NNAMCQ) *and under the assumption that* $\mathrm{int}T_C((\bar{x}, \bar{z}), D) \neq \emptyset$ (GMFCQ) *is equivalent to* (NNAMCQ) *with limiting normal cone of* D *replaced by the Clarke normal cone of* D.

Proof. Let $\eta = (\alpha, \beta)$. Then the condition (NNAMCQ) is equivalent to saying that there is no nonzero vector $(\gamma, \alpha, \beta) \in R^d_+ \times R^a \times R^b$ such that

$$\begin{aligned}
0 \in\ &\nabla\psi_M(\bar{x}, \bar{z}, \bar{u})^\top\gamma_M + \nabla G(\bar{x}, \bar{z}, \bar{u})^\top\alpha + \nabla H(\bar{x}, \bar{z}, \bar{u})^\top\beta\\
&+\{0\} \times \{0\} \times D^*N_{R^b_+}(\bar{u}, -H(\bar{x}, \bar{z}, \bar{u}))(\beta) + N((\bar{x}, \bar{z}), D) \times \{0\},
\end{aligned}$$

where $A^\top$ denotes the transpose of a matrix A. That is, there is no $(\gamma, \alpha, \beta) \neq 0$ such that $\gamma \geq 0$ and

$$\begin{aligned}
&-\nabla_{x,z}\psi_M(\bar{x}, \bar{z}, \bar{u})^\top\gamma_M - \nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u})^\top\alpha - \nabla_{x,z}H(\bar{x}, \bar{z}, \bar{u})^\top\beta \in N((\bar{x}, \bar{z}), D),\\
&(-\nabla_u\psi_M(\bar{x}, \bar{z}, \bar{u})^\top\gamma_M - \nabla_u G(\bar{x}, \bar{z}, \bar{u})^\top\alpha - \nabla_u H(\bar{x}, \bar{z}, \bar{u})^\top\beta, -\beta)\\
&\qquad \in N((\bar{u}, -H(\bar{x}, \bar{z}, \bar{u})), gphN_{R^b_+}).
\end{aligned}$$

Let $(w, -\beta) \in N((\bar{u}, -H(\bar{x}, \bar{z}, \bar{u})), gphN_{R^b_+})$. Then, by Proposition 3.7, $w_L = 0, \beta_{I_+} = 0$ and for any $i \in I_0$, either $w_i \beta_i = 0$ or $w_i < 0, \beta_i < 0$. So I_0 splits into the sets

$$P := \{i \in I_0 : w_i = 0\}, \quad Q := \{i \in I_0 : \beta_i = 0\}, \quad R := \{i \in I_0 : w_i < 0, \beta_i < 0\}.$$

Using this partition, condition (NNAMCQ) is equivalent to the following two conditions:

(i) For every partition of I_0 into the sets P, Q, R with $R \neq \emptyset$ there are no vectors $\gamma_M, w, \alpha, \beta_{L \cup P \cup R}$ satisfying the system

$$\begin{aligned}
&-\nabla_{x,z}\psi_M(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad\qquad -\nabla_{x,z}H_{L\cup P\cup R}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P\cup R} \in N((\bar{x}, \bar{z}), D) \\
&-\nabla_u \psi_{M,L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_u G_{A,L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad -\nabla_u H_{L\cup P\cup R, L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P\cup R} = 0, \\
&w_{I_+\cup Q\cup R} = -\nabla_u \psi_{M,I_+\cup Q\cup R}(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_u G_{A,I_+\cup Q\cup R}(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad\qquad -\nabla_u H_{L\cup P\cup R, I_+\cup Q\cup R}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P\cup R}, \\
&\gamma_M \geq 0, w_R < 0, \beta_R < 0;
\end{aligned}$$

(ii) For every partition of I_0 into the sets P, Q there are no vectors $\gamma_M, w, \alpha, \beta_{L\cup P}$ satisfying the system

$$\begin{aligned}
&-\nabla_{x,z}\psi_M(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_{x,z}G(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad -\nabla_{x,z}H_{L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P} \in N((\bar{x}, \bar{z}), D) \\
&-\nabla_u \psi_{M,L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_u G_{A,L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad -\nabla_u H_{L\cup P, L\cup P}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P} = 0, \\
&w_{I_+\cup Q} = -\nabla_u \psi_{M,I_+\cup Q}(\bar{x}, \bar{z}, \bar{u})^\top \gamma_M - \nabla_u G_{A,I_+\cup Q}(\bar{x}, \bar{z}, \bar{u})^\top \alpha \\
&\qquad -\nabla_u H_{L\cup P, I_+\cup Q}(\bar{x}, \bar{z}, \bar{u})^\top \beta_{L\cup P}, \\
&\gamma_M \geq 0,
\end{aligned}$$

where A denotes the index set $A := \{1, 2, \cdots, a\}$.

In the case where D is an open set, as in Outrata [18], the results follow from applying Motzkin's and Tucker's theorems of alternatives and the general case follows from applying the convex separation theorem. □

Remark. Note that in the case where $\Omega = R^m$, (OPVIC) is an ordinary nonlinear programming problem with equality, inequality constraints, and abstract constraints and (GMFCQ) is reduced to the condition that the matrix $\nabla F(\bar{x}, \bar{y})$ has full row rank and there exist vectors $k \in \mathrm{int}T_C((\bar{x}, \bar{y}), C)$ such that

$$\begin{aligned}
&\nabla \psi_M(\bar{x}, \bar{y})k < 0, \\
&\nabla F(\bar{x}, \bar{y})k = 0,
\end{aligned}$$

which is the generalized Mangasarian–Fromovitz constraint qualification for the nonlinear programming problems (see, e.g., Jourani [8]). Note that we can also deal with the case where the mappings ψ, F are not smooth but Lipschitz continuous only by replacing the gradient ∇ by the Clarke gradient ∂_C without any difficulty. The smoothness in the assumption is just for the easy exposition.

The following theorem extends a sufficient condition in [4, Theorem 3.3.1] for existence of a local error bound of an equality system to (CS). Note that as in the proof of [4, Theorem 3.3.8], we can prove that (NNAMCQ) is stronger than the following sufficient condition for existence of an local error bound.

THEOREM 4.6. *Let $(\bar{x}, \bar{y}) \in \Sigma(0,0)$, where Σ is the solution map* (4.2)*. Assume that the bounded constraint qualification condition (Bounded CQ) is satisfied at $(\bar{x}, \bar{y})$, i.e., there exist constants $\mu > 0$, $0 < \epsilon \leq \infty$, such that*

$$\begin{aligned} \mu^{-1} \leq \inf\{\|\xi\| :& \xi \in \partial\langle \psi, e_1\rangle(x,y) + \partial\langle F, e_2\rangle(x,y) \\ & + \{0\} \times D^* N_\Omega(y, q - F(x,y))(e_2) + N((x,y), C), \\ & \langle \psi(x,y) + p, e_1\rangle = 0, \|(e_1, e_2)\| = 1, e_1 \geq 0, \\ & (p,q) \neq 0, (x,y) \in \Sigma(p,q) \cap B_\epsilon(\bar{x}, \bar{y})\}. \end{aligned}$$

Then if $\epsilon < \infty$ $\forall$ $0 < \delta < \epsilon$,

$$d((x,y), \Sigma(0,0)) \leq \mu\|(p,q)\| \ \forall \ (x,y) \in \Sigma(p,q) \cap B_\delta(\bar{x}, \bar{y}), (p,q) \in (\epsilon - \delta)\mu^{-1} B$$

and if $\epsilon = \infty$,

$$d((x,y), \Sigma(0,0)) \leq \mu\|(p,q)\| \ \forall \ (x,y) \in \Sigma(p,q).$$

Proof. Observe that

$$\begin{aligned} \Sigma(0,0) &= \{(x,y) : 0 \in \Phi(x,y)\} \\ &= \{(x,y) : d(0, \Phi(x,y)) = 0\}, \end{aligned}$$

where $\Phi(x,y) := (-\psi(x,y), F(x,y)) + R^d_- \times N(y, \Omega) + \Delta_C(x,y)$ and Δ_C is the indicator mapping of set C defined by

$$\Delta_C(x,y) := \begin{cases} \{0\} & \text{if } (x,y) \in C, \\ \emptyset & \text{if } (x,y) \notin C. \end{cases}$$

It is obvious that the following claim will be useful.

Claim. Suppose the function $f(x) : R^n \to R \cup \{+\infty\}$ is nonnegative and lower semicontinuous. Let x_0 be a solution of $S = \{x : f(x) = 0\}$. Suppose that for some $\mu > 0, 0 < \epsilon \leq \infty$,

$$\|\xi\| \geq \mu^{-1} \ \forall \ \xi \in \partial^\pi f(x), 0 < f(x) < \infty, x \in B_\epsilon(x_0).$$

If $\epsilon < \infty$, then $\forall$ $0 < \delta < \epsilon$,

$$d(x,S) \leq \mu f(x) \qquad \forall x \in B_\delta(x_0), f(x) < (\epsilon - \delta)\mu^{-1}$$

and if $\epsilon = \infty$, then

$$d(x,S) \leq \mu f(x) \qquad \forall x \in R^n.$$

Proof of the claim. Taking $V = B_\epsilon(x_0)$ in [4, Theorem 3.3.1],

$$\min\{d(x, B_\epsilon(x_0)^C), d(x,S)\} \leq \mu f(x) \quad \forall x \in B_\epsilon(x_0),$$

where Ω^C denotes the complement of a set Ω.

Let $0 < \delta < \epsilon$ and $x \in B_\delta(x_0)$. Then obviously, $d(x, B_\epsilon(x_0)^C) > \epsilon - \delta$. Hence for all $x \in B_\delta(x_0)$ satisfying $f(x) < (\epsilon - \delta)\mu^{-1}$,

$$d(x, S) = \min\{d(x, B_\epsilon(x_0)^C), d(x, S)\} \le \mu f(x) < \epsilon - \delta.$$

In the case $\epsilon = \infty$, $d(x, B_\infty(x_0)^C) = \infty$, hence

$$d(x, S) \le \mu f(x) \quad \forall x.$$

The proof of the claim is complete.

Observe that

$$d(0, \Phi(x, y)) := \inf\{\|(p, q)\| : (p, q) \in \Phi(x, y)\} = \inf\{\|(p, q)\| + \Psi_{gph\Phi}(x, y, p, q)\},$$

where Ψ_E denotes the indicator function of set E. By the statement and the proof of [9, Theorem 5A.2], the function $(x, y) \to d(0, \Phi(x, y))$ is lower semicontinuous and

$$\begin{aligned}\partial^\pi d(0, \Phi(x, y)) \subseteq \{(\gamma, \eta) : (\gamma, \eta, 0, 0) \in \partial^\pi g(x, y, p, q)\\ \text{for some } (p, q) \text{ such that } d(0, \Phi(x, y)) = \|(p, q)\| + \Psi_{gph\Phi}(x, y, p, q)\},\end{aligned}$$

where $g(x, y, p, q) := \|(p, q)\| + \Psi_{gph\Phi}(x, y, p, q)$. At the point $(x, y, p, q) \in gph\Phi$ such that $0 < d(0, \Phi(x, y)) = \|(p, q)\|$, $\|(p, q)\|$ is smooth and the subgradient is the unit sphere S_{d+m}. By the sum rule Proposition 2.3, we have

$$\partial^\pi g(x, y, p, q) = \{0\} \times \{0\} \times S_{d+m} + N^\pi((x, y, p, q), gph\Phi).$$

Hence

$$\begin{aligned}\partial^\pi d(0, \Phi(x, y)) \subseteq \{(\gamma, \eta) : (\gamma, \eta, 0, 0) \in \{0\} \times \{0\} \times S_{d+m} + N^\pi((x, y, p, q), gph\Phi)\\ \text{for some } (p, q) \text{ such that } d(0, \Phi(x, y)) = \|(p, q)\| + \Psi_{gph\Phi}(x, y, p, q)\}.\end{aligned}$$

For any $(\gamma, \eta, 0, 0) \in \{0\} \times \{0\} \times S_{d+m} + N^\pi((x, y, p, q), gph\Phi)$, there exists $(e_1, e_2) \in S_{d+m}$ such that $(\gamma, \eta, e_1, -e_2) \in N^\pi((x, y, p, q), gph\Phi)$. By definition of the proximal normal cone, there exists $M > 0$ such that $\forall\ (x', y', p', q') \in gph\Phi$,

$$\langle(\gamma, \eta, e_1, -e_2), (x', y', p', q') - (x, y, p, q)\rangle \le M\|(x', y', p', q') - (x, y, p, q)\|^2.$$

That is, (x, y, p, q) is an optimal solution to

$$\begin{aligned}\min \quad & \langle-(\gamma, \eta, e_1, -e_2), (x', y', p', q')\rangle + M\|(x', y', p', q') - (x, y, p, q)\|^2\\ \text{s.t.} \quad & \psi(x', y') + p' \le 0, (x', y') \in C,\\ & q' \in F(x', y') + N(y', \Omega).\end{aligned}$$

One can easily verify that (NNAMCQ) for the above problem is satisfied. Applying Corollary 3.3, we conclude that

$$\begin{aligned}(\gamma, \eta) \in\ & \partial\langle\psi, e_1\rangle(x, y) + \partial\langle F, e_2\rangle(x, y)\\ & +\{0\} \times D^*N_\Omega(y, q - F(x, y))(e_2) + N(x, y, C),\\ & e_1 \ge 0, \langle\psi(x, y) + p, e_1\rangle = 0.\end{aligned}$$

Hence,

$$\begin{aligned}\partial^\pi d(0, \Phi(x, y)) \subseteq\{(\gamma, \eta) : (\gamma, \eta) \in\ & \partial\langle\psi, e_1\rangle(x, y) + \partial\langle F, e_2\rangle(x, y)\\ & +\{0\} \times D^*N_\Omega(y, q - F(x, y))(e_2) + N((x, y), C)\\ & \text{for some } (e_1, e_2) \in S_{d+m} \text{ such that } e_1 \ge 0, \langle p + \psi(x, y), e_1\rangle = 0,\\ & \text{and some } (p, q) \text{ such that } d(0, \Phi(x, y)) = \|(p, q)\| + \Psi_{gph\Phi}(x, y, p, q)\}.\end{aligned}$$

The proof of the theorem is completed after applying the claim. □

Now consider the case where the abstract constraint is independent of y, i.e., $C = D \times R^m$, there is no inequality constraint and $\forall\ x$ near $\bar{x}$ the solution map

$$y(x) := \{y \in R^m : 0 \in F(x,y) + N(y,\Omega)\}$$

is single-valued and Lipschitz on a neighborhood of $\bar{y}$. Then it is obvious that a local solution $(\bar{x},\bar{y})$ of (OPVIC) is also a local solution to the problem of minimizing $f(x,y(x))$ over D, and hence no other constraint qualifications are needed. A sufficient condition for the existence of such a Lipschitz continuous single-valued map is the strong regularity of the generalized equation

$$0 \in F(\bar{x},y) + N(y,\Omega) \tag{4.3}$$

at $\bar{y}$ in the sense of Robinson [22]. Indeed, in the following theorem we will show that strong regularity is stronger than the constraint qualification (NNAMCQ). The reader is referred to [22] for conditions of strong regularity. Since (4.3) is strongly regular in particular if F is locally strongly monotone in y uniformly in x, the following condition is weaker than the one in [26, Theorem 3.2 (b)]. Note that the result for the case $\Omega = R^m_+$ was proved by Outrata [18] using a different proof.

Theorem 4.7. *Let $(\bar{x},\bar{y})$ be a solution to the generalized equation. Assume that $F(x,y)$ is C^1 around $(\bar{x},\bar{y})$ and the generalized equation* (4.3) *is strongly regular at $\bar{y}$. Then the constraint qualification* (NNAMCQ) *is satisfied at $(\bar{x},\bar{y})$.*

Proof. Let $y := x$ and $f(q,y) := -q + F(\bar{x},y)$ in [22, Theorem 2.1 and Corollary 2.2]. Since the generalized equation (4.3) is strongly regular at $\bar{y}$, there exist neighborhoods N of 0 and W of $\bar{y}$, and a single-valued function $y(q) : N \to W$, such that for any $q \in N$, $y(q)$ is the unique solution in W of the inclusion

$$q \in F(\bar{x},y) + N(y,\Omega).$$

Further, $y(q)$ is Lipschitz continuous near 0. That is $\Sigma_{\bar{x}}(q) := \{y \in R^m : q \in F(\bar{x},y) + N(y,\Omega)\}$ is pseudo-Lipschitz continuous around $(0,\bar{y})$. Note that from [15, Theorem 5.8], $\Sigma_{\bar{x}}(q)$ is pseudo-Lipschitz continuous around $(0,\bar{y})$ if and only if there is no nonzero vector $\eta \in R^m$ such that

$$0 \in \nabla_y F(\bar{x},\bar{y})^\top \eta + D^* N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta).$$

Therefore there is no nonzero vector $\eta \in R^m$ such that

$$0 \in \nabla_x F(\bar{x},\bar{y})^\top \eta + N(\bar{x},D),$$
$$0 \in \nabla_y F(\bar{x},\bar{y})^\top \eta + D^* N_\Omega(\bar{y}, -F(\bar{x},\bar{y}))(\eta).$$

That is, (NNAMCQ) is satisfied. □

Corollary 4.8. *The following conditions are constraint qualifications:*

(1) [calmness constraint qualification (calmness CQ)]*: The problem* (GP) *is calm at $(\bar{x},\bar{y})$.*

(2) [error bound CQ]*:* (CS) *has a local error bound at $(\bar{x},\bar{y})$.*

(3) [linear constraint qualification (linear CQ)]*: The mappings ψ, F are affine, C is polyhedral, and Ω is a polyhedral convex set.*

(4) [strongly regular constraint qualification (SRCQ)]: *There is no inequality constraint $\psi(x,y) \le 0$. F is C^1 around the optimal solution $(\bar{x},\bar{y})$. $C = D \times R^m$, where D is a closed subset of R^n. The generalized equation*

$$0 \in F(\bar{x},y) + N(y,\Omega)$$

is strongly regular at $\bar{y}$.

(5) [no nonzero abnormal multiplier constraint qualification (NNAMCQ)]*: There is no nonzero vector* $(\gamma, \eta) \in R^d_+ \times R^m$ *such that*

$$0 \in \partial\langle \psi, \gamma\rangle(\bar{x}, \bar{y}) + \partial\langle F, \eta\rangle(\bar{x}, \bar{y}) + \{0\} \times D^* N_\Omega(\bar{y}, -F(\bar{x}, \bar{y}))(\eta) + N((\bar{x}, \bar{y}), C)$$
$$\langle \psi(\bar{x}, \bar{y}), \gamma\rangle = 0.$$

(6) [generalized Mangasarian–Fromovitz constraint qualification (GMFCQ)]*: Stated as in Proposition* 4.5*.*

(7) [bounded constraint qualification (bounded CQ)]*: There exist constants* $\mu > 0$*,* $0 < \epsilon \leq \infty$*, such that*

$$\begin{aligned} \mu^{-1} \leq \inf\{\|\xi\| : & \xi \in \partial\langle \psi, e_1\rangle(x, y) + \partial\langle F, e_2\rangle(x, y) \\ & + \{0\} \times D^* N_\Omega(y, q - F(x, y))(e_2) + N((x, y), C), \\ & \langle \psi(x, y) + p, e_1\rangle = 0, \|(e_1, e_2)\| = 1, e_1 \geq 0, \\ & (p, q) \neq 0, (x, y) \in \Sigma(p, q) \cap B_\epsilon(\bar{x}, \bar{y})\}. \end{aligned}$$

In summary, we have proved the following relationships between the constraint qualifications:

$$\begin{tikzcd}
& \text{(SRCQ)} \arrow[d, Rightarrow] & \\
& \text{(NNAMCQ)} \arrow[d, Rightarrow] & \text{(GMFCQ) when } \Omega = R^a \times R^b_+ \arrow[l, Rightarrow] \\
\text{Linear CQ} \arrow[d, Rightarrow] & \text{Bounded CQ} \arrow[dl, Rightarrow] & \\
\text{Error Bound CQ} \arrow[d, Rightarrow] & & \\
\text{Calmness CQ.} & &
\end{tikzcd}$$

5. Applications to bilevel programming problems.

The purpose of this section is to illustrate applications of the results obtained in the previous sections to the bilevel programming problems defined as follows:

$$\text{(BP)} \qquad \text{minimize } f(x, z) \qquad \text{s.t. } \psi(x, z) \leq 0, (x, z) \in D \text{ and } z \in S(x),$$

where $S(x)$ is the set of solutions of the problem (P_x):

$$(P_x) \qquad \text{minimize } g(x, z) \qquad \text{s.t. } \varphi(x, z) \leq 0$$

and $f : R^{n+a} \to R$, $\psi : R^{n+a} \to R^d$, $\varphi : R^{n+a} \to R^b$. For simplicity, we assume all functions f, g, ψ, φ are smooth enough.

Let $z \in S(x)$. If a certain constraint qualification holds for the lower level problem (P_x) at z, then there exists $u \in R^b$ such that

$$\begin{aligned} \nabla_z g(x, z) + u\nabla_z \varphi(x, z) = 0, \varphi(x, z) \leq 0, \\ u \geq 0, \langle u, \varphi(x, z)\rangle = 0, \end{aligned}$$

where $u\nabla_z\varphi(x, z) := \sum u_k \nabla_z \varphi_k(x, z)$. It is easy to see that the above Kuhn–Tucker conditions for (P_x) can be written as the generalized equation

$$0 \in ((\nabla_z g + u\nabla_z\varphi)^t(x, z), -\varphi(x, z)) + N((z, u), R^a \times R^b_+),$$

where a^t denotes the transpose of a vector a. Hence the original bilevel programming problem becomes an (OPVIC).

Applying Theorems 3.2 and 3.6 we now derive necessary optimality conditions for (BP).

THEOREM 5.1. *Assume that f and ψ are C^1, g, φ are twice continuously differentiable around $(\bar{x}, \bar{z})$. Further assume that g is pseudoconvex in z, φ is quasi-convex in z. Let $(\bar{x}, \bar{z})$ solve the problem* (BP). *For each feasible solution (x, z) of* (BP) *suppose that a certain constraint qualification holds for (P_x) at z and $\bar{u}$ is a corresponding multiplier associated with $(\bar{x}, \bar{z})$, i.e.,*

$$0 = \nabla_z g(\bar{x}, \bar{z}) + \bar{u}\nabla_z \varphi(\bar{x}, \bar{z}), \quad \bar{u} \geq 0, \quad \langle \varphi(\bar{x}, \bar{z}), \bar{u}\rangle = 0.$$

Then there exist $\lambda \geq 0$, $\gamma \in R^d_+$, $\alpha \in R^a, \beta \in R^b$ not all zero such that

$$0 \in \lambda \nabla f(\bar{x}, \bar{z}) + \gamma \nabla \psi(\bar{x}, \bar{z}) + \alpha \nabla(\nabla_z g + \bar{u}\nabla_z \varphi)^t(\bar{x}, \bar{z}) - \beta \nabla \varphi(\bar{x}, \bar{z}) + N((\bar{x}, \bar{z}), D),$$
$$\langle \psi(\bar{x}, \bar{z}), \gamma\rangle = 0, \quad (-\nabla_z \varphi(\bar{x}, \bar{z})\alpha, -\beta) \in N(\bar{u}, \varphi(\bar{x}, \bar{z})), gph N_{R^b_+}).$$

λ can be taken as 1 if one of the following constraint qualifications hold:

(a) *$\nabla_z g, \psi, \varphi$ are affine mappings and D is polyhedral.*

(b) *There is no nonzero vector $(\gamma, \alpha, \beta) \in R^d_+ \times R^a \times R^b$ such that*

$$0 \in \gamma \nabla \psi(\bar{x}, \bar{z}) + \alpha \nabla(\nabla_z g + \bar{u}\nabla_z \varphi)^t(\bar{x}, \bar{z}) - \beta \nabla \varphi(\bar{x}, \bar{z}) + N((\bar{x}, \bar{z}), D),$$
$$\langle \psi(\bar{x}, \bar{z}), \gamma\rangle = 0, \quad (-\nabla_z \varphi(\bar{x}, \bar{z})\alpha, -\beta) \in N((\bar{u}, \varphi(\bar{x}, \bar{z})), gph N_{R^b_+}).$$

(c) *There exist $\mu > 0$ and $\epsilon > 0$ such that*

$$\begin{aligned}\mu^{-1} \leq &\inf\{\|(\xi_1, \xi_2)\| :\\ &\xi_1 \in e_1 \nabla \psi(x, z) + e_2 \nabla(\nabla_z g + u\nabla_z \varphi)^t(x, z) - e_3 \nabla \varphi(x, z) + N((x, z), D),\\ &(\xi_2 - \nabla_z \varphi(x, z)e_2, -e_3) \in N((u, q + \varphi(x, z)), gph N_{R^b_+}),\\ &\langle \psi(x, z) + p, e_1\rangle = 0, \|(e_1, e_2, e_3)\| = 1, e_1 \geq 0,\\ &(p, q) \neq 0, (x, z, u) \in \Sigma(p, q) \cap B_\epsilon(\bar{x}, \bar{z}, \bar{u})\},\end{aligned}$$

where

$$\begin{aligned}\Sigma(p, q) := \{&(x, z, u) \in C \times R^b : \psi(x, z) + p \leq 0,\\ &q \in ((\nabla_z g + u\nabla_z \varphi)^t(x, z), -\varphi(x, z)) + N((z, u), R^a \times R^b_+)\}.\end{aligned}$$

(d) *$D = E \times R^a$, where E is a closed subset of R^n and there is no inequality constraint $\psi(x, z) \leq 0$. Furthermore the strong second order sufficient condition and the linear independence of binding constraints hold for the lower level problem $P_{\bar{x}}$ at $\bar{z}$, i.e., for any nonzero v such that*

$$\nabla_z \varphi_i(\bar{x}, \bar{z})^t v = 0, \quad i \in L,$$

$\langle v, (\nabla^2_z g(\bar{x}, \bar{z}) + \bar{u}\nabla^2_z \varphi(\bar{x}, \bar{z}))v\rangle > 0$, and gradients of the binding constraints $\{\nabla_z \varphi_i(\bar{x}, \bar{z}), i \in L \cup I_0\}$ are linearly independent, where

$$\bar{u}\nabla^2_z \varphi(\bar{x}, \bar{z}) := \sum \bar{u}_i \nabla^2_z \varphi_i(\bar{x}, \bar{z})$$

and

$$\begin{aligned}L := L(\bar{x}, \bar{z}, \bar{u}) &:= \{i : \bar{u}_i > 0, \varphi_i(\bar{x}, \bar{z}) = 0\},\\ I_0 := I_0(\bar{x}, \bar{z}, \bar{u}) &:= \{i : \bar{u}_i = 0, \varphi_i(\bar{x}, \bar{z}) = 0\},\\ I_+ := I_+(\bar{x}, \bar{z}, \bar{u}) &:= \{i : \bar{u}_i = 0, \varphi_i(\bar{x}, \bar{z}) < 0\}.\end{aligned}$$

Proof. Since the objective function of the lower level problem g is pseudoconvex in z and the constraint φ is quasi-convex in z, by Theorem 4.2.11 of Bazaraa and Shetty [2] the Kuhn–Tucker condition is a necessary and sufficient condition for optimality. Therefore from the discussion preceding Theorem 5.1 we know that $(\bar{x}, \bar{z})$ is a solution of the following problem:

$$\begin{array}{lll} & \min & f(x,z) \\ (5.1) & \text{s.t.} & 0 \in ((\nabla_z g + u\nabla_z \varphi)^t(x,z), -\varphi(x,z)) + N((z,u), R^a \times R^b_+), \\ & & \psi(x,z) \leq 0, (x,z) \in C. \end{array}$$

Condition (a) is the linear constraint qualification (Linear CQ). Condition (b) is the no nonzero abnormal multiplier constraint qualification (NNAMCQ). Condition (c) is the bounded constraint qualification (Bounded CQ). Condition (d) is a sufficient condition for the strong regularity of the generalized equation (5.1) by virtue of [22, Theorem 4.1]. □

Remark. In the case where $D = \{(x,z) : h(x,z) \leq 0\}$ and $h(x,z) : R^{n+a} \to R^q$, if h is an affine mapping, it is known that

$$N((\bar{x}, \bar{z}), D) = \{\zeta \nabla h(\bar{x}, \bar{z}) : \zeta \in R^q_+, \langle h(\bar{x}, \bar{z}), \zeta \rangle = 0\}.$$

In this case, the necessary optimality condition becomes the existence of $\lambda \geq 0$, $\gamma \in R^d_+$, $\alpha \in R^a, \beta \in R^b$ not all zero and $\zeta \in R^q_+$ such that

$$\begin{array}{l} 0 = \lambda \nabla f(\bar{x}, \bar{z}) + \gamma \nabla \psi(\bar{x}, \bar{z}) + \alpha \nabla (\nabla_z g + \bar{u} \nabla_z \varphi)^t(\bar{x}, \bar{z}) - \beta \nabla \varphi(\bar{x}, \bar{z}) + \zeta \nabla h(\bar{x}, \bar{z}), \\ \langle h(\bar{x}, \bar{z}), \zeta \rangle = 0, \langle \psi(\bar{x}, \bar{z}), \gamma \rangle = 0, \\ (-\nabla_z \varphi(\bar{x}, \bar{z})\alpha, -\beta) \in N(\bar{u}, \varphi(\bar{x}, \bar{z})), gphN_{R^b_+}). \end{array}$$

Hence incorporating an abstract constraint in (OPVIC) can be used as a useful device to handle linear and nonlinear constraints separately.

REFERENCES

[1] J.-P. AUBIN, *Lipschitz behavior of solutions to convex minimization problems*, Math. Oper. Res., 9 (1994), pp. 87–111.

[2] M.S. BAZARAA AND C.M. SHETTY, *Nonlinear Programming Theory and Algorithms*, John Wiley & Sons, New York, 1979.

[3] F.H. CLARKE, *Optimization and Nonsmooth Analysis*, Wiley-Interscience, New York, 1983.

[4] F.H. CLARKE, YU. S. LEDYAEV, R.J. STERN, AND P.R. WOLENSKI, *Nonsmooth Analysis and Control Theory*, Springer, New York, 1998.

[5] F.H. CLARKE, R.J. STERN, AND P.R. WOLENSKI, *Subgradient criteria for monotonicity, the Lipschitz condition, and convexity*, Canad. J. Math., 45 (1993), pp. 1167–1183.

[6] A.L. DONTCHEV AND R.T. ROCKAFELLAR, *Characterizations of strong regularity for variational inequalities over polyhedral convex sets*, SIAM J. Optim., 6 (1996), pp. 1087–1105.

[7] A.J. HOFFMAN, *On approximate solutions of systems of linear inequalities*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 263–265.

[8] A. JOURANI, *Constraint qualifications and Lagrange multipliers in nondifferentiable programming problems*, J. Optim. Theory Appl., 81 (1994), pp. 533–548.

[9] P.D. LOEWEN,*Optimal Control via Nonsmooth Analysis*, CRM Proc. Lecture Notes 2, AMS, Providence, RI, 1993.

[10] Z.Q. LUO, J.S. PANG AND D. RALPH, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, New York, 1996.

[11] B.S. MORDUKHOVICH, *Maximum principle in problems of time optimal control with nonsmooth constraints*, J. Appl. Math. Mech., 40 (1976), pp. 960–969.

[12] B.S. MORDUKHOVICH, *Metric approximation and necessary optimality conditions for general classes of nonsmooth extremal problems*, Soviet Math. Dokl., 22 (1980), pp. 526–530.

[13] B.S. Mordukhovich, *Approximation Methods in Problems of Optimization and Control*, Nauka, Moscow, 1988.

[14] B.S. Mordukhovich, *Generalized differential calculus for nonsmooth and set-valued mappings*, J. Math. Anal. Appl., 183 (1994), pp. 250–288.

[15] B.S. Mordukhovich, *Lipschitz stability of constraint systems and generalized equations*, Nonlinear Anal., 173 (1994), pp. 173–206.

[16] B.S. Mordukhovich, *Stability theory for parametric generalized equations and variational inequalities via nonsmooth analysis*, Trans. Amer. Math. Soc., 343 (1994), pp. 609–655.

[17] B.S. Mordukhovich and Y. Shao, *Extremal characterization of Asplund spaces*, Trans. Amer. Math. Soc., 124 (1996), pp. 197–205.

[18] J.V. Outrata, *Optimality conditions for a class of mathematical programs with equilibrium constraints*, Math. Oper. Res., 24 (1999), pp. 627–644.

[19] R.T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer-Verlag, Berlin, 1998.

[20] R.A. Poliquin and R.T. Rockafellar, *Tilt stability of a local minimum*, SIAM J. Optim., 8 (1998), pp. 287–299.

[21] S.M. Robinson, *Stability theory for systems of inequalities. Part* I*: Linear systems*, SIAM J. Numer. Anal., 12 (1975), pp. 754–769.

[22] S.M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

[23] S.M. Robinson, *Some continuity properties of polyhedral multifunctions*, Math. Programming Stud., 14 (1981), pp. 206–214.

[24] J.J. Ye and D.L. Zhu, *Optimality conditions for bilevel programming problems*, Optimization, 33 (1995), pp. 9–27.

[25] J.J. Ye, *Optimality conditions for optimization problems with complementarity constraints*, SIAM J. Optim., 9 (1999), pp. 374–387.

[26] J.J Ye and X.Y. Ye, *Necessary optimality conditions for optimization problems with variational inequality constraints*, Math. Oper. Res., 22 (1997), pp. 977–997.

[27] J.J. Ye, D.L. Zhu, and Q.J. Zhu, *Exact penalization and necessary optimality conditions for generalized bilevel programming problems*, SIAM J. Optim., 7 (1997), pp. 481–507.

SIAM J. Optim.
Vol. 10, No. 4, pp. 963–981

ON THE CONSTANT POSITIVE LINEAR DEPENDENCE CONDITION AND ITS APPLICATION TO SQP METHODS*

LIQUN QI† AND ZENGXIN WEI‡

Abstract. In this paper, we introduce a constant positive linear dependence condition (CPLD), which is weaker than the Mangasarian–Fromovitz constraint qualification (MFCQ) and the constant rank constraint qualification (CRCQ). We show that a limit point of a sequence of approximating Karush–Kuhn–Tucker (KKT) points is a KKT point if the CPLD holds there. We show that a KKT point satisfying the CPLD and the strong second-order sufficiency conditions (SSOSC) is an isolated KKT point. We then establish convergence of a general sequential quadratical programming (SQP) method under the CPLD and the SSOSC. Finally, we apply these results to analyze the feasible SQP method proposed by Panier and Tits in 1993 for inequality constrained optimization problems. We establish its global convergence under the SSOSC and a condition slightly weaker than the Mangasarian–Fromovitz constraint qualification, and we prove superlinear convergence of a modified version of this algorithm under the SSOSC and a condition slightly weaker than the linear independence constraint qualification.

Key words. constrained optimization, KKT point, constraint qualification, feasible SQP method, global convergence, superlinear convergence

AMS subject classifications. 90C30, 60K05

PII. S1052623497326629

1. Introduction.

Consider the *constrained optimization problem*

$$\min\{f(x) \mid x \in X\}, \tag{1.1}$$

where $X = \{x \in \Re^n \mid g(x) \leq 0, \quad h(x) = 0\}$, $f : \Re^n \to \Re$, $g : \Re^n \to \Re^m$, and $h : \Re^n \to \Re^p$ are continuously differentiable functions. Assume that $X \neq \emptyset$. Let $I = \{1, \ldots, m\}$ and $J = \{1, \ldots, p\}$. For a vector $d \in \Re^q$, we let $\operatorname{supp}(d) = \{j \mid d_j \neq 0\}$.

Let $x \in X$ be a given *feasible point* of (1.1). Let

$$I(x) = \{j \in I \mid g_j(x) = 0\},$$

$$S(x) = \{\nabla g_j(x) \mid j \in I(x)\},$$

and

$$T(x) = \{\nabla h_j(x) \mid j \in J\}.$$

We call a feasible point x a *Karush–Kuhn–Tucker (KKT) point* of (1.1) if there exist vectors $u \in \Re^m$ and $v \in \Re^p$ such that the following requirements are simultaneously

*Received by the editors August 29, 1997; accepted for publication (in revised form) October 12, 1999; published electronically June 6, 2000. This work was supported by the Australian Research Council.

http://www.siam.org/journals/siopt/10-4/32662.html

†School of Mathematics, The University of New South Wales, Sydney, New South Wales 2052, Australia (L.Qi@unsw.edu.au). Department of Applied Mathematics, Hong Kong Polytechnic University, Hong Kong.

‡Department of Mathematics, Guangxi University, Nanning, Guangxi, China (zxwei@gxu.edu.cn).

satisfied:

$$(1.2) \qquad \begin{cases} \nabla f(x) + \sum_{j\in I} u_j \nabla g_j(x) + \sum_{j\in J} v_j \nabla h_j(x) = 0; \\ u \geq 0; \\ u^T g(x) = 0. \end{cases}$$

We call the pair (u, v) a *Lagrange multiplier* at x and denote the set of all possible Lagrange multipliers associated with x by $M(x)$. For a given $x \in X$, if we regard (1.2) as the constraints of a linear program with (u, v) as variables, we see that if x is a KKT point, there is a $(u, v) \in M(x)$ such that the vectors $\{\nabla g_j(x) \mid j \in \text{supp}(u)\} \cup \{\nabla h_j(x) \mid j \in \text{supp}(v)\}$ are linearly independent. We call such a (u, v) a *regular Lagrange multiplier* of x.

For convenience, we set $M(x) = \emptyset$ if x is not a KKT point. We say that x is an *isolated KKT point* of (1.1) if there is a neighborhood of x such that x is the only KKT point in this neighborhood. Note that an isolated KKT point may have more than one Lagrange multiplier.

In *sequential quadratic programming (SQP) methods* [7, 10, 23, 24, 9] and KKT equation methods [26] for solving (1.1), at each step, an approximate KKT point of (1.1) is found. Is any limit point of a sequence of approximate KKT points a KKT point of (1.1)? If it is, will the whole sequence converge to it? Under which conditions is a KKT point stable with respect to perturbations? In the next section, we formally define an *approximate KKT point sequence* and introduce a regularity condition called the *constant positive linear dependence condition (CPLD).* The CPLD is weaker than the well-known Mangasarian–Fromovitz constraint qualification (MFCQ) [16] and the constant rank constraint qualification (CRCQ); moreover, the MFCQ and the CRCQ together are weaker than the linear independence constraint qualification (LICQ). We show that a limit point x^* of an approximate KKT point sequence is a KKT point of (1.1) if the CPLD holds at x^*. In section 3, we show that if a KKT point x satisfies the CPLD and the strong second-order sufficiency conditions (SSOSC) [31], then it is an isolated KKT point. Hence, a limit point x^* of an approximate KKT point sequence is a KKT point of (1.1) and the whole sequence will converge to it if both the CPLD and the SSOSC hold at x^*. We state in section 3 a Kojima theorem on perturbed KKT points under the MFCQ and the SSOSC. The Kojima theorem will be used in section 6.

SQP methods constitute an important class of methods for solving (1.1). They enjoy local superlinear convergence under mild conditions [7, 10, 23, 24, 9]. The superlinear convergence of SQP methods was first established [7, 10] under a set of conditions: the LICQ, the second-order sufficiency conditions, and the strict complementarity slackness. This set of conditions was first studied in [5] and is called the Jacobian uniqueness condition [10]. Robinson [31] reduced the second-order sufficiency conditions and the strict complementarity slackness to the SSOSC. Robinson's condition has been used for classical SQP methods and KKT equations methods in [1, 9, 26]. What is difficult is to relax the LICQ. The relaxation of the LICQ may result in multiple Lagrange multipliers. Only recently, several authors [6, 28, 35] began to study the convergence of algorithms on problems with nonunique Lagrange multipliers. In section 4, we apply the results in sections 2 and 3 to a general SQP method and establish its convergence under the CPLD and the SSOSC. In sections 5 and 6, we further apply these results to a feasible SQP method. For classical SQP methods, the iteration points may be infeasible, while feasible SQP methods take special precautions to guarantee that the iteration points are feasible. Panier and Tits [17, 18] proposed two feasible SQP methods in 1987 and 1993. They established global and superlinear convergence of their feasible SQP methods under the classical Jacobian uniqueness condition. In section 5, we establish global convergence of the 1993 Panier–Tits method [18] under the condition that the CPLD and the SSOSC hold for a limit point of the primal iterative sequence and the MFCQ holds at all non-KKT points in X. In section 6, we first modify the 1993 Panier–Tits algorithm slightly; then, with the help of the Kojima theorem, we establish superlinear convergence of the modified algorithm under the SSOSC and a condition slightly weaker than the LICQ. In this way, both the strict complementarity condition and the LICQ, assumed in [18], are relaxed. These results can be extended to the 1987 Panier–Tits method.

Throughout the paper, we denote the Euclidean norm of a vector v by $\|v\|$, the corresponding induced norm of a matrix H by $\|H\|$, and the cardinality of a finite set J by $|J|$ and let $\mathcal{N} \equiv \{1, 2, \ldots\}$.

2. Limiting point of an approximate KKT point sequence.

We first review the concept of positive linear independence for vectors [21, 32, 33, 29].

Definition 2.1. *Let $A = \{a^1, \ldots, a^l\}$ and $B = \{b^1, \ldots, b^r\}$ be two finite subsets of $\Re^n$ such that $A \cup B \neq \emptyset$. We say that (A, B) is positive-linearly dependent if there are $\alpha \in \Re^l$ and $\beta \in \Re^r$ such that $\alpha \geq 0, (\alpha, \beta) \neq 0$, and*

$$\sum_{j=1}^{l} \alpha_j a^j + \sum_{j=1}^{r} \beta_j b^j = 0.$$

Otherwise, we say that (A, B) is positive-linearly independent. *If $B = \emptyset$, we simply say that A is positive-linearly dependent or independent.*

Clearly, just as linearly independent and dependent sets, a subset pair of a positive-linearly independent set pair is always positive-linearly independent and a set pair with a positive-linearly dependent subset pair is always positive-linearly dependent.

Proposition 2.2. *Let $G_j : \Re^n \to \Re^n, j = 1, \ldots, l,$ and $H_j : \Re^n \to \Re^n, j = 1, \ldots, r,$ be continuous functions. If $(\{G_j(x)\}_{j=1}^l, \{H_j(x)\}_{j=1}^r)$ is positive-linearly independent for $x \in \Re^n$, then there is a neighborhood $N(x)$ of x such that for any $y \in N(x)$, $(\{G_j(y)\}_{j=1}^l, \{H_j(y)\}_{j=1}^r)$ is positive-linearly independent.*

Proof. If such a neighborhood does not exist, then there is a sequence $\{y^k\}_{k=1}^\infty \subset \Re^n$ with $y^k \to x$ as $k \to +\infty$ and $\alpha^k \geq 0, \|(\alpha^k, \beta^k)\| \equiv 1$, such that

$$\sum_{j=1}^{l} \alpha_j^k G_j(y^k) + \sum_{j=1}^{r} \beta_j^k H_j(y^k) = 0.$$

Without loss of generality, we may assume that $\alpha^k \to \alpha^*$ and $\beta^k \to \beta^*$ as $k \to +\infty$. Clearly,

$$\begin{cases} \sum_{j=1}^{l} \alpha_j^* G_j(x) + \sum_{j=1}^{r} \beta_j^* H_j(x) = 0; \\ \alpha^* \geq 0; \\ \|(\alpha^*, \beta^*)\| = 1. \end{cases}$$

This gives a contradiction. □

This proposition will be used in later sections.

PROPOSITION 2.3. *For any given $x \in X$, assume that $\nabla f(x)$, $\nabla g_j(x), j \in I(x)$, and $\nabla h_j(x), j \in J$, are not all zero. Then x is a KKT point of* (1.1), *i.e., $M(x) \neq \emptyset$, if and only if there is a subset $S_0(x) \subseteq S(x)$ and a subset $T_0(x)$ of $T(x)$ such that $(S_0(x), T_0(x))$ is positive-linearly independent while $(S_0(x) \bigcup \{\nabla f(x)\}, T_0(x))$ is positive-linearly dependent.*

Proof. The case when $\nabla f(x) = 0$ is trivial. Thus we assume that $\nabla f(x) \neq 0$.

[$\Rightarrow$]. If $M(x) \neq \emptyset$, then there exist vectors $u = u^1 \in \Re^m$ and $v = v^1 \in \Re^p$ such that (1.2) holds. Let $I_1 = \mathrm{supp}(u)$, $J_1 = \mathrm{supp}(v)$, $S_1 = \{\nabla g_j(x) \mid j \in I_1\}$, and $T_1 = \{\nabla h_j(x) \mid j \in J_1\}$. Since $\nabla f(x) \neq 0$, by the first equality of (1.2), $S_1 \cup T_1 \neq \emptyset$. If (S_1, T_1) is positive-linearly independent, then let $S_0(x) = S_1$ and $T_0(x) = T_1$, and the first equality in (1.2) implies that $(S_0(x) \bigcup \{\nabla f(x)\}, T_0(x))$ is positive-linearly dependent. If (S_1, T_1) is positive-linearly dependent, then we have $\alpha_j \geq 0, j \in I_1$, and $\beta_j, j \in J_1$, such that not all of α_j and β_j are zero and

$$\sum_{j \in I_1} \alpha_j \nabla g_j(x) + \sum_{j \in J_1} \beta_j \nabla h_j(x) = 0.$$

If some $\alpha_j \neq 0$, let $\lambda = \min\{\frac{u_j}{\alpha_j} \mid j \in I_1, \alpha_j \neq 0\}$; otherwise, there is $\bar{j} \in J_1$ such that $\beta_{\bar{j}} \neq 0$, and we then let $\lambda = \frac{v_{\bar{j}}}{\beta_{\bar{j}}}$. Let $u_j^2 = u_j - \lambda\alpha_j$ for $j \in I_1$, $u_j^2 = 0$ for $j \notin I_1$, $v_j^2 = v_j - \lambda\beta_j$ for $j \in J_1$, and $v_j^2 = 0$ for $j \notin J_1$. Then $(u, v) = (u^2, v^2)$ still satisfies (1.2) but its support set is strictly contained in $I_1 \cup J_1$, the support sets of S_1 and T_1. Repeat this process. Finally, we have a subset $S_0(x)$ of $S(x)$ and a subset $T_0(x)$ of $T(x)$, which satisfy the requirements.

[$\Leftarrow$]. Assume that $I_0 \subseteq I(x)$ and $J_0 \subseteq J$ such that $S_0(x) = \{\nabla g_j(x) \mid j \in I_0\}$ and $T_0(x) = \{\nabla h_j(x) \mid j \in J_0\}$ satisfy the requirements. The fact that $(S_0(x) \bigcup \{\nabla f(x)\}, T_0(x))$ is positive-linearly dependent implies that there are $\gamma \in \Re$, $\alpha \in \Re^{|I_0|}$, and $\beta \in \Re^{|J_0|}$ such that $\gamma \geq 0, \alpha \geq 0, (\gamma, \alpha, \beta) \neq 0$, and

$$\gamma \nabla f(x) + \sum_{j \in I_0} \alpha_j \nabla g_j(x) + \sum_{j \in J_0} \beta_j \nabla h_j(x) = 0.$$

These and the assumption that $(S_0(x), T_0(x))$ is positive-linearly independent imply that $\gamma > 0$. Let $u_j = \frac{\alpha_j}{\gamma}$ for $j \in I_0$, $u_j = 0$ for $j \notin I_0$, $v_j = \frac{\beta_j}{\gamma}$ for $j \in J_0$, and $v_j = 0$ for $j \notin J_0$. Then (u, v) satisfies (1.2). Hence, $M(x) \neq \emptyset$. □

A given feasible point $x \in X$ is said to satisfy the MFCQ [16] if $T(x)$ is linearly independent and there is a vector $z \in \Re^n$ such that

$$(\nabla g_{I(x)}(x))^T z < 0$$

and

$$(\nabla h(x))^T z = 0.$$

The following proposition was given in section 1.8 of [21].

PROPOSITION 2.4. *For any given $x \in X$, assume that $I(x) \cup J \neq \emptyset$. Then the MFCQ holds at x if and only if $(S(x), T(x))$ is positive-linearly independent.*

Proof. If $I(x) = \emptyset$, the conclusion is obvious; otherwise, the conclusion follows Motzkin's theorem of the alternative [16].

We now define an approximate KKT point sequence of (1.1).

DEFINITION 2.5. *We say that* $\{x^k\}_{k=1}^{\infty} \subset \Re^n$ *is an approximate KKT point sequence of* (1.1) *if there is a sequence* $\{(u^k, v^k, \epsilon^k, \delta^k, \lambda^k)\}_{k=1}^{\infty} \subset \Re^m \times \Re^p \times \Re^n \times \Re^m \times \Re$ *such that the following requirements are simultaneously satisfied for each* k*:*

$$(2.1) \qquad \begin{cases} \nabla f(x^k) + \sum_{j\in I} u_j^k \nabla g_j(x^k) + \sum_{j \in J} v_j^k \nabla h_j(x^k) = \epsilon^k; \\ g(x^k) \le \delta^k; \\ u^k \ge 0; \\ (u^k)^T(g(x^k) - \delta^k) = 0; \\ \|h(x^k)\| \le \lambda^k; \end{cases}$$

and $\{(\epsilon^k, \delta^k, \lambda^k)\}_{k=1}^{\infty}$ *converges to zero as* $k \to \infty$.

Such an approximate KKT point sequence is produced by SQP methods, KKT equations methods, and some other methods for solving (1.1). If x^* is a limit point of $\{x^k\}$, or without loss of generality, if $\{x^k\}$ converges to x^*, is x^* a KKT point of (1.1)? To answer this question, we introduce a regularity condition.

DEFINITION 2.6. *A given feasible point* $x \in X$ *is said to satisfy the CPLD if for any* $I_0 \subseteq I(x)$ *and* $J_0 \subseteq J$, *whenever* $(\{\nabla g_j(x) \mid j \in I_0\}, \{\nabla h_j(x) \mid j \in J_0\})$ *is positive-linearly dependent, there is a neighborhood* $N(x)$ *of* x *such that for any* $y \in N(x)$, $(\{\nabla g_j(y) \mid j \in I_0\}, \{\nabla h_j(y) \mid j \in J_0\})$ *is linearly dependent.*

Note that in the definition we do not require that $(\{\nabla g_j(y) \mid j \in I_0\}, \{\nabla h_j(y) \mid j \in J_0\})$ be positive-linearly dependent, which is stronger than our requirement here. By Propositions 2.2 and 2.4, the CPLD is weaker than the MFCQ.

It is said that the CRCQ [11, 15, 20, 34, 22] holds at $x \in X$ if there is a neighborhood $N(x)$ of x such that for every $I_0 \subseteq I(x)$ and $J_0 \subseteq J$, the family of gradient vectors

$$\{\nabla g_j(y) \mid j \in I_0\} \bigcup \{\nabla h_j(y) \mid j \in J_0\}$$

has the same rank (which depends on I_0 and J_0) for all vectors $y \in N(x)$. It is not difficult to see that the CRCQ holds at x if and only if for any $I_0 \subseteq I(x)$ and $J_0 \subseteq J$, whenever $\{\nabla g_j(x) \mid j \in I_0\} \bigcup \{\nabla h_j(x) \mid j \in J_0\}$ is linearly dependent, there is a neighborhood $N(x)$ of x such that for any $y \in N(x)$, $\{\nabla g_j(y) \mid j \in I_0\} \bigcup \{\nabla h_j(y) \mid j \in J_0\}$ is linearly dependent. Hence, the CPLD is also weaker than the CRCQ. Note [11] that neither the CRCQ implies the MFCQ nor the MFCQ implies the CRCQ. Furthermore, even the MFCQ and the CRCQ together are weaker than the LICQ. This can be seen from the following example: $n = m = 2, p = 0, g_1(x) = x_1 + x_2, g_2(x) = 2x_1 + 2x_2$, at $x = (0, 0)^T$. If x is a local minimum point of (1.1) and the CPLD holds at x, is x always a KKT point of (1.1)? If so, we may also call the CPLD a constraint qualification, but at this moment we only use the CPLD to derive the following result.

THEOREM 2.7. *If an approximate KKT point sequence* $\{x^k\}_{k=1}^{\infty}$ *converges to* x^* *as* $k \to \infty$ *and the CPLD holds at* x^*, *then* x^* *is a KKT point of* (1.1), *i.e., there are a* $u^* \in \Re^m$ *and a* $v^* \in \Re^p$ *such that* (x^*, u^*, v^*) *satisfies* (1.2).

Proof. By using the theory of linear programming, we may assume, without loss of generality, for any given k, there is $(\bar{u}^k, \bar{v}^k)$ satisfying (2.1) such that

$$\{\nabla g_j(x^k) \mid j \in \operatorname{supp}(\bar{u}^k)\} \bigcup \{\nabla h_j(x^k) \mid j \in \operatorname{supp}(\bar{v}^k)\}$$

is linearly independent. Let $I_k = \text{supp}(\bar{u}^k)$ and $J_k = \text{supp}(\bar{v}_k)$. Without loss of generality, we may assume that $I_0 \equiv I_k$ and $J_0 \equiv J_k$. Then $I_0 \subseteq I(x^*)$ and $J_0 \subseteq J$. If $\{(\bar{u}^k, \bar{v}^k)\}_{k=1}^{\infty}$ has a bounded subsequence, then, without loss of generality, we may assume that there are $u^* \in \Re^m$ and $v^* \in \Re^p$ such that $\bar{u}^k \to u^*$ and $\bar{v}^k \to v^*$ as $k \to \infty$. Letting k tend to infinity in (2.1), we see that (x^*, u^*, v^*) satisfies (1.2), and hence the conclusion holds for this case. We assume now that

$$\lim_{k\to\infty} \|(u^k, v^k)\| = +\infty.$$

Without loss of generality, we may assume that

$$\lim_{k\to\infty} \frac{(u^k, v^k)}{\|(u^k, v^k)\|} = (\alpha, \beta).$$

Then $\|(\alpha, \beta)\| = 1$, $\text{supp}(\alpha) \subseteq I_0$, $\text{supp}(\beta) \subseteq J_0$, and $\alpha \geq 0$. Dividing both sides of

$$\nabla f(x^k) + \sum_{j\in I} \bar{u}_j^k \nabla g_j(x^k) + \sum_{j\in J} \bar{v}_j^k \nabla h_j(x^k) = \epsilon^k$$

by $\|(u^k, v^k)\|$ and letting k tend to infinity in the above equality, we obtain

$$\sum_{j\in I_0} \alpha_j \nabla g_j(x^*) + \sum_{j\in J_0} \beta_j \nabla h_j(x^*) = 0.$$

This implies that $(\{\nabla g_j(x^*) \mid j \in I_0\}, \{\nabla h_j(x^*) \mid j \in J_0\})$ is positive-linearly dependent. By the assumptions that the CPLD holds at x^* and $x^k \to x^*$, we have that for all large k, $(\{\nabla g_j(x^k) \mid j \in I_0\}, \{\nabla h_j(x^k) \mid j \in J_0\})$ are linearly dependent. This contradicts the fact that $\{\nabla g_j(x^k) \mid j \in I_0\} \bigcup \{\nabla h_j(x^k) \mid j \in J_0\}$ are linearly independent for all k. □.

3. Isolated and stable KKT points. We now assume that f, g, and h are twice continuously differentiable.

For any $x \in \Re^n, u \in \Re^m$, and $v \in \Re^p$, we denote the Lagrange function of (1.1) by

$$L(x, u, v) = f(x) + u^T g(x) + v^T h(x).$$

By Robinson [31], a triplet (x, u, v) is said to satisfy the SSOSC if it satisfies the KKT conditions (1.2) and $\nabla_{xx} L(x, u, v)$ is positive definite on the subspace

$$G(x, u, v) = \{d \in \Re^n \mid \nabla f(x)^T d = 0, \;\; \nabla g_j(x)^T d = 0 \;\text{ for }\; j \in \text{supp}(u),$$
$$\nabla h_j(x)^T d = 0 \;\text{ for }\; j \in J\}.$$

Note that even under the second-order sufficiency conditions, x will be a strict local minimum of (1.1).

DEFINITION 3.1. *Suppose that x is a KKT point of* (1.1). *If for all Lagrange multipliers (u, v) of x, (x, u, v) satisfies the SSOSC, then we say that x satisfies the SSOSC.*

THEOREM 3.2. *Suppose that x^* is a KKT point of* (1.1). *If x^* satisfies the CPLD and the SSOSC, then x^* is an isolated KKT point of* (1.1).

Proof. Suppose that x^* is not an isolated KKT point of (1.1). Then there is a KKT point sequence $\{x^k\}_{k=1}^{\infty}$ such that $x^k \neq x^*$ and $\lim_{k\to\infty} x^k = x^*$. It follows from the theory of linear programming that for each k, there is a regular Lagrange multiplier (u^k, v^k) for x^k. Let $I_k = \mathrm{supp}(u^k)$ and $J_k = \mathrm{supp}(v_k)$. Without loss of generality, we may assume that $I_0 \equiv I_k$ and $J_0 \equiv J_k$ for all k. Then

$$\{\nabla g_j(x^k) \ \mid \ j \in I_0\} \bigcup \{\nabla h_j(x^k) \ \mid \ j \in J_0\}$$

is linearly independent for all k. By the CPLD at x^*,

$$(\{\nabla g_j(x^*) \ \mid \ j \in I_0\}, \{\nabla h_j(x^*) \ \mid \ j \in J_0\})$$

is positive-linearly independent.

If $\{(u_k, v_k)\}_{k=1}^{\infty}$ is unbounded, without loss of generality, we may assume that

$$\lim_{k\to\infty} \|(u^k, v^k)\| = +\infty,$$

$$\lim_{k\to\infty} \frac{(u^k, v^k)}{\|(u^k, v^k)\|} = (\alpha, \beta),$$

$\|(\alpha, \beta)\| = 1, \alpha \geq 0, \mathrm{supp}(\alpha) \subseteq I_0$, and $\mathrm{supp}(\beta) \subseteq J_0$. Then dividing

$$\nabla f(x^k) + \sum_{j\in I} u_j^k \nabla g_j(x^k) + \sum_{j\in J} v_j^k \nabla h_j(x^k) = 0$$

by $\|(u^k, v^k)\|$ and letting $k \to \infty$, we have

$$\sum_{j\in I_0} \alpha_j \nabla g_j(x^*) + \sum_{j\in J_0} \beta_j \nabla h_j(x^*) = 0.$$

This contradicts the fact that

$$(\{\nabla g_j(x^*) \ \mid \ j \in I_0\}, \{\nabla h_j(x^*) \ \mid \ j \in J_0\})$$

is positive-linearly independent.

Hence $\{(u^k, v^k)\}_{k=1}^{\infty}$ is bounded. Without loss of generality, we may assume that $u^k \to u^*$ and $v^k \to v^*$. Then $(u^*, v^*) \in M(x^*)$ is a Lagrange multiplier of x^*, $\mathrm{supp}(u^*) \subseteq I_0$, and $\mathrm{supp}(v^*) \subseteq J_0$. We may assume that

$$\lim_{k\to\infty} \frac{x^k - x^*}{\|x^k - x^*\|} = d.$$

Then $\|d\| = 1$. Since

$$g_j(x^k) - g_j(x^*) = 0, \qquad j \in I_0,$$

and

$$h_j(x^k) - h_j(x^*) = 0, \qquad j \in J,$$

we have, by Taylor's theorem, that

$$g_j(x^k) - g_j(x^*) = \nabla g_j(x^*)^T (x^k - x^*) + o(\|x^k - x^*\|), \qquad j \in I_0, \tag{3.1}$$

and

$$h_j(x^k) - h_j(x^*) = \nabla h_j(x^*)^T(x^k - x^*) + o(\|x^k - x^*\|), \qquad j \in J. \tag{3.2}$$

Dividing (3.1) and (3.2) by $\|x^k - x^*\|$ and letting $k \to \infty$, we have

$$\nabla g_j(x^*)^T d = 0, \qquad j \in I_0, \tag{3.3}$$

and

$$\nabla h_j(x^*)^T d = 0, \qquad j \in J. \tag{3.4}$$

On the other hand, since $(u^*, v^*) \in M(x^*)$, we have

$$\nabla f(x^*) + \sum_{j \in \mathrm{supp}(u^*)} u_j^* \nabla g_j(x^*) + \sum_{j \in \mathrm{supp}(v^*)} v_j^* \nabla h_j(x^*) = 0.$$

This formula, combined with (3.3), (3.4) and observing that $\mathrm{supp}(u^*) \subseteq I_0$ and $\mathrm{supp}(v^*) \subseteq J$, yields

$$\nabla f(x^*)^T d = 0. \tag{3.5}$$

From (3.3), (3.4), and (3.5), we have $d \in G(x^*, u^*, v^*)$. For any given k and $t \in [0,1]$, let

$$(x^t, u^t, v^t) = (1-t)(x^*, u^*, v^*) + t(x^k, u^k, v^k).$$

Then, Robinson's function [31, 8] is defined by

$$\begin{aligned} s(t) = (x^k - x^*)^T & \left[\nabla f(x^t) + \sum_{j \in I_0} u_j^t \nabla g_j(x^t) + \sum_{j \in J_0} v_j^t \nabla h_j(x^t) \right] \\ & - (u^k - u^*)^T g(x^t) - (v^k - v^*)^T h(x^t). \end{aligned}$$

The function $s : [0,1] \to \Re$ is clearly continuous on $[0,1]$ and continuously differentiable on $(0,1)$. Moreover, $s(0) = 0 = s(1)$. By the mean-value theorem, for any given k, there exists $t_k \in (0,1)$ such that $s'(t_k) = 0$, i.e.,

$$(x^k - x^*)^T \nabla_{xx} L(x^{t_k}, u^{t_k}, v^{t_k})(x^k - x^*) = 0.$$

Dividing this inequality by $\|x^k - x^*\|^2$ and passing to the limit $k \to \infty$, we obtain

$$d^T \nabla_{xx} L(x^*, u^*, v^*) d = 0.$$

This formula, combined with the facts that $d \in G(x^*, u^*, v^*)$ and $\nabla_{xx} L(x^*, u^*, v^*)$ is positive definite in $G(x^*, u^*, v^*)$, implies that $d = 0$, which contradicts the fact that $\|d\| = 1$. This proves the theorem. □

Remark. It is possible to reduce the requirement of twice differentiability of f, g, and h to semismoothness of $\nabla f, \nabla g$, and ∇h. Such an optimization problem is called an SC^1 optimization problem. For SC^1 optimization and its applications, see [24, 19, 4, 9, 3, 12, 27, 13, 2, 25].

THEOREM 3.3. *Suppose that x^* is a limit point of an approximate KKT point sequence $\{x^k\}_{k=1}^{\infty}$ of* (1.1) *and the CPLD and the SSOSC hold at x^*. If*

$$\lim_{k \to \infty} \|x^{k+1} - x^k\| = 0, \tag{3.6}$$

then $\lim_{k\to\infty} x^k = x^*$.

Proof. By Theorem 2.7, we have that x^* is a KKT point of (1.1) and every accumulation point of $\{x^k\}_{k=1}^{\infty}$ is a KKT point of (1.1). The assumptions that the CPLD and the SSOSC hold at x^* and Theorem 3.2 imply that x^* is an isolated KKT point of (1.1), i.e., there is $\epsilon > 0$ such that the ball $O(x^*, \epsilon) = \{x \in \Re^n, \ | \ \|x - x^*\| \leq \epsilon\}$ does not contain any KKT point other than x^*. On the other hand, (3.6) implies that for k large enough, $\|x^{k+1} - x^k\| < \frac{\epsilon}{4}$ and there exists a subsequence $\{x^k\}_{k\in\mathcal{K}}$ such that $\|x^k - x^*\| < \frac{\epsilon}{4}$ on $\mathcal{K}$. It is then impossible to leave $O(x^*, \epsilon)$ without creating another cluster point and hence a KKT point in that ball. □

In the remaining part of this section, as in [14], $\|\cdot\|$ is for the infinity norm instead of the Euclidean norm. Let $N(x, \delta) = \{y \in \Re^n : \|y - x\| \leq \delta\}$.

Consider the perturbed form of (1.1)

$$\min\{f(x) + \bar{f}(x) \ | \ x \in \bar{X}\}, \tag{3.7}$$

where $\bar{X} = \{x \in \Re^n \ | \ g(x) + \bar{g}(x) \leq 0, \quad h(x) + \bar{h}(x) = 0\}$, $f, \bar{f} : \Re^n \to \Re$, $g, \bar{g} : \Re^n \to \Re^m$, and $h, \bar{h} : \Re^n \to \Re^p$ are twice continuously differentiable functions.

DEFINITION 3.4. *Let x^* be a KKT point of* (1.1). *We call x^* a strongly stable KKT point of* (1.1) *if for some $\delta^* > 0$ and each $\delta \in (0, \delta^*]$ there exists an $\alpha > 0$ such that whenever twice continuously differentiable functions $\bar{f}, \bar{g}$, and $\bar{h}$ satisfy*

$$\sup_{\substack{\|x - x^*\| \leq \delta^* \\ i\in I, j\in J}} \{|\bar{f}(x)|, |\bar{g}_i(x)|, |\bar{h}_j(x)|, \|\nabla \bar{f}(x)\|, \|\nabla \bar{g}_i(x)\|, \|\nabla \bar{h}_j(x)\|,$$
$$\|\nabla^2 \bar{f}(x)\|, \|\nabla^2 \bar{g}_i(x)\|, \|\nabla^2 \bar{h}_j(x)\|\} \leq \alpha,$$

$N(x^*, \delta)$ contains a solution $\bar{x}^*$ of (3.7)*, which is unique in $N(x^*, \delta^*)$.*

The following theorem is Theorem 7.2 of [14]. We will use it in section 6.

THEOREM 3.5 (by Kojima [14]). *Suppose that x^* is a KKT point of* (1.1) *and that the MFCQ holds at x^*. Then x^* is a strongly stable KKT point of* (1.1) *if and only if for all $(u, v) \in M(x^*)$, (x^*, u, v) satisfies the SSOSC.*

Remark. The Kojima theorem can be regarded as an alternative to Robinson's perturbation theorem in [30]. Theorem 4.1 of [30] (together with Theorem 2.1 and Corollary 2.2 of the same paper) shows that under the SSOSC and the LICQ one has Lipschitzian behavior of the solution and the multipliers, with respect to perturbations, while the Kojima theorem shows that under the SSOSC and the MFCQ one has continuity of the solution and the multipliers, with respect to perturbations (but a counterexample in [31] shows that we cannot prove Lipschitz continuity in this situation). It is thus not surprising that in section 6 we must add the CRCQ to get our superlinear convergence result for a modified version of the 1993 Panier–Tits method. Note that the example in [31] does not satisfy the CRCQ. A question is, Is the Kojima theorem still true if the MFCQ is replaced by the CRCQ?

4. A general SQP method.

We describe a general SQP method as follows.

ALGORITHM A.

Let $C > 0$.

Data. $x^0 \in X, H_0 \in \Re^{n\times n}$, symmetric positive definite.

Step 0. (Initialization.) Set $k = 0$.

Step 1. (Computation of a search direction.) Compute d^k by solving the quadratic program

$$(QP)\left\{\begin{array}{ll} \min & \frac{1}{2}d^T H_k d + \nabla f(x^k)^T d \\ \\ \text{s.t.} & g_j(x^k) + \nabla g_j(x^k)^T d \leq 0, \quad j \in I, \\ \text{s.t.} & h_j(x^k) + \nabla h_j(x^k)^T d = 0, \quad j \in J. \end{array}\right.$$

If $d^k = 0$ stop.
Step 2. (Line search and additional correction.) Determine the steplength $\alpha_k \in (0,1)$ and a correction direction $\bar{d}^k$ such that

$$\|\bar{d}^k\| \leq C\|d^k\|. \tag{4.1}$$

Step 3. (Updates.) Compute a new symmetric positive definite approximation H_{k+1} to the Hessian of the Lagrangian. Set $x^{k+1} = x^k + \alpha_k d^k + \bar{d}^k$ and $k = k+1$. Go back to Step 1.

Algorithm A is a general model for SQP methods. For a specific SQP method, the rules for determining $\alpha_k, \bar{d}^k$, and H_k must be given. For classical SQP methods [23], $\bar{d}^k = 0$. We assume that the quadratic program (QP) is always solvable. This is obvious for feasible SQP methods since 0 is a feasible solution of (QP) in that case. Checking the KKT conditions of (QP) for $d = 0$, we have the following proposition.

PROPOSITION 4.1. *If Algorithm A stops in Step* 1*, then x^k is a KKT point of* (1.1).

Hence, we need only consider the case where Algorithm A generates an infinite sequence.

THEOREM 4.2. *Assume that Algorithm A generates an infinite sequence $\{x^k\}_{k=1}^\infty$ and that this sequence has an accumulation point x^*. Let $\mathcal{K}$ be a subsequence of $\mathcal{N}$ such that*

$$\lim_{k\in\mathcal{K}} x^k = x^*.$$

Suppose that the CPLD holds at x^* and that the Hessian estimates $\{H_k\}_{k=0}^\infty$ are bounded, i.e., there exists a scalar $C_1 > 0$ such that for all k

$$\|H_k\| \leq C_1. \tag{4.2}$$

If

$$\liminf_{k\in\mathcal{K}} \|d^k\| = 0, \tag{4.3}$$

then x^* is a KKT point of (1.1).

Proof. Without loss of generality, by passing to a subsequence if necessary, we may assume that

$$\lim_{k\in\mathcal{K}} \|d^k\| = 0. \tag{4.4}$$

By the KKT conditions of (QP), we have

$$\begin{cases} H_k d^k + \nabla f(x^k) + \nabla g(x^k)^T \bar{u}^k + \nabla h(x^k)^T u^k = 0; \\ g(x^k) + \nabla g(x^k)^T d^k \leq 0; \\ u^k \geq 0; \\ (u^k)^T (g(x^k) - \nabla g(x^k)^T d^k) = 0; \\ h(x^k) + \nabla h(x^k)^T d^k = 0. \end{cases}$$

By (4.2) and (4.4), as $k \to \infty$ for $k \in \mathcal{K}$, we have that

$$\epsilon_k \equiv -H_k d^k \to 0,$$

$$\delta_k \equiv -\nabla g(x^k)^T d^k \to 0,$$

and

$$\lambda_k \equiv \|\nabla h(x^k)^T d^k\| \to 0.$$

Then, by Theorem 2.7, x^* is a KKT point of (1.1). □

THEOREM 4.3. *Assume that the conditions of Theorem* 4.2 *hold. If, furthermore,* f, g, *and* h *are twice continuously differentiable, the SSOSC holds at* x^*, *and*

$$\lim_{k \to \infty} d^k = 0, \tag{4.5}$$

then $\lim_{k \to \infty} x^k = x^*$.

Proof. This follows from (4.1) and Theorem 3.3. □

We can establish superlinear convergence of the general SQP method by following, step by step, with minor modifications, the proofs of Lemma 3 to Theorem 1 of [23] and replacing $\nabla^2_{xx} L(x^*, u^*, v^*)$ with $\nabla^2_{xx} L(x^k, u^k, v^k)$ in (3.10) of [23]. We will see this more clearly in section 6.

To establish (4.3) or (4.5) one must use the properties of specific SQP methods. In the next section, we will establish these two conditions for a feasible SQP method.

5. Global convergence of a Panier–Tits method.

In this section, we establish the global convergence of the 1993 Panier–Tits feasible SQP method [18] under the SSOSC and a condition slightly weaker than the MFCQ. The global convergence of the 1987 Panier–Tits method [17] can be established in the same way. First of all, we describe the algorithm given in [18]. Keep in mind that the Panier–Tits methods are for inequality constrained optimization problems. Therefore, in this section and the next section, problem (1.1) becomes

$$\min\{f(x) \mid x \in X\}, \tag{5.1}$$

where $X = \{x \in \Re^n \mid g(x) \leq 0\}$.

5.1. A Panier–Tits method. In [18], a continuous map $d_1 : \Re^n \to \Re^n$ is needed in the algorithm such that

$$d_1(x) = 0 \ \text{ if } \ x \ \text{ is a KKT point of } \ (5.1), \tag{5.2}$$

$$\nabla f(x)^T d_1(x) < 0 \ \text{ if } \ x \ \text{ is not a KKT point of } \ (5.1), \tag{5.3}$$

and

$$\nabla g_j(x)^T d_1(x) < 0 \ \text{ if } \ x \ \text{ is not a KKT point of } \ (5.1) \ \text{ and } \ j \in I(x). \tag{5.4}$$

As indicated in [18], if the LICQ holds at x, then the continuous map $d_1(x)$ satisfying (5.2), (5.3), and (5.4), for example, can be obtained as the solution of

$$\min \ \frac{1}{2}\|d\|^2 + \max\{\nabla f(x)^T d; \ \ \max\{g_j(x) + \nabla g_j(x)^T d \ \mid \ j \in I\}\}. \tag{5.5}$$

We see from (5.4) that the existence of such a $d_1(x)$ implies that the MFCQ holds at all non-KKT points. On the other hand, if the MFCQ holds at all non-KKT points, then such a continuous map still exists (see section 2.6 of [21]). However, this does not require that the MFCQ hold at KKT points.

In the method of [18], it is necessary to have a map $\rho : \Re^n \to [0, 1]$ that is bounded away from zero outside every neighborhood of zero, and for v small

$$\rho(v) = O(\|v\|^2).$$

Since the existence of the map ρ is independent of problem (5.1), for sake of simplicity, we choose

$$\rho(v) = \frac{\|v\|^2}{1 + \|v\|^2}.$$

Establishing the convergence properties of the algorithm presents no difficulty when choosing other such maps.

The 1993 Panier–Tits method is as follows.

Algorithm B.

Let $C > 0, \tau_1 \in (0, \frac{1}{2}), \tau_2 \in (0, 1), \tau_3 \in (2, 3)$.

Data. $x^0 \in X, H_0 \in \Re^{n \times n}$, symmetric positive definite.

Step 0. (Initialization.) Set $k = 0$.

Step 1. (Computation of a search arc.)

(i) Compute d_0^k by solving the quadratic program

$$(QP_1) \begin{cases} \min \ \frac{1}{2} d^T H_k d + \nabla f(x^k)^T d \\ \text{s. t. } \ g_j(x^k) + \nabla g_j(x^k)^T d \le 0, \quad j \in I. \end{cases}$$

If $d_0^k = 0$, stop.

(ii) Let d_1^k be the solution of (5.5), $\rho_k = \rho(d_0^k)$, and $d^k = (1 - \rho_k) d_0^k + \rho_k d_1^k$.

(iii) Compute a correction $\tilde{d}^k$ as the solution of the problem

$$(QP_2) \begin{cases} \min \ \frac{1}{2}(d^k + d)^T H_k (d + d^k) + \nabla f(x^k)^T (d + d^k) \\ \text{s.t. } \ g_j(x^k + d^k) + \nabla g_j(x^k)^T d \le -\|d^k\|^{\tau_3}, \quad j \in I, \end{cases}$$

if it exists and has norm less than min $\{\|d^k\|, C\}$ and $\tilde{d}^k = 0$ otherwise. Hence, in any case, we have

$$\|\tilde{d}^k\| \leq \min \{\|d^k\|, C\}. \tag{5.6}$$

Step 2. (Arc search.)

Compute t_k, the first number t of the sequence $\{1, \tau_2, \tau_2^2, \ldots\}$ satisfying

$$f(x^k + td^k + t^2\tilde{d}^k) \leq f(x^k) + \tau_1 t \nabla f(x^k)^T d^k$$

and

$$g_j(x^k + td^k + t^2\tilde{d}^k) \leq 0, \qquad j \in I.$$

Step 3. (Updates.) Compute a new symmetric positive definite approximation H_{k+1} to the Hessian of the Lagrangian. Set $x^{k+1} = x^k + t_k d^k + t_k^2 \tilde{d}^k$ and $k = k+1$. Go back to Step 1.

We see that Algorithm B is a special case of Algorithm A with d_0^k in Algorithm B playing the role of d^k in Algorithm A. The following two propositions show that Algorithm B is well defined and either stops at a KKT point of (5.1) or generates a sequence $\{x^k\}_{k=1}^{\infty}$.

PROPOSITION 5.1 (Proposition 3.1 of [18]). *If Algorithm B stops at Step* 1(i), *then x^k is a KKT point of* (5.1). *If x^k is not a KKT point of* (5.1), *d_0^k satisfies*

$$\nabla f(x^k)^T d_0^k < 0 \tag{5.7}$$

and

$$\nabla g_j(x^k)^T d_0^k \leq 0 \ \ \textit{for all} \ \ j \in I(x^k).$$

PROPOSITION 5.2 (Proposition 3.2 of [18]). *The line search yields a step $t_k = \tau_2^i$ for some finite $i = i(k)$.*

5.2. Global convergence of Algorithm B.

In order to prove the convergence properties of Algorithm B, we assume that

(H1) the Hessian estimates $\{H_k\}_{k=0}^{\infty}$ are bounded, i.e., there exists a scalar $C_1 > 0$ such that for all k, $\|H_k\| \leq C_1$;

(H2) the MFCQ holds at all non-KKT points in X.

As discussed in subsection 5.1, (H2) implies that (5.5) has a continuous solution for x. By Proposition 4.1 or Proposition 5.1, we may assume that Algorithm B generates an infinite sequence $\{x^k\}_{k=1}^{\infty}$ and $\{x^k\}_{k=1}^{\infty}$ has an accumulation point x^*. Furthermore, we assume that

(H3) the CPLD holds at x^*.

(H2) and (H3) together are slightly weaker than the condition that the MFCQ holds at all points in X.

THEOREM 5.3. *Assume that the hypotheses* (H1)–(H3) *hold. Then x^* is a KKT point of* (5.1).

Proof. We assume that there is $\mathcal{K}$ such that

$$\lim_{k \in \mathcal{K}} x^k = x^*.$$

By Theorem 4.2, we only need to prove that

$$\liminf_{k \in \mathcal{K}} \|d_0^k\| = 0.$$

Assume that this does not hold. Then there exists a subsequence $\mathcal{K}' \subset \mathcal{K}$ and a scalar $c > 0$ such that for all $k \in \mathcal{K}'$, $\|d_0^k\| \geq c$. Suppose, by contradiction, that x^* is not a KKT point of (5.1). Then from the definitions of ρ_k and ρ, there exists a number $c_0 > 0$ such that for all $k \in \mathcal{K}'$, $\rho_k \geq c_0$. Therefore, using (5.3), (5.4), (5.7), and the definition of d^k in Step 1(ii) of Algorithm B, we have

$$\nabla f(x^k)^T d^k \leq c_0 \nabla f(x^k)^T d_1^k. \tag{5.8}$$

Similarly, for $j \in I$, we have

$$\nabla g_j(x^k)^T d^k \leq -g_j(x^k) + c_0 \nabla g_j(x^k)^T d_1^k. \tag{5.9}$$

Since x^* is not a KKT point, we may assume that

$$\lim_{k \in \mathcal{K}'} d_1^k = d_1^*, \tag{5.10}$$

$$\nabla f(x^*)^T d_1^* \leq -3c_1, \tag{5.11}$$

and

$$\nabla g_j(x^*)^T d_1^* \leq -3c_1 \quad \text{for} \quad j \in I(x^*) \tag{5.12}$$

for some $c_1 > 0$. (5.10) and (5.11) imply that, for $k \in \mathcal{K}'$ large enough,

$$\nabla f(x^k)^T d_1^k \leq -2c_1. \tag{5.13}$$

Similarly, from (5.10) and (5.12), we have, for $k \in \mathcal{K}'$ large enough, that

$$\nabla g_j(x^k)^T d_1^k \leq -2c_1 \quad \text{for} \quad j \in I(x^*). \tag{5.14}$$

Therefore, by viewing (5.8) and (5.13), (5.9) and (5.14), we have $c_2 > 0$ such that, for all $k \in \mathcal{K}'$ large enough,

$$\nabla f(x^k)^T d^k < -c_2,$$

$$\nabla g_j(x^k)^T d^k < -c_2 \quad \text{for} \quad j \in I(x^*),$$

and, by continuity of g,

$$g_j(x^k) \leq -c_2 \quad \text{for} \quad j \in I \setminus I(x^*).$$

From the definitions of ρ and d^k, we see that $\{d^k\}_{k=1}^{\infty}$ is bounded. From (5.6), $\{\tilde{d}^k\}_{k=1}^{\infty}$ is also bounded. The argument used in the proof of Proposition 3.2 of [17] implies that, in this case, the step performed by the line search is bounded away from zero. This and the monotonic decrease of $f(x^k)$ imply therefore that $\{f(x^k)\}_{k \in \mathcal{K}'}$ is unbounded, which contradicts the facts that $x^k \to x^*$ as $k \in \mathcal{K}'$ and $k \to \infty$ and the continuity of f. Hence the proof of this theorem is complete. □

In addition to (H1)–(H3), we further assume that

(H4) f and g are twice continuously differentiable;

(H5) there exists a scalar $C_2 > 0$ such that, for all k, the Hessian estimates satisfy

$$d^T H_k d \geq C_2 \|d\|^2 \quad \text{for any} \quad d \in \Re^n; \tag{5.15}$$

(H6) x^* satisfies the SSOSC.

PROPOSITION 5.4. *Assume that* (H1)–(H6) *hold and* $\{x^k\}_{k=1}^{\infty}$ *is generated by Algorithm B. Then*

$$\lim_{k\to\infty} x^k = x^*. \tag{5.16}$$

Proof. The argument used in the proof of Proposition 3.4 in [18] showed that

$$\lim_{k\to\infty} \|x^{k+1} - x^k\| = 0,$$

which combined with Theorem 3.3 yields that (5.16) holds. □

PROPOSITION 5.5. *Assume that* (H1)–(H6) *hold. Then*

$$\lim_{k\to\infty} d_0^k = 0.$$

Proof. By Proposition 4.3, $\{x_k\}_{k=1}^{\infty}$ is bounded. From

$$-\|\nabla f(x^k)\|\|d_0^k\| \le \nabla f(x^k)^T d_0^k \le -\frac{1}{2}(d_0^k)^T H_k d_0^k \le -\frac{1}{2}C_2\|d_0^k\|^2$$

we have

$$\|d_0^k\| \le \frac{2}{C_2}\|\nabla f(x^k)\|^2,$$

which implies that $\{d_0^k\}_{k=1}^{\infty}$ is bounded.

Suppose, by contradiction, that there exists a subsequence $\{d_0^k\}_{k\in\mathcal{K}}$ such that

$$\lim_{k\in\mathcal{K}} d_0^k = d_0^* \ne 0. \tag{5.17}$$

Since x^* is a KKT point of (1.1), there is a $u^* \ge 0$ such that

$$\begin{cases} \nabla f(x^*) + \sum_{j\in I(x^*)} u_j^* \nabla g_j(x^*) = 0, \\ u_j^* g_j(x^*) = 0 \ \text{ for } \ j \in I(x^*), \end{cases}$$

which combined with the facts that $\nabla f(x^*)^T d_0^* \le 0$, $\nabla g_j(x^*)^T d_0^* \le 0$ for $j \in I(x^*)$ and $u^* \ge 0$ implies that

$$\nabla f(x^*)^T d_0^* = 0.$$

On the other hand, from $\nabla f(x^k)^T d_0^k \le -\frac{1}{2}C_2\|d_0^k\|^2$, $C_2 > 0$, Proposition 4.3, and $\lim_{k\in\mathcal{K}} d_0^* = d_0^*$, we have

$$0 = \nabla f(x^*)^T d_0^* \le -\frac{1}{2}C_2\|d_0^*\|^2,$$

which contradicts (5.17). This completes the proof. □

6. Superlinear convergence of a modified Panier–Tits method. We begin by modifying Algorithm B in subsection 6.1 to enable us to prove its superlinear convergence. Then we establish the superlinear convergence of the modified algorithm in subsection 6.2.

6.1. A modified Panier–Tits method. In Algorithm B, let u^k be a regular Lagrange multiplier of d_0^k with respect to (QP_1). Let $\hat{I}_k$ be the active constraint set of (QP_1). Then there is a subset $I_k \subset \hat{I}_k$ such that $u_j^k = 0$ if $j \notin I_k$ and $\{\nabla g_j(x^k) \mid j \in I_k\}$ is a maximum linearly independent subset of $\{\nabla g_j(x^k) \mid j \in \hat{I}_k\}$. We now replace (QP_2) in Algorithm B by

$$(QP_3) \begin{cases} \min \ \frac{1}{2}(d^k + d)^T H_k (d + d^k) + \nabla f(x^k)^T (d + d^k) \\ \\ \text{s.t. } \ g_j(x^k + d^k) + \nabla g_j(x^k)^T d = -\|d^k\|^{\tau_3}, \quad j \in I_k, \\ \qquad g_j(x^k + d^k) + \nabla g_j(x^k)^T d \le -\|d^k\|^{\tau_3}, \quad j \in I \setminus I_k. \end{cases}$$

We call the resulting algorithm Algorithm C. This modification forces I_k to be a part of the active constraints of (QP_3), which is necessary for the proof of superlinear convergence without assuming the LICQ and the strict complementarity slackness. Checking the proofs of subsection 5.2, we see that this modification does not affect the global convergence of the algorithm since only (5.6) is required for $\tilde{d}^k$ in the global convergence analysis in subsection 5.2. We did not make this modification in section 5 since there it was not needed.

Let R_k be the $n \times |I_k|$ matrix whose columns consist of $\nabla g_j(x^k)$ for $j \in I_k$. Note that $R_k^T R_k$ is invertible in view of the definition of the regular Lagrange multiplier. Let

$$P_k = I - R_k (R_k^T R_k)^{-1} R_k^T$$

and

$$\nabla_{xx}^2 L(x^k, u^k) = \nabla^2 f(x^k) + \sum_{j \in I_k} u_j^k \nabla^2 g_j(x^k).$$

6.2. Superlinear convergence of Algorithm C. In the following analysis, we assume that $\{x^k\}_{k=1}^\infty$ converges to a point x^*. It follows from the preceding discussion that x^* is a KKT point of (5.1). In addition to (H1)–(H6), we assume that the following hypotheses hold:

(H7) x^* satisfies the CRCQ;

(H8) whenever $B \subset I(x^*)$ and vectors in $\{\nabla g_j(x^*) | j \in B\}$ are linearly independent, $(\{\nabla g_j(x^*) | j \in I(x^*) \setminus B\}, \{\nabla g_j(x^*) | j \in B\})$ is positive-linearly independent;

(H9)

$$\frac{\|P_k(H_k - \nabla_{xx}^2 L(x^k, u^k)) P_k d^k\|}{\|d^k\|} \to 0 \text{ as } k \to \infty.$$

Note that the LICQ implies both (H7) and (H8). Thus, even (H7) and (H8) together are slightly weaker than the LICQ at x^*.

PROPOSITION 6.1. *Assume that* (H1)–(H9) *hold and that* $\{x^k\}_{k=1}^\infty$ *is generated by Algorithm C. Then for* k *large enough, the step size* t_k *is one.*

Proof. Since I_k are finite sets for all k, we may partition $\mathcal{N} \equiv \{1, 2, \ldots\}$ into $l + 1$ disjoint subsets $\mathcal{K}_i$ for $i = 0, 1, \ldots, l$ such that $\mathcal{K}_0$ is finite, while other $\mathcal{K}_i$ are infinite and $I_k \equiv \bar{I}_i$ if $k \in \mathcal{K}_i$ and $i > 0$. For $i = 1, \ldots, l$, let $\bar{R}_i$ be the $n \times |\bar{I}_i|$ matrix whose columns consist of $\nabla g_j(x^*)$ for $j \in \bar{I}_i$. Note that by (H7), $\bar{R}_i^T \bar{R}_i$ is also invertible. By the equality part of the KKT conditions for (QP_1),

$$u_{I_k}^k = -(R_k^T R_k)^{-1} R_k^T (H_k d_0^k + \nabla f(x^k)).$$

Then, as $k \to \infty$ for $k \in \mathcal{K}_i$,

$$u^k_{I_k} \to u^*_{\bar{I}_i} = -(\bar{R}_i^T \bar{R}_i)^{-1} \bar{R}_i^T \nabla f(x^*).$$

Let $u_i^* = 0$ if $i \notin \bar{I}_i$. Then $u^* \in M(x^*)$. We see that 0 is a KKT point of

$$(\bar{QP}_i) \begin{cases} \min \ \frac{1}{2} d^T d + \nabla f(x^*)^T d \\ \text{s.t.} \ g_j(x^*) + \nabla g_j(x^*)^T d = 0, \quad j \in \bar{I}_i, \\ \quad\quad g_j(x^*) + \nabla g_j(x^*)^T d \le 0, \quad j \in I \setminus \bar{I}_i, \end{cases}$$

with a Lagrange multiplier u^*. Because of (H7), vectors in $\{\nabla g_j(x^*) | j \in \bar{I}_i\}$ are linearly independent. Then (H8) implies that the MFCQ holds at x^* for $(\bar{QP}_i)$. It is easy to see that the SSOSC holds at x^* for $(\bar{QP}_i)$ too. Applying the Kojima theorem (Theorem 3.5), we see that

$$(QP_4) \begin{cases} \min \ \frac{1}{2} d^T d + \nabla f(x^k)^T d \\ \text{s.t.} \ g_j(x^k + d^k) + \nabla g_j(x^k)^T d = -\|d^k\|^{\tau_3}, \quad j \in I_k, \\ \quad\quad g_j(x^k + d^k) + \nabla g_j(x^k)^T d \le -\|d^k\|^{\tau_3}, \quad j \in I \setminus I_k, \end{cases}$$

is feasible for k large enough, since (QP_4) is a perturbed form of $(\bar{QP}_i)$. Since (QP_3) has the same constraints as (QP_4), (QP_3) is also feasible for k large enough. Hence, and because of (H5) and (H7), and because of the fact that if the CRCQ holds at a point then it holds at a neighborhood of that point, (QP_3) has a KKT point $\tilde{d}^k$ for k large enough. Now, we may follow the proof of Proposition 3.6 of [18] step by step with minor modification for each i satisfying $1 \le i \le l$. Note that l is finite. The conclusion follows. □

Finally, two-step superlinear convergence follows. As in [18], the proof is not given as it follows step by step, with minor modifications, that of Lemma 3 to Theorem 1 in [23]. Note that with (H9), (H8), and (H7), we do not need to invoke Lemmas 1 and 2 in [23], which rely on the LICQ and the strict complementarity slackness.

THEOREM 6.2. *Under the stated assumptions, the convergence is two-step superlinear, i.e.,*

$$\lim_{k \to \infty} \frac{\|x^{k+2} - x^*\|}{\|x^k - x^*\|} = 0.$$

Remark. Similarly, as we mentioned in section 4, the conditions of Powell's theorem on the SQP method, Theorem 1 of [23], may be reduced to the SSOSC and the CPLD, by replacing (3.10) in [23] with (H9).

Again, the result in this section can also be extended to the 1987 Panier–Tits algorithm.

Acknowledgments. The authors are grateful to Professors Olvi Mangasarian, Jorge Nocedal, Jong-Shi Pang, Elijah Polak, and Stephen Robinson and two referees for their helpful comments.

REFERENCES

[1] J. F. BONNANS, *Local study of Newton type algorithms for constrained problems*, in Optimization—Fifth French-German Conference, S. Dolecki, ed., Springer-Verlag, Berlin, 1989, pp. 13–24.

[2] X. Chen, *Convergence of the BFGS methods for LC^1 convex constrained optimization*, SIAM J. Control Optim., 34 (1996), pp. 2051–2063.

[3] X. Chen, L. Qi, and R. Womersley, *Newton's method for quadratic stochastic programs with recourse*, J. Comput. Appl. Math., 60 (1995), pp. 29–46.

[4] F. Facchinei, *Minimization of SC^1 functions and the Maratos effect*, Oper. Res. Lett., 17 (1995), pp. 131–137.

[5] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, 1968.

[6] A. Fischer, *Modified Wilson Method for Nonlinear Programs with Nonunique Multipliers*, preprint, MATH-NM-04-1997, Institute for Numerical Mathematics, Technical University of Dresden, Dresden, Germany, February 1997.

[7] U. C. Garcia Palomares and O. L. Mangasarian, *Superlinearly convergent quasi-Newton algorithms for nonlinearly constrained optimization problems*, Math. Programming, 11 (1976), pp. 1–13.

[8] M. S. Gowda and J. S. Pang, *Stability analysis of variational inequalities and nonlinear complementarity problems, via the mixed linear complementarity problems and degree theory*, Math. Oper. Res., 19 (1994), pp. 831–879.

[9] J. Han and D. Sun, *Superlinear convergence of approximate Newton methods for LC^1 optimization problems without strict complementarity*, in Recent Advances in Nonsmooth Optimization, D. Du, L. Qi, and R. S. Womersley, eds., World Scientific, River Edge, NJ, 1995, pp. 141–158.

[10] S. P. Han, *Superlinearly convergent variable metric algorithms for general nonlinear programming problems*, Math. Programming, 11 (1976), pp. 263–282.

[11] R. Janin, *Direction derivative of the marginal function in nonlinear programming*, Math. Programming Study, 21 (1984), pp. 127–138.

[12] H. Jiang and L. Qi, *Globally and superlinearly convergent trust region algorithm for convex SC^1 minimization problems and its application to stochastic programs*, J. Optim. Theory Appl., 90 (1996), pp. 653–673.

[13] H. Jiang, L. Qi, X. Chen, and D. Sun, *Semismoothness and superlinear convergence in nonsmooth optimization and nonsmooth equations*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum, New York, 1996, pp. 197–212.

[14] M. Kojima, *Strongly stable stationary solutions in nonlinear programming*, in Analysis and Computation of Fixed Points, S. M. Robinson, ed., Academic Press, New York, 1980, pp. 93–138.

[15] Z. Q. Luo, J. S. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, New York, 1996.

[16] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[17] E. R. Panier and A. L. Tits, *A superlinearly convergent feasible method for the solution of inequality constrained optimization problems*, SIAM J. Control Optim., 25 (1987), pp. 934–950.

[18] E. R. Panier and A. L. Tits, *On combining feasibility, descent and superlinear convergence in inequality constrained optimization*, Math. Programming, 59 (1993), pp. 261–276.

[19] J. S. Pang and L. Qi, *A globally convergent Newton method for convex SC^1 minimization problems*, J. Optim. Theory Appl., 85 (1995), pp. 633–648.

[20] J. S. Pang and D. Ralph, *Piecewise smoothness, local invertibility, and parametric analysis of normal maps*, Math. Oper. Res., 21 (1996), pp. 401–426.

[21] E. Polak, *Optimization: Algorithms and Consistent Approximations*, Springer-Verlag, New York, 1997.

[22] E. Polak and L. Qi, *A globally and superlinearly convergent scheme for minimizing a normal merit function*, SIAM J. Control Optim., 36 (1998), pp. 1005–1019.

[23] M. J. D. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, in Nonlinear Programming 3, O. L. Mangasarian, R. R. Meyer, and S. M. Robinson, eds., Academic Press, New York, 1978, pp. 27–63.

[24] L. Qi, *Superlinearly convergent approximate Newton methods for LC^1 optimization problems*, Math. Programming, 64 (1994), pp. 277–294.

[25] L. Qi, *LC^1 functions and LC^1 optimization*, in Operations Research and Its Application, D. Z. Du, X. S. Zhang, and K. Chen, eds., World Publishing, Beijing, 1996, pp. 4–13.

[26] L. Qi and H. Jiang, *Semismooth Karush-Kuhn-Tucker equations and convergence analysis of Newton methods and quasi-Newton methods for solving these equations*, Math. Oper. Res., 22 (1997), pp. 301–325.

[27] L. Qi and R. Womersley, *An SQP Algorithm for extended linear-quadratic problems in*

stochastic programming, Ann. Oper. Res., 56 (1995), pp. 251–285.

[28] D. Ralph and S. J. Wright, *Superlinear convergence of an interior-point method for monotone variational inequalities*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J. S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 345–385.

[29] S. M. Robinson, *First order conditions for general nonlinear optimization*, SIAM J. Appl. Math., 30 (1976), pp. 597–607.

[30] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

[31] S. M. Robinson, *Generalized equations and their solutions, part* II*: Applications to nonlinear programming*, Math. Programming Study, 19 (1982), pp. 200–221.

[32] S. M. Robinson and R. R. Meyer, *Lower semicontinuity of multivalued linearization mappings*, SIAM J. Control, 11 (1973), pp. 525–533.

[33] J. Stoer and C. Witzgall, *Convexity and Optimization in Finite Dimensions.* I, Springer-Verlag, New York, 1970.

[34] D. Sun, M. Fukushima and L. Qi, *A computable generalized Hessian of the D-gap function and Newton-type methods for variational inequality problems*, in Complementarity and Variational Problems: State of the Art, M. C. Ferris and J. S. Pang, eds., SIAM, Philadelphia, PA, 1997, pp. 452–473.

[35] S. Wright, *Superlinear convergence of a stabilized SQP method to a degenerate solution*, Comput. Optim. Appl., 11 (1998), pp. 253–275.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 982–998

AN OPTIMIZATION PROBLEM FOR PREDICTING THE MAXIMAL EFFECT OF DEGRADATION OF MECHANICAL STRUCTURES*

W. ACHTZIGER†, M. P. BENDSØE‡, AND J. E. TAYLOR§

Abstract. This paper deals with a nonlinear nonconvex optimization problem that models prediction of degradation in discrete or discretized mechanical structures. The mathematical difficulty lies in equality constraints of the form $\sum_{i=1}^{m} \frac{1}{y_i} A_i x = b$, where A_i are symmetric and positive semidefinite matrices, b is a vector, and x, y are the vectors of unknowns. The linear objective function to be maximized is $(x, y) \mapsto b^T x$.

In a first step we investigate the problem properties such as existence of solutions and the differentiability of related marginal functions. As a by-product, this gives insight in terms of a mechanical interpretation of the optimization problem. We derive an equivalent convex problem formulation and a convex dual problem, and for dyadic matrices A_i a quadratic programming problem formulation is developed. A nontrivial numerical example is included, based on the latter formulation.

Key words. nonlinear optimization, structural optimization, variational methods

AMS subject classifications. 49A55, 65K10, 73C60, 73K40

PII. S1052623497328768

1. Introduction and problem formulation.

The topic of this paper is the investigation of mathematical properties of a recently proposed model for the evaluation of the maximal effect of degradation in mechanical structures (cf. [2]). The model takes the form of an optimization problem over two sets of variables, one being the state variables (displacements or forces) for the structure, the other being so-called inner state variables characterizing the degradation of the structure.

The model is based on the use of structural compliance (flexibility) as a global measure of the effect of degradation and on an interpretation of local degradation as a loss of stiffness of elements or material in the structure, with the maximal degradation effect characterized by the distribution of degradation giving the upper bound on this global measure. For evolution of degradation, sequential solutions to this problem predict patterns of evolving local degradation and local deformation corresponding to the respective stage in a process (a time-stepping approach). We note that the model does not reflect explicit considerations (i.e., the cause for degradation) that arise in studies of damage mechanics, but there are many analogies to models used in the field of continuum damage mechanics; the reader is referred to the recent monographs [7, 8] for surveys on damage mechanics. Also, for a discussion on certain mathematical problems arising in continuum damage mechanics we refer to [9] and the literature cited therein.

In the following we introduce the main optimization problem considered in this paper. In order to enable readers familiar with mechanics to make their own interpre-

*Received by the editors October 15, 1997; accepted for publication (in revised form) July 6, 1999; published electronically June 6, 2000. This work was supported in part by the Danish Technical Research Council through the Programme of Research on Computer Aided Design (MPB, JET) and by Danish Natural Sciences Research Council grant 77-990011 (MPB, WA).

http://www.siam.org/journals/siopt/10-4/32876.html

†University of Erlangen-Nuremberg, Institute of Applied Mathematics, D-91058 Erlangen, Germany (achtzig@am.uni-erlangen.de).

‡Technical University of Denmark, Department of Mathematics, DK-2800 Lyngby, Denmark (m.p.bendsoe@mat.dtu.dk).

§University of Michigan, Department of Aerospace Engineering, Ann Arbor, MI 48109-2118 (janos@umich.edu).

tations, we use common notation from this field. However, the general mathematical structure of the problem is illustrated in Remark 1.2.

We consider the following optimization problem:

$$\text{(P1)}\qquad \max_{\beta\in\mathbb{R}^m,\ u\in\mathbb{R}^n} \tfrac{1}{2}f^T u$$

$$\text{subject to (s.t.)}\quad \sum_{i=1}^{m}\left[(1-\beta_i)\tfrac{1}{E_i}+\beta_i\tfrac{1}{E_i^D}\right]^{-1} a_i\ell_i K_i u = f,$$

$$0\le\beta_i\le 1\quad \text{for all } i=1,\dots,m,$$

$$\sum_{i=1}^{m} a_i\ell_i\beta_i\le D.$$

In this problem, $f\in\mathbb{R}^n$ is a given vector. To exclude trivial cases, we assume $f\neq 0$. The matrices $K_1,\dots,K_m\in\mathbb{R}^{n\times n}$ are given with

$$\text{(A1)}\qquad K_i \text{ positive semidefinite for all } i=1,\dots,m.$$

The given constants a_i,ℓ_i,E_i,E_i^D, $i=1,\dots,m$, are positive, with

$$\text{(A2)}\qquad 0<E_i^D<E_i\quad\text{for all } i=1,\dots,m.$$

Throughout the paper we put $V:=\sum_{i=1}^m a_i\ell_i$, and for the given constant D we assume

$$\text{(A3)}\qquad 0<D\le V.$$

Finally, we assume that

$$\text{(A4)}\qquad f\in\text{Range}\left(\sum_{i=1}^m K_i\right).$$

Together with (A1) this assumption guarantees that for any fixed $\beta\in[0,1]^m$ the equality constraints can be solved for u. Moreover, (A2) to (A4) guarantee that the set of feasible points of (P1) is nonempty.

Application 1.1. Problem (P1) models one time-step of the computation of upper bounds on the progressive degradation of elastic structures. This is outlined in the following. We mainly concentrate on trusses for simplicity.

(i) *Truss structures.* Trusses are pin-jointed frameworks consisting of long slender bar elements. Denote the number of bars by m, and let the material of the ith bar be linearly elastic with Young's modulus E_i. Similarly, denote by ℓ_i the length and by a_i the cross-sectional area of this bar. The matrix K_i contains some geometrical properties of the ith bar (see also below). Then $u\in\mathbb{R}^n$ is the vector of nodal displacements under the load f that is applied at the nodal points (i.e., points where bars are connected). For a more precise description see, e.g., [3, 6]. Let E_i^D be a Young's modulus smaller than E_i (cf. (A2)), i.e., characterizing weaker material. Then the variable β_i determines the effective stiffness of the material in the ith bar, controlling the "degree of degradation": For fixed β_i, the effective Young's modulus is given by $E_i^{\text{eff}}=[(1-\beta_i)\frac{1}{E_i}+\beta_i\frac{1}{E_i^D}]^{-1}$. If $\beta_i=0$, then we get the effective material constant $E_i^{\text{eff}}=E_i$ (i.e., the original material), and for $\beta_i=1$ we obtain $E_i^{\text{eff}}=E_i^D$, i.e., weak material. For $0<\beta_i<1$ we get an intermediate stiffness expressed as what is called the Reuss lower bound on stiffness of a mixture of materials, corresponding to springs in series; see, e.g., [4].

The total amount of material degradation in the structure is controlled by the inequality constraint $\sum a_i \ell_i \beta_i \leq D$; that is, the total volume of degraded material is limited by D, where D is some part of the total volume V of the structure (cf. (A3)).

The objective function measures the total displacement of the structure along the force vector f: if $f^T u$ is small, then the structure is "stiff"; i.e., its nodal points hardly move along the applied forces. The term $f^T u$ is called *compliance.*

In this way, formulation (P1) is a model for predicting local material degradation in a structure in a worst case sense. That means, given some global limitation of degradation (by D), that we look for that degradation contribution in the truss bars that causes the total structure to be as weak as possible.

Of course, degradation is a time-dependent process and (P1) models the (maximally possible) degradation process of a particular time point only. Thus for progressive degradation, i.e., for computing the evolution of degradation in the structure, we have to solve a sequence of problems of type (P1). This is the topic of section 5.

(ii) *Continuum elastic structures.* For continuum structures, (P1) constitutes the problem form obtained after discretization with finite elements. □

Remark 1.2. Substituting $y_i := (1-\beta_i)\frac{1}{E_i} + \beta_i \frac{1}{E_i^D}$ for all $i = 1, \dots, m$, problem (P1) becomes

$$\begin{aligned} &\max_{y \in \mathbb{R}^m,\ u \in \mathbb{R}^n} \tfrac{1}{2} f^T u \\ \text{s.t.}\quad &\sum_{i=1}^m \tfrac{1}{y_i} \tilde{K}_i u = f, \\ &\tfrac{1}{E_i} \leq y_i \leq \tfrac{1}{E_i^D} \quad \text{for all } i = 1, \dots, m, \\ &\sum_{i=1}^m c_i y_i \leq \tilde{D} \end{aligned}$$

with suitable values for $c_i, \tilde{K}_i$, and $\tilde{D}$. Thus formulation (P1) is a special form of a problem with equality constraints of the type $\sum \frac{1}{y_i} \tilde{K}_i u = f$. □

Due to the interpretation of problem (P1) in Application 1.1, we call β the *degradation distribution* or *degradation field.* The constraint $\sum a_i \ell_i \beta_i \leq D$ is called "degradation volume constraint."

2. Problem investigation and analysis of degradation behavior. In this section we provide several useful problem properties that are directly derived from the particular given problem structure of (P1). Moreover, this will lead to an interpretation of the mechanical degradation behavior modeled by problem (P1).

We define the following functions and notations:

$$\begin{aligned} \mathcal{B}_D \quad &:= \left\{ \beta \in \mathbb{R}^m \,\middle|\, 0 \leq \beta_i \leq 1 \text{ for all } i,\ \sum_{i=1}^m a_i \ell_i \beta_i \leq D \right\}, \\ \tilde{\psi}(\beta, u) &:= \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^m \left[(1-\beta_i)\tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D} \right]^{-1} a_i \ell_i u^T K_i u \right\}, \\ \psi(\beta) \quad &:= \max_{u \in \mathbb{R}^n} \left\{ \tfrac{1}{2} f^T u \,\middle|\, \sum_{i=1}^m \left[(1-\beta_i)\tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D} \right]^{-1} a_i \ell_i K_i u = f \right\}. \end{aligned}$$

Proposition 2.1.

(i) *For all* $\beta \geq 0$ *we have the identity* $\psi(\beta) = \max_{u \in \mathbb{R}^n} \tilde{\psi}(\beta, u)$.

(ii) *The function* $\tilde{\psi}$ *is concave on* $[0,1]^m \times \mathbb{R}^n$ *and differentiable.*

(iii) *The function* ψ *is finite and concave on* $[0,1]^m$.

(iv) *Problem* (P1) *always possesses a solution.*

Proof. By (A2) the set

$$\mathcal{D} := \left\{ \delta \in \mathbb{R}^m \mid \tfrac{E_i^D}{E_i^D - E_i} < \delta_i \text{ for all } i \right\}$$

is well defined. It's easy to see that $\mathcal{B}_D \subset \mathcal{D}$ and that

$$(1 - \delta_i)\frac{1}{E_i} + \delta_i \frac{1}{E_i^D} > 0 \quad \text{for all } \delta \in \mathcal{D}. \tag{2.1}$$

To avoid technical difficulties, we prove assertions (i) to (iii) for the open set $\mathcal{D}$ instead of $[0,1]^m$ (note $[0,1]^m \subset \mathcal{D}$; see above).

Let $\delta \in \mathcal{D}$ be arbitrary. By assumption (A1) and by (2.1), a vector u^* maximizes the function $\tilde{\psi}(\delta, \,.\,)$ if and only if $\sum [(1 - \delta_i)\frac{1}{E_i} + \delta_i \frac{1}{E_i^D}]^{-1} a_i \ell_i K_i u = f$ (by (A4) this is solvable). By this, explicit calculation of the optimal function value yields $\frac{1}{2} f^T u^*$, and (i) is proved.

The proof of (ii) is a straightforward calculation: The Hessian of $\tilde{\psi}$ is negative semidefinite. Differentiability is obvious.

By (i), ψ is the pointwise supremum function of the concave functions $\tilde{\psi}(\,.\,, u)$, and by (ii) $\tilde{\psi}$ is concave as a function in both variables (δ, u). Moreover, for each fixed δ the supremum in (i) is attained by (A4). Therefore, ψ is concave as well.

By (i) we conclude for all $\delta \in \mathcal{D}$ that $\infty > \psi(\delta)$. Since $\mathcal{D}$ is open, the concavity of ψ (cf. (ii)) yields that ψ is finite and continuous on $\mathcal{D}$. Together with $\mathcal{B}_D \subset \mathcal{D}$ this shows that the supremum of ψ on $\mathcal{B}_D$ is attained since $\mathcal{B}_D$ is a compact set. By (i) this means that there exists a solution of (P1). □

By Proposition 2.1(i), problem (P1) may be written in the variable β as

$$\max_{\beta \in \mathcal{B}_D} \psi(\beta)\ . \tag{P1$'$}$$

This is a convex, differentiable, and linearly constrained problem. (If $\operatorname{rank}(\sum K_i) = n$, then the differentiability of ψ easily follows from an implicit-function theorem. The case $\operatorname{rank}(\sum K_i) < n$ can be deduced from the first case by using the standard projection onto the nullspace of $\sum K_i$.) Alternatively, Proposition 2.1(i) and (ii) show that (P1) may also be rewritten in the form

$$\max_{\beta \in \mathcal{B}_D,\, u \in \mathbb{R}^n} \tilde{\psi}(\beta, u), \tag{P1$''$}$$

which again is convex, smooth, and nonlinear with only linear constraints.

Remark 2.2. It is perhaps surprising that problem (P1) can be rewritten as the convex problem (P1$'$) (resp., (P1$''$)). This is due to the special structure of the problem. We note that—though convex—formulations (P1$'$) and (P1$''$) may not be suitable for numerical computations in practical applications: the number m of elements will be of the order several thousands (n is of the same order), and this will cause standard solvers to break down. More tractable formulations are studied below. □

Application 2.3. In engineering, the formulation (P1$''$) is referred to as a formulation in *potential energy* since the term $-\tilde{\psi}(\beta, u)$ is the potential energy of the structure under the displacements u.

Supplementary to Proposition 2.1(iv) it can be proved (analogously to a proof in [1]) that there is always a solution β^* with $\sum a_i \ell_i \beta_i^* = D$; i.e., there exists an optimal degradation distribution making use of the maximally permitted amount of degraded material. This is sensible from a modeling point of view. □

The following theorem reformulates (P1) (resp., (P1$'$), (P1$''$)) as a partly dual problem. This derivation gives insight into the degradation behavior of particular parts of the structure. For this, it is necessary to isolate an energy term for each so-called "member" of the structure. The energy contained in each structural member is called *member strain energy.*

Theorem 2.4 (formulation in member strain energies). *For the optimal function value of* (P1) *we have*

$$\text{(P1SE)} \qquad \max_{\beta \in \mathcal{B}_D} \psi(\beta) = \max_{u \in \mathbb{R}^n} \min_{\Lambda \geq 0} \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^{m} a_i \ell_i \Psi_i(u, \Lambda) + \Lambda D \right\},$$

where for all $i = 1, \ldots, m$

$$\Psi_i(u, \Lambda) := \begin{cases} E_i u^T K_i u & \text{if } u^T K_i u \leq 2\Lambda \frac{E_i^D}{E_i(E_i - E_i^D)}, \\ E_i^D u^T K_i u + 2\Lambda & \text{if } u^T K_i u \geq 2\Lambda \frac{E_i}{E_i^D(E_i - E_i^D)}, \\ 2\sqrt{\dfrac{2\,\Lambda\, E_i E_i^D}{E_i - E_i^D}}\sqrt{u^T K_i u} - \dfrac{2\,\Lambda\, E_i^D}{E_i - E_i^D} & \text{otherwise.} \end{cases}$$

Proof. We start with the application of Proposition 2.1(i) and get

$$\begin{aligned} &\max_{\beta \in \mathcal{B}_D} \psi(\beta) \\ &= \max_{u \in \mathbb{R}^n} \max_{\beta \in [0,1]^m,\ \sum a_i \ell_i \beta_i \leq D} \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^{m} \left[(1 - \beta_i) \tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D} \right]^{-1} a_i \ell_i u^T K_i u \right\} \end{aligned}$$

(introduce the Lagrangian multiplier $\Lambda \geq 0$ for the degradation volume constraint, and apply Lagrangian relaxation)

$$\begin{aligned} = \max_{u \in \mathbb{R}^n} \max_{\beta \in [0,1]^m} \inf_{\Lambda \geq 0} \Big\{ & f^T u - \tfrac{1}{2} \sum_{i=1}^{m} \left[(1 - \beta_i) \tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D} \right]^{-1} a_i \ell_i u^T K_i u \\ & -\Lambda \sum_{i=1}^{m} a_i \ell_i \beta_i + \Lambda D \Big\} \end{aligned}$$

(since the inner term in brackets is concave in β by Proposition 2.1(ii), we may apply a well-known minmax theorem (see, e.g., [11, Cor. 37.3.2]) interchanging $\max_\beta$ with $\inf_\Lambda$)

$$\begin{aligned} = \max_{u \in \mathbb{R}^n} \inf_{\Lambda \geq 0} \Big\{ & f^T u - \tfrac{1}{2} \sum_{i=1}^{m} a_i \ell_i \min_{\beta_i \in \mathbb{R},\ \beta_i \in [0,1]} \{ [(1 - \beta_i) \tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D}]^{-1} u^T K_i u + 2\Lambda \beta_i \} \\ & + \Lambda D \Big\}. \end{aligned}$$

It can now be verified that for all fixed u, Λ, and i,

$$\min_{\beta_i \in [0,1]} \{ [(1 - \beta_i) \tfrac{1}{E_i} + \beta_i \tfrac{1}{E_i^D}]^{-1} u^T K_i u + 2\Lambda \beta_i \} = \Psi_i(u, \Lambda)$$

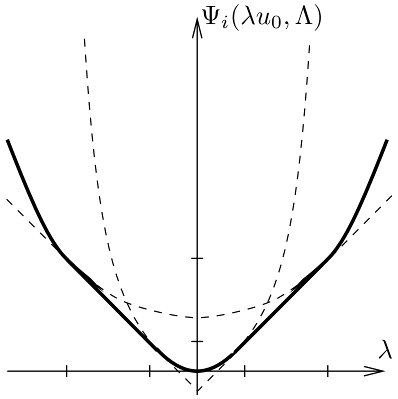


FIG. 2.1. *Specific member strain energy Ψ_i for proportionally increasing displacements $u \equiv \lambda u_0$.*

with Ψ_i from above. Finally, it is easy to see that the infimum over Λ is attained for each fixed u. □

Application 2.5. Theorem 2.4 allows the following interpretation of the model. The degraded structure has a specific potential energy ("specific" meaning per volume unit; see Figure 2.1) corresponding to a nonlinear material, which at low strains (measured by $u^T K_i u$) is healthy with Young's modulus E_i, while at high strains it is completely degraded with Young's modulus E_i^D (see also Application 3.4). □

3. Formulations in stresses. In this section, we assume (in addition to (A1)) that each matrix K_i is given as a symmetric dyadic product; i.e., we assume that there exist vectors $b_1, \ldots, b_m \in \mathbb{R}^n$ such that

$$\text{(A5)} \qquad K_i = \frac{1}{\ell_i^2} b_i b_i^T \quad \text{for all } i = 1, \ldots, m.$$

By collecting all vectors b_i as columns in a matrix, we get

$$B := (b_1 b_2 \cdots b_m) \in \mathbb{R}^{n \times m}.$$

Application 3.1. The dyadic structure in (A5) is valid for truss structures (see, e.g., [6]). Then the vector b_i contains the cosines of the angle between the ith bar and the axis of a local coordinate system (i.e., in two or three dimensions). The matrix B is called the *compatibility matrix* since it translates bar forces into nodal forces, and B^T translates nodal displacements into bar elongations. For other discretized structural models the stiffness matrix is the product $K_i = B_i B_i^T$ of a (low-rank) matrix B_i corresponding to a discrete divergence operator. In the following we concentrate, for simplicity, on the dyadic case for trusses. The derivations can also be performed for the more general case, as well as for their continuum equivalents (cf. [2]). □

Using the dyadic form above, we can reformulate problem (P1) (resp., (P1′) or (P1″)) to (see Theorem 3.3 below)

$$\text{(D1)} \qquad \min_{q \in \mathbb{R}^m,\ \Lambda \in \mathbb{R}} \left\{ \tfrac{1}{2} \sum_{i=1}^m \tfrac{\ell_i}{a_i E_i} q_i^2 + \tfrac{1}{2} \sum_{i=1}^m a_i \ell_i \max\{0, \tfrac{E_i - E_i^D}{E_i E_i^D} \tfrac{q_i^2}{a_i^2} - 2\Lambda\} + \Lambda D \right\}$$
$$\text{s.t.} \quad Bq = f,$$
$$\Lambda \geq 0.$$

This problem is a dual of (P1) (see below). The contained "partial duality" with respect to (w.r.t.) u is considered separately in the following proposition.

PROPOSITION 3.2. *Let $\bar{E}_1, \ldots, \bar{E}_m$ be positive real numbers. Then*

$$\max_{u \in \mathbb{R}^n} \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^m \bar{E}_i a_i \ell_i u^T K_i u \right\} = \min_{q \in \mathbb{R}^m} \left\{ \tfrac{1}{2} \sum_{i=1}^m \frac{\ell_i}{a_i \bar{E}_i} q_i^2 \mid Bq = f \right\}.$$

Proof. Both quadratic problems are dual to each other: The vector u of the max-problem simply plays the role of Lagrange multipliers for the linear constraints $Bq = f$ in the min-problem (note that by (A4) and (A5) the feasible set of the min-problem is nonempty). □

THEOREM 3.3 (dual problem: formulation in complementary energy). *Problems* (P1) *and* (D1) *are related in the following way:*

(i) *Problem* (D1) *possesses a solution.*

(ii) *The optimal function values coincide, i.e.,*

$$\max \text{(P1)} = \min \text{(D1)} .$$

(iii) *A couple $(q^*, \Lambda^*) \in \mathbb{R}^m \times \mathbb{R}$ is optimal for problem* (D1) *if and only if there exist $u^* \in \mathbb{R}^n$, $\rho^*, \delta^* \in \mathbb{R}^m$, and $\kappa^* \in \mathbb{R}$ such that with*

$$r_i^* = \max \left\{ 0, \frac{E_i - E_i^D}{2E_i E_i^D} \frac{q_i^{*2}}{a_i^2} - \Lambda^* \right\} \quad \textit{for all } i = 1, \ldots, m \tag{3.1}$$

the following conditions are satisfied:

$$\frac{\ell_i}{a_i E_i} q_i^* - b_i^T u^* + \delta_i^* \frac{E_i - E_i^D}{E_i E_i^D a_i^2} q_i^* = 0 \quad \textit{for all } i = 1, \ldots, m, \tag{3.2}$$

$$a_i \ell_i - \rho_i^* - \delta_i^* = 0 \quad \textit{for all } i = 1, \ldots, m, \tag{3.3}$$

$$D - \sum_{i=1}^m \delta_i^* - \kappa^* = 0, \tag{3.4}$$

$$r_i \rho_i^* = 0 \quad \textit{for all } i = 1, \ldots, m, \tag{3.5}$$

$$\delta_i^* \Big(\frac{E_i - E_i^D}{2E_i E_i^D a_i^2} q_i^{*2} - \Lambda^* \Big) = \delta_i^* r_i \quad \textit{for all } i = 1, \ldots, m, \tag{3.6}$$

$$\Lambda^* \kappa^* = 0, \tag{3.7}$$

$$\rho_i^* \geq 0 \quad \textit{for all } i = 1, \ldots, m, \tag{3.8}$$

$$\delta_i^* \geq 0 \quad \textit{for all } i = 1, \ldots, m, \tag{3.9}$$

$$\kappa^* \geq 0, \tag{3.10}$$

$$Bq^* = f, \tag{3.11}$$

$$\Lambda^* \geq 0 . \tag{3.12}$$

(iv) *Let (q^*, Λ^*) be optimal for* (D1), *and let $u^* \in \mathbb{R}^n$, $\rho^*, \delta^* \in \mathbb{R}^m$, $\kappa^* \in \mathbb{R}$ be corresponding multipliers as in* (iii). *Then (β^*, u^*) is optimal for problem* (P1), *where*

$$\beta_i^* := \frac{\delta_i^*}{a_i \ell_i} \quad \textit{for all } i = 1, \ldots, m. \tag{3.13}$$

In particular,

$$\beta_i^* = \begin{cases} 1 & \textit{if } \frac{q_i^{*2}}{a_i^2} > \frac{2E_i E_i^D}{E_i - E_i^D} \Lambda^*, \\ 0 & \textit{if } \frac{q_i^{*2}}{a_i^2} < \frac{2E_i E_i^D}{E_i - E_i^D} \Lambda^*. \end{cases} \tag{3.14}$$

Proof. By (A4) and (A5) there exists a feasible point $(\bar{q}, \bar{\Lambda})$ of (D1). For any $(q^T, \Lambda)^T \in \mathbb{R}^{m+1}$ we denote the objective function of problem (D1) by Γ,

$$\Gamma(q, \Lambda) := \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i E_i} q_i^2 + \tfrac{1}{2} \sum_{i=1}^{m} a_i \ell_i \max \left\{ 0, \frac{E_i - E_i^D}{E_i E_i^D} \frac{q_i^2}{a_i^2} - 2\Lambda \right\} + \Lambda D.$$

Then, clearly,

$$\Gamma(q, \Lambda) \geq \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i E_i} q_i^2 + \Lambda D \quad \text{for all } (q^T, \Lambda)^T \in \mathbb{R}^{m+1}.$$

Therefore, each sequence $(q^j, \Lambda^j)_{j \in \mathbb{N}}$ of feasible points with $\|(q^{jT}, \Lambda^j)^T\|_2 \longrightarrow +\infty$ leads to $\lim_{j \to \infty} \Gamma(q, \Lambda) = +\infty$. This shows that for any feasible point, e.g., for $(\bar{q}, \bar{\Lambda})$ from above, the set

$$\mathcal{Z}(\bar{q}, \bar{\Lambda}) := \left\{ (q^T, \Lambda)^T \in \mathbb{R}^{m+1} \mid Bq = f,\ \Lambda \geq 0,\ \Gamma(q, \Lambda) \leq \Gamma(\bar{q}, \bar{\Lambda}) \right\}$$

is bounded. Since Γ is continuous, we even get that $\mathcal{Z}(\bar{q}, \bar{\Lambda})$ is a compact set. Since we may consider problem (D1) as minimization on the set $\mathcal{Z}(\bar{q}, \bar{\Lambda})$, we have proved that the minimum is attained, i.e., (i).

For the proof of (ii) we take as a starting point Proposition 2.1(i) together with Proposition 3.2 for each fixed $\beta \in \mathcal{B}_D$:

$$\begin{aligned} &\max\,(\text{P1}) \\ &= \max_{\beta \in \mathcal{B}_D} \min_{q \in \mathbb{R}^m :\, Bq = f} \left\{ \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i} \Big[(1 - \beta_i) \frac{1}{E_i} + \beta_i \frac{1}{E_i^D} \Big] q_i^2 \right\} \end{aligned}$$

(analogously to the proof of Theorem 2.4: apply minmax theorem and introduce multiplier $\Lambda \geq 0$ for volume constraint)

$$= \inf_{q \in \mathbb{R}^m :\, Bq = f} \max_{\beta \in [0,1]^m} \inf_{\Lambda \geq 0} \left\{ \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i} \Big[(1 - \beta_i) \frac{1}{E_i} + \beta_i \frac{1}{E_i^D} \Big] q_i^2 + \Lambda \Big(D - \sum_{i=1}^{m} a_i \ell_i \beta_i \Big) \right\}$$

(apply minmax theorem for fixed q and use the separability of maximization over the β_i)

$$\begin{aligned} &= \inf_{q \in \mathbb{R}^m :\, Bq = f} \inf_{\Lambda \geq 0} \left\{ \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i E_i} q_i^2 + \tfrac{1}{2} \sum_{i=1}^{m} a_i \ell_i \max \left\{ 0, \frac{E_i - E_i^D}{E_i E_i^D} \frac{q_i^2}{a_i^2} - 2\Lambda \right\} + \Lambda D \right\} \\ &= \min\,(\text{D1}). \end{aligned}$$

This proves (ii) (note that $\inf_{q,\Lambda}$ is attained by (i)).

With the auxiliary variables $r_1, \ldots, r_m \in \mathbb{R}$, the problem (D1) can be equivalently rewritten in the form

$$\begin{aligned} (3.15) \qquad & \min_{q \in \mathbb{R}^m,\, \Lambda \in \mathbb{R},\, r \in \mathbb{R}^m} \left\{ \tfrac{1}{2} \sum_{i=1}^{m} \tfrac{\ell_i}{a_i E_i} q_i^2 + \sum_{i=1}^{m} a_i \ell_i r_i + \Lambda D \right\} \\ \text{s.t.} \quad & Bq = f, \\ & -r_i \leq 0 \qquad \text{for all } i = 1, \ldots, m, \\ & \tfrac{E_i - E_i^D}{2 E_i E_i^D} \tfrac{q_i^2}{a_i^2} - \Lambda - r_i \leq 0 \quad \text{for all } i = 1, \ldots, m, \\ & \Lambda \geq 0\,. \end{aligned}$$

By (A4) and (A5) there is a $\bar{q} \in \mathbb{R}^m$ with $B\bar{q} = f$. This shows that the feasible set of (3.15) is nonempty (choose r_i and Λ large enough). If (q^*, Λ^*, r^*) is optimal for problem (3.15), then it is clear that (3.1) is satisfied. Since the equality constraints in (3.15) are linear, it is easy to prove that a type of generalized Slater's constraint qualification is satisfied. Therefore, the Karush–Kuhn–Tucker (KKT) optimality conditions are satisfied in (q^*, Λ^*, r^*) with suitable multipliers $u^*, \rho^*, \delta^*, \kappa^*$. These conditions are listed in (3.2) to (3.12).

Vice versa, let (3.2) to (3.12) be satisfied. Since (3.15) is a convex problem (in the variables (q, Λ, r)), the point (q^*, Λ^*, r^*) is a global minimizer. Thus (3.1) must be satisfied. This proves (iii).

The proof of (iv) is a rather long but simple exercise only using conditions (3.2) to (3.12) and the definition of β^*. We only sketch the main steps of an exact proof: Feasibility of (β^*, u^*) for (P1) is derived from conditions (3.2) to (3.4) and (3.8) to (3.11). Thus—since (q^*, Λ^*) is optimal for (D1) by assumption, and by (i) and (ii)—it suffices to verify the identity

$$\tag{3.16} \tfrac{1}{2} f^T u^* = \tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i E_i} q_i^{*2} + \sum_{i=1}^{m} a_i \ell_i r_i^* + \Lambda^* D,$$

where r^* is defined by (3.1). We start by using (3.2) and get

$$\begin{aligned} &\tfrac{1}{2} \sum_{i=1}^{m} \frac{\ell_i}{a_i E_i} q_i^{*2} + \sum_{i=1}^{m} a_i \ell_i r_i^* + \Lambda^* D \\ &= \sum_{i=1}^{m} \left(\tfrac{1}{2} b_i^T u^* q_i^* - \tfrac{1}{2} \delta_i^* \frac{E_i - E_i^D}{a_i^2 E_i E_i^D} q_i^{*2} \right) + \sum_{i=1}^{m} a_i \ell_i r_i^* + \Lambda^* D \end{aligned}$$

(use (3.6), (3.11), (3.3), and (3.4); then apply (3.5) and (3.7))

$$\begin{aligned} &= \tfrac{1}{2} f^T u^* - \sum_{i=1}^{m} r_i^* (a_i \ell_i - \rho_i^*) - \Lambda^* (D - \kappa^*) + \sum_{i=1}^{m} a_i \ell_i r_i^* + \Lambda^* D \\ &= \tfrac{1}{2} f^T u^* \, . \end{aligned}$$

This shows (3.16), and thus (β^*, u^*) is optimal for (P1). In particular, if $r_i^* > 0$, then (3.5) shows $\rho_i^* = 0$, and by (3.3) we get $\delta_i^* = a_i \ell_i$, and thus $\beta_i^* = 1$. Analogously, if $r_i^* = 0 > (\frac{1}{2}(E_i - E_i^D) q_i^{*2} / (a_i^2 E_i E_i^D) - \Lambda^*)$, then (3.6) yields $\delta_i^* = 0$, i.e., $\beta_i^* = 0$. By the definition of r_i^*, this shows the rest of assertion (iv). □

This theorem shows that formulation (D1) is a "proper" dual of (P1) as the solution of (P1) (up to a scaling) is given directly by the multipliers of problem (D1).

Application 3.4. For truss structures the equality constraint $Bq = f$ in Theorem 3.3 expresses the *equilibrium of forces*, i.e., relating internal bar forces q with the external loads f (applied at the nodal points). Moreover, problem (D1) is a so-called complementary energy principle (cf. Proposition 3.2), here for the degraded truss. Thus Theorem 3.3(ii) expresses the same mechanical duality for the degraded truss as does Proposition 3.2 for a truss consisting of nondegradable, linearly elastic material.

The objective function of (D1) can be written as the sum

$$\sum_{i=1}^{m} a_i \ell_i \Xi_i(q, \Lambda) + \Lambda D,$$

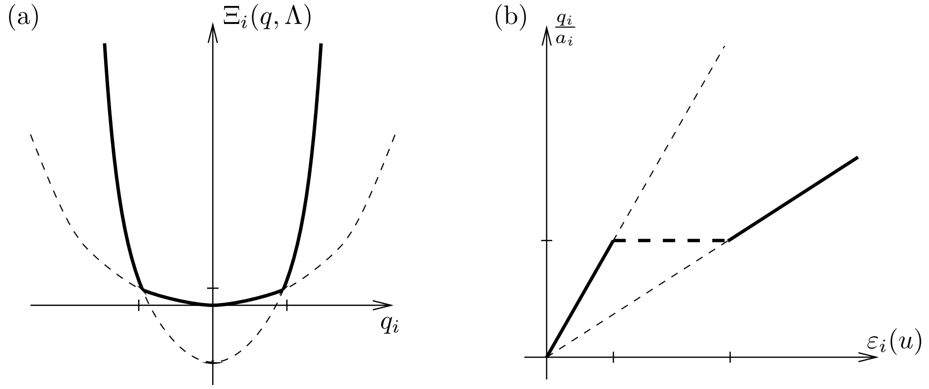


FIG. 3.1. *Specific member complementary energy* Ξ_i (a) *and stress-strain diagram* (b).

where for all $i = 1, \ldots, m$

$$\Xi_i(q,\Lambda) = \tfrac{1}{2}\max\left\{\frac{1}{E_i}\frac{q_i^2}{a_i^2}, \frac{1}{E_i^D}\frac{q_i^2}{a_i^2} - 2\Lambda\right\}$$

is the specific member complementary energy of the degraded truss. Its plot in q_i is shown in Figure 3.1(a). For this complementary energy we have a member stress-strain diagram (i.e., q_i/a_i expressed as a function of strains $\varepsilon_i(u) = b_i^T u/\ell_i$, where the displacements u are the multipliers for the equation $Bq = f$) as shown in Figure 3.1(b). This illustrates the nonlinear material behavior modeled through the original formulation (P1).

We finally note that problem (D1) is the dual of problem (P1SE) of Theorem 2.4 when this is written in the form

$$\max_{u\in\mathbb{R}^n,\ \varepsilon\in\mathbb{R}^m:\ \varepsilon_i = b_i^T u/\ell_i\ \forall i} \left\{ f^T u - \tfrac{1}{2}\sum_{i=1}^m a_i \ell_i \Psi_{i,\Lambda}^{\mathbb{R}}(\varepsilon_i) + \Lambda D \right\},$$

where $\Psi_{i,\Lambda}^{\mathbb{R}}$ is given by $\Psi_i(\,.\,,\Lambda)$ through the substitution $u^T K_i u = \varepsilon_i^2$.

Also, (P1SE) and (D1) have the same stress-strain diagram, and this diagram is a plot of the derivatives of $\Psi_{i,\Lambda}^{\mathbb{R}}$; cf. Figure 2.1. □

4. A QP formulation for trusses.

In the particular situation of a truss structure, we can further manipulate problem (D1) so as to achieve problems that are numerically more tractable. This seems to be a necessity—at least for larger dimension m—since the inner nonsmooth terms would destroy any reasonable numerical performance. We outline in Theorem 4.1 that (D1) is equivalent to the following quadratic programming problem (QP):

$$\text{(D1QP)} \qquad \min_{q\in\mathbb{R}^m,\ \mu\in\mathbb{R},\ s\in\mathbb{R}^m} \left\{ \tfrac{1}{2}\sum_{i=1}^m \tfrac{\ell_i}{a_i E_i} q_i^2 + \tfrac{1}{2}\sum_{i=1}^m a_i \ell_i s_i^2 + \tfrac{1}{2}(D - V)\mu^2 \right\}$$

$$\begin{aligned} \text{s.t.}\quad & Bq = f, \\ & 0 \le \mu \le s_i && \text{for all } i = 1, \ldots, m, \\ & -s_i \le \tfrac{\sqrt{E_i - E_i^D}}{a_i\sqrt{E_i E_i^D}} q_i \le s_i && \text{for all } i = 1, \ldots, m. \end{aligned}$$

In what follows we call the third group of constraints the *stress constraints*, because for trusses q_i/a_i denotes the stress of the ith bar, which here is limited by the values $\pm\sqrt{E_iE_i^D}s_i/\sqrt{E_i-E_i^D}$.

We already mention that (D1QP) is a *non*convex quadratic problem since the Hessian of the objective function is a diagonal matrix that has only positive entries apart from one negative coefficient $(D-V)$ (cf. (A3)).

The following theorem parallels Theorem 3.3. Because of lack of space we suppress the pathological case $\mu^*=0$ in assertion (iii).

THEOREM 4.1 (dual problem: formulation as QP).

(i) *The quadratic programming problem* (D1QP) *possesses a solution.*

(ii) *The optimal function values of problems* (D1QP) *and* (D1) *coincide, i.e.,*

$$\min(\text{D1QP}) = \min(\text{D1}).$$

(iii) *Let* (q^*,μ^*,s^*) *be a KKT point of* (D1QP) *with multipliers* $u^*\in\mathbb{R}^n$ *for the equilibrium constraints and multipliers* $\delta^{-*},\delta^{+*}\in\mathbb{R}^m$ *for the stress constraints. Moreover, we assume* $\mu^*>0$.

Then $(q^*,\frac{1}{2}(\mu^*)^2)$ *is a global optimizer of problem* (D1), *and* (β^*,u^*) *is a global optimizer of problem* (P1), *where*

$$\beta_i^* := \begin{cases} \frac{\sqrt{E_iE_i^D}}{\ell_i|q_i^*|\sqrt{E_i-E_i^D}}(\delta_i^{+*}+\delta_i^{-*}) & \text{if } (\delta_i^{+*}+\delta_i^{-*})>0, \\ 0 & \text{if } (\delta_i^{+*}+\delta_i^{-*})=0. \end{cases} \tag{4.1}$$

In particular,

$$\beta_i^* = \begin{cases} 1 & \text{if } \frac{|q_i^*|}{a_i} > \frac{\sqrt{E_iE_i^D}}{\sqrt{E_i-E_i^D}}\mu^*, \\ 0 & \text{if } \frac{|q_i^*|}{a_i} < \frac{\sqrt{E_iE_i^D}}{\sqrt{E_i-E_i^D}}\mu^*. \end{cases} \tag{4.2}$$

Moreover, (q^*,μ^*,s^*) *is a global optimizer of* (D1QP).

Proof. We take as our starting point (3.15). Then the substitutions

$$\mu := \sqrt{2\Lambda}, \qquad s_i := \sqrt{2\Lambda+2r_i} \quad \text{for all } i=1,\dots,m \tag{4.3}$$

show that for each (q,Λ,r) that is feasible for (3.15),

$$\sum_{i=1}^m a_i\ell_ir_i + \Lambda D = \tfrac{1}{2}\sum_{i=1}^m a_i\ell_is_i^2 + \tfrac{1}{2}(D-V)\mu^2\ . \tag{4.4}$$

Vice versa, the reverse substitutions

$$\Lambda := \tfrac{1}{2}\mu^2, \qquad r_i := \tfrac{1}{2}s_i^2 - \tfrac{1}{2}\mu^2 \quad \text{for all } i=1,\dots,m \tag{4.5}$$

apply, yielding (4.4) for any $\mu\geq 0$. The inequality constraints in problem (3.15) become

$$\mu^2 \leq s_i^2, \qquad \frac{E_i-E_i^D}{E_iE_i^D}\frac{q_i^2}{a_i^2} \leq s_i^2 \quad \text{for all } i=1,\dots,m. \tag{4.6}$$

By taking square roots and deleting redundant constraints we get the constraints in (D1QP). Summarizing, problems (D1) and (D1QP) are equivalent through the substitutions (4.3) and (4.5). This shows (ii). Together with Theorem 3.3(i) and substitutions (4.3) we obtain the proof of (i). For the proof of (iii) lengthy but simple calculations show that the KKT conditions of (D1QP) become (3.1) to (3.12). □

Remark 4.2. As already mentioned above, problem (D1QP) is nonconvex, though Theorem 4.1 proves its equivalence to the convex problem (D1). This indicates that convexity is somehow hidden in (D1QP). Indeed, the supplementary assertion in Theorem 4.1(iii) tells us that the KKT conditions are sufficient as in convex problems. It is known that for QP formulations the strong assumptions on convexity of the objective function can be weakened while sufficiency of KKT conditions remains valid (see, e.g., [5]). This is the case here (apart from the situation $\mu^* = 0$, which requires a sophisticated treatment). Formulation (D1QP) gives a nice tool for numerical computations: Efficient and well-tuned implementations exist, and thus we can easily accept $2m$ instead of m variables. However, the method used must be able to deal with the nonconvexity of (D1QP) (see also Example 5.1 for more comments). □

5. Progressive degradation and numerical optimization. Problem (P1) represents the degradation of the structure corresponding to one finite interval of possible global degradation (given by D). Modeling of progressive degradation as in a discrete time series requires that such finite interval models have to be solved successively. Here we say that at any step in a progressive degradation the result of the prior step gives a lower bound on the degradation that follows. Thus if the degradation parameter at step $k-1$ is given as β^{k-1} (the solution of (P1)), the next step of degradation is given as β^k, where $\beta^k \geq \beta^{k-1}$ has to be guaranteed. By this, $\beta^k - \beta^{k-1} \geq 0$ will be an increment in member degradation, and its amount (i.e., the length of the time-step) is controlled by the constant $D = D^k$ representing the permitted degradation volume interval of time-step k. For the total model to make sense, the sum of all global increments should satisfy $\sum D^k \leq V$. In addition, we also consider time-dependent loads; i.e., the acting forces change from step to step, $f = f^k$.

Summarizing, we get the following procedure for modeling progressive degradation. Choose the number K of total degradation steps, and choose degradation volume intervals $D^1, \ldots, D^K$, one for each degradation step, such that $\bar{D} := \sum D^k \leq V$. Moreover, let the load vectors $f^1, \ldots, f^K \in \mathbb{R}^n$ be given which apply at the structure during time-steps $k = 1, \ldots, K$, respectively. Put $\bar{\beta}^0 := 0 \in \mathbb{R}^m$.

For each degradation step $k = 1, \ldots, K$ solve the problem (in variables β^k, u^k)

$$(\mathrm{P1}^k) \qquad \begin{aligned} &\max_{\beta^k \in \mathbb{R}^m,\ u^k \in \mathbb{R}^n} \tfrac{1}{2} f^{kT} u^k \\ \text{s.t.}\quad &\sum_{i=1}^m \left[(1 - \bar{\beta}_i^{k-1} - \beta_i^k) \tfrac{1}{E_i} + (\bar{\beta}_i^{k-1} + \beta_i^k) \tfrac{1}{E_i^D} \right]^{-1} a_i \ell_i K_i u^k = f^k, \\ &0 \leq \beta_i^k \leq 1 - \bar{\beta}_i^{k-1} \quad \text{for all } i = 1, \ldots, m, \\ &\sum_{i=1}^m a_i \ell_i \beta_i^k \leq D^k, \end{aligned}$$

and put $\bar{\beta}^k := \bar{\beta}^{k-1} + \beta_k$. Then $\bar{\beta}^k$ corresponds to the degradation field that contains degradation during all the steps $1, \ldots, k$, while β^k denotes the incremental degradation field occurring in the particular step k.

Example 5.1. We consider a simple two-dimensional truss example of the main-span truss supporting a railway bridge. This truss is shown in Figure 5.1. It consists

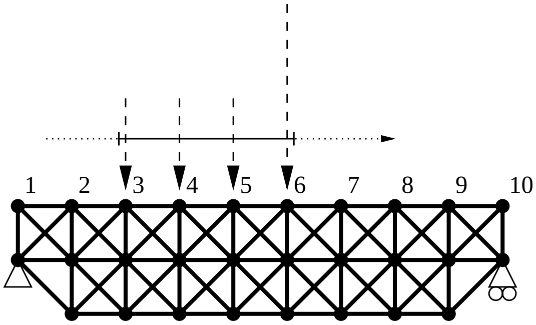


FIG. 5.1. *Structure and loads.*

TABLE 5.1
Time dependent load nodes.

Time-steps			Force nodes			
1	to	3	$\underline{1}$			
4	to	6	1	$\underline{2}$		
7	to	9	1	2	$\underline{3}$	
10	to	12	1	2	3	$\underline{4}$
13	to	15	2	3	4	$\underline{5}$
16	to	18	3	4	5	$\underline{6}$
19	to	21	4	5	6	$\underline{7}$
22	to	24	5	6	7	$\underline{8}$
25	to	27	6	7	8	$\underline{9}$
28	to	30	7	8	9	$\underline{10}$
31	to	33		8	9	10
34	to	36			9	10
37	to	39				10

of $m = 77$ bars and 28 nodes. One support node (left; cf. Figure 5.1) is completely fixed while the other support node (right) allows for a horizontal displacement, leading to $n = (28 \cdot 2) - 3 = 53$ degrees of freedom for the nodal displacements.

We consider all truss bars to consist of the same material, so $E_i := 1.0$ for all $i = 1, \ldots, m$. We regard the material as fully degraded if its stiffness is only 10% of the original one; i.e., we put $E_i^D := 0.1$ for all $i = 1, \ldots, m$. We compute $K = 39$ steps until half of the total volume $V := \sum a_i \ell_i$ of the structure represents fully degraded material, i.e., $D^k := V/(2K)$ for all $k = 1, \ldots, K$, and $\bar{D} = \frac{1}{2}V$.

We consider vertical forces mimicking the weight of a train running over the bridge, causing the degradation. One of the forces is doubled, simulating a heavy locomotive. By modeling degradation as progressing "faster" than the load changes, we perform three subsequent degradation steps before changing the loads. In Figure 5.1 the loads are displayed by dashed arrows while their movement in time is indicated by a small horizontal dotted arrow. The numbers in Figure 5.1 correspond to Table 5.1, which shows the nodes loaded in each time-step. The node number is underlined, which corresponds to the doubled force. The situation in Figure 5.1 shows the loads applied during time-steps $k = 16, 17, 18$.

In Figure 5.2 the results of all steps are displayed. For each time-step we have used the formulation (D1QP) which was tackled by the routine E04NAF (from the NAG library [10]), which is able to deal with indefinite QPs. A solution of (P1) was then obtained via Theorem 4.1.

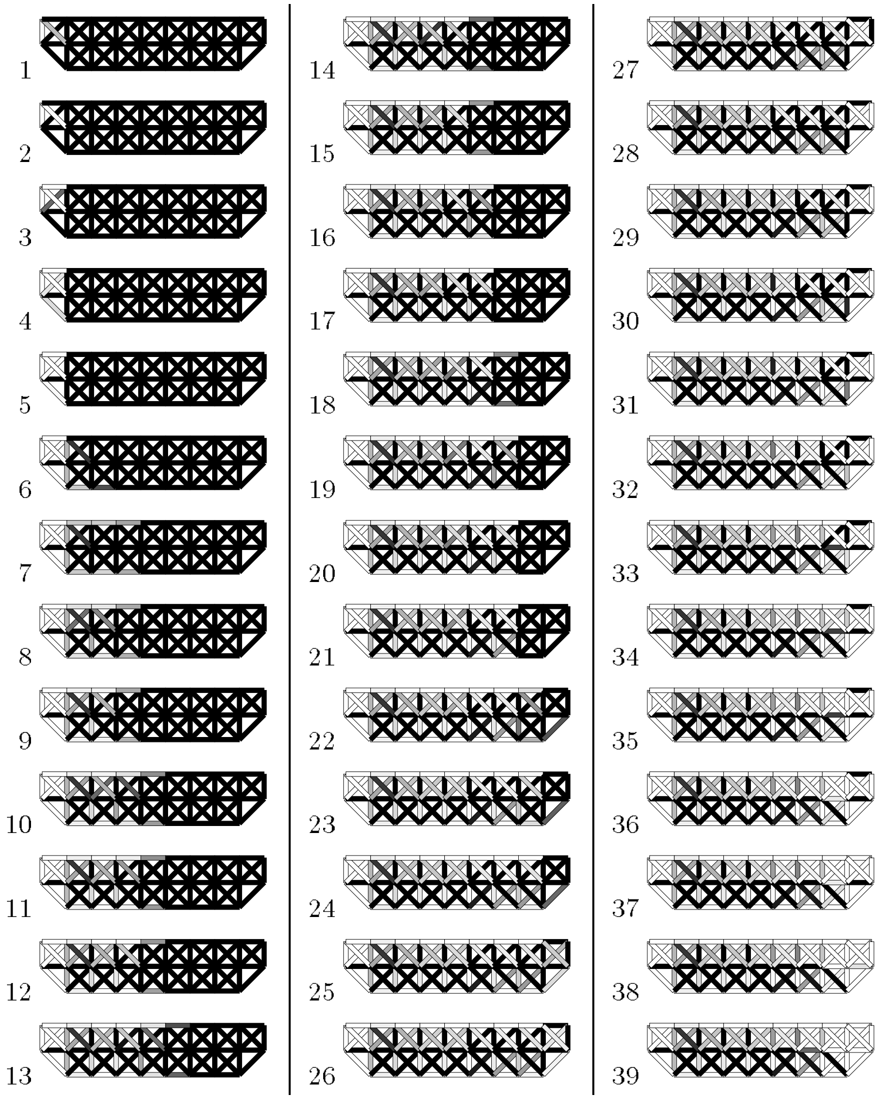


Fig. 5.2. *Progressive degradation pattern for maximal degradation of a bridge with moving loads.*

For each step k we show the state of the structure, i.e., the effective material stiffness constants E_i^{eff} including the accumulated degradation $\bar{\beta}^k$. The computed degradation is indicated by a gray scale: black bars mean nondegraded material (i.e., $\bar{\beta}_i^k = 0$), while white means fully degraded material (i.e., $\bar{\beta}_i^k = 1$). Values of E_i^{eff} in between are indicated by a linear gray scale from black to white.

One can see how degradation spreads from left to right "through the structure" as the train is running over the bridge. Note that all degradation pictures represent global solutions of the problems (P1^k) for $k = 1, \ldots, K$, respectively (cf. Theorem 4.1). □

6. Relation to an alternative degradation model. In this section we briefly compare our approach with an alternative method for including the internal parameter in the model: The range of material stiffness parameters between given values E_i^D and E_i is here *linearly* parameterized by an internal variable α_i (such a scheme is investigated in many works in solid mechanics; see, e.g., [7, 8]). For fixed α_i, the effective material constant is thus given by $E_i^{\text{eff}} = (1-\alpha_i)E_i + \alpha_i E_i^D$ (corresponding to the so-called Voigt upper bound on stiffness of mixtures of material; see, e.g., [4]). As above, we get the material constant $E_i^{\text{eff}} = E_i$ if $\alpha_i = 0$, and we obtain $E_i^{\text{eff}} = E_i^D$ if $\alpha_i = 1$. Again, total degradation in the structure is controlled by a volume constraint. This leads to the following model problem for a single time-step of degradation:

$$\text{(P2)} \qquad \begin{aligned} &\max_{\alpha\in\mathbb{R}^m,\ u\in\mathbb{R}^n} \tfrac{1}{2} f^T u \\ \text{s.t.}\quad &\sum_{i=1}^m \Big[(1-\alpha_i)E_i + \alpha_i E_i^D\Big] a_i \ell_i K_i u = f, \\ &0 \le \alpha_i \le 1 \quad \text{for all } i = 1,\dots,m, \\ &\sum_{i=1}^m a_i \ell_i \alpha_i \le D. \end{aligned}$$

Parallel to the definitions in section 2 we put

$$\phi(\alpha) := \max_{u\in\mathbb{R}^n} \left\{ \tfrac{1}{2} f^T u \,\Big|\, \sum_{i=1}^m \Big[(1-\alpha_i)E_i + \alpha_i E_i^D\Big] a_i \ell_i K_i u = f \right\}.$$

Moreover, almost analogously to Proposition 2.1 we may prove Proposition 6.1.

PROPOSITION 6.1.

(i) *For all* $\alpha \ge 0$*,*

$$\phi(\alpha) = \max_{u\in\mathbb{R}^n} \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^m \Big[(1-\alpha_i)E_i + \alpha_i E_i^D\Big] a_i \ell_i u^T K_i u \right\}.$$

(ii) *The function* ϕ *is convex on* $\mathbb{R}^m$.

(iii) *Problem* (P2) *always possesses a solution.*

Parallel to formulation (P1′) we get the following reformulation of (P2):

$$\text{(P2}'\text{)} \qquad \max_{\alpha\in\mathcal{B}_D} \phi(\alpha).$$

Remark 6.2. Note that due to the convexity of ϕ (cf. Proposition 6.1(ii)) the formulation (P2′) is a nonconvex problem (because we are *max*imizing). Thus model (P2′) (resp., (P2)) is not as attractive as formulation (P1′) (resp., (P1)) for numerical purposes.

A problem closely related to (P2) has been considered in [1], where an algorithm is proposed that finds a local optimizer of the formulation corresponding to (P2′) in a finite number of steps. However, we cannot expect these computed local optima to be global ones. □

Completely analogously to Theorem 2.4 we get Theorem 6.3.

THEOREM 6.3 (formulation in member strain energies). *For the optimal value of problem* (P2) *we have*

$$\max_{\alpha\in\mathcal{B}_D} \phi(\alpha) = \max_{u\in\mathbb{R}^n} \min_{\Lambda\ge 0} \left\{ f^T u - \tfrac{1}{2} \sum_{i=1}^m a_i \ell_i \Phi_i(u,\Lambda) + \Lambda D \right\},$$

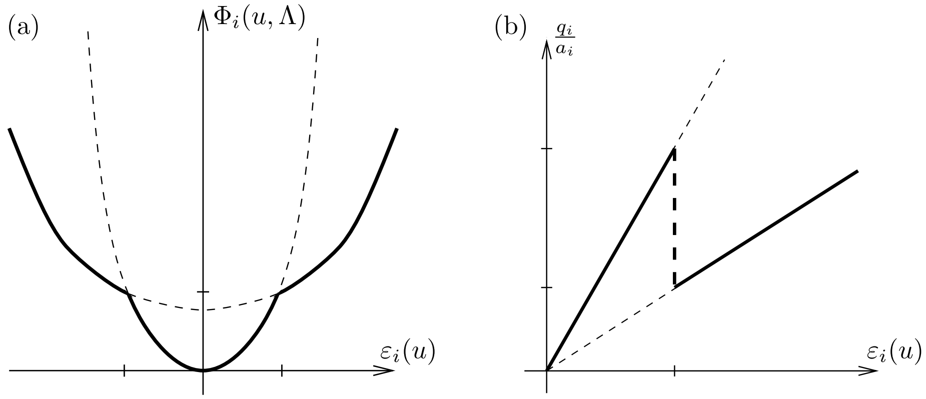


FIG. 6.1. *Specific member strain energy* Φ_i (a) *and member stress-strain diagram* (b).

where for all $i = 1, \ldots, m$

$$\Phi_i(u,\Lambda) := \begin{cases} E_i u^T K_i u & \textit{if } u^T K_i u \le \frac{2\Lambda}{E_i - E_i^D}, \\ E_i^D u^T K_i u + 2\Lambda & \textit{otherwise.} \end{cases}$$

Application 6.4. For trusses, the bar strain energy is displayed in Figure 6.1(a), which parallels Figure 2.1. One can see the nonconvex nature hidden in problem (P2) (contrary to (P1); cf. Remark 6.2). Analogously to Figure 3.1(b), the stress-strain diagram corresponding to Φ_i is displayed in Figure 6.1(b). We see the nonlinear material behavior degrading at a certain stage of strain (determined by Λ). □

The problem in inverse stiffness is in a weak sense the convexification of the problem with linear interpolation. To see this, consider the transformation $T(\alpha) := (T_1(\alpha), \ldots, T_m(\alpha))^T$, where

$$T_i(\alpha) := \frac{\alpha_i E_i^D}{(1-\alpha_i)E_i + \alpha_i E_i^D} \quad \text{for all } i = 1, \ldots, m. \tag{6.1}$$

Then the following holds. (The details are basic analysis and thus are skipped for brevity.)

THEOREM 6.5 (relations of (P1) and (P2)).

(i) $\max(\text{P2}) \le \max(\text{P1})$.

(ii) *Problem* (P1) *can be regarded as an outer convex approximation of* (P2) *in the sense that*

$$(\text{P2}) \equiv \max_{\alpha \in \mathcal{B}_D} \phi(\alpha) = \max_{\beta \in T(\mathcal{B}_D)} \psi(\beta) \approx \max_{\beta \in \mathcal{B}_D} \psi(\beta) \equiv (\text{P1}),$$

where $T(\mathcal{B}_D) \subset \operatorname{conv}(T(\mathcal{B}_D)) \subset \mathcal{B}_D$ *(and "*conv*" denotes the convex hull).*

The relation between the problems is further highlighted by noting that for the specific strain energy functions for the degraded structures defined in Theorems 2.4 and 6.3 we have that $\Psi_i(\,.\,,\Lambda) : \mathbb{R}^n \longrightarrow \mathbb{R}$ is differentiable and convex, and it is the convex envelope of $\Phi_i(\,.\,,\Lambda) : \mathbb{R}^n \longrightarrow \mathbb{R}$ (also compare Figure 2.1 with Figure 6.1(a)).

Acknowledgment. Useful suggestions from an anonymous referee on the material presented in this paper are gratefully acknowledged.

REFERENCES

[1] W. Achtziger and M. P. Bendsøe, *Design for maximal flexibility as a simple computational model of damage*, Struct. Opt., 10 (1995), pp. 258–268.

[2] W. Achtziger, M. P. Bendsøe, and J. E. Taylor, *Bounds on the effect of progressive structural degradation*, J. Mech. Phys. Solids, 46 (1998), pp. 1055–1087.

[3] A. Ben-Tal and M. P. Bendsøe, *A new method for optimal truss topology design*, SIAM J. Optim., 3 (1993), pp. 322–358.

[4] R. M. Christensen, *Mechanics of Composite Materials*, Wiley Interscience, New York, 1979.

[5] L. B. Contesse, *Une caractérisation complète des minimeaux en programmation quadratique*, Numer. Math., 34 (1980), pp. 315–332.

[6] E. J. Haug and J. S. Arora, *Applied Optimal Design*, Wiley, New York, 1979.

[7] D. Krajcinovic, *Damage Mechanics*, Elsevier Science B.V., Amsterdam, 1996.

[8] J. Lemaitre, *A Course on Damage Mechanics*, 2nd ed., Springer-Verlag, Heidelberg, Berlin, 1996.

[9] G. A. Francfort and J.-J. Marigo, *Stable damage evolution in a brittle continuous medium*, European J. Mech. A Solids, 12 (1993), pp. 149–189.

[10] *NAG (Numerical Algorithms Group Limited) FORTRAN Library Manual, Mark* 13, NAG Ltd., Oxford, UK, 1988.

[11] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

SIAM J. Optim.
Vol. 10, No. 4, pp. 999–1019

ON MINIMIZING AND CRITICAL SEQUENCES IN NONSMOOTH OPTIMIZATION[*]

L. R. HUANG[†], K. F. NG[‡], and J.-P. PENOT[§]

Abstract. Let f be a bounded below, lower semicontinuous function from a Banach space into $R \cup \{+\infty\}$. We study the relationships between minimizing and critical sequences of f, where the criticality condition is given in terms of some subdifferential ∂. Here the objective function f is not supposed to be convex or smooth. Our work extends that of Auslender and Crouzeix and that of Chou, Ng, and Pang.

Key words. asymptotically well-behaved functions, critical sequences, minimizing sequences, stationary sequences, subdifferential, well behavior

AMS subject classifications. 26B25, 26A96, 90C30, 90C33

PII. S1052623498341259

1. Introduction.

The question of determining conditions ensuring that a critical (or stationary, cf. Definition 3.1) sequence of a given function is minimizing is crucial for algorithms. Recall that a sequence (x_n) of some normed vector space X is said to be *critical* (or stationary) for a differentiable function f on X if $(f'(x_n)) \to 0$, where f' is the derivative of f; it is said to be *minimizing* if $(f(x_n)) \to \inf f(X)$. This question has been considered in the convex case by Auslender [2], Auslender and Crouzeix [4], Auslender, Cominetti, and Crouzeix [5], Lemaire [26], and Angleraud [1]. Here we tackle the case of a nonconvex, nonsmooth function f, a more intricate case. The problem of minimizing f over a feasible subset F of X, where F is a closed subset of X, can be reduced to the preceding unconstrained problem by using different techniques such as the penalization method. The problem also becomes an unconstrained one if f is replaced by f_F defined by $f_F = f$ on F and $f_F = +\infty$ elsewhere (i.e., $f_F := f + \iota_F$, where $\iota_F := 0_F$ is the indicator function of F). In both cases the objective function becomes nonsmooth, even if the initial data are very smooth. This is the reason why we treat the nonsmooth case from the outset. For the sake of versatility, we use an unspecified subdifferential satisfying some basic properties, owing to the facts that a given problem often imposes a particular space as a natural framework and that not all subdifferentials have nice properties in an arbitrary space. Thus the choice of a subdifferential may be dictated by the problem at hand. Since we avoid any specific construction, the reader interested in the unconstrained, smooth case only will not be confused by particular constructions or long developments; readers may suppose throughout that $\partial f(x) = \{f'(x)\}$.

[*] Received by the editors June 26, 1998; accepted for publication (in revised form) November 2, 1999; published electronically June 6, 2000. This work was supported by a direct grant and an earmarked grant from the Research Grant Council of Hong Kong.

http://www.siam.org/journals/siopt/10-4/34125.html

[†] Department of Mathematics, South China Normal University, Guangzhou, People's Republic of China (huanglr@scnu.edu.cn). The research of this author was supported by The Chinese University of Hong Kong through a postdoctorate fellowship and by the Natural Science Foundation of Guang Dong Province, China.

[‡] Department of Mathematics, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong (kfng@math.cuhk.edu.hk).

[§] Department of Mathematics, Laboratoire de Mathématiques Appliquées, CNRS UPRES A 5033, Faculté des Sciences, Av. de l'Université, 64000 Pau, France (jean-paul.penot@univ-pau.fr).

Our results also have their origin in [11], where the authors considered smooth functions and $X = \mathbb{R}^n$ and focused their attention on the constrained case (whereas we only deal with the unconstrained case, possibly after one of the reductions described above). Moreover, we study the condition (which is weaker than usual convexity)

$$\lim_{n\to\infty} \frac{f(y_n) - f(z_n)}{\|y_n - z_n\|} = 0 \tag{1.1}$$

whenever $(y_n), (z_n)$ are critical sequences and are such that $y_n \neq z_n$ for each n. For f satisfying this condition, any critical sequence (x_n) for f is minimizing if and only if, for some sequence $(\lambda_n) \searrow \inf\{f(x) : x \in X\}$, one has

$$\sup_n \operatorname{dist}(x_n, L(\lambda_n)) < +\infty,$$

where $L(\lambda_n) := \{x : f(x) \leq \lambda_n\}$ and for $x \in X$ and a subset S of X $\operatorname{dist}(x, S) := d_S(x) := \inf\{\|x - y\| : y \in S\}$. This characterization and others displayed in section 6 extend results of [4], where the authors considered convex functions. In [17], the study of the relationships between minimizing sequences and stationary sequences was given in terms of the lower (Dini–) Hadamard directional derivatives for functions defined on a special class of Banach spaces (admitting a Lipschitz smooth bump function). Here we use a condition on the subdifferentials related to variational principles.

The class of functions satisfying condition (1.1) (we call them C-convex functions) has some analogy with the class of Legendre functions (i.e., the class of functions for which the Legendre transform is well defined) as it contains the class of convex functions and the class of quadratic functions. Note that when $\partial(-f)(x) = -\partial f(x)$ for each x (and this property occurs when f is Lipschitzian and one takes the Clarke's subdifferential [12] or the moderate subdifferential of Michel and Penot [30], [31]), the class of C-convex functions also contains the class of concave functions. It also contains functions which are neither convex nor concave and, more generally, the function $x \mapsto (x + r)^n$ for any positive integer n and any $r \in \mathbb{R}$. Under appropriate assumptions, it also contains the important class of distance functions that is known to play a crucial role in nonsmooth analysis. Since this class enjoys reasonable stability properties (in particular under composition with mappings of class C^1), this class is rich enough. In fact, if ∂ is the Fenchel subdifferential, or the Plastria subdifferential, any function is C-convex. Moreover, the smaller the subdifferential ∂ is, the larger is the class of C-convex functions. Of course, the Fenchel (resp., Plastria) subdifferential is an extreme instance, which is of interest essentially in the class of convex (resp., quasi-convex) functions.

We also consider related classes of functions such as the class of critical functions, which is a natural class for studying the relationships between critical sequences and minimizing sequences.

Let us end this introduction with an observation pertaining to constrained problems, which could give rise to further study. When f is a function of class C^1, for most subdifferentials ∂, the subdifferential of the function $f_F := f + \iota_F$ introduced above is given by

$$\partial f_F(x) = f'(x) + \partial\iota_F(x) = f'(x) + N(F, x),$$

where $N(F, x)$ is the normal cone to F at $x \in F$ (in a sense related to ∂ by the formula $N(F, x) := \partial\iota_F(x)$). Therefore (x_n) is a critical sequence in the sense of Definition 3.1 below if and only if the distance $\operatorname{dist}(-f'(x_n), N(F, x_n))$ of $-f'(x_n)$ to $N(F, x_n)$ goes to 0. This remark enables one to detect a link between our approach and the results of [11]. However, that paper relied on the notion of residual functions and on the study of error bounds, which are somewhat outside the subject of the present paper; see [3], [25], [27], [33], [42] for recent contributions and references on this important topic which appeared after the first version of the present paper had been written.

2. Subdifferentials.

Throughout we consider lower semicontinuous (l.s.c.) extended real-valued functions f defined on a Banach space X whose dual space is denoted by X^*. As usual, we write $B(x,r)$ for the open ball with center x and radius r and B_{X^*} for the closed unit ball in the dual space X^* of X, and we set

$$\mathrm{dom} f := \{x \in X : -\infty < \; f(x) < +\infty\}.$$

Since we deal with nonconvex, nondifferentiable functions, we have to replace the derivative by a subdifferential. However, the reader who is just interested in the smooth case may suppose throughout that only differentiable functions are considered and take as the subdifferential at x of a function f the singleton $\{f'(x)\}$. There are several ways of presenting subdifferentials (see, for instance, [6], [21], [23], [41], and their references). As noticed by several authors, a unified approach through a set of general properties is convenient: in such a way specific constructions can be avoided. In what follows we denote by $\mathcal{X}$ a class of normed vector spaces (n.v.s), for instance, the class of all Banach spaces, the class of separable spaces, or the class of Asplund spaces. For X in $\mathcal{X}$, $\mathcal{F}(X)$ denotes a subset of the set of functions from X to $\mathbb{R}^{\cdot} = \mathbb{R} \cup \{\infty\}$.

We consider a *subdifferential* ∂ associated with the classes $\mathcal{X}, \mathcal{F}$ as a mapping which associates to any X in $\mathcal{X}$, $f \in \mathcal{F}(X)$, $x \in X$ a subset $\partial f(x)$ of X^*. In the usual examples that follow, a number of useful calculus rules are satisfied. Here we do not impose such rules. However, we will occasionally use some of them.

Given a subdifferential ∂ associated with the classes $\mathcal{X}, \mathcal{F}$, given X in $\mathcal{X}$, let $\mathcal{S}(X)$ be the family of subsets S of X such that the distance function d_S to S belongs in $\mathcal{F}(X)$; then the normal cone to S can be introduced as

$$N(S,x) := \mathbb{R}_+ \partial d_S(x).$$

If $\mathcal{F}(X)$ is the class of l.s.c. functions on X and if S is closed, an alternative definition uses the indicator function ι_S of S. If $\mathcal{F}(X)$ is the class of Lipschitzian functions on X, the subdifferential can be extended to any l.s.c. function f by setting

$$\partial f(x) := \{x^* \in X^* : (x^*, -1) \in N(E_f, x_f)\},$$

where $E_f = \{(x,r) : r \geq f(x)\}$ is the epigraph of f and $x_f := (x, f(x))$.

Examples. The main examples that are frequently used, besides the Fenchel–Moreau subdifferential, the lower subdifferential of Plastria, the proximal subdifferential, the subdifferential of Clarke [12], and the moderate subdifferential of Michel and Penot [31], are the following ones.

(1) *The infradifferential* of Gutiérrez f at x is the set $\partial^{\leq} f(x)$ of all vectors x^* satisfying

$$f(u) \geq f(x) + \langle x^*, u - x\rangle \qquad \forall u \in \{v : f(v) \leq f(x)\}.$$

(2) *The Fréchet subdifferential* of f at x is the set $\partial^- f(x)$ of all vectors x^* satisfying

$$\liminf_{\|u\| \to 0_+} \|u\|^{-1}(f(x+u) - f(x) - \langle x^*, u\rangle) \geq 0.$$

(3) *The Hadamard subdifferential* $\partial^{!}$, which consists of all x^* satisfying

$$\langle x^*, u\rangle \leq df(x;u) := \liminf_{(t,v)\to(0_+,u)} t^{-1}(f(x+tv)-f(x)) \;\; \forall\, u \in X.$$

(4) *The viscosity subdifferentials*, which are obtained by taking the set of derivatives at x of differentiable functions g verifying $g \leq f,\ g(x) = f(x)$. Such a process defines several classes, since the differentiability assumption can be given in different senses.

(5) *The Ioffe subdifferential.* For a Lipschitzian function f and a vector subspace $W \subset X$ containing $u \in X$, one sets

$$\partial_W f(u) = \{u^* :\ \langle u^*, w\rangle \leq df(u,w) \;\; \forall w \in W\}$$

and one defines the approximate subdifferential of f at x by

$$\widehat{\partial} f(x) = \bigcap_{W\in\mathcal{L}} \limsup_{u\to x,\ u\in W} \partial_W f(u) \bigcap cB_{X^*},$$

where $\mathcal{L}$ is the collection of finite dimensional linear subspaces of X and c is any number greater than the Lipschitz rate of f near x. In the general case, one defines $\partial f(x)$ as above using the normal cone to the epigraph of f.

(6) *The limiting or stabilized subdifferentials.* Using any subdifferential as a starting point, one can define a new subdifferential $\overline{\partial}$ called the *limiting* or stabilized subdifferential associated with ∂. Namely, one says that x^* belongs to $\overline{\partial} f(x)$ if it is a weak* cluster point of a sequence (x_n^*) such that $x_n^* \in \partial f(x_n)$ for each n, where $(x_n) \to x$ and $(f(x_n)) \to f(x)$. A similar definition holds for normal cones.

(7) *The extended Clarke subdifferential* of f at x is the set $\partial^0 f(x)$ of all $x^* \in X^*$ such that $x^* \leq f^0(x,\cdot)$, where

$$f^0(x,v) := \limsup_{\substack{(t,w)\to(0_+,x)\\ f(w)<\infty}} t^{-1}(f(w+tv)-f(w)).$$

This is not the most sensible way of extending the classical definition to nonlocally Lipschitz functions (see [12] for better proposals). However, we will use this notion as a bound, imposing an assumption of the form $\partial f(x) \subset \partial^0 f(x)$ in some statements. The fact that $\partial^0 f(x)$ is usually very large makes this assumption rather mild.

Additional properties of subdifferentials will be needed. The following notion which has been pointed out recently [42], [43] proves to be a convenient substitute to a combination of a variational principle with a fuzzy sum rule.

Definition 2.1. *A subdifferential ∂ is said to be variational on a Banach space X for a class $\mathcal{F}(X)$ if for any bounded below l.s.c. function $f \in \mathcal{F}(X)$ and for any $\varepsilon, \lambda, \rho \in\ \mathbb{P} :=]0,\infty[$ and for any $x \in X$ such that $f(x) < \inf f(X) + \lambda\rho$ there exist $w \in B(x,\rho)$ and $w^* \in \partial f(w)$ such that $\|w^*\| \leq \lambda,\ f(w) \leq f(x) + \varepsilon$.*

As observed in [42], this property is satisfied when ∂ is reliable in the sense of the following definition, which can be considered as a variant of the notion of trustworthiness due to Ioffe [20]. Thus, this property holds whenever a sum rule of weak type (even weaker than the one considered in [41]) is satisfied. In fact, as shown below, it suffices to combine Ekeland's variational principle with a fuzzy sum rule (or even a basic fuzzy principle in the sense of [24]) to get this property. When f is convex and ∂ is the Fenchel subdifferential, it is a consequence of the Brøndsted–Rockafellar theorem [10] that ∂ is variational.

Definition 2.2 (see [37], [40]). *Given a class of spaces $\mathcal{X}$, a class of functions $\mathcal{F}$, and a subdifferential ∂ associated with the classes $\mathcal{X}$, $\mathcal{F}$, a Banach space X in $\mathcal{X}$ is said to be a ∂-reliable space (or a ∂-$\mathcal{F}$-reliable space if there is any risk of confusion) if for any l.s.c. function $f : X \to \mathbb{R} \cup \{\infty\}$ in $\mathcal{F}(X)$, for any convex Lipschitzian function $g \in \mathcal{F}(X)$, for any $x \in \operatorname{dom} f$ at which $f + g$ attains its infimum, and for any $\varepsilon > 0$ one has*

$$0 \in \partial f(u) + \partial g(v) + \varepsilon B_{X^*}$$

for some $u, v \in B(x, \varepsilon)$ such that $\mid f(u) - f(x) \mid < \varepsilon$. We also say that ∂ is reliable on X (for the class $\mathcal{F}$).

If ∂ is the Fréchet or the Hadamard subdifferential, any Asplund space is ∂-reliable; in particular, any reflexive Banach space and any Banach space whose dual is separable is ∂-reliable. If ∂ is some viscosity subdifferential and if X has a smooth enough bump function [13], then X is ∂-reliable. If ∂ is the Clarke subdifferential or the Ioffe subdifferential, then any Banach space is ∂-reliable. For the class of l.s.c. convex functions and the usual subdifferential ∂ of convex analysis, any Banach space is ∂-reliable. The same is true for the Hadamard subdifferential and the class of tangentially convex functions, a function f being tangentially convex if its Hadamard lower derivative at any point x is convex. Let us observe that quite often X and $\mathcal{F}(X)$ are given and one chooses a subdifferential ∂ such that X is reliable for ∂ in order to have the useful property described in the preceding definition.

We are ready to prove that a reliable subdifferential is variational.

Proposition 2.3. *Given a member X of a class of spaces $\mathcal{X}$, a class of functions $\mathcal{F}$, and a subdifferential ∂ associated with the classes $\mathcal{X}$, $\mathcal{F}$ is variational on X if it is reliable on X.*

Proof. Given a bounded below l.s.c. function $f \in \mathcal{F}(X), \varepsilon, \lambda, \rho, \gamma \in \mathbb{P} :=]0, \infty[$ with $\gamma \leq \lambda\rho$ and $x \in X$ such that $f(x) < \inf f(X) + \gamma$, taking $\alpha > 0$ such that $\gamma < (\lambda - \alpha)(\rho - \alpha)$, the Ekeland's variational principle yields some $u \in B(x, \rho - \alpha)$ such that $f(u) \leq f(x),\ f(u) \leq f(v) + (\lambda - \alpha)\|v - u\|$ for each $v \in X$. Since the function h given by $h(v) := f(v) + (\lambda - \alpha)\|v - u\|$ attains its minimum on X at u, and since ∂ is reliable on X, one can find $w, z \in B(u, \alpha),\ w^* \in \partial f(w),\ z^* \in (\lambda - \alpha)B_{X^*}$ such that $f(w) \leq f(u) + \varepsilon,\ \|w^* + z^*\| \leq \alpha$. Then we have $\|w^*\| \leq \lambda,\ w \in B(x, \rho)$. □

We will make occasional use of the following rule which is more exacting than reliability, but we will impose it for two specific functions only.

Definition 2.4. *Given a subdifferential ∂, two functions g, h in $\mathcal{F}(X)$ are said to satisfy the fuzzy sum rule if for any x in the domain of their sum f, any $x^* \in \partial f(x)$, and any $\varepsilon > 0$, there exist $y, z \in B(x, \varepsilon),\ y^* \in \partial g(y),\ z^* \in \partial h(z)$ such that $\mid g(y) - g(x) \mid < \varepsilon, \mid h(z) - h(x) \mid < \varepsilon,\ \|y^* + z^* - x^*\| < \varepsilon$.*

We can also say that ∂ satisfies the fuzzy sum rule for g and h. A related property is contained in the following definition; it could be restricted to two functions, as above.

Definition 2.5. *The subdifferential ∂ is said to satisfy the fuzzy composition rule if for any X, Y in $\mathcal{X}$, for any $g \in \mathcal{F}(Y)$, and any continuously differentiable map $h : X \to Y$ with a surjective derivative at each point, then $f := g \circ h \in \mathcal{F}(X)$ and for any $x \in X$, $x^* \in \partial f(x)$, and any $\varepsilon > 0$ there exist $y \in B(h(x), \varepsilon)$, $y^* \in \partial g(y)$, such that $\mid g(y) - g(h(x)) \mid < \varepsilon$, $\|A^T y^* - x^*\| < \varepsilon$, where $A = h'(x)$.*

If for any $x \in X$ one has $\partial f(x) \subset h'(x)^T(\partial g(h(x)))$, one says that ∂ satisfies the composition rule.

3. Critical and minimizing sequences. In what follows we suppose that a class of spaces $\mathcal{X}$, a class of functions $\mathcal{F}$, and a subdifferential ∂ associated with the classes $\mathcal{X}, \mathcal{F}$ are given. Having a notion of subdifferential, it is natural to extend to nonsmooth functions the notions of critical sequence and of critical point.

Definition 3.1. *Given a subdifferential ∂ relative to the classes $\mathcal{X}, \mathcal{F}$, a member X of $\mathcal{X}$, and a function f in $\mathcal{F}(X)$, we say that a point x of X is a critical point if $0 \in \partial f(x)$. A real number r is a critical value of f if there exists a critical point x such that $r = f(x)$. A sequence (x_n) is critical or, more precisely, ∂-critical for f if there exists a sequence (x_n^*) such that*

$$x_n^* \in \partial f(x_n) \text{ and } (x_n^*) \to 0. \tag{3.1}$$

A critical sequence is often called stationary, but we prefer to keep this term for the case when (x_n) is critical for f and $-f$. Note that a critical sequence (x_n) may be such that none of the x_n's is critical. This situation occurs in most numerical experiences with optimization algorithms. Moreover, a critical sequence may not approach a critical point. As recalled above, a sequence (x_n) is called a *minimizing sequence* if $(f(x_n)) \to \inf f$ $(:= \inf_{x \in X} f(x))$ (see [17]). Note that minimizing sequences always exist and that critical sequences do not always exist but frequently appear in using algorithms. Thus, relating both notions is an important matter.

The following theorem from [36] refines a recent result in [11] which extends a method known in the differentiable case [15, p. 455]. It shows that for any minimizing sequence of f one can always find "nearby" a sequence that is both critical and minimizing. We give a proof for completeness.

Theorem 3.2. *Let X be a Banach space in $\mathcal{X}$, let ∂ be a variational subdifferential ∂ for X, and let $f \in \mathcal{F}(X)$ be bounded below and l.s.c. Let (x_n) be a minimizing sequence for f. Then there exist sequences (w_n) and (w_n^*) with $w_n^* \in \partial f(w_n)$ for each n such that*

(a) $\lim_{n\to\infty} f(w_n) = \inf f$;

(b) $\lim_{n\to\infty} \|x_n - w_n\| = 0$;

(c) $\lim_{n\to\infty} w_n^* = 0$.

Proof. Since the sequence (x_n) is minimizing, there exist sequences (ε_n), $(\delta_n) \downarrow 0$ such that $\varepsilon_n < \delta_n$ for each n and

$$f(x_n) \le \inf_{x\in X} f(x) + \varepsilon_n \quad \forall n.$$

Thus, as ∂ is variational, for $\lambda_n = \rho_n = \delta_n^{\frac{1}{2}} > 0$, there exists $w_n \in X$, $w_n^* \in X^*$ such that

(i) $\|x_n - w_n\| \le \lambda_n$;

(ii) $f(w_n) \le f(x_n) + \varepsilon_n$;

(iii) $w_n^* \in \partial f(w_n)$ with $\|w_n^*\| \le \lambda_n$.

Therefore (a)–(c) hold. □

Now let us turn to the question, Under which conditions is a minimizing sequence (x_n) critical? Simple examples in $\mathbb{R}$ show it is not always the case. In order to present a positive answer, let us introduce a suitable uniform continuity condition on the map ∂f. We formulate it in the framework of set-valued analysis, but we are conscious that the main realistic cases of application concern the single-valued case.

Definition 3.3. *Let X, Y be metric spaces and let $F : X \rightrightarrows Y$ be a multifunction. We say that it is uniformly upper semicontinuous near a sequence (x_n) of X if for* *any $\epsilon > 0$, there exist $\delta > 0$, m such that, for all $n \geq m$, $w \in B(x_n, \delta)$ one has*

$$F(w) \subseteq F(x_n) + B(0, \epsilon).$$

Example 3.1. (a) If F is a uniformly continuous mapping, then it is uniformly upper semicontinuous near any sequence (x_n) of X.

(b) If F is a mapping that is continuous at some $x \in X$ and if $(x_n) \to x$, then F is uniformly upper semicontinuous near the sequence (x_n). This example can be extended to multimappings that are continuous for the Pompeiu–Hausdorff metric.

(c) The function f on $\mathbb{R}$ given by $f(x) = x^2 \sin x$ for $x \in \bigcup_{k \in Z}[(3k-1)\pi, 3k\pi]$, $f(x) = 0$ otherwise, is not uniformly continuous on $\mathbb{R}$, but it is uniformly upper semicontinuous near the sequence (x_n) given by $x_n = (3k+1)\pi$.

(d) If $F = \partial f$, where f is the one-variable function given by $f(x) =| x |$, then F is uniformly upper semicontinuous near any critical sequence (x_n) of f.

(e) Consider the Weierstrass function $f(x) = \sum_{n=0}^{\infty} b^n \cos(a^n \pi x)$, where $0 < b < 1$ and a is an odd integer with $ab > 1 + \frac{3}{2}\pi$. It is well known that f is continuous and nowhere differentiable [16, pp. 404–405]. One can show that $\partial^0 f(x) = \mathbb{R}$ for any $x \in \mathbb{R}$. Thus the subdifferential mapping of f, $x \longmapsto \partial^0 f(x)$ is uniformly upper semicontinuous.

(f) Another example for the uniform upper semicontinuity of the map $x \mapsto \partial f(x)$ is the function $f : \mathbb{R} \to \mathbb{R} \cup \{+\infty\}$ defined by

$$f(r) = \begin{cases} \frac{1}{n}, & r = n, \quad n \in \mathbb{N}\backslash\{0\}, \\ +\infty & \text{otherwise.} \end{cases}$$

Then f is a l.s.c. function with an unbounded minimizing sequence (n) such that $f(n) \to \inf f = 0$. Note that $\partial f(n) = \mathbb{R}$ for all $n \in \mathbb{N}$ whenever ∂ is larger than the Fréchet subdifferential. Hence the subdifferential map $x \mapsto \partial f(x)$ is uniformly upper semicontinuous near any critical sequence.

LEMMA 3.4. *Let X and f be as in Theorem* 3.2 *and suppose that the subdifferential map ∂f is uniformly upper semicontinuous near a minimizing sequence (x_n). Then (x_n) is critical: there exists $x_n^* \in \partial f(x_n)$ such that $(\|x_n^*\|) \to 0$.*

Proof. By Theorem 3.2, we can take sequences $(w_n), (w_n^*)$ satisfying (a), (b), and (c). Let $(\epsilon_k) \downarrow 0$. For each ϵ_k, by the upper semicontinuity condition, there exist $\delta_k > 0$ and m_k such that

$$\partial f(x) \subseteq \partial f(x_n) + B(0, \epsilon_k)$$

whenever $n \geq m_k$ and $\|x_n - x\| < \delta_k$. We do this for each k, and we manage to have $(\delta_k) \downarrow 0$. Since $\|x_n - w_n\| \to 0$, there exists $n_k \geq m_k$ such that

$$\|x_n - w_n\| < \delta_k \qquad \forall n > n_k,$$

and so,

$$w_n^* \in \partial f(w_n) \subseteq \partial f(x_n) + B(0, \epsilon_k) \qquad \forall n > n_k. \tag{3.2}$$

We may suppose that (n_k) is increasing. Therefore, for $n_k < n \leq n_{k+1}$ we can pick $x_n^* \in \partial f(x_n)$ such that

$$\|x_n^* - w_n^*\| \leq \epsilon_k.$$

By (c) it follows that $(x_n^*) \to 0$. □

4. Critical functions and minimizing sequences. Extending a notion studied by Auslender [2], Auslender and Crouzeix [4], and Auslender, Cominetti, and Crouzeix [5] (where these authors considered the convex case, with $X = \mathbb{R}^n$), an l.s.c. function f on X into $(-\infty, +\infty]$ is said to be (asymptotically) *well behaved* if all of its critical sequences are minimizing (see [42]). We write $f \in \mathcal{W}$. For convex functions, this behavior can be characterized in terms of an estimate of the distance between critical sequences and the level sets (see [4] for details). Here we go a little further by dealing with functions that are not necessarily convex: we shall consider a class larger than the class of convex l.s.c. asymptotically well-behaved functions studied in [4].

In the present section we study a class of functions that is closely linked with the study of critical sequences.

Definition 4.1. *A function $f : X \to (-\infty, +\infty]$ is said to be ∂-critical, or critical for the subdifferential ∂, or, in short, critical if there is no risk of confusion, if for any critical sequence (x_n) the sequence $(f(x_n))$ of values converges in $\mathbb{R}$.*

Let us stress that this notion depends on the choice of the subdifferential. The one-variable function f given by $f(x) = \min(e^x, e^{-x})$ is critical for the Fréchet and the Hadamard subdifferentials but is not critical for the Clarke subdifferential or for the moderate subdifferential of Michel and Penot. Note that this function has no critical point for the Hadamard subdifferential (hence no minimizer). Also, the function f given by $f(x) := \min(x_+, 1)$, where $x_+ = \max(x, 0)$, is critical for the Fenchel–Moreau subdifferential but not for the lower subdifferential of Plastria.

Clearly, not all functions are critical: sin, cos, arctan are not critical functions when one considers the class of differentiable functions and takes as subdifferential the derivative. However, the class of critical functions contains significant elements, as the following examples and results show.

Example 4.1. Let f be a quadratic function with positive definite Hessian on a Hilbert space. Then, by the Lax–Milgram theorem, f is critical. In fact f is well behaved, and any bounded below well-behaved convex function is critical for the Fenchel–Moreau subdifferential (hence for most subdifferentials).

Example 4.2. If f is a quadratic function with positive semidefinite Hessian A on a Hilbert space X, it may happen that f is not critical. In fact, taking $X = \ell_2$ with its canonical basis (e_n), setting $x_n = ne_n$, and assuming that $Ae_n = n^{-2}e_n$, we see that (x_n) is critical for f given by $f(x) = (Ax \mid x)$, but not minimizing.

Example 4.3. Let S be a nonempty closed subset of an arbitrary Banach space X and let $f := d_S$ be the distance function to S. Take for subdifferential the Fréchet subdifferential ∂^-. Since for any $x \in X \backslash S$, $x^* \in \partial^- f(x)$ one has $\|x^*\| = 1$ (see [9], [39]) any critical sequence (x_n) is eventually in S and $(f(x_n)) \to 0$. Thus f is a critical function.

Example 4.4. Let $f : W \to \mathbb{R}$ be a ∂-invex function on an open subset W of X; this means that there exists a mapping $v : W \times W \to X$ such that $f(w) - f(x) \geq \langle x^*, v(w, x)\rangle$ for each $(w, x) \in X^2$ and each $x^* \in \partial f(x)$. In the differentiable case (and for $\partial f(x) = \{f'(x)\}$), this class of functions has been the subject of numerous studies; it has the pleasant property that any critical point is a minimizer. Suppose further that v is bounded or else that it is bounded on bounded subsets of W and each critical sequence is bounded. Then f is ∂-critical: since for each critical sequence (x_n) of f there is a constant m such that $\|v(x_p, x_q)\| \leq m$ for any $p, q \in \mathbb{N}$, taking $x_n^* \in \partial f(x_n)$ such that $(x_n^*) \to 0$, we get $|f(x_p) - f(x_q)| \leq m \max(\|x_p^*\|, \|x_q^*\|) \to 0$, so that $(f(x_n))$ is a Cauchy sequence, and hence it converges.

Other examples can be given by applying the stability properties below or the criteria given in the next section.

An immediate consequence of the preceding definition is the following property, which is also shared by convex functions, pseudoconvex functions, and invex functions.

PROPOSITION 4.2. *A critical function has at most one critical value. Conversely, if f is a function of class C^1, if its derivative f' is proper at* 0 *(i.e., if any critical sequence has a cluster point), and if f has at most one critical value, then f is critical.*

Proof. Let us show that f is constant on the set Z of critical points of f. Given x, $x' \in Z$, let us define the sequence (x_n) by $x_{2p} = x,\ x_{2p+1} = x'$. Then (x_n) is critical. It follows that $f(x)$ and $f(x')$ must coincide with the limit of $(f(x_n))$.

For the converse, we suppose that $\partial f(x) = \{f'(x)\}$, a natural (and implicit) assumption when f is of class C^1. Given a critical sequence (x_n), taking a subsequence if necessary, we may assume that (x_n) converges; then $(f(x_n))$ converges to the unique critical value of f. □

As observed above, any well-behaved function is critical if it is bounded below. The following result presents a partial converse.

PROPOSITION 4.3. *Let $f \in \mathcal{F}(X)$ be a l.s.c. function bounded below on X and let ∂ be a variational subdifferential on $\mathcal{F}(X)$. Then, if f is critical, any critical sequence is minimizing: f is well behaved.*

Proof. Let (x_n) be a critical sequence. We know from Theorem 3.2 that there exists a minimizing sequence (w_n) which is also a critical sequence. The sequence (z_n) given by $z_{2n} = x_n,\ z_{2n+1} = w_n$ is critical. Our assumption on f ensures that $(f(z_n))$ converges to $\lim f(w_n) = \inf\ f(X)$. Thus $(f(x_n)) \to \inf\ f(X)$. □

The interest of the class of critical functions lies in the simplicity of its definition and its links with well behavior. Also, it enjoys some stability properties. Let us start with a composition property. Here we use the *openness index* (or Banach constant [22]) of a linear mapping $A : X \to Y$, given by

$$\operatorname{open}(A) := \sup\{\inf\{\|u\| : A(u) = v\} :\ v \in Y,\ \|v\| = 1\}.$$

PROPOSITION 4.4. *Let $h : X \to Y$ be a continuously differentiable map between two Banach spaces and let $g \in \mathcal{F}(Y)$. Suppose there exists a constant $c > 0$ such that for each $x \in X$ one has* $\operatorname{open}(h'(x)) < c$. *Suppose either ∂ satisfies the composition rule or ∂ satisfies the fuzzy composition rule and g is uniformly continuous. Then, if g is critical, the function $f := g \circ h$ is critical.*

Proof. Let $c > 0$ be as above. Given $x \in X$, let $A := h'(x)$. For each $v \in Y$ there exists $u \in A^{-1}(v)$ satisfying $\|u\| \le c\|v\|$. Thus, if $x^* = A^T(y^*)$ for some $y^* \in Y^*$ one has

$$\langle y^*, v\rangle = \langle y^*, A(u)\rangle = \langle x^*, u\rangle \le c\|x^*\|\|v\|,$$

so that $\|y^*\| \le c\|x^*\|$. Let (x_n) be a critical sequence for f and let $x_n^* \in \partial f(x_n)$ be such that $(x_n^*) \to 0$. Using the fuzzy composition rule we can find (y_n) in Y, $y_n^* \in \partial g(y_n)$ such that $(y_n - h(x_n)) \to 0$, $\left(x_n^* - h'(x_n)^T(y_n^*)\right) \to 0$. What precedes shows that $(y_n^*) \to 0$. As g is critical, $(g(y_n))$ converges. As g is uniformly continuous, it follows that $(f(x_n)) = (g(h(x_n)))$ converges. When the exact composition rule holds, we can take $y_n = h(x_n)$ and there is no need to suppose that g is uniformly continuous. □

PROPOSITION 4.5. *Let $f = h \circ g$, where $g : X \to \mathbb{R}$ is critical and $h : \mathbb{R} \to \mathbb{R}$ is differentiable. Suppose there exists $a > 0$ such that $h'(r) \ge a$ for each $r \in \mathbb{R}$ and suppose $\partial f(x) \subset h'(g(x))\partial g(x)$ for each $x \in X$. Then f is critical.*

Proof. Let (x_n) be a critical sequence for f and let $x_n^* \in \partial f(x_n)$ be such that $(x_n^*) \to 0$. Using our assumption, we can find $y_n^* \in \partial g(x_n)$ such that $x_n^* = h'(r_n)y_n^*$ with $r_n := g(x_n)$. Since $(h'(r_n))$ is bounded below by a, the sequence (x_n) is critical for g. Thus $(f(x_n))$ converges, g being critical and h being continuous. □

Let us observe that the relation $\partial f(x) \subset h'(g(x))\partial g(x)$ is satisfied by a number of subdifferentials such as the Fréchet and the Hadamard subdifferentials; if h is of class C^1 it is also satisfied by the Clarke subdifferential.

Now let us turn to stability with respect to addition. We first consider the case of separable functions.

Proposition 4.6. *Let $X := Y \times Z$ be the product of two Banach spaces and let g, h be l.s.c. critical functions on Y and Z, respectively. Let $f := \overline{g} + \overline{h}$, where $\overline{g}$ is the function given by $\overline{g}(y,z) = g(y)$ and $\overline{h}(y,z) = h(z)$. Suppose $\partial\overline{g}(y,z) \subset \partial g(y) \times \{0\}$, $\partial\overline{h}(y,z) \subset \{0\} \times \partial h(z)$, and the fuzzy sum rule is satisfied by ∂ for these functions. Then f is critical.*

Proof. Our assumption implies that if $(x_n) = (u_n, v_n)$ is a critical sequence of f, and if $(x_n^*) \to 0$ is such that $x_n^* \in \partial f(x_n)$ for each n, then we can find $(y_n,\ z_n) \in X$, $(y_n^*, z_n^*) \in X^*$ with $(g(y_n) - g(u_n)) \to 0$, $(h(z_n) - h(v_n)) \to 0$, $y_n^* \in \partial g(y_n)$, and $z_n^* \in \partial h(z_n)$ such that $x_n^* = w_n^* + (y_n^*, 0) + (0, z_n^*)$ for some sequence $(w_n^*) \to 0$ in X^*. Then (y_n) and (z_n) are critical for g and h, respectively, so that $(h(y_n))$ and $(g(z_n))$ converge. It follows that $(f(x_n))$ converges. □

In order to present another stability result, let us recall that a closed convex cone Q in a Banach space Y is said to be *normal* if for any sequences (y_n), (z_n) in Q such that $(y_n + z_n) \to 0$ one has $(y_n) \to 0$. It is known that if Y is the dual of a Banach space X and if Q is the dual cone

$$P^* := \{x^* \in X^* : \forall\, x \in P \;\; x^*(x) \geq 0\}$$

of closed convex cone P in X, then Q is normal in Y if and only if P is generating in X (in the sense that $P - P = X$); cf. [32, Proposition 3.5 and Theorem 5.16], [44], [48]. Let us also recall that the (pre)order induced by P is defined by $x \leq x'$ if and only if $x' - x \in P$.

Proposition 4.7. *Let P be a closed convex cone in X such that the dual cone P^* is normal (equivalently, such that P is generating in X). Let ∂ be a subdifferential on X contained in the Clarke subdifferential ∂^0 or in the infradifferential $\partial^{\leq}$. Let g, h be l.s.c. functions on X that are nondecreasing for the order induced by P. Suppose g and h satisfy the fuzzy sum rule for ∂. Then, if g and h are critical, their sum $f := g + h$ is critical.*

Proof. We first note that for each $z \in X$ and each $z^* \in \partial h(z)$ we have $z^* \in P^*$. When $\partial h(z) \subset \partial^{\leq} h(z)$ this follows from the fact that for each $w \in -P$ we have

$$\langle z^*, w\rangle \leq h(z+w) - h(z) \leq 0.$$

When $\partial h(z) \subset \partial^0 h(z)$ this follows from the fact that for each $w \in -P$ we have

$$\langle z^*, w\rangle \leq h^0(z,w) \leq 0.$$

By assumption, we note that, if (x_n) is a critical sequence of $f = g + h$, and if $(x_n^*) \to 0$ is such that $x_n^* \in \partial f(x_n)$ for each n, then we can find $y_n,\ z_n \in X$ with $(g(y_n) - g(x_n)) \to 0$, $(h(z_n) - h(x_n)) \to 0$, $y_n^* \in \partial g(y_n)$, and $z_n^* \in \partial h(z_n)$ such that $x_n^* = w_n^* + y_n^* + z_n^*$ for some sequence $(w_n^*) \to 0$. Then, with the order induced by P^*,

$$0 \leq y_n^* = x_n^* - w_n^* - z_n^* \leq x_n^* - w_n^* \to 0,$$

hence $(y_n^*) \to 0$, so that (y_n) is a critical sequence of g. Similarly (z_n) is a critical sequence of h. Since g and h are critical, there exist real numbers $c_g,\ c_h$ such that $(g(y_n)) \to c_g,\ (h(z_n)) \to c_h$. Thus $(f(x_n)) \to c_f := c_g + c_h$, and f is critical. □

Taking successively the class of differentiable functions and the class of closed convex functions we get the following consequences.

COROLLARY 4.8. *Let X be an arbitrary Banach space. Let P be a generating closed convex cone in X. Let g, h be differentiable functions on X that are nondecreasing for the order induced by P. Then, if g and h are critical, their sum $f := g+h$ is critical.*

COROLLARY 4.9. *Let X be an arbitrary Banach space. Let P be a generating closed convex cone in X. Let g, h be nondecreasing functions on X and g be differentiable in the sense of Fréchet (resp., Hadamard). Then, if g and h are critical, their sum $f := g + h$ is critical for the Fréchet (resp., Hadamard) subdifferential.*

Proof. The result stems from the relation $\partial f(x) = g'(x) + \partial h(x)$, which is valid under our assumptions. It would hold for any subdifferential ∂ for which this relation is satisfied. □

COROLLARY 4.10. *Let X be a reflexive Banach space. Let P be a generating closed convex cone in X. Let g, h be closed convex functions on X which are nondecreasing for the order induced by P. Then, if g and h are critical, their sum $f := g + h$ is critical.*

Proof. In such a case the fuzzy sum rule is satisfied (see [38] for a recent contribution). □

5. Critically convex functions. In the present section we introduce a class of functions for which the characterization of well behavior in terms of sublevel sets can be extended.

DEFINITION 5.1. *A function f from X into $\mathbb{R} \cup \{+\infty\}$ is said to be critically convex, in short C-convex, if it satisfies the following property: for any pair of critical sequences $(x_n), (y_n)$ with $x_n \neq y_n$ one has*

$$\lim_{n\to\infty} \frac{|f(x_n) - f(y_n)|}{\|x_n - y_n\|} = 0. \tag{5.1}$$

Again, for the Fréchet subdifferential, any distance function is critically convex. (It may not be so for other subdifferentials.) Other important examples of critically convex functions are given in the following lemmas.

Let us first note obvious relationships with the class of critical functions.

PROPOSITION 5.2. (a) *If f is of class C^1 and critical and if any critical sequence has a cluster point (in particular, if X is finite dimensional and if any critical sequence is bounded), then f is C-convex.*

(b) *If f is C-convex, if f has at least one critical point, and if any critical sequence of f is bounded, then f is critical.*

Proof. (a) Let $(x_n), (y_n)$ with $x_n \neq y_n$ be a pair of critical sequences of f. Taking subsequences if necessary, we may assume that (x_n) and (y_n) converge to some x and y, respectively. When $x \neq y$, since $(f(x_n))$ and $(f(y_n))$ have the same limit, condition (5.1) is satisfied. When $x = y$, the mean value theorem yields some $z_n \in [x_n, y_n]$ such that $|f(x_n) - f(y_n)| = |f'(z_n)(x_n - y_n)| \leq \|f'(z_n)\| \, \|x_n - y_n\|$. Since $(z_n) \to x$ and since x is a critical point, we have $\|f'(z_n)\| \to 0$, and condition (5.1) is satisfied.

(b) Let (x_n) be a critical sequence of f and let z be a critical point of f. Let us prove that any subsequence $(x_i)_{i\in I}$ of (x_n) contains a subsequence $(x_k)_{k\in K}$ such that $(f(x_k))_{k\in K}$ converges to $f(z)$. That will show that $(f(x_n))$ converges to $f(z)$. The conclusion is obvious if $J := \{j \in I : x_j = z\}$ is infinite. Otherwise we take $K = I \backslash J$ so that we have $x_k \neq z$ for $k \in K$. Then, denoting by r the radius of a ball centered at z containing the critical sequence (x_n) and taking for (y_n) the constant sequence with value z, we have

$$\mid f(x_k) - f(z) \mid \leq \varepsilon_k r$$

for some sequence (ε_k) with limit 0. Thus $(f(x_k))_{k \in K}$ converges to $f(z)$ and the whole sequence converges to $f(z)$. □

A proof similar to the preceding one establishes the following variant.

Proposition 5.3. *If f is C-convex and has a critical sequence (z_n) whose values are bounded, and if any critical sequence of f is bounded, then f is critical.*

Proof. In the preceding proof one replaces $f(z)$ by $f(z_k)$, where $(f(z_k))$ is a subsequence of $(f(z_n))$ which converges in $\mathbb{R}$. □

The following result partially justifies the terminology we adopt.

Lemma 5.4. *If f is convex and ∂f is the subdifferential of convex analysis, then f is C-convex.*

In fact, the proof below shows that for the Fenchel subdifferential, any function is C-convex. However, this subdifferential is essentially adapted to the class of convex functions. A similar observation holds for the next lemma.

Proof. Let $(x_n), (y_n)$ be two critical sequences with $x_n \neq y_n$. We choose $x_n^* \in \partial f(x_n)$, $y_n^* \in \partial f(y_n)$ with $(x_n^*) \to 0$ and $(y_n^*) \to 0$. Then, as ∂f is the subdifferential of convex analysis, one has

$$y_n^*(x_n - y_n) \leq f(x_n) - f(y_n) \leq x_n^*(x_n - y_n)$$

so that, setting $r \vee s = \max(r, s)$, one gets

$$|f(x_n) - f(y_n)| \leq (\|x_n^*\| \vee \|y_n^*\|) \|x_n - y_n\| .$$

Thus, $\lim_{n \to \infty} \|x_n - y_n\|^{-1} |f(x_n) - f(y_n)| = 0$. □

Lemma 5.5. *If f is quasi-convex and if ∂ is the lower subdifferential $\partial^<$ of Plastria, then f is C-convex.*

Proof. Let $(x_n), (y_n)$ be two critical sequences with $x_n \neq y_n$ and let $x_n^* \in \partial f(x_n)$, $y_n^* \in \partial f(y_n)$ with $(x_n^*) \to 0$ and $(y_n^*) \to 0$. Without loss of generality, we may suppose $f(x_n) < f(y_n)$. Then the definition of the lower subdifferential $\partial^<$ yields $f(x_n) - f(y_n) \geq y_n^*(x_n - y_n)$, hence

$$- \|y_n^*\| \|x_n - y_n\| \leq y_n^*(x_n - y_n) \leq f(x_n) - f(y_n) < 0,$$

so that $\lim_{n \to \infty} \|x_n - y_n\|^{-1} |f(x_n) - f(y_n)| = 0$. □

The following example makes a link with the notion of ∂-invexity considered in the preceding section.

Example 5.1. Let f be a ∂-invex function on an open subset W of X with associated mapping $v : f(w) - f(x) \geq \langle x^*, v(w, x) \rangle$ for each $(w, x) \in X^2$ and each $x^* \in \partial f(x)$. Suppose there exists a constant $k > 0$ such that $\|v(w, x)\| \leq k \|w - x\|$ for each $(w, x) \in X^2$; this condition is obviously satisfied in the convex case for which one takes $v(w, x) = w - x$. Then f is C-convex.

Whenever ∂ satisfies $\partial(-f)(x) = -\partial f(x)$ for f locally Lipschitzian (this is the case for the Clarke subdifferential and for the moderate subdifferential of Michel and Penot [30], [31]), any continuous concave function is also C-convex. Note that whenever the subdifferential reduces to the ordinary derivative for a function of class C^1 one can exhibit C-convex functions that are neither convex nor concave, such as the one-variable functions $x \mapsto x^k$ $(k \geq 1)$ or $x \mapsto x^k \mid x \mid^s$ for k a positive even integer and $s \in [0, 1[$. These examples are special cases of the following criterion. (Note that for a polynomial function of several variables, coercivity can be ensured by requiring that the higher order term is coercive; for the square of a one-variable polynomial of positive degree, this condition is automatic.)

Example 5.2. Let X be finite dimensional and let $f : X \to \mathbb{R}$ be of class C^1 and such that $x \to \|f'(x)\|$ is coercive (or semicoercive in the sense of [39] that $\liminf_{\|x\|\to\infty} \|f'(x)\| > \inf_{x\in X} \|f'(x)\|$). Then, if f has at most one critical value, it is C-convex. This criterion follows from Proposition 5.2 and the fact that f is critical.

Note that the nonsmooth nonconvex function $x \mapsto || \, x \mid -1|$ is also C-convex.

The following class of examples is important too.

LEMMA 5.6. *Any quadratic function is C-convex.*

The derivative of a quadratic function being affine, the result is a consequence of the following lemma in which the segment with end points x, y is denoted by $[x, y]$.

LEMMA 5.7. *Let f be defined and differentiable on an open convex subset and such that the following property is satisfied: if (x_n) and (y_n) are critical sequences, then any sequence (z_n) such that $z_n \in [x_n, y_n]$ is critical. Then f is C-convex.*

Proof. The result follows from the mean value theorem: if (x_n) and (y_n) are critical sequences, then for some sequence (z_n) such that $z_n \in [x_n, y_n]$ one has

$$\mid f(x_n) - f(y_n) \mid = \mid f'(z_n)(x_n - y_n) \mid \leq \|f'(z_n)\| \|x_n - y_n\|$$

and $(f'(z_n)) \to 0$. □

The example of the exponential function on $\mathbb{R}$ shows that nonquadratic functions may satisfy the preceding condition. However, it is convex, but the functions f, g, h given by $f(x) = \ln x$, $g(x) = 1/x$, and $h(x) = e^x(\sin x + \cos x + 3)$ are examples of functions satisfying the preceding condition without being convex or quadratic.

Let us present some stability results for the class of C-convex functions that are similar to the results for critical functions but often require reinforced assumptions.

PROPOSITION 5.8. *Let P be a generating closed convex cone in X. Let ∂ be a subdifferential on X contained in the Clarke subdifferential ∂^0 or in the infradifferential $\partial^{\leq}$. Let g, h be functions on X that are nondecreasing for the order induced by P and satisfy the fuzzy sum rule. Then, if g and h are C-convex, their sum $f := g + h$ is C-convex.*

Proof. Let (x_n) and (x'_n) be critical sequences of $f = g + h$ such that $r_n := \|x_n - x'_n\| > 0$ for each n. The proof of Proposition 4.7 shows that we can find sequences (y_n), (z_n) that are critical for g and h, respectively, and are such that

$$\|y_n - x_n\| \leq 2^{-n} r_n, \quad \|z_n - x_n\| \leq 2^{-n} r_n,$$

$$\mid g(y_n) - g(x_n) \mid \leq 2^{-n} r_n, \quad \mid h(z_n) - h(x_n) \mid \leq 2^{-n} r_n,$$

and sequences (y'_n) and (z'_n) which have similar properties with (x_n) replaced by (x'_n). Then

$$\|x_n - x'_n\| \geq \|y_n - y'_n\| - 2^{-n+1} r_n$$

and $(1 + 2^{-n+1}) r_n \geq \|y_n - y'_n\|$, and similarly $r_n^{-1} \leq (1 + 2^{-n+1}) \|z_n - z'_n\|^{-1}$, so that

$$\begin{aligned} & r_n^{-1} \mid f(x_n) - f(x'_n) \mid \\ & \leq (1 + 2^{-n+1})(\|y_n - y'_n\|^{-1} \mid g(y_n) - g(y'_n) \mid + \|z_n - z'_n\|^{-1} \mid h(z_n) - h(z'_n) \mid) + 2^{-n+1}, \end{aligned}$$

hence $r_n^{-1}(f(x_n) - f(x'_n)) \to 0$ as g and h are C-convex. □

Similarly, we have a stability property for a sum of separable functions.

Proposition 5.9. *Let $X := Y \times Z$ be the product of two Banach spaces and let g, h be l.s.c. C-convex functions on Y and Z, respectively. Let $f := \overline{g} + \overline{h}$, where $\overline{g}$ is the function given by $\overline{g}(y, z) = g(y)$ and $\overline{h}(y, z) = h(z)$. Suppose $\partial\overline{g}(y, z) \subset \partial g(y) \times \{0\}$, $\partial\overline{h}(y, z) \subset \{0\} \times \partial h(z)$, and the fuzzy sum rule is satisfied by ∂ for these functions. Then f is C-convex.*

Proposition 5.10. *Suppose ∂ satisfies the fuzzy composition rule. Let $h : X \to Y$ be a differentiable map between two Banach spaces which is Lipschitzian with rate ℓ and such that for some $c > 0$ one has* $\operatorname{open}(h'(x)) < c$ *for each $x \in X$. Then, for any C-convex function g on Y, the function $f := g \circ h$ is C-convex.*

Proof. Let (x_n) and (x'_n) be critical sequences of $f = g \circ h$ such that $r_n := \|x_n - x'_n\| > 0$ for each n. If $s_n := \|h(x_n) - h(x'_n)\|$ is 0 for n in an infinite subset N of $\mathbb{N}$, then $r_n^{-1} \mid f(x_n) - f(x'_n) \mid = 0$ for each $n \in \mathbb{N}$. Thus, without loss of generality, we may suppose that $s_n > 0$ for each $n \in \mathbb{N}$. The fuzzy composition rule and the proof of Proposition 4.4 show that there exist critical sequences (y_n), (y'_n) for g such that

$$\|y_n - h(x_n)\| \leq 2^{-n} s_n, \quad \|y'_n - h(x'_n)\| \leq 2^{-n} s_n,$$

$$\mid g(y_n) - g(h(x_n)) \mid \leq 2^{-n} s_n, \quad \mid g(y'_n) - g(h(x'_n)) \mid \leq 2^{-n} s_n.$$

Thus

$$\frac{\mid f(x_n) - f(x'_n) \mid}{\|x_n - x'_n\|} \leq \ell \frac{\mid g(y_n) - g(y'_n) \mid}{\|y_n - y'_n\|} + 2^{-n+1}\ell \to 0$$

as g is C-convex. □

Proposition 5.11. *Let $f = h \circ g$, where $g : X \to \mathbb{R}$ is C-convex and $h : \mathbb{R} \to \mathbb{R}$ is differentiable and Lipschitzian. Suppose there exists $a > 0$ such that $h'(r) \geq a$ for each $r \in \mathbb{R}$ and suppose $\partial f(x) \subset h'(g(x))\partial g(x)$ for each $x \in X$. Then f is C-convex.*

Proof. Let (x_n), (x'_n) be critical sequences for f. The proof of Proposition 4.5 shows that these sequences are critical for g. If ℓ is the Lipschitz rate of h we get

$$\frac{\mid f(x_n) - f(x'_n) \mid}{\|x_n - x'_n\|} \leq \ell \frac{\mid g(x_n) - g(x'_n) \mid}{\|x_n - x'_n\|} \to 0$$

as g is C-convex. □

In several cases of interest, critical sequences are bounded or their values are bounded. In such cases, the following variants of C-convexity coincide with the genuine notion of C-convexity. In turn, these variants will serve to get verifiable criteria.

Definition 5.12. *The function f is said to be boundedly critically convex (BC-convex) if it satisfies the following property: for any pair of bounded critical sequences $(x_n), (y_n)$ with $x_n \neq y_n$ one has*

$$\lim_{n \to \infty} \frac{|f(x_n) - f(y_n)|}{\|x_n - y_n\|} = 0. \tag{5.2}$$

The class of BC-convex functions being larger than the class of C-convex functions, we get that any convex function and any quadratic function is BC-convex for any subdifferential that coincides with the Fenchel–Moreau subdifferential on the class of convex functions and that reduces to the derivative when the function is of class C^1. Moreover, any function whose critical sequences are unbounded is BC-convex. (The exponential function on $\mathbb{R}$ is an example of such a function.) On the other hand, one-variable polynomial functions are C-convex if and only if they are BC-convex (since their critical sequences are bounded).

PROPOSITION 5.13. *Suppose X is reflexive and ∂-reliable and suppose f is l.s.c. and satisfies the following assumptions:*

(a) *f is constant on the set Z of its critical points;*

(b) *if (z_n) is a critical sequence weakly converging to some z, then $z \in Z$ and $(f(z_n)) \to f(z)$;*

(c) *for any critical point z, any sequence (z_n) weakly converging to z and any $z_n^* \in \partial f(z_n)$, one has $(z_n^*) \to 0$.*

Then f is BC-convex.

Proof. Suppose on the contrary that there exist bounded critical sequences (x_n) and (y_n) and $c > 0$ such that

$$| f(x_n) - f(y_n) | \geq c\|x_n - y_n\| > 0 \tag{5.3}$$

for each n. Without loss of generality we may suppose $f(x_n) - f(y_n) > 0$. Taking subsequences if necessary, we may suppose (x_n) and (y_n) converge weakly. By (b) their respective limits x and y belong to Z. Conditions (a) and (b) ensure that $(f(x_n) - f(y_n)) \to 0$, so that relation (5.3) implies that $x = y := z$. Let (δ_n) be a sequence of positive numbers with limit 0. Using the mean value theorem of [34], which, among other ones [6], [29], is adapted to the present setting, for each n we can find $z_n \in X$, $z_n' \in [x_n, y_n]$, and $z_n^* \in \partial f(z_n)$ such that $\|z_n - z_n'\| < \delta_n$, $| f(z_n) - f(z_n') | < \delta_n$, and

$$z_n^*(x_n - y_n) \geq f(x_n) - f(y_n) - \delta_n\|x_n - y_n\|.$$

Then (z_n) converges weakly to z and we have

$$\|z_n^*\|\|x_n - y_n\| \geq c\|x_n - y_n\| - \delta_n\|x_n - y_n\|,$$

a contradiction with condition (c). □

Clearly, condition (a) is necessary for f to be BC-convex. Let us note that condition (c) implies that any critical point z is continuously critical in the sense that $\partial f(z) = \{0\}$ and $\partial f(\cdot)$ is upper semicontinuous for the weak topology on X, a rather stringent assumption. This assumption is satisfied when f is of class C^1 and X is finite dimensional.

COROLLARY 5.14. *Suppose X is finite dimensional and f is of class C^1 and is constant on the set of its critical points. Then f is BC-convex.*

A variant of the preceding proposition deals with C-convexity.

PROPOSITION 5.15. *Suppose X is ∂-reliable and suppose f is l.s.c. and satisfies the following assumptions:*

(a) *f is BC-convex;*

(b) *if (z_n) is a critical sequence, then $(f(z_n))$ is bounded*;

(c) *for any critical sequence (w_n) such that $(\|w_n\|) \to \infty$ and for any $z_n \in X$, $z_n^* \in \partial f(z_n)$ such that $(z_n - w_n)$ is bounded, one has $(z_n^*) \to 0$.*

Then f is C-convex.

Proof. Suppose on the contrary that there exist critical sequences (x_n) and (y_n) and $c > 0$ such that

$$| f(x_n) - f(y_n) | \geq c\|x_n - y_n\| > 0 \quad \text{for each } n.$$

Without loss of generality, since f is BC-convex, we may suppose $(\|x_n\|), (\|y_n\|) \to \infty$ and $f(x_n) - f(y_n) > 0$. Again using the mean value theorem, given a sequence (δ_n) of positive numbers with limit 0, for each n we can find $z_n \in X$, $z'_n \in [x_n, y_n]$, and $z^*_n \in \partial f(z_n)$ such that $\|z_n - z'_n\| < \delta_n$, $| f(z_n) - f(z'_n) | < \delta_n$, and

$$z^*_n(x_n - y_n) \geq f(x_n) - f(y_n) - \delta_n\|x_n - y_n\|.$$

Then, as $(f(x_n))$ and $(f(y_n))$ are bounded by assumption (b), $(x_n - y_n)$ is bounded and thus $(z_n - x_n)$ is bounded too. It follows from (c) with $w_n = x_n$ that $(z^*_n) \to 0$, and we have

$$\|z^*_n\|\|x_n - y_n\| \geq c\|x_n - y_n\| - \delta_n\|x_n - y_n\|,$$

a contradiction. □

6. Characterizations of minimizing sequences in terms of sublevel sets.

In the present section we relate minimizing sequences and the distance to the sublevel sets of a function. We do not intend to give results of practical interest for algorithms. We just explore the role of convexity in results that deal with crucial questions about minimizing sequences and error estimates. For $\lambda \in \mathbb{R}$, we use $L(\lambda)$ or $L_f(\lambda)$ to denote the λ-sublevel set of f, that is,

$$L(\lambda) := \{x : \ f(x) \leq \lambda\}.$$

Since f is assumed to be l.s.c., each λ-sublevel set is closed. The following result provides an answer to the question, When is a critical sequence minimizing?

PROPOSITION 6.1. *Let X be a ∂-variational Banach space and let f be a C-convex l.s.c. function on X (to $\mathbb{R} \cup \{+\infty\}$) which is bounded below. Let (x_n) be a critical sequence. Then (x_n) is a minimizing sequence if and only if there exists a sequence $(\lambda_n) \searrow \inf f$ such that*

$$\sup_{n \in N} \operatorname{dist}(x_n, L(\lambda_n)) < +\infty. \tag{6.1}$$

Note that in this statement we cannot change "there exists" into "for any" as shown by the example $f(x) = e^{-x}$, $x_n = n$, $\lambda_n = e^{-2n}$. Also note that the assumption that f is C-convex is crucial: for $f(x) = \cos(2\pi x^2)$, $x_n = n$, $\lambda_n = -1$, one has $\operatorname{dist}(x_n, L(\lambda_n)) \leq d(x_n, y_n) \to 0$, where $y_n := (n^2 + \frac{1}{2})^{1/2}$ but (x_n) is not minimizing.

Proof. As necessity is obtained by taking $\lambda_n = f(x_n)$, it is enough to show sufficiency. Let (x_n) be a critical sequence such that (6.1) holds for some $(\lambda_n) \searrow \inf f$. Take $y_n \in L(\lambda_n)$ such that

$$\|x_n - y_n\| < \operatorname{dist}(x_n, L(\lambda_n)) + \frac{1}{n}. \tag{6.2}$$

Then $f(y_n) \leq \lambda_n \to \inf f$, that is, (y_n) is minimizing. It follows from Theorem 3.2 that there exist sequences (w_n) and (w^*_n) such that

(i) (w_n) is minimizing;
(ii) $w^*_n \in \partial f(w_n)$, $(w^*_n) \to 0$;
(iii) $\|y_n - w_n\| \to 0$.

It follows from (5.1) that there exists some $(\delta_n) \downarrow 0$ with

$$|f(x_n) - f(w_n)| \leq \delta_n \, \|x_n - w_n\| \quad \text{for each } n$$

(considering separately the case $x_n = w_n$ and the case $x_n \neq w_n$). Then

$$|f(x_n) - f(w_n)| \leq \delta_n(\|x_n - y_n\| + \|y_n - w_n\|) \to 0$$

as $n \to \infty$. Thus (x_n) is minimizing because (w_n) is so by (i). □

Let us turn now to criteria for a sequence to be minimizing which may be considered as valid for convex functions only. Recall that the ϵ-approximate subdifferential of a function $f : X \to \mathbb{R} \cup \{+\infty\}$ at x is defined by

$$\partial_\epsilon f(x) := \{x^* \in X^* : \; x^*(y - x) \leq f(y) - f(x) + \epsilon, \;\; \forall y \in X\}.$$

In general, this notion is used only for convex functions. However, it has a meaning for any function. For instance, if (x_n) is a minimizing sequence of an arbitrary function f, then there exists a sequence $(\varepsilon_n) \to 0$ such that $0 \in \partial_{\varepsilon_n} f(x_n)$ for each n. Let us note that this notion is a global one and not a local one.

Lemma 6.2. *Let X be a Banach space and let $f : X \to \mathbb{R} \cup \{+\infty\}$ be a function. Let $(\epsilon_n), (x_n), (x_n^*)$ be sequences such that $\epsilon_n \geq 0$, $(\epsilon_n) \to 0$, $x_n^* \in \partial_{\epsilon_n} f(x_n)$, and $(x_n^*) \to 0$. Then (x_n) is a minimizing sequence for f if and only if for any $\lambda > \inf f$, one has*

$$\sup_{n \in N} \operatorname{dist}(x_n, L(\lambda)) < +\infty. \tag{6.3}$$

Proof. It suffices to prove sufficiency. Assume by way of contradiction that there exist scalars $\alpha > \beta > \inf f$ and a subsequence (x_{n_k}) of (x_n) such that $f(x_{n_k}) > \alpha$ for all k. Take $y_k \in L(\beta)$ such that

$$\|x_{n_k} - y_k\| < \operatorname{dist}(x_{n_k}, L(\beta)) + \frac{1}{k}. \tag{6.4}$$

By the definition of $\partial_\epsilon f$, one has that

$$\begin{aligned} \left\|x_{n_k}^*\right\| \|x_{n_k} - y_k\| &\geq x_{n_k}^*(x_{n_k} - y_k) \\ &\geq f(x_{n_k}) - f(y_k) - \epsilon_{n_k} \\ &> \alpha - \beta - \epsilon_{n_k} \end{aligned} \tag{6.5}$$

for large k. This is not possible as $(\epsilon_{n_k}) \to 0$, $(\left\|x_{n_k}^*\right\|) \to 0$, and $(\|x_{n_k} - y_k\|)$ is bounded by (6.4) and (6.3). □

The following result establishes the equivalence between the conditions (6.1), (6.3) and a condition of Auslender and Crouzeix for critical sequences of a convex function. Here f is an arbitrary function.

Proposition 6.3. *Let X be a Banach space and let $f : X \to \mathbb{R} \cup \{+\infty\}$ be a l.s.c. function. Let $(\epsilon_n), (x_n), (x_n^*)$ be sequences such that $\epsilon_n \geq 0$, $(\epsilon_n) \to 0$, $x_n^* \in \partial_{\epsilon_n} f(x_n)$, and $(x_n^*) \to 0$. Then the following conditions are equivalent:*

(a) *there exists a sequence $(\lambda_n) \to \inf f$ such that*

$$\sup_{n \in N} \operatorname{dist}(x_n, L(\lambda_n)) < +\infty;$$

(b) *for any $\lambda > \inf f$,*

$$\sup_{n \in N} \operatorname{dist}(x_n, L(\lambda)) < +\infty;$$

(c) *for any* $\lambda > \inf f$, *there exists* $c > 0$ *such that for all* n,

$$\operatorname{dist}(x_n, L(\lambda)) \leq c(f(x_n) - \lambda)_+; \tag{6.6}$$

(d) *for any* $\lambda > \inf f$, *there exists* $h : \mathbb{R}_+ \to \mathbb{R}_+$ *locally bounded such that* $\limsup_{r\to\infty} r^{-1}h(r) < \infty$ *and for all* n,

$$\operatorname{dist}(x_n, L(\lambda)) \leq h(f(x_n) - \lambda); \tag{6.7}$$

(e) (x_n) *is a minimizing sequence of* f.

Proof. By Lemma 6.2, (b) and (e) are equivalent. As (a) obviously implies (b) and is satisfied with $\lambda_n = f(x_n)$ when (e) holds, the equivalence of (a) and (e) follows. If $\lambda > \inf f$ and (x_n) is minimizing, then $x_n \in L(\lambda)$ for large n. Hence (e) implies (c). Clearly, (c) implies (d) as $h(r) := r_+ := \max(r, 0)$ is a special case of the assumption of (d). Finally, we show that (d) implies (b). Assume by way of contradiction that there exists $\lambda_0 > \inf f$ such that

$$\sup_{n\in N} \operatorname{dist}(x_n, L(\lambda_0)) = +\infty. \tag{6.8}$$

By (d), take h satisfying (6.7) with $\lambda = \lambda_0$. By (6.7), (6.8), considering a subsequence if necessary, we may assume that

$$\lim_{n\to\infty} f(x_n) = +\infty. \tag{6.9}$$

Take $y_n \in L(\lambda_0)$ such that $\|x_n - y_n\| \leq \operatorname{dist}(x_n, L(\lambda_0)) + \frac{1}{n}$. It follows that

$$\begin{array}{rcl} \|x_n^*\|\, h(f(x_n) - \lambda_0) & \geq & \|x_n^*\|\,(\|x_n - y_n\| - \frac{1}{n}) \\ & \geq & x_n^*(x_n - y_n) - \frac{1}{n}\,\|x_n^*\| \\ & \geq & f(x_n) - f(y_n) - \epsilon_n - \frac{1}{n}\,\|x_n^*\| \\ & \geq & f(x_n) - \lambda_0 - \epsilon_n - \frac{1}{n}\,\|x_n^*\|, \end{array}$$

where the third inequality holds because $x_n^* \in \partial_{\epsilon_n} f(x_n)$. Dividing by the positive number $f(x_n) - \lambda_0$ (with n large), we have

$$\|x_n^*\| \limsup_n (f(x_n) - \lambda_0)^{-1} h(f(x_n) - \lambda_0) \geq 1 - \frac{n\epsilon_n + \|x_n^*\|}{n(f(x_n) - \lambda_0)},$$

which is impossible as the left-hand side converges to zero and the right-hand side converges to 1 by (6.9) and the assumption that $(\|x_n^*\|) \to 0$. □

The conditions in (c) and (d) are error bounds conditions. We refer to [27], [11], and [33] and their bibliographies for more information about the significance and the uses of such estimates.

7. Links with well-set problems. The following notion of well-set function was introduced in [8] (see also [7], [42]) as a modification of the famous notion of Tychonov well-posed problems (see [14], [45]). It takes into account the fact that the set of minimizers of a function f may contain more than one point while the minimization problem of f can be considered as an easy, gentle problem.

DEFINITION 7.1. *A function* f *on* X *is said to be metrically well-set (M-well-set) if its set of minimizers* S *is nonempty and if for any minimizing sequence* (x_n) *one has* $(\operatorname{dist}(x_n, S)) \to 0$.

It is easy to see [35] that f is M-well-set if and only if there exists a modulus μ (i.e., a one-variable nondecreasing function that has limit 0 at 0) such that $d(x, S) \leq \mu(f(x) - \inf f)$.

The following variant was introduced by Lemaire [26] (essentially in the convex case).

DEFINITION 7.2. *A function f on X is said to be very well behaved if its set of minimizers S is nonempty and if for any critical sequence (x_n) one has* $(\operatorname{dist}(x_n, S)) \rightarrow 0$. *We denote by $\mathcal{V}(X)$ the set of very well behaved functions on X.*

The following propositions describe some relationships between these two notions.

PROPOSITION 7.3. *Suppose ∂ is variational for X and $\mathcal{F}(X)$. If f is l.s.c., bounded below, and very well behaved, then f is M-well-set.*

Proof. Let $f \in \mathcal{V}(X)$ be l.s.c. and bounded below. Given a minimizing sequence (x_n) of f, we can find a sequence (w_n) that is critical and such that $(d(w_n, x_n)) \rightarrow 0$. Since $f \in \mathcal{V}(X)$ we have $(d(w_n, S)) \rightarrow 0$. Thus $(d(x_n, S)) \rightarrow 0$. □

A partial converse is as follows.

PROPOSITION 7.4. *If a critical function f is M-well-set, then it is very well behaved.*

Proof. Let f be a critical function that is M-well-set and let (x_n) be a critical sequence of f. Since any minimizer z of f (and the set S of such points is nonempty) is critical, we have $(f(x_n)) \rightarrow \min f$. Thus (x_n) is minimizing, and as f is M-well-set we have $(d(x_n, S)) \rightarrow 0$. □

COROLLARY 7.5. *If a well-behaved function f is M-well-set, then it is very well behaved.*

The following observation completes the preceding results and partially justifies the terminology.

PROPOSITION 7.6. *If f is very well behaved and if f is uniformly continuous around S, then f is critical and well behaved.*

Proof. Let $f \in \mathcal{V}(X)$. The uniform continuity of our assumption means that for each $\varepsilon > 0$ there exists $\delta > 0$ such that for each $x \in S$ and for each $w \in X$ satisfying $d(w, x) < \delta$ one has $\mid f(w) - f(x) \mid < \varepsilon$. It implies that for each critical sequence (x_n) we have $(f(x_n)) \rightarrow \min f$ since $(d(x_n, S)) \rightarrow 0$ as f is in $\mathcal{V}(X)$. □

In particular, for a variational subdifferential and for a Lipschitzian function f whose set of minimizers S is nonempty, one has that f is very well behaved if and only if f is well behaved and M-well-set.

Acknowledgments. We are grateful to two anonymous referees for their criticisms, which led to an improved presentation of the results. The second author wishes to thank Professor Szymon Dolecki for helpful discussions. The third author acknowledges the support and the hospitality of The Chinese University of Hong Kong and the help of the French Consulate in Hong Kong.

REFERENCES

[1] P. ANGLERAUD, *Caractérisation duale du bon comportement de fonctions convexes*, C.R. Acad. Sci. Paris Sér. I Math., 314 (1992), pp. 583–586.

[2] A. AUSLENDER, *Convergence of critical sequences for variational inequalities with maximal monotone operators*, Appl. Math. Optim., 28 (1993), pp. 161–172.

[3] A. AUSLENDER, *How to deal with the unbounded in optimization: Theory and algorithms*, Math. Programming, 79 (1997), pp. 3–18.

[4] A. AUSLENDER AND J.-P. CROUZEIX, *Well behaved asymptotical convex functions*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 6 (1989), pp. 101–121.

[5] A. Auslender, R. Cominetti, and J.-P. Crouzeix, *Convex functions with unbounded level sets and applications to duality theory*, SIAM J. Optim., 3 (1993), pp. 669–687.

[6] D. Aussel, J.-N. Corvellec, and M. Lassonde, *Mean value property, and subdifferential criteria for l.s.c. functions*, Trans. Amer. Math. Soc., 347 (1995), pp. 4147–4161.

[7] E. Bednarczuk and J.-P. Penot, *On the position of the notions of well-posed minimization problems*, Bollet. Un. Mat. Ital. B (7), 6 (1992), pp. 665–683.

[8] E. Bednarczuk and J.-P. Penot, *Metrically well-set minimization problems*, Appl. Math. Optim., 26 (1992), pp. 273–285.

[9] J. M. Borwein and J. R. Giles, *The proximal normal formula in Banach space*, Trans. Amer. Math. Soc., 302 (1987), pp. 371–381.

[10] A. Brøndsted and R. T. Rockafellar, *On the subdifferentiability of convex functions*, Proc. Amer. Math. Soc., 16 (1965), pp. 605–611.

[11] C.-C. Chou, K.-F. Ng, and J.-S. Pang, *Minimizing and stationary sequences of constrained optimization problems*, SIAM J. Control Optim., 36 (1998), pp. 1908–1936.

[12] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley-Interscience, New York, 1983.

[13] R. Deville, G. Godefroy, and V. Zizler, *A smooth variational principle with applications to Hamilton–Jacobi equations in infinite dimensions*, J. Funct. Anal., 111 (1993), pp. 197–212.

[14] A. L. Dontchev and T. Zolezzi, *Well-Posed Optimization Problems*, Lecture Notes in Math. 1543, Springer-Verlag, Berlin, 1991.

[15] I. Ekeland, *Nonconvex minimization problems*, Bull. Amer. Math. Soc., 1 (1979), pp. 443–474.

[16] E. W. Hobson, *The Theory of Functions of a Real Variable and the Theory of Fourier's Series,* Vol. II, Dover, New York, 1957.

[17] L. R. Huang and X. B. Li, *Minimizing Sequences in Nonsmooth Optimization*, preprint.

[18] L. R. Huang and K. F. Ng, *Second-order necessary and sufficient conditions in nonsmooth optimization*, Math. Programming, 66 (1994), pp. 379–402.

[19] L. R. Huang and K. F. Ng, *On second order directional derivatives in nonsmooth optimization*, in Recent Advances in Nonsmooth Optimization, D. Z. Du, L. Qi, and R. S. Womersley, eds., World Scientific, Singapore, 1995, pp. 159–171.

[20] A. D. Ioffe, *On subdifferentiability spaces*, Ann. New York Acad. Sci., 410 (1983), pp. 107–119.

[21] A. D. Ioffe, *Calculus of Dini subdifferentials of functions and contingent coderivatives of set-valued maps*, Nonlinear Anal., 8 (1984), pp. 517–539.

[22] A. D. Ioffe, *On the local surjection property,* Nonlinear Anal., 11 (1987), pp. 565–592.

[23] A. D. Ioffe, *Codirectional compactness, metric regularity and subdifferential calculus*, in Constructive, Experimental and Nonlinear Analysis, M. Théra, ed., Canad. Math. Soc. Ser. Monogr. Adv. Texts, to appear.

[24] A. D. Ioffe and J.-P. Penot, *Subdifferentials of performance functions and calculus of coderivatives of set-valued mappings*, Serdica Math. J., 22 (1996), pp. 359–384.

[25] D. Klatte, *Error bounds for solutions of linear equations and inequalities*, Math. Methods Oper. Res., 41 (1995), pp. 191–214.

[26] B. Lemaire, *Bonne position, conditionnement, et bon comportement asymptotique*, Sém. Anal. Convexe, 22 (1992), pp. 5.1–5.12.

[27] A. S. Lewis and J.-S. Pang, *Error bounds for convex inequality systems*, in Generalized Convexity, Generalized Monotonicity, J.-P. Crouzeix, J.-E. Martinez, and M. Volle, eds., Kluwer, Dordrecht, the Netherlands, 1998, pp. 75–110.

[28] Y. Li and S. Shi, *A generalization of Ekeland's ϵ-variational principle and of its Borwein-Preiss' smooth variant*, J. Math. Anal. Appl., to appear.

[29] Ph. Loewen, *A mean value theorem for Fréchet subgradients*, Nonlinear Anal., 23 (1994), pp. 1365–1381.

[30] Ph. Michel and J.-P. Penot, *Calcul sous-différentiel pour des fonctions lipschitziennes et non lipschitziennes*, C. R. Acad. Sci. Paris Sér. I Math., 298 (1984), pp. 269–272.

[31] Ph. Michel and J.-P. Penot, *A generalized derivative for calm and stable functions*, Differential Integral Equations, 5 (1992), pp. 433–454.

[32] K. F. Ng and Y.-C. Wong, *Partially Ordered Topological Vector Spaces*, Oxford Math. Monogr., Clarendon Press, Oxford, UK, 1973.

[33] J. S. Pang, *Error bounds in mathematical programming*, Math. Programming Ser. B, 79 (1997), pp. 299–332.

[34] J.-P. Penot, *A mean value theorem with small subdifferentials*, J. Optim. Theory Appl., 94 (1997), pp. 209–221.

[35] J.-P. Penot, *Conditioning convex and nonconvex problems*, J. Optim. Theory Appl., 90 (1997), pp. 539–558.

[36] J.-P. Penot, *Palais-Smale Condition and Coercivity*, preprint, Univ. of Pau, France, 1994.

[37] J.-P. Penot, *Miscellaneous incidences of convergence theories in optimization and nonlinear analysis, Part* II: *Applications in nonsmooth analysis*, in Recent Advances in Nonsmooth Optimization, D.-Z. Du, L. Qi, and R. S. Womersley, eds., World Scientific, Singapore, 1995, pp. 289–321.

[38] J.-P. Penot, *Subdifferential calculus without qualification assumptions*, J. Convex Anal., 3 (1996), pp. 207–220.

[39] J.-P. Penot, *Proximal mappings*, J. Approx. Theory, 94 (1998), pp. 203–221.

[40] J.-P. Penot, *Compactness properties, openness criteria and coderivatives*, Set-Valued Anal., 6 (1998), pp. 363–380.

[41] J.-P. Penot, *Subdifferential calculus and subdifferential compactness*, in Proceedings of the Second Catalan Days on Applied Mathematics, M. Sofonea and J.-N. Corvellec, eds., Presses Universitaires de Perpignan, Perpignan, France, 1995, pp. 209–226.

[42] J.-P.-Penot, *Well-behavior, well-posedness and nonsmooth analysis*, Pliska Stud. Math. Bulgar., 12 (1998), pp. 141–190.

[43] J.-P. Penot, *A Variational Subdifferential for Quasiconvex Functions*, submitted.

[44] A. L. Peressini, *Ordered Topological Vector Spaces*, Harper and Row, New York, 1967.

[45] J. Revalski, *Various aspects of well-posedness of optimization problems*, in Recent Developments in Well-Posed Variational Problems, R. Lucchetti and J. Revalski, eds., Kluwer, Dordrecht, the Netherlands, 1995, pp. 229–256.

[46] R. T. Rockafellar, *Directionally Lipschitzian functions and subdifferential calculus*, Proc. London Math. Soc., 39 (1979), pp. 145–154.

[47] R. T. Rockafellar, *Generalized directional derivatives and subgradients of nonconvex functions*, Canad. J. Math., 32 (1980), pp. 257–280.

[48] H. Schaefer, *Topological Vector Spaces*, Macmillan, London, 1966.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1020–1048

SUBDIFFERENTIAL CALCULUS AND NONSMOOTH CRITICAL POINT THEORY[*]

INES CAMPA[†] AND MARCO DEGIOVANNI[†]

Abstract. A general critical point theory for continuous functions defined on metric spaces has been recently developed. In this paper a new subdifferential, related to that theory, is introduced. In particular, results on the subdifferential of a sum are proved. An example of application to PDEs is sketched. Detailed applications to PDEs are developed in separate papers.

Key words. subdifferentials, nonsmooth critical point theory

AMS subject classifications. Primary, 49J52; Secondary, 58E05

PII. S1052623499353169

1. Introduction.

In the last 20 years, several efforts have been devoted to extending the classical critical point theory of [10, 31, 32, 36] to some classes of nondifferentiable functions. The case of locally Lipschitzian functions was treated in [9], while suitable families of lower semicontinuous functions were considered in [18, 20, 30] and in [37].

More recently, a general critical point theory, for continuous functions f defined on metric spaces, has been independently developed, with some variant, in [14, 16, 21] and in [27, 28, 29]. It is based on a generalized notion of norm of the derivative, denoted by $|df|$. This theory contains both the classical and the locally Lipschitz cases. Moreover, in [16, 21] some classes of lower semicontinuous functions are also considered, including those of [18, 20] and of [37].

This abstract framework has been applied to several problems in PDEs and variational inequalities. For instance, consider a functional $f : H^1_0(\Omega) \to \mathbb{R}$ of the form

$$f(u) = \frac{1}{2}\int_\Omega \sum_{i,j=1}^n a_{ij}(x,u) D_{x_i} u D_{x_j} u \, dx - \int_\Omega G(x,u)\, dx,$$

where Ω is an open subset of $\mathbb{R}^n$. Under reasonable assumptions on a_{ij} and G, the functional f turns out to be continuous. However, f is not locally Lipschitzian, unless the a_{ij}s are independent of u or $n = 1$. We refer the reader to [2, 4, 5, 6, 7, 8, 13, 15, 19, 23] and references therein for papers applying the theory of [14, 16, 21] to functionals that are not locally Lipschitzian.

Although the abstract approach seems to be satisfactory from the point of view of nonsmooth critical point theory, it is clear that $|df|$, being a generalization of the norm of the derivative, cannot have a rich calculus. Therefore, when f is defined on a normed space, it is more comfortable to work with a subdifferential, provided that it is suitable for critical point theory. For instance, if we know that

$$(1.1)\quad |df|(x) < +\infty \implies \big(\partial f(x) \neq \emptyset \quad \text{and} \quad |df|(x) \geq \min\{\|\alpha\| : \alpha \in \partial f(x)\}\big),$$

[*]Received by the editors March 10, 1999; accepted for publication (in revised form) October 12, 1999; published electronically June 6, 2000.

http://www.siam.org/journals/siopt/10-4/35316.html

[†]Dipartimento di Matematica e Fisica, Università Cattolica del Sacro Cuore, Via Trieste 17, I 25121 Brescia, Italy (m.degiovanni@dmf.bs.unicatt.it). The research of the second author was supported by Ministero dell'Università e della Ricerca Scientifica e Tecnologica.

then the condition $|df|(x) = 0$ implies $0 \in \partial f(x)$ and for every sequence (x_h) with $|df|(x_h) \to 0$ we may find a sequence $\alpha_h \to 0$ with $\alpha_h \in \partial f(x_h)$. Thus each critical point result in terms of $|df|$ implies a corresponding result in terms of such a subdifferential.

Now, if f is locally Lipschitzian, then (1.1) holds true for the Clarke subdifferential (see [21, Theorem (2.17)]). On the other hand, it is well known that, in such a case, the Clarke subdifferential is suitable for critical point theory: it was just the tool used in [9]. On the contrary, if f is, say, only continuous, the situation is quite different. Consider, for instance, the function $f : \mathbb{R} \to \mathbb{R}$ defined by $f(x) = x - \sqrt{|x|}$. Then we have $|df|(0) = 0$—it must be so, if we want to keep the mountain pass theorem [1] also for continuous functions. On the contrary, the Clarke subdifferential of f at 0 is empty and, of course, the same result a fortiori holds if we consider other subdifferentials, like those of [17, 25], which are even smaller than that of Clarke.

A related example can be built up in the following way: let

$$C = \left\{(x, y) \in \mathbb{R}^2 : \, y = f(x)\right\}$$

and let $\alpha : \mathbb{R}^2 \to \mathbb{R}$ be defined by $\alpha(x, y) = y$. Again we have $\left|d\left(\alpha_{|C}\right)\right|(0, 0) = 0$. On the other hand, since α is a continuous linear functional, we may expect a formula like

$$\partial\left(\alpha + I_C\right)(0, 0) = \alpha + \mathrm{N}_C\left(0, 0\right),$$

where I_C denotes the indicator function of C and $\mathrm{N}_C\left(0, 0\right)$ the normal cone to C at $(0, 0)$. Such a formula is in fact true, say, for Clarke's calculus and will hold also for the notions we are going to introduce, according to Corollary 5.3. Thus, to get $(0, 0) \in \partial\left(\alpha + I_C\right)(0, 0)$, it is necessary to have $(0, -1) \in \mathrm{N}_C\left(0, 0\right)$. But for the Clarke normal cone we have

$$\mathrm{N}_C\left(0, 0\right) = \left\{(\lambda, \mu) \in \mathbb{R}^2 : \, \mu \geq 0\right\}.$$

At this point it may seem hard to reconcile all these requirements. In fact, for several subdifferentials (see, e.g., [26]) the construction may be performed in three steps:

(i) definition of the subdifferential when the function is locally Lipschitzian;

(ii) definition of the normal cone $\mathrm{N}_C\left(x\right)$ to a subset C at $x \in C$ as the weak* closure of $\bigcup_{s>0}\left(s\,\partial\varrho_C(x)\right)$, where $\varrho_C(\xi) = \inf\left\{\|\xi - y\| : \, y \in C\right\}$;

(iii) definition of the subdifferential in the general case through the formula

$$\partial f(x) := \left\{\alpha \in X^* : \, (\alpha, -1) \in \mathrm{N}_{\mathrm{epi}(f)}\left(x, f(x)\right)\right\},$$

where epi (f) is the epigraph of f.

Now, if we keep these three points, it seems to be necessary, for our purposes, to enlarge the Clarke subdifferential even in the locally Lipschitz case, which may appear unconvenient.

The aim of this paper is to introduce new notions of tangent cone, normal cone, and subdifferential, which are conveniently related to $|df|$ (in particular, (1.1) holds according to Theorem 4.13) and thus suitable for critical point theory. Actually, our subdifferential agrees with that of Clarke in the locally Lipschitz case and point (iii) above is still true. Thus we violate point (ii). Our strategy will be to define first the tangent cone by a modification of the geometric construction of [12], then the normal cone in a standard way, and finally the subdifferential through (iii). A more direct approach to the subdifferential will also be presented, via a modification of the construction of [35].

In section 2 we recall, in a form suitable for our purposes, some notions from [16, 21, 29]. In section 3 and section 4 we introduce the main definitions and provide some general properties. As we have already mentioned, a subdifferential may be useful because of its calculus rules. Therefore, in section 5 we prove some results on the subdifferential of a sum $f+g$. We are in particular interested in the case in which f is a general functional, while g is either locally Lipschitzian or is the indicator of some "nice" set (see Corollaries 5.3, 5.4, and 5.9). Finally, in section 6 we study the abstract notions we have introduced in the case of functionals of the calculus of variations, like those we have mentioned before. We will see, at least in that case, that our subdifferential is not too large, although it is possibly larger than Clarke's. We also sketch, following [24], an application to nonlinear PDEs. For detailed applications of our subdifferential to variational problems involving PDEs, we refer the reader to [22, 24].

The main definitions and some results of this paper were announced in [19].

2. The weak slope.

In this section we recall, following an equivalent approach, some notions from [16, 21, 29].

Let X be a metric space endowed with the metric d, and let $f : X \to \overline{\mathbb{R}}$ be a function. If $Y \subseteq X$, we denote by $\overline{Y}$, $\mathrm{int}\,(Y)$, and ∂Y the closure, the interior, and the boundary of Y in X, respectively. We also denote by $\mathrm{B}_r\,(x)$ the open ball of center x and radius r and we set

$$\mathrm{epi}\,(f) = \{(x,\lambda) \in X \times \mathbb{R} :\ f(x) \le \lambda\}\,.$$

In the following, $X \times \mathbb{R}$ will be endowed with the metric

$$d\left((x,\lambda)\,,(y,\mu)\right) = \left(d(x,y)^2 + (\lambda-\mu)^2\right)^{\frac{1}{2}}$$

and $\mathrm{epi}\,(f)$ with the induced metric. Finally, as in [17] we define a continuous function $\mathcal{G}_f : \mathrm{epi}\,(f) \to \mathbb{R}$ by $\mathcal{G}_f(x,\lambda) = \lambda$.

Definition 2.1. *For every $x \in X$ with $f(x) \in \mathbb{R}$, we denote by $|df|\,(x)$ the supremum of the σ's in $[0,+\infty[$ such that there exist $\delta > 0$ and a continuous map*

$$\mathcal{H} : (\mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times [0,\delta] \to X$$

satisfying

$$d(\mathcal{H}((\xi,\mu),t),\xi) \le t\,, \qquad f(\mathcal{H}((\xi,\mu),t)) \le \mu - \sigma t,$$

whenever $(\xi,\mu) \in \mathrm{B}_\delta\,(x,f(x)) \cap \mathrm{epi}\,(f)$ and $t \in [0,\delta]$.

The extended real number $|df|\,(x)$ is called the weak slope *of f at x.*

The next proposition shows that the above definition agrees with that of [16, 21, 29] when f is real valued and continuous.

Proposition 2.2. *Let $f : X \to \mathbb{R}$ be a continuous function. Then $|df|\,(x)$ is the supremum of the σ's in $[0,+\infty[$ such that there exist $\delta > 0$ and a continuous map*

$$\mathcal{H} : \mathrm{B}_\delta\,(x) \times [0,\delta] \to X$$

satisfying

$$d(\mathcal{H}(\xi,t),\xi) \le t\,, \qquad f(\mathcal{H}(\xi,t)) \le f(\xi) - \sigma t,$$

whenever $\xi \in \mathrm{B}_\delta\,(x)$ and $t \in [0,\delta]$.

Proof. If

$$\mathcal{H} : (\mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta] \to X$$

is a map as in Definition 2.1, taking into account the continuity of f we may define

$$\mathcal{K} : \mathrm{B}_{\delta'}\,(x) \times [0, \delta'] \to X$$

by $\mathcal{K}(\xi, t) = \mathcal{H}((\xi, f(\xi)), t)$ for some small $\delta' > 0$. It is easy to see that $\mathcal{K}$ has the properties required in the statement of the proposition.

Conversely, let

$$\mathcal{K} : \mathrm{B}_\delta\,(x) \times [0, \delta] \to X$$

be a map as in the statement of the proposition. Then

$$\mathcal{H} : (\mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta] \to X$$

defined by $\mathcal{H}((\xi, \mu), t) = \mathcal{K}(\xi, t)$ has the properties required by Definition 2.1, as

$$f\,(\mathcal{H}((\xi, \mu), t)) = f\,(\mathcal{K}(\xi, t)) \le f(\xi) - \sigma t \le \mu - \sigma t.$$

Therefore equality holds. □

The next proposition shows that our notion agrees with that of [16, 21], also in the general case. Observe that $|d\mathcal{G}_f|\,(x, \lambda) \le 1$ for any $(x, \lambda) \in \mathrm{epi}\,(f)$, as $\mathcal{G}_f$ is Lipschitzian of constant 1.

Proposition 2.3. *For every $x \in X$ with $f(x) \in \mathbb{R}$, we have*

$$|df|\,(x) = \begin{cases} \dfrac{|d\mathcal{G}_f|\,(x, f(x))}{\sqrt{1 - |d\mathcal{G}_f|\,(x, f(x))^2}} & \textit{if } |d\mathcal{G}_f|\,(x, f(x)) < 1, \\ +\infty & \textit{if } |d\mathcal{G}_f|\,(x, f(x)) = 1. \end{cases}$$

Proof. First we prove that

$$|df|\,(x) \ge \begin{cases} \dfrac{|d\mathcal{G}_f|\,(x, f(x))}{\sqrt{1 - |d\mathcal{G}_f|\,(x, f(x))^2}} & \text{if } |d\mathcal{G}_f|\,(x, f(x)) < 1, \\ +\infty & \text{if } |d\mathcal{G}_f|\,(x, f(x)) = 1. \end{cases} \tag{2.1}$$

If $|d\mathcal{G}_f|\,(x, f(x)) = 0$, the assertion is evident. Otherwise, let $0 < \sigma < |d\mathcal{G}_f|\,(x, f(x))$. Since $\mathcal{G}_f$ is continuous, there exists

$$\mathcal{H} : (\mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta] \to \mathrm{epi}\,(f)$$

as in Proposition 2.2. Let $\delta' > 0$ be such that $\delta' < \delta\sqrt{1 - \sigma^2}$ and let

$$\mathcal{K} : (\mathrm{B}_{\delta'}\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta'] \to X$$

be defined by

$$\mathcal{K}((\xi, \mu), t) = \mathcal{H}_1\left((\xi, \mu), \frac{t}{\sqrt{1 - \sigma^2}}\right),$$

where $\mathcal{H}_1$ is the first component of $\mathcal{H}$. The map $\mathcal{K}$ is clearly continuous and

$$\begin{aligned} d(\mathcal{K}((\xi,\mu),t),\xi)^2 &= d\left(\mathcal{H}_1\left((\xi,\mu),\frac{t}{\sqrt{1-\sigma^2}}\right),\xi\right)^2 \\ &\leq \frac{t^2}{1-\sigma^2} - \left|\mathcal{H}_2\left((\xi,\mu),\frac{t}{\sqrt{1-\sigma^2}}\right)-\mu\right|^2 \\ &\leq \frac{t^2}{1-\sigma^2} - \frac{\sigma^2 t^2}{1-\sigma^2} = t^2\,. \end{aligned}$$

Moreover, we have

$$\begin{aligned} f(\mathcal{K}((\xi,\mu),t)) &= f\left(\mathcal{H}_1\left((\xi,\mu),\frac{t}{\sqrt{1-\sigma^2}}\right)\right) \leq \mathcal{H}_2\left((\xi,\mu),\frac{t}{\sqrt{1-\sigma^2}}\right) \\ &= \mathcal{G}_f\left(\mathcal{H}\left((\xi,\mu),\frac{t}{\sqrt{1-\sigma^2}}\right)\right) \leq \mathcal{G}_f(\xi,\mu) - \frac{\sigma}{\sqrt{1-\sigma^2}}\,t \\ &= \mu - \frac{\sigma}{\sqrt{1-\sigma^2}}\,t. \end{aligned}$$

Therefore it is

$$|df|\,(x) \geq \frac{\sigma}{\sqrt{1-\sigma^2}}$$

and inequality (2.1) follows from the arbitrariness of σ.

Now let us prove the opposite inequality. If $|df|\,(x)=0$ or $|d\mathcal{G}_f|\,(x,f(x))=1$, the assertion is evident. Otherwise, let $0<\sigma<|df|\,(x)$ and let

$$\mathcal{H} : (\mathrm{B}_\delta\,(x,f(x)) \cap \mathrm{epi}\,(f)) \times [0,\delta] \to X$$

be as in Definition 2.1. Define

$$\mathcal{K} : (\mathrm{B}_\delta\,(x,f(x)) \cap \mathrm{epi}\,(f)) \times [0,\delta] \to \mathrm{epi}\,(f)$$

by

$$\mathcal{K}((\xi,\mu),t) = \left(\mathcal{H}\left((\xi,\mu),\frac{t}{\sqrt{1+\sigma^2}}\right),\mu - \frac{\sigma}{\sqrt{1+\sigma^2}}t\right).$$

Since

$$f\left(\mathcal{H}\left((\xi,\mu),\frac{t}{\sqrt{1+\sigma^2}}\right)\right) \leq \mu - \frac{\sigma}{\sqrt{1+\sigma^2}}\,t,$$

we actually have $\mathcal{K}((\xi,\mu),t) \in \mathrm{epi}\,(f)$. Of course, $\mathcal{K}$ is continuous and

$$\begin{aligned} d(\mathcal{K}((\xi,\mu),t),(\xi,\mu)) &= \left(d\left(\mathcal{H}\left((\xi,\mu),\frac{t}{\sqrt{1+\sigma^2}}\right),\xi\right)^2 + \left(\frac{\sigma}{\sqrt{1+\sigma^2}}t\right)^2\right)^{\frac{1}{2}} \\ &\leq \left(\frac{t^2}{1+\sigma^2} + \frac{\sigma^2 t^2}{1+\sigma^2}\right)^{\frac{1}{2}} = t. \end{aligned}$$

Moreover we have

$$\mathcal{G}_f(\mathcal{K}((\xi,\mu),t)) = \mu - \frac{\sigma}{\sqrt{1+\sigma^2}}\,t = \mathcal{G}_f(\xi,\mu) - \frac{\sigma}{\sqrt{1+\sigma^2}}\,t.$$

Therefore, it is

$$|d\mathcal{G}_f|(x, f(x)) \geq \frac{\sigma}{\sqrt{1+\sigma^2}},$$

namely,

$$\sigma \leq \frac{|d\mathcal{G}_f|(x, f(x))}{\sqrt{1 - |d\mathcal{G}_f|(x, f(x))^2}}.$$

From the arbitrariness of σ, the assertion follows. □

3. Tangent and normal cones. Throughout this section, X will denote a real normed space and C a subset of X.

DEFINITION 3.1. *For every $x \in C$, we denote by $\mathrm{T}_C(x)$ the set of the v's in X such that for every $\varepsilon > 0$ there exist $\delta > 0$ and a continuous map*

$$\mathcal{V} : (\mathrm{B}_\delta(x) \cap C) \times]0, \delta] \to \mathrm{B}_\varepsilon(v)$$

satisfying $\xi + t\mathcal{V}(\xi, t) \in C$ whenever $\xi \in \mathrm{B}_\delta(x) \cap C$ and $t \in]0, \delta]$.

We say that $\mathrm{T}_C(x)$ is the tangent cone *to C at x.*

If we drop the continuity condition on $\mathcal{V}$, we get exactly the tangent cone in the sense of Clarke (see [12, Theorem 2.4.5]). Therefore, $\mathrm{T}_C(x)$ is contained in the tangent cone in the sense of Clarke.

THEOREM 3.2. *For every $x \in C$, the set $\mathrm{T}_C(x)$ is a closed convex cone in X with vertex at the origin.*

Proof. It is easy to see that $\mathrm{T}_C(x)$ is a closed cone with vertex at the origin. If $v_0, v_1 \in \mathrm{T}_C(x)$ and $\varepsilon > 0$, let $\mathcal{V}_0 : (\mathrm{B}_{\delta_0}(x) \cap C) \times]0, \delta_0] \to \mathrm{B}_{\frac{\varepsilon}{2}}(v_0)$ and $\mathcal{V}_1 : (\mathrm{B}_{\delta_1}(x) \cap C) \times]0, \delta_1] \to \mathrm{B}_{\frac{\varepsilon}{2}}(v_1)$ be as in Definition 3.1. Choose $0 < \delta \leq \min\{\delta_0, \delta_1\}$ satisfying $\|\xi + t\mathcal{V}_0(\xi, t) - x\| < \delta_1$ whenever $\|\xi - x\| < \delta$ and $0 < t \leq \delta$. Then the map

$$\mathcal{V} : (\mathrm{B}_\delta(x) \cap C) \times]0, \delta] \to X$$

defined by

$$\mathcal{V}(\xi, t) = \mathcal{V}_0(\xi, t) + \mathcal{V}_1(\xi + t\mathcal{V}_0(\xi, t), t)$$

is continuous and satisfies

$$\|\mathcal{V}(\xi, t) - v_0 - v_1\| \leq \|\mathcal{V}_0(\xi, t) - v_0\| + \|\mathcal{V}_1(\xi + t\mathcal{V}_0(\xi, t), t) - v_1\| < \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon,$$

$$\xi + t\left(\mathcal{V}_0(\xi, t) + \mathcal{V}_1(\xi + t\mathcal{V}_0(\xi, t), t)\right) = (\xi + t\mathcal{V}_0(\xi, t)) + t\mathcal{V}_1(\xi + t\mathcal{V}_0(\xi, t), t) \in C,$$

so that $(v_0 + v_1) \in \mathrm{T}_C(x)$. □

DEFINITION 3.3. *For every $x \in C$, we set*

$$\mathrm{N}_C(x) := \{\nu \in X^* : \langle \nu, v \rangle \leq 0 \quad \forall v \in \mathrm{T}_C(x)\}.$$

We say that $\mathrm{N}_C(x)$ is the normal cone *to C at x.*

Of course, $\mathrm{N}_C(x)$ is a convex cone with vertex at the origin and is weak* closed in X^*. Moreover, $\mathrm{N}_C(x)$ contains the normal cone in the sense of Clarke (see [12, Chapter 2.4]).

Theorem 3.4. *Assume that C is convex. Then, for every $x \in C$, $\mathrm{T}_C(x)$ and $\mathrm{N}_C(x)$ agree with the tangent and normal cones in the sense of convex analysis, i.e., we have*

$$\mathrm{T}_C(x) = \overline{\bigcup_{s>0} s\,(C-x)},$$
$$\mathrm{N}_C(x) = \{\nu \in X^* : \langle \nu, y-x \rangle \le 0 \quad \forall y \in C\}.$$

Proof. Let $s > 0$ and $y \in C$. Given $\varepsilon > 0$, let $\delta > 0$ be such that $\delta s \le \min\{\varepsilon, 1\}$. If we define $\mathcal{V} : (\mathrm{B}_\delta(x) \cap C) \times]0,\delta] \to \mathrm{B}_\varepsilon(s(y-x))$ by

$$\mathcal{V}(\xi, t) = s(y-\xi),$$

then $\mathcal{V}$ is continuous and satisfies

$$\|\mathcal{V}(\xi,t) - s(y-x)\| = s\|\xi - x\| < s\delta \le \varepsilon.$$

From the convexity of C it also follows that

$$\xi + t\mathcal{V}(\xi,t) = \xi + ts(y-\xi) \in C,$$

so $s(y-x) \in \mathrm{T}_C(x)$. Since $\mathrm{T}_C(x)$ is closed in X, we deduce that

$$\overline{\bigcup_{s>0} s\,(C-x)} \subseteq \mathrm{T}_C(x).$$

The opposite inclusion is obvious, as $\mathrm{T}_C(x)$ is contained in the tangent cone in the sense of Clarke.

The formula for $\mathrm{N}_C(x)$ follows from that for $\mathrm{T}_C(x)$. □

Now we want to investigate some particular situations in which our notions agree with those of Clarke. For this purpose, let us recall a well-known concept (see, e.g., [12, Chapter 2.4]).

Definition 3.5. *For every $x \in C$, we denote by $\mathrm{Hyp}_C(x)$ the set of the v's in X such that there exists $\delta > 0$ satisfying*

$$(\mathrm{B}_\delta(x) \cap C) +]0,\delta] \cdot \mathrm{B}_\delta(v) \subseteq C. \tag{3.1}$$

We say that $\mathrm{Hyp}_C(x)$ is the hypertangent cone *to C at x.*

If $v \in \mathrm{Hyp}_C(x)$ and δ satisfies (3.1), it is readily seen that

$$\left(\mathrm{B}_\delta(x) \cap \overline{C}\right) +]0,\delta] \cdot \mathrm{B}_\delta(v) = (\mathrm{B}_\delta(x) \cap C) +]0,\delta] \cdot \mathrm{B}_\delta(v) \subseteq \mathrm{int}\,(C), \tag{3.2}$$

$$\mathrm{B}_\delta(x) \cap \overline{\mathrm{int}\,(C)} = \mathrm{B}_\delta(x) \cap \overline{C}, \qquad \mathrm{B}_\delta(x) \cap \mathrm{int}\,\left(\overline{C}\right) = \mathrm{B}_\delta(x) \cap \mathrm{int}\,(C). \tag{3.3}$$

Theorem 3.6. *Let $x \in C$ with $\mathrm{Hyp}_C(x) \ne \emptyset$. Then $\mathrm{T}_C(x)$ agrees with the corresponding tangent cone in the sense of Clarke and we have $\mathrm{Hyp}_C(x) = \mathrm{int}\,(\mathrm{T}_C(x))$ and $\mathrm{T}_C(x) = \overline{\mathrm{Hyp}_C(x)}$.*

Proof. If we denote by $\widetilde{\mathrm{T}}_C(x)$ the tangent cone in the sense of Clarke, we clearly have $\mathrm{Hyp}_C(x) \subseteq \mathrm{T}_C(x) \subseteq \widetilde{\mathrm{T}}_C(x)$. Then the assertion follows from Rockafellar's Theorem (see, e.g., [12, Theorem 2.4.8]). □

Lemma 3.7. *Let D be another subset of X and let $x \in C \cap D$ with $\mathrm{Hyp}_C(x) \ne \emptyset$. Assume there exists $r > 0$ such that $\mathrm{B}_r(x) \subseteq C \cup D$ and $\mathrm{B}_r(x) \cap D \cap \mathrm{int}\,(C) = \emptyset$.*

Then

$$\mathrm{Hyp}_D(x) = -\mathrm{Hyp}_C(x).$$

Proof. Let $v \in \mathrm{Hyp}_C(x)$ and, according to (3.2), let $\varepsilon \in]0,r]$ be such that

$$(\mathrm{B}_\varepsilon(x) \cap C) +]0,\varepsilon] \cdot \mathrm{B}_\varepsilon(v) \subseteq \mathrm{int}(C).$$

Take $\delta > 0$ with

$$(\mathrm{B}_\delta(x) \cap D) +]0,\delta] \cdot \mathrm{B}_\delta(-v) \subseteq \mathrm{B}_\varepsilon(x) \subseteq \mathrm{B}_r(x).$$

We want to show that

$$(\mathrm{B}_\delta(x) \cap D) +]0,\delta] \cdot \mathrm{B}_\delta(-v) \subseteq D.$$

By contradiction, let $\xi \in \mathrm{B}_\delta(x) \cap D$, $t \in]0,\delta]$ and $w \in \mathrm{B}_\delta(v)$ with $\xi - tw \in C$. Then

$$\xi = (\xi - tw) + tw \in \mathrm{int}(C),$$

which contradicts $\mathrm{B}_r(x) \cap D \cap \mathrm{int}(C) = \emptyset$. Therefore $-\mathrm{Hyp}_C(x) \subseteq \mathrm{Hyp}_D(x)$.

In particular, $\mathrm{Hyp}_D(x) \neq \emptyset$. Moreover, from (3.3) it follows that $\mathrm{B}_\varepsilon(x) \cap C \cap \mathrm{int}(D) = \emptyset$. Therefore the opposite inclusion also follows. □

Theorem 3.8. *Let $x \in C \cap \partial C$ with $\mathrm{Hyp}_C(x) \neq \emptyset$. Then the following facts hold:*

(i) $\mathrm{Hyp}_C(x) + \mathrm{T}_{\partial C}(x) = \mathrm{Hyp}_C(x)$;

(ii) $\mathrm{T}_{\partial C}(x)$ *agrees with the corresponding tangent cone in the sense of Clarke and we have* $\mathrm{T}_{\partial C}(x) = \mathrm{T}_C(x) \cap (-\mathrm{T}_C(x))$*; in particular,* $\mathrm{T}_{\partial C}(x)$ *is a closed linear subspace of* X*.*

Proof. Denote by $\widetilde{\mathrm{T}}_{\partial C}(x)$ the tangent cone in the sense of Clarke. First of all, we want to show that

$$\mathrm{Hyp}_C(x) + \mathrm{T}_{\partial C}(x) = \mathrm{Hyp}_C(x) + \widetilde{\mathrm{T}}_{\partial C}(x) = \mathrm{Hyp}_C(x). \tag{3.4}$$

Since $0 \in \mathrm{T}_{\partial C}(x)$ and $\mathrm{T}_{\partial C}(x) \subseteq \widetilde{\mathrm{T}}_{\partial C}(x)$, it is sufficient to prove that

$$\mathrm{Hyp}_C(x) + \widetilde{\mathrm{T}}_{\partial C}(x) \subseteq \mathrm{Hyp}_C(x).$$

Let $v_0 \in \mathrm{Hyp}_C(x)$ and let $\delta_0 > 0$ be as in Definition 3.5. Let also $v_1 \in \widetilde{\mathrm{T}}_{\partial C}(x)$ and let $\delta_1 > 0$ be such that

$$\forall \xi \in \mathrm{B}_{\delta_1}(x) \cap \partial C, \forall t \in]0,\delta_1], \exists w \in \mathrm{B}_{\frac{\delta_0}{2}}(v_1) : \xi + tw \in \partial C. \tag{3.5}$$

Finally, let

$$\delta_2 = \min\left\{\frac{\delta_0}{2}, \frac{\delta_0}{1 + \frac{1}{2}\delta_0 + \|v_1\|}, \delta_1\right\}.$$

We claim that

$$(\mathrm{B}_{\delta_2}(x) \cap \partial C) +]0,\delta_2] \cdot \mathrm{B}_{\delta_2}(v_0 + v_1) \subseteq C. \tag{3.6}$$

Actually, if $\xi \in \mathrm{B}_{\delta_2}(x) \cap \partial C$, $t \in]0, \delta_2]$, and $v \in \mathrm{B}_{\delta_2}(v_0 + v_1)$, let $w \in \mathrm{B}_{\frac{\delta_0}{2}}(v_1)$ be as in (3.5). We have $\xi + tw \in \partial C \subseteq \overline{C}$ and also $\xi + tw \in \mathrm{B}_{\delta_0}(x)$, as

$$\begin{aligned}\|\xi + tw - x\| &= \|\xi + tw - tv_1 + tv_1 - x\| \\ &\le \|\xi - x\| + t\|w - v_1\| + t\|v_1\| \\ &< \delta_2 + \delta_2 \frac{\delta_0}{2} + \delta_2\|v_1\| \le \delta_0.\end{aligned}$$

On the other hand, $v - w \in \mathrm{B}_{\delta_0}(v_0)$, as

$$\begin{aligned}\|v - w - v_0\| &= \|v - v_0 - v_1 + v_1 - w\| \\ &\le \|v - v_0 - v_1\| + \|v_1 - w\| < \delta_2 + \frac{\delta_0}{2} \le \delta_0.\end{aligned}$$

Therefore, according to (3.2), we have

$$\xi + tv = (\xi + tw) + t(v - w) \in C$$

and (3.6) follows.

Now let $\delta > 0$ be such that

$$\delta + \delta^2 + \delta\|v_0\| + \delta\|v_1\| \le \delta_2.$$

According to (3.2) and (3.3), to prove that $v_0 + v_1 \in \mathrm{Hyp}_C(x)$ it is sufficient to show that

$$\left(\mathrm{B}_\delta(x) \cap \overline{C}\right) +]0, \delta] \cdot \mathrm{B}_\delta(v_0 + v_1) \subseteq \mathrm{B}_{\delta_0}(x) \cap \overline{C}.$$

Let $\xi \in \mathrm{B}_\delta(x) \cap \overline{C}$, $t \in]0, \delta]$, and $v \in \mathrm{B}_\delta(v_0 + v_1)$. Clearly, we have $\xi + tv \in \mathrm{B}_{\delta_0}(x)$. If, by contradiction, $\xi + tv \notin \overline{C}$, there exists $\tau \in [0, t[$ with $\xi + \tau v \in \partial C$. Of course we have $0 < t - \tau \le \delta_2$ and $v \in \mathrm{B}_{\delta_2}(v_0 + v_1)$. Moreover, it is

$$\begin{aligned}\|\xi + \tau v - x\| &\le \|\xi - x\| + \tau\|v - v_0 - v_1\| + \tau\|v_0\| + \tau\|v_1\| \\ &< \delta + \delta^2 + \delta\|v_0\| + \delta\|v_1\| \le \delta_2.\end{aligned}$$

From (3.6) we deduce that

$$\xi + tv = (\xi + \tau v) + (t - \tau)v \in C,$$

whence a contradiction. Therefore (3.4) follows.

Since $sv_0 \in \mathrm{Hyp}_C(x)$ for any $s > 0$, we also have $\widetilde{\mathrm{T}}_{\partial C}(x) \subseteq \mathrm{T}_C(x)$. Now let $D = \overline{X \setminus C}$. From (3.3) it follows that $\mathrm{B}_{\delta_0}(x) \cap \partial C = \mathrm{B}_{\delta_0}(x) \cap \partial D$. Combining this fact with Lemma 3.7, we deduce that

$$\widetilde{\mathrm{T}}_{\partial C}(x) = \widetilde{\mathrm{T}}_{\partial D}(x) \subseteq \mathrm{T}_D(x) = -\mathrm{T}_C(x);$$

hence

$$\mathrm{T}_{\partial C}(x) \subseteq \widetilde{\mathrm{T}}_{\partial C}(x) \subseteq \mathrm{T}_C(x) \cap (-\mathrm{T}_C(x)).$$

Finally, let $z \in \mathrm{T}_C(x) \cap (-\mathrm{T}_C(x))$ and let again $v_0 \in \mathrm{Hyp}_C(x)$. Given $\varepsilon > 0$, let $\varepsilon' \in]0, 1]$ with $\varepsilon'\|v_0\| < \varepsilon$. From Lemma 3.7 we deduce that $z + \varepsilon' v_0 \in \mathrm{Hyp}_C(x)$

and $z - \varepsilon' v_0 \in \mathrm{Hyp}_D(x)$. Let $\delta_3 > 0$ be associated with v_0, $z + \varepsilon' v_0$ and $z - \varepsilon' v_0$, according to Definition 3.5, and let $\delta_4 > 0$ with

$$\delta_4 + \delta_4\|z\| + \delta_4\|v_0\| \le \delta_3.$$

If $\xi \in \mathrm{B}_{\delta_4}(x) \cap \partial C$ and $0 < t \le \delta_4$, we have

$$\xi + t(z + \varepsilon' v_0) \in \mathrm{int}\,(C)\,.$$

Therefore, if $\xi + tz \notin \mathrm{int}\,(C)$, there exists $\tau^+ \in [0,1[$ with

$$\xi + t(z + \tau^+ \varepsilon' v_0) \in \partial C.$$

Such a τ^+ is unique. Otherwise, if $0 \le \tau_1^+ < \tau_2^+ < 1$ have this property, it follows that

$$\begin{aligned}\|\xi + tz + t\tau_1^+\varepsilon' v_0 - x\| &\le \|\xi - x\| + t\|z\| + t\|v_0\| \\ &< \delta_4 + \delta_4\|z\| + \delta_4\|v_0\| \le \delta_3;\end{aligned}$$

hence

$$\xi + t(z + \tau_2^+\varepsilon' v_0) = (\xi + tz + t\tau_1^+\varepsilon' v_0) + t(\tau_2^+ - \tau_1^+)\varepsilon' v_0 \in \mathrm{int}\,(C)\,,$$

which is absurd. Therefore we may define a continuous map $\tau^+ : C^+ \to [0,1[$ with

$$C^+ = \{(\xi,t) \in (\mathrm{B}_{\delta_4}(x) \cap \partial C) \times]0,\delta_4] : \xi + tz \notin \mathrm{int}\,(C)\}$$

such that

$$\xi + t(z + \tau^+(\xi,t)\varepsilon' v_0) \in \partial C,$$

$$\xi + tz \in \partial C \implies \tau^+(\xi,t) = 0.$$

In a similar way, it is possible to define a continuous map $\tau^- : C^- \to [0,1[$ with

$$C^- = \left\{(\xi,t) \in (\mathrm{B}_{\delta_4}(x) \cap \partial C) \times]0,\delta_4] : \xi + tz \in \overline{C}\right\}$$

such that

$$\xi + t(z - \tau^-(\xi,t)\varepsilon' v_0) \in \partial C,$$

$$\xi + tz \in \partial C \implies \tau^-(\xi,t) = 0.$$

Moreover we have

$$\|z \pm \tau^\pm(\xi,t)\varepsilon' v_0 - z\| \le \varepsilon'\|v_0\| < \varepsilon.$$

Therefore we may define a continuous map $\mathcal{Z} : (\mathrm{B}_{\delta_4}(x) \cap \partial C) \times]0,\delta_4] \to \mathrm{B}_\varepsilon(z)$ by

$$\mathcal{Z}(\xi,t) = \begin{cases} z + \tau^+(\xi,t)\varepsilon' v_0 & \text{if } \xi + tz \notin \mathrm{int}\,(C), \\ z - \tau^-(\xi,t)\varepsilon' v_0 & \text{if } \xi + tz \in \overline{C}. \end{cases}$$

From the construction it follows that $\xi + t\mathcal{Z}(\xi,t) \in \partial C$, so that $z \in \mathrm{T}_{\partial C}(x)$. Therefore

$$\mathrm{T}_C(x) \cap (-\mathrm{T}_C(x)) \subseteq \mathrm{T}_{\partial C}(x)$$

and the proof is complete. □

4. The subdifferential. Throughout this section, X will denote a real normed space and $f : X \to \overline{\mathbb{R}}$ a function. The linear space $X \times \mathbb{R}$ will be endowed with the norm

$$\|(x, \lambda)\| = \left(\|x\|^2 + \lambda^2\right)^{\frac{1}{2}} .$$

Definition 4.1. *For every $x \in X$ with $f(x) \in \mathbb{R}$, we set*

$$\partial f(x) := \left\{\alpha \in X^* : \, (\alpha, -1) \in \mathrm{N}_{\mathrm{epi}(f)}\,(x, f(x))\right\} .$$

We say that $\partial f(x)$ is the subdifferential *of f at x.*

If $C \subseteq X$ and I_C denotes the indicator function of C, namely,

$$I_C(\xi) = \begin{cases} 0 & \text{if } \xi \in C, \\ +\infty & \text{if } \xi \in X \setminus C, \end{cases}$$

it is easy to see that $\partial I_C(x) = \mathrm{N}_C\,(x)$ for every $x \in C$. Moreover, $\partial f(x)$ contains the subdifferential in the sense of Clarke (see [12]).

Theorem 4.2. *Assume that f is convex. Then, for every $x \in X$ with $f(x) \in \mathbb{R}$, $\partial f(x)$ agrees with the subdifferential of convex analysis, i.e., we have*

$$\partial f(x) = \{\alpha \in X^* : \, f(y) \geq f(x) + \langle \alpha, y - x\rangle \quad \forall y \in X\} .$$

Proof. The proof follows from Theorem 3.4. □

Definition 4.3. *Let $x \in X$ with $f(x) \in \mathbb{R}$. For every $v \in X$ and $\varepsilon > 0$ let $f^0_\varepsilon\,(x; v)$ be the infimum of the r's in $\mathbb{R}$ such that there exist $\delta > 0$ and a continuous map*

$$\mathcal{V} : (\mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)) \times]0, \delta] \to \mathrm{B}_\varepsilon\,(v)$$

satisfying

$$f\left(\xi + t\mathcal{V}((\xi, \mu), t)\right) \leq \mu + rt$$

whenever $(\xi, \mu) \in \mathrm{B}_\delta\,(x, f(x)) \cap \mathrm{epi}\,(f)$ and $t \in]0, \delta]$ (we agree that $\inf \emptyset = +\infty$).

Let also

$$f^0\,(x; v) := \sup_{\varepsilon > 0} f^0_\varepsilon\,(x; v) = \lim_{\varepsilon \to 0^+} f^0_\varepsilon\,(x; v) .$$

We say that $f^0\,(x; v)$ is the generalized directional derivative *of f at x with respect to v.*

Again, if we drop the continuity condition on $\mathcal{V}$, we get exactly the generalized directional derivative in the sense of Rockafellar (see [12, 35]). Therefore $f^0\,(x; v)$ is greater than or equal to the generalized directional derivative of Rockafellar.

Proposition 4.4. *Let $f : X \to \mathbb{R}$ be continuous and let $x \in X$. Then for every $v \in X$ and $\varepsilon > 0$ we have that $f^0_\varepsilon\,(x; v)$ is the infimum of the r's in $\mathbb{R}$ such that there exist $\delta > 0$ and a continuous map $\mathcal{V} : \mathrm{B}_\delta\,(x) \times]0, \delta] \to \mathrm{B}_\varepsilon\,(v)$ satisfying*

$$f(\xi + t\mathcal{V}(\xi, t)) \leq f(\xi) + rt$$

whenever $\xi \in \mathrm{B}_\delta\,(x)$ and $t \in]0, \delta]$.

Proof. The proof is a variant of that of Proposition 2.2. Therefore, we omit it. □

THEOREM 4.5. *For every $x \in X$ with $f(x) \in \mathbb{R}$ we have*

$$\mathrm{T}_{\mathrm{epi}(f)}\left(x, f(x)\right) = \mathrm{epi}\left(f^0\left(x; \cdot\right)\right).$$

Proof. Let $(v, r) \in \mathrm{T}_{\mathrm{epi}(f)}\left(x, f(x)\right)$, let $\varepsilon > 0$, and let

$$\mathcal{W} : \left(\mathrm{B}_\delta\left(x, f(x)\right) \cap \mathrm{epi}\left(f\right)\right) \times \left]0, \delta\right] \to \mathrm{B}_\varepsilon\left(v, r\right)$$

be a map as in Definition 3.1. Then the first component

$$\mathcal{W}_1 : \left(\mathrm{B}_\delta\left(x, f(x)\right) \cap \mathrm{epi}\left(f\right)\right) \times \left]0, \delta\right] \to \mathrm{B}_\varepsilon\left(v\right)$$

is continuous and satisfies

$$\frac{f\left(\xi + t\mathcal{W}_1((\xi, \mu), t)\right) - \mu}{t} \leq \mathcal{W}_2((\xi, \mu), t) < r + \varepsilon.$$

It follows that $f^0_\varepsilon\left(x; v\right) \leq r + \varepsilon$, hence $f^0\left(x; v\right) \leq r$ by the arbitrariness of ε.

Conversely, let $f^0\left(x; v\right) \leq r$. Given $\varepsilon > 0$, let

$$\mathcal{V} : \left(\mathrm{B}_\delta\left(x, f(x)\right) \cap \mathrm{epi}\left(f\right)\right) \times \left]0, \delta\right] \to \mathrm{B}_{\frac{\varepsilon}{\sqrt{2}}}\left(v\right)$$

be a continuous map satisfying

$$\frac{f(\xi + t\mathcal{V}((\xi, \mu), t)) - \mu}{t} \leq r + \frac{\varepsilon}{\sqrt{2}}.$$

Define

$$\mathcal{W} : \left(\mathrm{B}_\delta\left(x, f(x)\right) \cap \mathrm{epi}\left(f\right)\right) \times \left]0, \delta\right] \to X \times \mathbb{R}$$

by

$$\mathcal{W}((\xi, \mu), t) = \left(\mathcal{V}((\xi, \mu), t), r + \frac{\varepsilon}{\sqrt{2}}\right).$$

Then $\mathcal{W}$ is continuous and satisfies $\mathcal{W}((\xi, \mu), t) \in \mathrm{B}_\varepsilon\left(v, r\right)$. Moreover one has

$$f(\xi + t\mathcal{V}((\xi, \mu), t)) \leq \mu + t\left(r + \frac{\varepsilon}{\sqrt{2}}\right),$$

namely, $(\xi, \mu) + t\mathcal{W}((\xi, \mu), t) \in \mathrm{epi}\left(f\right)$. Therefore $(v, r) \in \mathrm{T}_{\mathrm{epi}(f)}\left(x, f(x)\right)$. □

COROLLARY 4.6. *For every $x \in X$ with $f(x) \in \mathbb{R}$, the function*

$$\left\{v \longmapsto f^0\left(x; v\right)\right\}$$

is convex, lower semicontinuous, and positively homogeneous of degree 1. *Moreover it is $f^0\left(x; 0\right) \in \{0, -\infty\}$.*

Proof. It follows from Theorem 4.5. □

COROLLARY 4.7. *Let $x \in X$ with $f(x) \in \mathbb{R}$. Then the following facts hold:*

(i) $\partial f(x) = \left\{\alpha \in X^* : \langle \alpha, v\rangle \leq f^0\left(x; v\right) \quad \forall v \in X\right\}$;

(ii) $\partial f(x) = \emptyset \Longleftrightarrow f^0\left(x; 0\right) = -\infty$;

(iii) $\partial f(x)$ *is convex and weak* closed in* X^*;

(iv) *if $\partial f(x) \neq \emptyset$, it is*

$$\forall v \in X : f^0(x;v) = \sup\{\langle \alpha, v\rangle : \alpha \in \partial f(x)\}.$$

Proof. Statement (i) follows from Theorem 4.5. The other properties are consequences of (i) and Corollary 4.6. □

Now let us recall a well-known notion (see, e.g., [12, Definition 2.9.2]).

DEFINITION 4.8. *Let $x \in X$ with $f(x) \in \mathbb{R}$. We set*

$$\forall v \in X : f^+(x;v) := \limsup_{\substack{(\xi,\mu)\to(x,f(x))\\ (\xi,\mu)\in \mathrm{epi}\,(f)\\ w\to v,\ t\to 0^+}} \frac{f(\xi+tw)-\mu}{t},$$

$$\mathrm{D}_f(x) := \left\{v \in X : f^+(x;v) < +\infty\right\}.$$

The function f is said to be directionally Lipschitzian *at x if $\mathrm{D}_f(x) \neq \emptyset$.*

THEOREM 4.9. *Let $x \in X$ with $f(x) \in \mathbb{R}$. Assume that f is directionally Lipschitzian at x.*

Then $\partial f(x)$ and $f^0(x;\cdot)$ agree with the corresponding notions of Clarke–Rockafellar and we have

$$\mathrm{D}_f(x) = \mathrm{int}\left(\left\{v \in X : f^0(x;v) < +\infty\right\}\right),$$

$$\forall v \in \mathrm{D}_f(x) : f^+(x;v) = f^0(x;v).$$

Proof. From [12, Proposition 2.9.3] we deduce that $\mathrm{Hyp}_{\mathrm{epi}(f)}(x,f(x)) \neq \emptyset$. Therefore $\partial f(x)$ and $f^0(x;\cdot)$ agree with the corresponding notions of Clarke–Rockafellar by Theorems 3.6 and 4.5. The last formulas follow from [12, Theorem 2.9.5]. □

COROLLARY 4.10. *Let $f : X \to \mathbb{R}$ be locally Lipschitzian and let $x \in X$. Then $\partial f(x)$ and $f^0(x;\cdot)$ agree with the corresponding notions of Clarke.*

Remark 4.11. If we consider the function $f : \mathbb{R} \to \mathbb{R}$ and the subset C of $\mathbb{R}^2$ mentioned in the introduction, it is easy to see that $\mathrm{T}_C(0,0) = \{(0,0)\}$ while $\partial f(0)$ and $\mathrm{N}_C(0,0)$ are the whole space. Therefore our notions of subdifferential, tangent cone, and normal cone do not agree, in general, with those of Clarke. Nevertheless, when f is locally Lipschitzian, the two approaches turn out to be equivalent. This is possible, as we do not impose, for instance, that

$$v \in \mathrm{T}_C(x) \iff (\varrho_C)^0(x;v) \leq 0,$$

where ϱ_C is the Lipschitzian function defined by

$$\varrho_C(\xi) = \inf\{\|\xi - y\| : y \in C\}.$$

This is closely related to the fact that, in our approach, condition (ii) mentioned in the introduction is not fulfilled.

In the next result we state without proof some simple calculus rules.

THEOREM 4.12. *The following facts hold:*

(i) *For every $x \in X$ with $f(x) \in \mathbb{R}$ and for every $s > 0$ we have*

$$\forall v \in X : (sf)^0(x;v) = sf^0(x;v),$$

$$\partial(sf)(x) = s\partial f(x).$$

(ii) *If Y is another real normed space, $\varphi : Y \to X$ a diffeomorphism, and $y \in Y$ is such that $f(\varphi(y)) \in \mathbb{R}$, then we have*

$$\forall w \in Y : \; (f \circ \varphi)^0 (y; w) = f^0 (\varphi(y); \varphi'(y)w) ,$$

$$\partial (f \circ \varphi) (y) = \{\alpha \circ \varphi'(y) : \, \alpha \in \partial f (\varphi(y))\} .$$

Now we can prove the main result which has motivated the introduction of a new subdifferential.

THEOREM 4.13. *For every $x \in X$ with $f(x) \in \mathbb{R}$ the following facts hold:*

(i) *we have*

$$\forall \varepsilon > 0 : \quad \sup\left\{-f^0_\varepsilon (x; v) : \, v \in X, \, \|v\| \le 1\right\} \le (1+\varepsilon) \, |df| \, (x),$$
$$\sup\left\{-f^0 (x; v) : \, v \in X, \, \|v\| \le 1\right\} \le |df| \, (x);$$

(ii) $|df| \, (x) < +\infty \Longleftrightarrow \partial f(x) \ne \emptyset$;

(iii) $|df| \, (x) < +\infty \Longrightarrow |df| \, (x) \ge \min\{\|\alpha\| : \, \alpha \in \partial f(x)\}$.

Proof. (i) Let $\varepsilon > 0$ and $v \in X$ with $\|v\| \le 1$. To prove the first inequality, it is sufficient to show that

$$f^0_\varepsilon (x; v) \ge -(1+\varepsilon) \, |df| \, (x). \tag{4.1}$$

If $f^0_\varepsilon (x; v) \ge 0$, the assertion is evident. Otherwise, let $f^0_\varepsilon (x; v) < r < 0$ and let

$$\mathcal{V} : (\mathrm{B}_\delta (x, f(x)) \cap \mathrm{epi} (f)) \times]0, \delta] \to \mathrm{B}_\varepsilon (v)$$

be as in Definition 4.3. Define

$$\mathcal{H} : (\mathrm{B}_\delta (x, f(x)) \cap \mathrm{epi} (f)) \times [0, \delta] \to X$$

by

$$\mathcal{H}((\xi, \mu), t) = \begin{cases} \xi + \dfrac{t}{1+\varepsilon} \mathcal{V}\left((\xi, \mu), \dfrac{t}{1+\varepsilon}\right) & \text{if } t \ne 0, \\ \xi & \text{if } t = 0. \end{cases}$$

Since

$$\left\|\mathcal{V}\left((\xi, \mu), \frac{t}{1+\varepsilon}\right)\right\| \le \left\|\mathcal{V}\left((\xi, \mu), \frac{t}{1+\varepsilon}\right) - v\right\| + \|v\| < \varepsilon + 1,$$

it is easy to see that $\mathcal{H}$ is continuous. Moreover we have

$$\|\mathcal{H}((\xi, \mu), t) - \xi\| \le t,$$

$$f(\mathcal{H}((\xi, \mu), t)) \le \mu + \frac{r}{1+\varepsilon} t,$$

so that $|df| \, (x) \ge -\frac{r}{1+\varepsilon}$. It follows that $r \ge -(1+\varepsilon) \, |df| \, (x)$, hence (4.1) by the arbitrariness of r.

The second inequality in (i) is a consequence of the first one.

(ii) and (iii) Assume that $|df|(x) < +\infty$. Combining property (i) with Corollary 4.6, we see that

$$\forall v \in X : \, f^0(x;v) \geq -|df|(x)\|v\|.$$

From [37, Lemma 1.3] we deduce that there exists $\alpha \in \partial f(x)$ with $\|\alpha\| \leq |df|(x)$. Therefore $\partial f(x) \neq \emptyset$ and (iii) holds.

Finally, suppose that $|df|(x) = +\infty$. Let $\varepsilon, \sigma > 0$ and let

$$\mathcal{H} : (\mathrm{B}_\delta(x, f(x)) \cap \mathrm{epi}(f)) \times [0,\delta] \to X$$

be a map as in Definition 2.1. Set $\delta' = \min\left\{\delta, \frac{2\delta}{\varepsilon}\right\}$ and define

$$\mathcal{V} : (\mathrm{B}_{\delta'}(x, f(x)) \cap \mathrm{epi}(f)) \times]0,\delta'] \to X$$

by

$$\mathcal{V}((\xi,\mu),t) = \frac{\mathcal{H}\left((\xi,\mu), \frac{\varepsilon}{2}t\right) - \xi}{t}.$$

Of course, $\mathcal{V}$ is continuous and $\mathcal{V}((\xi,\mu),t) \in \mathrm{B}_\varepsilon(0)$, as

$$\|\mathcal{V}((\xi,\mu),t)\| \leq \frac{\frac{\varepsilon}{2}t}{t} < \varepsilon.$$

Moreover

$$f(\xi + t\mathcal{V}((\xi,\mu),t)) = f\left(\mathcal{H}\left((\xi,\mu), \frac{\varepsilon}{2}t\right)\right) \leq \mu - \frac{\varepsilon\sigma}{2}t,$$

so that $f^0_\varepsilon(x;0) \leq -\frac{\varepsilon}{2}\sigma$. By the arbitrariness of σ, it follows that $f^0_\varepsilon(x;0) = -\infty$, hence $f^0(x;0) = -\infty$ by the arbitrariness of ε. Therefore $\partial f(x) = \emptyset$. □

Example 4.14. The inequality in (iii) may be strict, even if f is Lipschitzian (so that the usual Clarke subdifferential is involved). The first counterexample in this sense has been provided in [33]. As a variant, consider $f : \mathbb{R}^2 \to \mathbb{R}$ defined by

$$f(x,y) = a\Big|y - m|x|\Big| - \sigma x$$

with $a, m, \sigma > 0$ and $am \geq \sigma$. Then $0 \in \partial f(0,0)$, but $|df|(0,0) > 0$.

5. The subdifferential of a sum.

Throughout this section, X will denote a real Banach space and $f : X \to \overline{\mathbb{R}}$ a function.

The next result is an adaptation to our setting of Rockafellar's theorem (see, e.g., [12, Theorem 2.9.8]).

THEOREM 5.1. *Let $g : X \to \mathbb{R} \cup \{+\infty\}$ be a function, and let $x \in X$ with $f(x) \in \mathbb{R}$ and $g(x) \in \mathbb{R}$. Assume that the restriction of g to $\{\xi \in X : g(\xi) < +\infty\}$ is continuous and that there exists $v_0 \in \mathrm{D}_g(x)$ with $f^0(x;v_0) < +\infty$.*

Then the following facts hold:

(i) *for every $v \in X$ we have*

$$(f+g)^0(x;v) \leq f^0(x;v) + g^0(x;v);$$

(ii) $\partial f(x) + \partial g(x)$ *is weak** *closed in* X^* *and we have*

$$\partial(f+g)(x) \subseteq \partial f(x) + \partial g(x).$$

(We agree that $+\infty + (-\infty) = -\infty + (+\infty) = +\infty$*.)*

Proof. (i) Consider first $v \in \mathrm{D}_g(x)$ with $f^0(x;v) < +\infty$. By Theorem 4.9, we have $g^+(x;v) = g^0(x;v)$. Given $r > f^0(x;v)$, $s > g^0(x;v)$, and $\varepsilon > 0$, let $\delta \in]0,\varepsilon]$ be such that

$$\forall \xi \in \mathrm{B}_\delta(x),\ \forall t \in]0,\delta],\ \forall w \in \mathrm{B}_\delta(v) :\ g(\xi + tw) \le g(\xi) + st.$$

Then let $\delta' \in]0,\delta]$ and let $\mathcal{V} : (\mathrm{B}_{\delta'}(x, f(x)) \cap \mathrm{epi}(f)) \times]0,\delta'] \to \mathrm{B}_\delta(v)$ be a continuous map satisfying

$$f(\xi + t\mathcal{V}((\xi,\mu),t)) \le \mu + rt.$$

Finally, let $\delta'' \in]0,\delta']$ be such that

$$\forall (\xi,\mu) \in \mathrm{B}_{\delta''}(x, f(x)+g(x)) \cap \mathrm{epi}(f+g) :\ \|\xi - x\|^2 + (\mu - g(\xi) - f(x))^2 < {\delta'}^2.$$

Define

$$\mathcal{W} : (\mathrm{B}_{\delta''}(x, f(x)+g(x)) \cap \mathrm{epi}(f+g)) \times]0,\delta''] \to \mathrm{B}_\varepsilon(v)$$

by

$$\mathcal{W}((\xi,\mu),t) = \mathcal{V}((\xi,\mu - g(\xi)),t).$$

Of course, $\mathcal{W}$ is continuous and we have

$$\begin{aligned} f(\xi + t\mathcal{W}((\xi,\mu),t)) &+ g(\xi + t\mathcal{W}((\xi,\mu),t)) \\ &= \mu + f(\xi + t\mathcal{V}((\xi,\mu - g(\xi)),t)) - (\mu - g(\xi)) \\ &\quad + g(\xi + t\mathcal{V}((\xi,\mu - g(\xi)),t)) - g(\xi) \le \mu + (r+s)t, \end{aligned}$$

so that $(f+g)^0_\varepsilon(x;v) \le r + s$. From the arbitrariness of ε, it follows that $(f+g)^0(x;v) \le r+s$. Going to the limit as $r \to f^0(x;v)$ and $s \to g^0(x;v)$, we conclude that

$$(f+g)^0(x;v) \le f^0(x;v) + g^0(x;v).$$

Now let $v \in X$ with $f^0(x;v) < +\infty$ and $g^0(x;v) < +\infty$ and let

$$v_h = \left(1 - \frac{1}{h}\right) v + \frac{1}{h} v_0.$$

From Theorem 4.9, it follows that $f^0(x;v_h) < +\infty$ and $v_h \in \mathrm{D}_g(x)$. From the previous step, we deduce that

$$(f+g)^0(x;v_h) \le f^0(x;v_h) + g^0(x;v_h),$$

hence the assertion, going to the limit as $h \to \infty$.

(ii) If $f^0(x;0) = -\infty$ or $g^0(x;0) = -\infty$, from (i) we deduce that $(f+g)^0(x;0) = -\infty$; hence

$$\partial(f+g)(x) = \partial f(x) + \partial g(x) = \emptyset.$$

Therefore, let $f^0(x;0) = g^0(x;0) = 0$. Define $\varphi, \psi : X \to \mathbb{R} \cup \{+\infty\}$ by $\varphi(v) = f^0(x;v)$ and $\psi(v) = g^0(x;v)$. From Corollary 4.6 we know that φ and ψ are convex and lower semicontinuous with $\varphi(0) = \psi(0) = 0$. Moreover, from Theorem 4.9 and (i) we deduce that

$$v_0 \in \mathrm{Dom}(\varphi) \cap \mathrm{int}(\mathrm{Dom}(\psi)),$$

$$\forall v \in X : (f+g)^0(x;v) \le \varphi(v) + \psi(v).$$

It follows that $\mathrm{Dom}(\psi) - \mathrm{Dom}(\varphi) = X$; hence, taking into account [3, Corollary (2.1)],

$$\partial(f+g)(x) \subseteq \partial(\varphi+\psi)(0) = \partial\varphi(0) + \partial\psi(0) = \partial f(x) + \partial g(x).$$

In particular, $\partial f(x) + \partial g(x)$ is weak* closed in X^*. □

The next example shows that the continuity condition for g on

$$\{\xi \in X : g(\xi) < +\infty\}$$

cannot be omitted.

Example 5.2. Define $f, g : \mathbb{R}^2 \to \mathbb{R}$ by

$$f(x,y) = \begin{cases} 0 & \text{if } x \le -y^2, \\ -1 & \text{if } x > -y^2, \end{cases}$$

$$g(x,y) = \begin{cases} 0 & \text{if } x \ge y^2, \\ 1 & \text{if } (x,y) = \left(0, \frac{1}{n}\right),\ n \ge 1, \\ 2 & \text{otherwise.} \end{cases}$$

Then $(1,0) \in \mathrm{D}_f(0,0) \cap \mathrm{D}_g(0,0)$, but we have

$$f^0((0,0);(1,0)) = g^0((0,0);(1,0)) = 0,$$

$$\partial(f+g)(0,0) = \mathbb{R}^2,$$

so that

$$\partial(f+g)(0,0) \not\subseteq \partial f(0,0) + \partial g(0,0).$$

On the other hand, f is only upper semicontinuous, while g is only lower semicontinuous.

COROLLARY 5.3. *Let $g : X \to \mathbb{R}$ be locally Lipschitzian and let $x \in X$ with $f(x) \in \mathbb{R}$.*

Then the following facts hold:

(i) *for every $v \in X$ we have $(f+g)^0(x;v) \le f^0(x;v) + g^0(x;v)$;*

(ii) *$\partial f(x) + \partial g(x)$ is weak* closed in X^* and we have*

$$\partial(f+g)(x) \subseteq \partial f(x) + \partial g(x).$$

Moreover, if g is of class C^1, then equality holds in (i) *and* (ii).

Proof. Since $\mathrm{D}_g(x) = X$, we have $0 \in \mathrm{D}_g(x)$, $f^0(x;0) < +\infty$, and the locally Lipschitz case follows from Theorem 5.1. If g is of class C^1, from Corollary 4.10 we deduce that $\partial g(x) = \{g'(x)\}$. Then it is sufficient to apply the previous case also to the decomposition $f = (f+g) + (-g)$. □

COROLLARY 5.4. *Let $C \subseteq X$ and let $x \in C$ with $f(x) \in \mathbb{R}$. Assume there exists $v_0 \in \mathrm{Hyp}_C(x)$ with $f^0(x;v_0) < +\infty$.*

Then the following facts hold:

(i) *for every $v \in \mathrm{T}_C(x)$ we have $(f+I_C)^0(x;v) \le f^0(x;v)$;*

(ii) *$\partial f(x) + \mathrm{N}_C(x)$ is weak* closed in X^* and we have*

$$\partial(f+I_C)(x) \subseteq \partial f(x) + \mathrm{N}_C(x).$$

Proof. Let $g = I_C$. It is readily seen that $(v_0,1) \in \mathrm{Hyp}_{\mathrm{epi}(g)}(x,0)$. From [12, Proposition 2.9.3] it follows that $v_0 \in \mathrm{D}_g(x)$. From Theorem 5.1 we deduce (ii). Moreover, if $v \in \mathrm{T}_C(x)$, we have $(v,0) \in \mathrm{T}_{\mathrm{epi}(g)}(x,0)$; hence $g^0(x;v) \le 0$. Then (i) also follows. □

DEFINITION 5.5. *Let $x \in X$ with $f(x) \in \mathbb{R}$. For every $v \in X$ and $\varepsilon > 0$ let $\overline{f}^0_\varepsilon(x;v)$ be the infimum of the r's in $\mathbb{R}$ such that there exist $\delta > 0$ and a continuous map*

$$\mathcal{H} : (\mathrm{B}_\delta(x,f(x)) \cap \mathrm{epi}(f)) \times [0,\delta] \to X$$

satisfying $\mathcal{H}((\xi,\mu),0) = \xi$,

$$\left\| \frac{\mathcal{H}((\xi,\mu),t_1) - \mathcal{H}((\xi,\mu),t_2)}{t_1 - t_2} - v \right\| < \varepsilon,$$

$$f(\mathcal{H}((\xi,\mu),t)) \le \mu + rt$$

whenever $(\xi,\mu) \in \mathrm{B}_\delta(x,f(x)) \cap \mathrm{epi}(f)$ and $t, t_1, t_2 \in [0,\delta]$ with $t_1 \ne t_2$. (We agree that $\inf\emptyset = +\infty$.*)*

Let also

$$\overline{f}^0(x;v) := \sup_{\varepsilon>0} \overline{f}^0_\varepsilon(x;v) = \lim_{\varepsilon\to 0^+} \overline{f}^0_\varepsilon(x;v).$$

Remark 5.6. This variant of the generalized directional derivative will be used to express a qualification condition in Theorem 5.8 and in Corollary 5.9.

It is easy to see that the map

$$\mathcal{V}((\xi,\mu),t) = \frac{\mathcal{H}((\xi,\mu),t) - \xi}{t}$$

satisfies the properties required by Definition 4.3. Therefore we always have

$$f^0(x;v) \le \overline{f}^0(x;v).$$

Moreover, we will see in the next theorem that, for a restricted class of functions f, the condition $f^0(x;v) < +\infty$ is equivalent to $\overline{f}^0(x;v) < +\infty$.

THEOREM 5.7. *Assume that $f = f_0 + f_1$, where $f_0 : X \to \overline{\mathbb{R}}$ is convex and $f_1 : X \to \mathbb{R}$ is locally Lipschitzian, and let $x \in X$ with $f(x) \in \mathbb{R}$.*

Then we have

$$\forall v \in X : \overline{f}^{\,0}(x;v) < +\infty \iff f^0(x;v) < +\infty,$$

$$\forall y \in X : f(y) < +\infty \implies \overline{f}^{\,0}(x;y-x) < +\infty.$$

Proof. If L is a Lipschitz constant for f_1 in a neighborhood of x, it is readily seen that

$$\forall v \in X : \overline{f}^{\,0}(x;v) \le \overline{f_0}^{\,0}(x;v) + L\,, \quad f^0(x;v) \ge {f_0}^0(x;v) - L.$$

Therefore it is sufficient to treat the case $f_1 = 0$.

First of all, we show that

$$\forall v \in X : \overline{f}^{\,0}(x;v) = f^0(x;v)\,. \tag{5.1}$$

Let $r > f^0(x;v)$, $\varepsilon > 0$, and let $\mathcal{V} : (\mathrm{B}_\delta(x, f(x)) \cap \mathrm{epi}(f)) \times]0,\delta] \to \mathrm{B}_{\frac{\varepsilon}{2}}(v)$ be a map as in Definition 4.3. Set $w = x + \delta\mathcal{V}((x,f(x)),\delta)$, take

$$0 < \delta' \le \min\left\{\delta, \frac{\delta\varepsilon}{2}\right\},$$

and define $\mathcal{H} : (\mathrm{B}_{\delta'}(x, f(x)) \cap \mathrm{epi}(f)) \times [0,\delta'] \to X$ by $\mathcal{H}((\xi,\mu),t) = \xi + (t/\delta)(w-\xi)$. We have

$$\begin{aligned}\left\|\frac{\mathcal{H}((\xi,\mu),t_1) - \mathcal{H}((\xi,\mu),t_2)}{t_1 - t_2} - v\right\| &= \left\|\frac{w-\xi}{\delta} - v\right\| \\ &\le \frac{\|x-\xi\|}{\delta} + \|\mathcal{V}((x,f(x)),\delta) - v\| \\ &< \frac{\delta'}{\delta} + \frac{\varepsilon}{2} \le \varepsilon\end{aligned}$$

and

$$\begin{aligned}f(\mathcal{H}((\xi,\mu),t)) &\le \left(1 - \frac{t}{\delta}\right) f(\xi) + \frac{t}{\delta} f(w) \le \mu + \frac{t}{\delta}(f(w) - \mu) \\ &\le \mu + \frac{t}{\delta}(f(x) + r\delta - f(x) + \delta') = \mu + t\left(r + \frac{\delta'}{\delta}\right).\end{aligned}$$

It follows that $\overline{f}^{\,0}_{\varepsilon}(x;v) \le r + (\delta'/\delta)$, hence $\overline{f}^{\,0}_{\varepsilon}(x;v) \le r$ by the arbitrariness of δ'. Finally, we deduce that $\overline{f}^{\,0}(x;v) \le r$ by the arbitrariness of ε and (5.1) follows from the arbitrariness of r.

Now, if $y \in X$ and $f(y) < +\infty$, we have $(y - x, f(y) - f(x)) \in \mathrm{T}_{\mathrm{epi}(f)}(x, f(x))$ by Theorem 3.4; hence $f^0(x;y-x) < +\infty$ by Theorem 4.5. Therefore the proof is complete. □

THEOREM 5.8. *Let $C \subseteq X$ and let $x \in C \cap \partial C$ with $f(x) \in \mathbb{R}$. Assume there exist $v_+, v_- \in \mathrm{Hyp}_C(x)$ such that*

$$\overline{f}^{\,0}(x;v_+) < +\infty, \qquad \overline{f}^{\,0}(x;-v_-) < +\infty.$$

Then the following facts hold:

(i) $(f + I_{\partial C})^0 (x; v) \leq f^0 (x; v)$ *for any* $v \in \mathrm{T}_{\partial C}(x)$;

(ii) $\partial (f + I_{\partial C})(x) \neq \emptyset \implies \partial f(x) \neq \emptyset$.

Proof. (i) Let $v \in \mathrm{T}_{\partial C}(x)$. If $f^0 (x; v) = +\infty$, the fact is obvious. Otherwise, let $r > f^0 (x; v)$. Given $\varepsilon > 0$ and

$$0 < \varepsilon' \leq \min \left\{ 1, \frac{\varepsilon}{3\|v_+\|}, \frac{\varepsilon}{3\|v_-\|} \right\},$$

it follows from Lemma 3.7 and Theorem 3.8 that $w_+ = v + \varepsilon' v_+ \in \mathrm{Hyp}_C(x)$ and $w_- = v - \varepsilon' v_- \in \mathrm{Hyp}_{\overline{X \setminus C}}(x)$. Let $\delta > 0$ be such that

$$\left(\mathrm{B}_\delta (x) \cap \overline{C}\right) +]0, \delta] \cdot \mathrm{B}_\delta (v_+) \subseteq \mathrm{int}\,(C),$$

$$\left(\mathrm{B}_\delta (x) \cap \overline{C}\right) +]0, \delta] \cdot \mathrm{B}_\delta (w_+) \subseteq \mathrm{int}\,(C),$$

$$\left(\mathrm{B}_\delta (x) \cap \overline{X \setminus C}\right) +]0, \delta] \cdot \mathrm{B}_\delta (-v_-) \subseteq X \setminus \overline{C},$$

$$\left(\mathrm{B}_\delta (x) \cap \overline{X \setminus C}\right) +]0, \delta] \cdot \mathrm{B}_\delta (w_-) \subseteq X \setminus \overline{C}.$$

Given $s_+ > \overline{f}^{\,0} (x; v_+)$ and $s_- > \overline{f}^{\,0} (x; -v_-)$, let

$$\varepsilon'' = \min \left\{ \frac{\delta}{2}, \frac{\varepsilon}{3} \right\},$$

let $\delta' \in]0, \delta]$, and let

$$\mathcal{H}_+ : (\mathrm{B}_{\delta'} (x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta'] \to X,$$

$$\mathcal{H}_- : (\mathrm{B}_{\delta'} (x, f(x)) \cap \mathrm{epi}\,(f)) \times [0, \delta'] \to X,$$

$$\mathcal{V} : (\mathrm{B}_{\delta'} (x, f(x)) \cap \mathrm{epi}\,(f)) \times]0, \delta'] \to \mathrm{B}_{\varepsilon''} (v)$$

be three continuous maps such that $\mathcal{H}_+((\xi, \mu), 0) = \xi$, $\mathcal{H}_-((\xi, \mu), 0) = \xi$,

$$\left\| \frac{\mathcal{H}_+((\xi, \mu), t_1) - \mathcal{H}_+((\xi, \mu), t_2)}{t_1 - t_2} - v_+ \right\| < \varepsilon'',$$

$$f(\mathcal{H}_+((\xi, \mu), t)) \leq \mu + s_+ t,$$

$$\left\| \frac{\mathcal{H}_-((\xi, \mu), t_1) - \mathcal{H}_-((\xi, \mu), t_2)}{t_1 - t_2} + v_- \right\| < \varepsilon'',$$

$$f(\mathcal{H}_-((\xi, \mu), t)) \leq \mu + s_- t,$$

$$f(\xi + t\mathcal{V}((\xi, \mu), t)) \leq \mu + rt$$

for any $(\xi,\mu) \in \mathrm{B}_{\delta'}\,(x, f(x)) \cap \mathrm{epi}\,(f)$, $t \in]0,\delta']$ and $t_1, t_2 \in [0,\delta']$ with $t_1 \neq t_2$. Set $\mathcal{K}((\xi,\mu),t) = \xi + t\mathcal{V}((\xi,\mu),t)$ and choose $\delta'' \in]0,\delta']$ such that

$$\delta''\,(2\varepsilon'' + 1 + \|v_+\| + \|v_-\| + \|v\|) \leq \delta,$$

$$\|\mathcal{K}((\xi,\mu),t) - x\|^2 + (\mu + rt - f(x))^2 < {\delta'}^2$$

for any $(\xi,\mu) \in \mathrm{B}_{\delta''}\,(x, f(x)) \cap \mathrm{epi}\,(f)$ and $t \in]0,\delta'']$.

If $(\xi,\mu) \in \mathrm{B}_{\delta''}\,(x, f(x)) \cap \mathrm{epi}\,(f)$, $t \in]0,\delta'']$ and $\xi \in \partial C$, we have

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t), \mu + rt), \varepsilon' t)$$

$$= \xi + t\left(\mathcal{V}((\xi,\mu),t) + \varepsilon'\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon' t) - \mathcal{K}((\xi,\mu),t)}{\varepsilon' t}\right) \in \mathrm{int}\,(C)\,,$$

as

$$\left\|\mathcal{V}((\xi,\mu),t) + \varepsilon'\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon' t) - \mathcal{K}((\xi,\mu),t)}{\varepsilon' t} - w_+\right\|$$

$$\leq \varepsilon'\left\|\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon' t) - \mathcal{K}((\xi,\mu),t)}{\varepsilon' t} - v_+\right\| + \|\mathcal{V}((\xi,\mu),t) - v\|$$

$$< \varepsilon'\varepsilon'' + \varepsilon'' \leq \delta.$$

Therefore, if $\mathcal{K}((\xi,\mu),t) \notin \mathrm{int}\,(C)$, there exists $\tau^+ \in [0,1[$ such that

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau^+ t) \in \partial C.$$

We claim that such a τ^+ is unique. By contradiction, let τ_1^+ and τ_2^+ be such that $0 \leq \tau_1^+ < \tau_2^+ < 1$ and

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_j^+ t) \in \partial C.$$

We have

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_2^+ t) = \mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+ t)$$

$$+\varepsilon'(\tau_2^+ - \tau_1^+)t\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_2^+ t) - \mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+ t)}{\varepsilon'(\tau_2^+ - \tau_1^+)t}.$$

But $\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+ t) \in \mathrm{B}_\delta\,(x)$, as

$$\|\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+ t) - x\|$$

$$\leq \varepsilon'\tau_1^+ t\left\|\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+ t) - \mathcal{K}((\xi,\mu),t)}{\varepsilon'\tau_1^+ t} - v_+\right\|$$

$$+t\|\mathcal{V}((\xi,\mu),t) - v\| + \|\xi - x\| + \|\varepsilon'\tau_1^+ t v_+ + tv\| < \delta''\varepsilon'' + \delta''\varepsilon'' + \delta'' + \delta''\|v_+\| + \delta''\|v\| \leq \delta.$$

Moreover $0 < \varepsilon' \left(\tau_2^+ - \tau_1^+\right) t < \delta$ and

$$\frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_2^+t) - \mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_1^+t)}{\varepsilon'(\tau_2^+ - \tau_1^+)t} \in \mathrm{B}_\delta\left(v_+\right).$$

It follows that

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau_2^+t) \in \operatorname{int}(C),$$

which is absurd.

Therefore we may define a continuous map $\tau^+ : C^+ \to [0,1[$ with

$$C^+ = \{((\xi,\mu),t) \in (\mathrm{B}_{\delta''}(x,f(x)) \cap \operatorname{epi}(f)) \times]0,\delta''] : \xi \in \partial C, \mathcal{K}((\xi,\mu),t) \notin \operatorname{int}(C)\}$$

such that for every $((\xi,\mu),t) \in C^+$ the following facts hold:

$$\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon_1\tau^+((\xi,\mu),t)t) \in \partial C,$$

$$\mathcal{K}((\xi,\mu),t) \in \partial C \implies \tau^+((\xi,\mu),t) = 0.$$

On the other hand, if $(\xi,\mu) \in \mathrm{B}_{\delta''}(x,f(x)) \cap \operatorname{epi}(f)$, $t \in]0,\delta'']$ and $\xi \in \partial C$, as previously we have

$$\mathcal{H}_-((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon't) \in X \setminus \overline{C}.$$

In a similar way we may define a continuous map $\tau^- : C^- \to [0,1[$ with

$$C^- = \left\{((\xi,\mu),t) \in (\mathrm{B}_{\delta''}(x,f(x)) \cap \operatorname{epi}(f)) \times]0,\delta''] : \xi \in \partial C, \mathcal{K}((\xi,\mu),t) \in \overline{C}\right\}$$

such that for every $((\xi,\mu),t) \in C^-$ the following facts hold:

$$\mathcal{H}_-((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau^-((\xi,\mu),t)t) \in \partial C,$$

$$\mathcal{K}((\xi,\mu),t) \in \partial C \implies \tau^-((\xi,\mu),t) = 0.$$

Consider now the continuous map

$$\mathcal{V}' : (\mathrm{B}_{\delta''}(x,f(x)) \cap \operatorname{epi}(f) \cap (\partial C \times \mathbb{R})) \times]0,\delta''] \to \mathrm{B}_\varepsilon(v)$$

defined by

$$\mathcal{V}'((\xi,\mu),t) = \frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau^+((\xi,\mu),t)t) - \xi}{t}$$

if $\mathcal{K}((\xi,\mu),t) \notin \operatorname{int}(C)$ and

$$\mathcal{V}'((\xi,\mu),t) = \frac{\mathcal{H}_-((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau^-((\xi,\mu),t)t) - \xi}{t}$$

if $\mathcal{K}((\xi,\mu),t) \in \overline{C}$.

Actually, if $\mathcal{K}((\xi,\mu),t) \notin \mathrm{int}\,(C)$, we have

$$\|\mathcal{V}'((\xi,\mu),t)-v\|$$

$$\leq \varepsilon' \left\| \frac{\mathcal{H}_+((\mathcal{K}((\xi,\mu),t),\mu+rt),\varepsilon'\tau^+((\xi,\mu),t)t)-\mathcal{K}((\xi,\mu),t)}{\varepsilon' t} - v_+ \right\|$$

$$+\,\varepsilon'\|v_+\| + \|\mathcal{V}((\xi,\mu),t)-v\| < \varepsilon'\varepsilon'' + \varepsilon'' + \varepsilon'\|v_+\| \leq \varepsilon,$$

and a similar inequality holds when $\mathcal{K}((\xi,\mu),t) \in \overline{C}$. Moreover

$$f(\xi + t\mathcal{V}'((\xi,\mu),t)) \leq \mu + rt + \max\{s_+, s_-, 0\}\varepsilon' t.$$

It follows that

$$(f + I_{\partial C})^0_\varepsilon\,(x;v) \leq r + \max\{s_+, s_-, 0\}\,\varepsilon';$$

hence, going to the limit as $\varepsilon' \to 0^+$,

$$(f + I_{\partial C})^0_\varepsilon\,(x;v) \leq r.$$

Then assertion (i) follows from the arbitrariness of r and ε.

(ii) Since $(f + I_{\partial C})^0\,(x;0) > -\infty$ implies $f^0\,(x;0) > -\infty$, the assertion follows from Corollary 4.7. □

As a corollary, we can now deduce a Lagrange multipliers theorem. When f belongs to suitable functional classes, related results are contained in [8, 11].

Corollary 5.9. *Let U be an open subset of X and let $x \in \partial U$ with $f(x) \in \mathbb{R}$. Assume that ∂U is of class C^1, and denote by $\nu(x) \in X^* \setminus \{0\}$ an outer normal vector to U at x. Suppose there exist $v_+, v_- \in X$ such that*

$$\langle \nu(x), v_+ \rangle < 0, \qquad \langle \nu(x), v_- \rangle < 0,$$

$$\overline{f}^{\,0}\,(x;v_+) < +\infty, \qquad \overline{f}^{\,0}\,(x;-v_-) < +\infty.$$

Then the following facts hold:

(i) $(f + I_{\partial U})^0\,(x;v) \leq f^0\,(x;v)$ *for any* $v \in \mathrm{T}_{\partial U}\,(x)$*;*

(ii) $\partial f(x) + \mathbb{R}\nu(x)$ *is weak* closed in* X^* *and we have*

$$\partial\,(f + I_{\partial U})\,(x) \subseteq \partial f(x) + \mathbb{R}\nu(x).$$

Proof. (i) If we set $C = \overline{U}$, it is readily seen that $\partial C = \partial U$ and

$$\mathrm{Hyp}_C\,(x) = \{v \in X : \langle \nu(x), v \rangle < 0\}\,,$$

$$\mathrm{T}_{\partial U}\,(x) = \{v \in X : \langle \nu(x), v \rangle = 0\}\,.$$

Then the assertion is a particular case of Theorem 5.8.

(ii) If $f^0\,(x;0) = -\infty$, from (i) we deduce that $(f + I_{\partial U})^0\,(x;0) = -\infty$. By Corollary 4.7 we have

$$\partial f(x) = \emptyset, \qquad \partial\,(f + I_{\partial U}) = \emptyset,$$

so that (ii) clearly follows.

Otherwise, observe that the functions $f^0(x;\cdot)$ and $I_{\mathrm{T}_{\partial U}(x)}$ are convex, lower semicontinuous, and with values in $\mathbb{R}\cup\{+\infty\}$. Moreover, we have

$$\mathrm{Dom}\left(f^0(x;\cdot)\right) - \mathrm{T}_{\partial U}(x) = X.$$

From [3, Corollary (2.1)] it follows that

$$\partial\left(f^0(x;\cdot) + I_{\mathrm{T}_{\partial U}(x)}\right)(0) = \partial\left(f^0(x;\cdot)\right)(0) + \partial I_{\mathrm{T}_{\partial U}(x)}(0).$$

On the other hand,

$$\partial\left(f^0(x;\cdot)\right)(0) + \partial I_{\mathrm{T}_{\partial U}(x)}(0) = \partial f(x) + \mathbb{R}\nu(x),$$

so that $\partial f(x) + \mathbb{R}\nu(x)$ is weak* closed in X^*.

Let now $\alpha \in \partial(f + I_{\partial U})(x)$. From (i) we deduce that

$$\forall v \in \mathrm{T}_{\partial U}(x) : \langle \alpha, v\rangle \le f^0(x;v).$$

Since $f^0(x;0) = 0$, we have

$$\alpha \in \partial\left(f^0(x;\cdot) + I_{\mathrm{T}_{\partial U}(x)}\right)(0);$$

hence

$$\alpha \in \partial f(x) + \mathbb{R}\nu(x)$$

and the proof is complete. □

Example 5.10. Define a lower semicontinuous function $f : \mathbb{R}^3 \to \mathbb{R}\cup\{+\infty\}$ by

$$f(x,y,z) = \begin{cases} -\sqrt{xy} & \text{if } x \ge 0 \text{ and } y \ge 0, \\ \sqrt{-y} & \text{if } x \ge 0 \text{ and } y \le 0, \\ +\infty & \text{otherwise} \end{cases}$$

and set

$$C = \left\{(x,y,z) \in \mathbb{R}^3 : |y| \le z\right\}.$$

Then the assumptions of Theorem 5.8 are satisfied with respect to the origin, but

$$\partial\left(f + I_{\partial C}\right)(0,0,0) \not\subseteq \partial f(0,0,0) + \mathrm{N}_{\partial C}(0,0,0).$$

More precisely, we have

$$\left(f + I_{\partial C}\right)^0((0,0,0);(u,v,w)) = \begin{cases} 0 & \text{if } u \ge 0 \text{ and } v = w = 0, \\ +\infty & \text{otherwise,} \end{cases}$$

$$f^0((0,0,0);(u,v,w)) = \begin{cases} -\sqrt{uv} & \text{if } u \ge 0 \text{ and } v \ge 0, \\ +\infty & \text{otherwise,} \end{cases}$$

and $\mathrm{T}_{\partial C}(0,0,0) = \{(u,0,0) : u \in \mathbb{R}\}$, according to Theorem 3.8. Therefore it follows that $(0,0,0) \in \partial\left(f + I_{\partial C}\right)(0,0,0)$, while $(0,\lambda,\mu) \notin \partial f(0,0,0)$ for any $\lambda, \mu \in \mathbb{R}$.

6. Functionals of the calculus of variations and PDEs. In this section we consider a typical functional of the calculus of variations for which the generalized directional derivative and the subdifferential can be estimated in a useful way. In particular, the subdifferential has the expression one may expect and, therefore, is not too large. Moreover we sketch, following [24], an example of application to nonlinear PDEs.

Let Ω be an open subset of $\mathbb{R}^n$, let $1 \leq p < n$ and let $L : \Omega \times \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ be a function such that

(i) for almost everywhere (a.e.) $x \in \Omega$ the function $\{(s, \xi) \longmapsto L(x, s, \xi)\}$ is of class C^1 on $\mathbb{R} \times \mathbb{R}^n$;

(ii) for every $(s, \xi) \in \mathbb{R} \times \mathbb{R}^n$ the function $\{x \longmapsto L(x, s, \xi)\}$ is measurable on Ω.

Let $f : W_0^{1,p}(\Omega) \to \overline{\mathbb{R}}$ be the functional defined by

$$f(u) = \int_\Omega L(x, u, \nabla u)\, dx,$$

where, as in [34], we agree that $f(u) = +\infty$ whenever

$$\int_\Omega (L(x, u, \nabla u))^+ \, dx = \int_\Omega (L(x, u, \nabla u))^- \, dx = +\infty.$$

Assume there exist $a \in L^1_{loc}(\Omega)$ and $b \in L^\infty_{loc}(\Omega)$ such that

$$|D_{\xi_j} L(x, s, \xi)| \leq a(x) + b(x) \left(|s|^{\frac{np}{n-p}} + |\xi|^p\right),$$

$$|D_s L(x, s, \xi)| \leq a(x) + b(x) \left(|s|^{\frac{np}{n-p}} + |\xi|^p\right),$$

for a.e. $x \in \Omega$ and every $s \in \mathbb{R}$, $\xi \in \mathbb{R}^n$.

From the Sobolev theorem, it is easy to deduce that

$$D_{\xi_j} L(x, u, \nabla u) \in L^1_{loc}(\Omega), \qquad D_s L(x, u, \nabla u) \in L^1_{loc}(\Omega)$$

for every $u \in W_0^{1,p}(\Omega)$, so that

$$-\sum_{j=1}^n D_{x_j}\left[D_{\xi_j} L(x, u, \nabla u)\right] + D_s L(x, u, \nabla u)$$

defines a distribution on Ω.

Theorem 6.1. *Let $u \in W_0^{1,p}(\Omega)$ with $f(u) \in \mathbb{R}$. Then the following facts hold:*

(i) *For every $v \in C_c^\infty(\Omega)$ we have*

$$f^0(u; v) \leq \overline{f}^0(u; v) \leq \int_\Omega \left(\sum_{j=1}^n D_{\xi_j} L(x, u, \nabla u)\, D_{x_j} v + D_s L(x, u, \nabla u)\, v\right) dx.$$

(ii) *If $\partial f(u) \neq \emptyset$, we have*

$$-\sum_{j=1}^n D_{x_j}\left[D_{\xi_j} L(x, u, \nabla u)\right] + D_s L(x, u, \nabla u) \in W^{-1,p'}(\Omega),$$

$$\partial f(u) = \left\{-\sum_{j=1}^n D_{x_j}\left[D_{\xi_j} L(x, u, \nabla u)\right] + D_s L(x, u, \nabla u)\right\}.$$

Proof. (i) Let $v \in C_c^\infty(\Omega)$, let $\varepsilon > 0$, and let

$$r > \int_\Omega \sum_{j=1}^n D_{\xi_j} L(x, u, \nabla u)\, D_{x_j} v + D_s L(x, u, \nabla u)\, v\, dx.$$

There exists $\delta > 0$ such that

$$\forall w \in W_0^{1,p}(\Omega) : \|w - u\|_{1,p} < \delta \implies \\ \int_\Omega \sum_{j=1}^n D_{\xi_j} L(x, w, \nabla w)\, D_{x_j} v + D_s L(x, w, \nabla w)\, v\, dx < r,$$

where $\|\cdot\|_{1,p}$ denotes the norm in $W_0^{1,p}(\Omega)$. Let $\delta' > 0$ be such that $\delta' + \delta'\|v\|_{1,p} < \delta$, and let

$$\mathcal{H} : (\mathrm{B}_{\delta'}\,(u, f(u)) \cap \mathrm{epi}\,(f)) \times [0, \delta'] \to W_0^{1,p}(\Omega)$$

be defined by $\mathcal{H}((w, \mu), t) = w + tv$. For every $w \in W_0^{1,p}(\Omega)$ with $\|w - u\|_{1,p} < \delta'$ and $f(w) < +\infty$ and every $t \in [0, \delta']$ we have

$$\begin{aligned} L(x, w + tv, \nabla w + t\nabla v) = L(x, w, \nabla w) & \\ + t\Bigg[\int_0^1 \Bigg(\sum_{j=1}^n & D_{\xi_j} L(x, w + \vartheta tv, \nabla w + \vartheta t\nabla v)\, D_{x_j} v \\ & + D_s L(x, w + \vartheta tv, \nabla w + \vartheta t\nabla v) v\Bigg)\, d\vartheta\Bigg]. \end{aligned}$$

It follows that $f(w + tv) < +\infty$ and

$$\begin{aligned} \int_\Omega L(x, w + tv, \nabla w + t\nabla v)\, dx = & \int_\Omega L(x, w, \nabla w)\, dx \\ & + t\Bigg[\int_0^1 \int_\Omega \Bigg(\sum_{j=1}^n D_{\xi_j} L(x, w{+}\vartheta tv, \nabla w{+}\vartheta t\nabla v)\, D_{x_j} v \\ & \qquad + D_s L(x, w + \vartheta tv, \nabla w + \vartheta t\nabla v) v\Bigg)\, dx d\vartheta\Bigg] \\ \le & \int_\Omega L(x, w, \nabla w)\, dx + rt, \end{aligned}$$

hence

$$f(\mathcal{H}((w, \mu), t)) \le f(w) + rt \le \mu + rt.$$

Therefore we have $\overline{f}_\varepsilon^{\,0}\,(x; v) \le r$ and the assertion follows from the arbitrariness of r and ε.

(ii) Let $\alpha \in \partial f(u) \subseteq W^{-1,p'}(\Omega)$. For every $v \in C_c^\infty(\Omega)$ we have

$$\langle \alpha, v\rangle \le f^0\,(x; v) \le \int_\Omega \sum_{j=1}^n D_{\xi_j} L(x, u, \nabla u)\, D_{x_j} v + D_s L(x, u, \nabla u)\, v\, dx.$$

Since we may exchange v with $-v$, we deduce that

$$\forall v \in C_c^\infty(\Omega) : \int_\Omega \sum_{j=1}^n D_{\xi_j} L(x, u, \nabla u)\, D_{x_j} v + D_s L(x, u, \nabla u)\, v\, dx = \langle \alpha, v \rangle$$

and the assertion follows. □

Now let Ω be a bounded open subset of $\mathbb{R}^n$ with $n \geq 3$, let $2 < q < \frac{2n}{n-2}$, let $a \in L^1(\Omega)$, and let $g : \mathbb{R} \to \mathbb{R}$ be an odd continuous function with compact support.

Theorem 6.2. *The semilinear elliptic problem*

$$\begin{cases} -\Delta u = |u|^{q-2}u + a(x)g(u) & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega \end{cases} \tag{6.1}$$

admits a sequence (u_h) of weak solutions with $\|u_h\|_{W_0^{1,2}} \to \infty$.

The above result is a special case of [24, Theorem 6.2]. If $a \in L^{\frac{2n}{n+2}}(\Omega)$, then it follows from a celebrated theorem of Ambrosetti and Rabinowitz [1, 32, 36]. To have a sketch of the proof, consider, as usual, the functional $f : W_0^{1,2}(\Omega) \to \mathbb{R}$ defined by

$$f(u) = \frac{1}{2} \int_\Omega |\nabla u|^2\, dx - \int_\Omega \left(\frac{1}{q}\, |u|^q + a(x)G(u) \right) dx\,, \qquad G(s) = \int_0^s g(t)\, dt.$$

If $a \in L^{\frac{2n}{n+2}}(\Omega)$, then f turns out to be of class C^1 and classical critical point theory applies. On the contrary, if $a \in L^1(\Omega)$ we can ensure only the continuity of f. However, nonsmooth critical point theory still yields the existence of a sequence (u_h) in $W_0^{1,2}(\Omega)$ with $|df|\,(u_h) = 0$ and $\|u_h\|_{W_0^{1,2}} \to \infty$. From Theorem 4.13 it follows that $0 \in \partial f(u_h)$. Then Theorem 6.1 allows us to conclude that each u_h is a weak solution of (6.1).

On the contrary, we do not know if there exist u's in $W_0^{1,2}(\Omega)$ with $0 \in \partial_C f(u)$, where ∂_C denotes Clarke's subdifferential.

REFERENCES

[1] A. Ambrosetti and P. H. Rabinowitz, *Dual variational methods in critical point theory and applications*, J. Funct. Anal., 14 (1973), pp. 349–381.

[2] G. Arioli and F. Gazzola, *Quasilinear elliptic equations at critical growth*, NoDEA Nonlinear Differential Equations Appl., 5 (1998), pp. 83–97.

[3] H. Attouch and H. Brezis, *Duality for the sum of convex functions in general Banach spaces*, in Aspects of Mathematics and its Applications, J. A. Barroso, ed., North–Holland Math. Library 34, North–Holland, Amsterdam, New York, 1986, pp. 125–133.

[4] A. Canino, *Multiplicity of solutions for quasilinear elliptic equations*, Topol. Methods Nonlinear Anal., 6 (1995), pp. 357–370.

[5] A. Canino, *On a variational approach to some quasilinear problems*, in Well-Posed Problems and Stability in Optimization, Y. Sonntag, ed., Serdica Math. J. 22, 1996, pp. 399-426.

[6] A. Canino, *On a jumping problem for quasilinear elliptic equations*, Math. Z., 226 (1997), pp. 193–210.

[7] A. Canino and M. Degiovanni, *Nonsmooth critical point theory and quasilinear elliptic equations*, in Topological Methods in Differential Equations and Inclusions, A. Granas, M. Frigon, and G. Sabidussi, eds., NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci. 472, Kluwer Academic Publishers, Dordrecht, 1995, pp. 1–50.

[8] A. Canino and U. Perri, *Constrained problems in Banach spaces with an application to variational inequalities*, Nonlinear Anal., 24 (1995), pp. 839–856.

[9] K. C. Chang, *Variational methods for non-differentiable functionals and their applications to partial differential equations*, J. Math. Anal. Appl., 80 (1981), pp. 102–129.

[10] K. C. Chang, *Infinite Dimensional Morse Theory and Multiple Solution Problems*, Progress in Nonlinear Differential Equations and Their Applications 6, Birkhäuser, Boston, 1993.

[11] G. Chobanov, A. Marino, and D. Scolozzi, *Evolution equation for the eigenvalue problem for the Laplace operator with respect to an obstacle*, Rend. Accad. Naz. Sci. XL Mem. Mat., 14 (1990), pp. 139–162.

[12] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Canadian Mathematical Society Series of Monographs and Advanced Texts, John Wiley & Sons, New York, 1983.

[13] M. Conti and F. Gazzola, *Positive entire solutions of quasilinear elliptic problems via nonsmooth critical point theory*, Topol. Methods Nonlinear Anal., 8 (1996), pp. 275–294.

[14] J.-N. Corvellec, *Morse theory for continuous functionals*, J. Math. Anal. Appl., 196 (1995), pp. 1050–1072.

[15] J.-N. Corvellec and M. Degiovanni, *Nontrivial solutions of quasilinear equations via nonsmooth Morse theory*, J. Differential Equations, 136 (1997), pp. 268–293.

[16] J.-N. Corvellec, M. Degiovanni, and M. Marzocchi, *Deformation properties for continuous functionals and critical point theory*, Topol. Methods Nonlinear Anal., 1 (1993), pp. 151–171.

[17] E. De Giorgi, A. Marino, and M. Tosques, *Problemi di evoluzione in spazi metrici e curve di massima pendenza*, Atti Accad. Naz. Lincei Rend. Cl. Sci. Fis. Mat. Natur. (8), 68 (1980), pp. 180–187.

[18] M. Degiovanni, *Homotopical properties of a class of nonsmooth functions*, Ann. Mat. Pura Appl. (4), 156 (1990), pp. 37–71.

[19] M. Degiovanni, *Nonsmooth critical point theory and applications*, in Second World Congress of Nonlinear Analysts, V. Lakshmikantham, ed., Nonlinear Anal., 30 (1997), pp. 89–99.

[20] M. Degiovanni, A. Marino, and M. Tosques, *Evolution equations with lack of convexity*, Nonlinear Anal., 9 (1985), pp. 1401–1443.

[21] M. Degiovanni and M. Marzocchi, *A critical point theory for nonsmooth functionals*, Ann. Mat. Pura Appl. (4), 167 (1994), pp. 73–100.

[22] M. Degiovanni, M. Marzocchi, and V. D. Rădulescu, *Multiple solutions of hemivariational inequalities with area-type term*, Calc. Var. Partial Differential Equations, to appear.

[23] M. Degiovanni and F. Schuricht, *Buckling of nonlinearly elastic rods in the presence of obstacles treated by nonsmooth critical point theory*, Math. Ann., 311 (1998), pp. 675–728.

[24] M. Degiovanni and S. Zani, *Multiple solutions of semilinear elliptic equations with one-sided growth conditions*, in Advanced Topics in Nonlinear Operator Theory, Math. Comput. Modelling, to appear.

[25] A. Ioffe, *Approximate subdifferentials and applications. The metric theory*, Mathematika, 36 (1989), pp. 1–38.

[26] A. Ioffe, *Non-smooth subdifferentials: Their calculus and applications*, in World Congress of Nonlinear Analysts '92, V. Lakshmikantham, ed., de Gruyter, Berlin, New York, 1996, pp. 2299–2310.

[27] A. Ioffe and E. Schwartzman, *Metric critical point theory* 1. *Morse regularity and homotopic stability of a minimum*, J. Math. Pures Appl., 75 (1996), pp. 125–153.

[28] A. Ioffe and E. Schwartzman, *Metric critical point theory* 2. *Deformation techniques*, in New Results in Operator Theory and Its Applications, I. Gohberg and Yu. Lyubich, eds., Oper. Theory Adv. Appl., 98, Birkhäuser, Basel, 1997, pp. 131–144.

[29] G. Katriel, *Mountain pass theorems and global homeomorphism theorems*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 11 (1994), pp. 189-209.

[30] A. Marino and D. Scolozzi, *Geodetiche con ostacolo*, Boll. Un. Mat. Ital. B (6), 2 (1983), pp. 1–31.

[31] J. Mawhin and M. Willem, *Critical Point Theory and Hamiltonian Systems*, Appl. Math. Sci., 74, Springer-Verlag, New York, Berlin, 1989.

[32] P. H. Rabinowitz, *Minimax Methods in Critical Point Theory with Applications to Differential Equations*, CBMS Regional Conf. Ser. Math. 65, AMS, Providence, RI, 1986.

[33] N. K. Ribarska, Ts. Y. Tsachev, and M. I. Krastanov, *Speculating about mountains*, in Well-Posed Problems and Stability in Optimization, Y. Sonntag, ed., Serdica Math. J. 22, 1996, pp. 341–358.

[34] R. T. Rockafellar, *Integral functionals, normal integrands and measurable selections*, in Nonlinear Operators and the Calculus of Variations, J. P. Gossez, E. J. Lami Dozo, J.

Mawhin, and L. Waelbroeck, eds., Lecture Notes in Math. 543, Springer, Berlin, New York, 1976, pp. 157–207.

[35] R. T. Rockafellar, *Generalized directional derivatives and subgradients of nonconvex functions*, Canad. J. Math., 32 (1980), pp. 257–280.

[36] M. Struwe, *Variational Methods*, Springer-Verlag, Berlin, 1990.

[37] A. Szulkin, *Minimax principles for lower semicontinuous functions and applications to nonlinear boundary value problems*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 3 (1986), pp. 77–109.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1049–1057

ON FINITELY TERMINATING BRANCH-AND-BOUND ALGORITHMS FOR SOME GLOBAL OPTIMIZATION PROBLEMS[*]

FAIZ A. AL-KHAYYAL[†] AND HANIF D. SHERALI[‡]

Abstract. Global optimization algorithms are typically terminated with an ϵ-approximate solution after a finite number of iterations. This paper shows how *existing* infinitely convergent branch-and-bound algorithms can be augmented to guarantee finite termination with an *exact* global solution for problems having extreme point solutions.

Key words. finite convergence, global optimization, branch-and-bound, concave, bilinear

AMS subject classifications. 65B99, 65K05, 90C26

PII. S105262349935178X

1. Introduction.

Nonlinear programming distinguishes between the notions of global convergence (convergence from any starting point) and local convergence (order or rate of convergence in a neighborhood of a local solution). Global convergence is crucial from a theoretical point of view (to ensure that the limit point is stationary), while a high order of convergence is desirable from a practical standpoint (to reduce computing time). Cauchy's gradient method of steepest descent is globally convergent, while Newton's method enjoys second-order (quadratic) local convergence when started sufficiently close to a local minimizer (under certain conditions). Most algorithms are designed to satisfy a descent property which guarantees objective function decrease at each iteration. Newton's method in its purest form does not satisfy this descent property, but enforcing a line search in the Newton direction is one variant that does. A locally convergent descent algorithm (or even a simple descent routine that guarantees nonworsening objective values) can be augmented (every finite number of iterations) with an additional spacer step (of a known globally convergent method) to obtain a globally convergent variant that possesses the same local convergence behavior.

The counterpart issues in global optimization are convergence (to ensure that the limit point is a global minimizer) and finite convergence to an exact (as opposed to an ϵ-approximate) global minimizer. A finitely convergent algorithm, however, may not be the fastest or the most computationally efficient. Some of the best algorithms can make rapid progress to a neighborhood of a global optimizer and then spend most of the time identifying and verifying a global minimizer. Such algorithms are typically terminated with an ϵ-approximate global optimum after a finite number of iterations.

This paper is concerned with studying problem structures and mechanisms conducive to finite convergence of branch-and-bound global optimization algorithms. In particular, our aim is to establish how *existing* infinitely convergent descent algo-

[*] Received by the editors February 11, 1999; accepted for publication (in revised form) October 15, 1999; published electronically June 6, 2000.
http://www.siam.org/journals/siopt/10-4/35178.html

[†] School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332. This author was supported in part by the National Science Foundation under grant DMI-9908216 (faiz@isye.gatech.edu).

[‡] Department of Industrial and Systems Engineering, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061 (hanifs@vt.edu). This author was supported in part by the National Science Foundation under grant DMI-9812047 and by the Air Force Office of Scientific Research under grant F49620-96-1-0274.

rithms can be augmented to achieve finite convergence for certain classes of problems that possess extreme-point global optimizers. We note that such an augmentation preserves the underlying convergence process of the overall procedure while serving to enhance the speed of convergence to an exact (not ϵ-approximate) global minimizer. Two specialized algorithmic approaches are presented. The first method assumes pseudoconcave or bilinear polynomial objective functions being minimized over polyhedral feasible sets, and the second method extends the results to the more general case of any problem that has an extreme-point global minimizer.

A number of authors have addressed the issue of finite termination in both the nonlinear programming and the global optimization literature. Concave minimization is perhaps the most studied problem with respect to finite convergence, including works by Benson [4], Benson and Sayin [5], Hamami and Jacobsen [10], Nast [16], and Tam and Bam [21], to name only a few. A recent paper by Shectman and Sahinidis [18], more closely related to the work herein, developed a branching technique that partitions the problem, based on the incumbent solution when it is contained in the relative interior of the current node subproblem's bounding region, to yield a finite procedure for separable concave minimization problems. Good computational results are reported by also incorporating a number of domain reduction strategies. Cutting plane approaches have been augmented to be finitely convergent for quasi-concave minimization problems by Majthay and Whinston [15] and for separably constrained bilinear programs by Sherali and Shetty [19], based on a scheme for finitely exploring the facial structure of polytopes. Here the concept of extreme faces that generalizes extreme points is developed, and methodologies for detecting and deleting extreme faces are designed to compose finitely convergent methods. Sherali and Tünçbilek [20] developed a finitely convergent branch-and-bound algorithm for location-allocation problems using squared Euclidean distance-based separation penalties by partitioning on the dichotomy that a variable either is basic and strictly between its bounds (nondegenerate) or is at one of its bounds, at any basic feasible solution. Each of these dichotomous conditions was shown to permit certain domain reduction strategies that were integrated into the algorithm. While the foregoing procedures are *inherently finite*, our primary concern in this paper is to develop means for converting *given infinitely* convergent algorithms into finitely convergent procedures for the stated classes of problems.

Al-Khayyal and Falk [2] employed an auxiliary subproblem as a spacer step to accelerate convergence to *nonextreme*-point global optimizers for bilinear programs over general (nonseparable) polyhedra. Later, Al-Khayyal [1] noted that this same procedure, without modification, is finite when applied to linear complementarity problems. Al-Khayyal and Kyparisis [3] established that the key property for finiteness appears to be that the target minimizer is an isolated local solution, and they developed first-order sufficient conditions for when an infinitely convergent algorithm can be made finite by the solution of an auxiliary spacer step subproblem. Shapiro and Al-Khayyal [17] extend some of these results by showing that first-order sufficient conditions for strict local optimizers guarantee isolated local solutions provided a constraint qualification holds and the feasible set is *nearly* convex at the solution.

In this paper we introduce a novel construction for ensuring the finite convergence of branch-and-bound methods for minimizing pseudoconcave polynomial objective functions over polyhedral sets, bilinear functions over separably constrained polyhedral sets, and general objective functions that guarantee extreme-point optimality (such as quasi-concave functions) over polytopes. The principal notion introduced here is an extension of the *purification scheme* of Charnes and Kortanek [6], which is used in Khachiyan's ellipsoid algorithm (see Gacs and Lovász [8]) as well as in Karmarkar's [14] interior point method and its many variants, from the context of linear programming to the domain of nonlinear (nonconvex) optimization. This permits the fathoming of nodes once lower and upper bounds are within a specified tolerance *without* losing the *exact optimum* (as in ϵ-approximation methods) and, in some contexts, permits the collapsing of variable intervals into one of the end-points, thereby producing a finite algorithm for finding global solutions.

The remainder of this paper is organized as follows. In the next section, we prove that incorporating a purification step, within any branch-and-bound algorithm that conforms with a stated Property 1, yields a finite procedure for separably constrained bilinear programs or for global optimization problems having pseudoconcave objective functions. In section 3, we examine any objective function that enjoys extreme-point global optimality and augment Property 1 to include algorithms that enforce a longest edge interval bisection every finitely many iterations. For this class of problems and algorithms, we propose an augmentation routine that guarantees finite convergence to a global minimizer. Finally, in section 4, we address the issue of selecting a value for a key parameter on which our analysis is largely based. For brevity, we assume in what follows that the reader is familiar with the main concepts of branch-and-bound algorithms for global optimization as presented in a number of sources including [13, 12, 22].

2. Polynomial and bilinear programs. Consider the problem, which will be referred to as *Problem P*,

$$\text{minimize}\{f(x) : x \in X\}, \tag{2.1}$$

where the feasible set $X := \{x : Ax = b, -\infty < l \le x \le u < \infty\}$ is nonempty, A is a real $m \times n$ matrix with rank $m < n$, and we assume that the data $A, b, l,$ and u are all integer. (An appropriate scaling can convert a problem in rational data into an equivalent one in all-integer data.)

In this paper we will first consider polynomial functions f of degree δ with all-integer coefficients and exponents, and we will study the finite convergence of branch-and-bound algorithms for solving Problem P that satisfy the following property.

Property 1. The algorithm

(i) partitions each node into a finite number of subproblems;

(ii) selects nodes for partitioning via the least lower bound rule; and

(iii) is infinitely convergent in the sense that if it does not terminate finitely with an optimum, then along any infinite path in the branch-and-bound tree generated, the difference between the incumbent solution value and the lower bounds generated tends to zero.

We will show that if the following assumption holds true, then any algorithm for Problem P that satisfies Property 1 can be made to terminate finitely.

Assumption 1. Either f is pseudoconcave, or P is a separably constrained bilinear program.

Toward this end, we first introduce a particular polynomial-time step within the algorithm. Noting that in the cited cases there exists an extreme-point solution to P, this step finds an extreme point of X that has at least as good an objective value as any feasible solution. In analogy with a similar procedure used in the context of linear programming (cf. Charnes, Kortanek, and Raike [7]), we refer to this as a *purification step*.

Given a point $\tilde{x} \in X$, this step finds an $\bar{x} \in vert(X)$, where $vert(X)$ denotes the set of vertices of X, such that $f(\bar{x}) \leq f(\tilde{x})$.

Purification step.

Case (i). f is pseudoconcave on X.

If there are n linearly independent defining hyperplanes of X that are binding at $\tilde{x}$, then $\tilde{x} \in vert(X)$; stop with $\bar{x} = \tilde{x}$. Else, find a direction $d \neq 0$ lying in the null space of the constraints that are binding at $\tilde{x}$. Assume that $\nabla f(\bar{x})^t d \leq 0$, else, replace d by $-d$ so that this condition holds true. Move along d until this motion is blocked by a defining constraint of X, and repeat this step with $\tilde{x}$ equal to this resulting solution.

Case (ii). P is a separably constrained bilinear program.

In this case, suppose that $x = (u, v)$, and that $X = U \times V$, where $x \in X \Leftrightarrow u \in U$ and $v \in V$. Given $\tilde{x} = (\tilde{u}, \tilde{v}) \in U \times V$, if this is not an extreme point, then alternately solve P as a linear program over u and over v, with the other variable vector fixed as in the current solution, each time finding an optimal extreme point solution over U and V, respectively, until an extreme point $\bar{x} = (\bar{u}, \bar{v})$ of $U \times V$ is obtained. (This step can be further continued, if so desired, until the value of f fails to improve, i.e., a fixed point of the algorithmic map is obtained. However, this is not necessary for the required purification step, and moreover, such a fixed point might not be obtained in a polynomial number of steps.)

LEMMA 2.1. *Both cases of the foregoing Purification Step will result in an extreme point $\bar{x} \in X$ such that $f(\bar{x}) \leq f(\tilde{x})$, with this step being executable in polynomial time.*

Proof. In Case (i), given d such that $\nabla f(\tilde{x})^t d \leq 0$, by the pseudoconcavity of f, $f(\tilde{x} + \lambda d) \leq f(\tilde{x})\ \forall\ \lambda \geq 0$. Hence, the revised $\tilde{x}$ found by the procedure has an objective value that is at least as good as that at $\tilde{x}$. Moreover, at least one additional linearly independent defining hyperplane of X is binding at this revised solution. Since this step would be repeated at most n times, and since each step is polynomially bounded, the assertion holds true in this case.

Furthermore, in Case (ii), since we solve at most two linear programs over X, where the sizes of these problems are polynomially related to the size of Problem P, and since each linear program does not worsen the objective value, the assertion again holds true. This completes the proof. □

The following result prescribes the design of a *finitely* convergent variant of a given algorithm that satisfies Property 1.

THEOREM 2.2. *Consider Problem P and suppose that Assumption* 1 *holds true. Define*

$$L \geq \lceil 1 + \log_2 |\det_{\max}| \rceil, \tag{2.2}$$

where $|\det_{\max}|$ is the largest determinant of a basis of A in absolute value. Suppose that a branch-and-bound algorithm that satisfies Property 1 *is used to solve Problem P, where an extreme-point incumbent solution yielding an upper bound UB is maintained by using the foregoing Purification Step, if necessary, and where a node is fathomed if its lower bound LB satisfies $UB - LB \leq 2^{-2\delta L}$. Then this algorithm will finitely converge to a global optimal solution of Problem P.*

Proof. Since the given algorithm is infinitely convergent, and since an ϵ-termination criterion is being used, where $\epsilon \equiv 2^{-2\delta L} > 0$, the algorithm will terminate finitely. Hence, noting the existence of an extreme-point optimum under Assumption 1 and that the given UB at any stage is equal to $f(\bar{x})$, where $\bar{x} \in vert(X)$ by the Purification Step, we only need to show that whenever we fathom a node for which the lower bound LB satisfies $UB - LB \leq 2^{-2\delta L}$, then this node does not admit a feasible solution $\hat{x}$, where $\hat{x} \in vert(X)$ and $f(\hat{x}) < f(\bar{x})$. On the contrary, if such a solution $\hat{x}$ exists, then let D_1 and D_2, respectively, be the absolute values of the determinants of bases representing the basic feasible solutions $\bar{x}$ and $\hat{x}$. By the nature of f and the data of Problem P, the values $f(\bar{x})$ and $f(\hat{x})$ are of the form N_1/D_1^δ and N_2/D_2^δ, where N_1 and N_2 are integers. Since $f(\hat{x}) < f(\bar{x})$, we have

$$UB - LB \geq f(\bar{x}) - f(\hat{x}) \geq \frac{1}{(D_1 D_2)^\delta} > 2^{-2\delta L}, \tag{2.3}$$

which contradicts $UB - LB \leq 2^{-2\delta L}$. This completes the proof. □

Remark 1. By the proof of Theorem 2.2, it should be evident that the key to deriving a finite variant of an infinitely convergent branch-and-bound algorithm for solving Problem P by our approach lies in two aspects. First, there should exist some discrete set among which incumbent solutions and an optimum to Problem P reside. In our case, this is ensured by Assumption 1 along with the Purification Step (Lemma 1). Second, there should exist some $\epsilon > 0$ such that for any two elements $\bar{x}$ and $\hat{x}$ of the aforementioned discrete set,

$$\text{if } f(\bar{x}) \neq f(\hat{x}), \text{ then we have } |f(\bar{x}) - f(\hat{x})| > \epsilon. \tag{2.4}$$

In our analysis, this was ensured by the nature of f and X that led to (2.3), which corresponds to (2.4) with $\epsilon \equiv 2^{-2\delta L}$.

Note that we might have other forms of f that are not polynomial or integer valued for integer x but for which (2.4) holds true. For example, consider the concave function $f(x) = ln(e^t x)$, where $e = (1, 1, \ldots, 1)^t$ and where $e^t x$ is positive on X. In this case, let $U = \text{maximum}\{e^t x : x \in X\}$. For any pair of vertices $\bar{x}$ and $\hat{x}$ of X such that $f(\bar{x}) \neq f(\hat{x})$, since $e^t \bar{x}$ and $e^t \hat{x}$ differ by more than 2^{-2L} by the same argument as in the proof of Theorem 2.2, we have by the concave nature of $ln(\cdot)$ that

$$|f(\bar{x}) - f(\hat{x})| > [ln(U) - ln(U - 2^{-2L})] \equiv \epsilon. \tag{2.5}$$

Hence, (2.4) is satisfied, and so, Theorem 2.2 would hold true for this function f as well. Note that this is generalizable to the composite objective function $g[h(x)]$, where $h : \Re^n \to \Re$ is a concave function that satisfies the property asserted for f in (2.1), and where $g : \Re^n \to \Re$ is an increasing, concave function. In this case, (2.4) holds true with $\epsilon = g(U) - g(U - 2^{-2\delta L})$, where $U \geq \max\{h(x) : x \in X\}$. However, minimizing $g[h(x)]$ would be equivalent to minimizing $h(x)$ itself.

3. Alternative scheme for achieving finiteness. Consider any algorithm that satisfies Property 1, and suppose that this is achieved by performing interval bisections on the longest interval every N iterations. (We will refer to such an algorithm as satisfying Property 1′.) Furthermore, assume that the objective function enjoys extreme point optimality (e.g., when f is quasi-concave) but is not necessarily polynomial or integer valued as assumed before.

At any stage of the algorithm, commencing with variable intervals $[l_j, u_j]$, $j = 1, \ldots, n$, let $[l'_j, u'_j]$ denote the current interval bounds, and define

$$S_l := \{j : l'_j = l_j\}, S_u := \{j : u'_j = u_j\}, \text{ and } S_0 := \{j : l'_j = u'_j\}.$$

Now, in the bounding step of the algorithm, in addition to checking the ubiquitous fathoming criterion $UB \geq LB$, suppose that we augment this step by the following routine that is executed *before* computing the lower bound LB. In what follows we denote

$$X' := \{x : Ax = b, l' \leq x \leq u'\}.$$

Augmentation routine.

Step (i). If $|S_l \cup S_u| < n - m$, fathom the node subproblem.

Step (ii). If $|S_l \cup S_u| = n - m$ and $S_l \cap S_u = \emptyset$, then if the solution obtained by letting x_j be nonbasic at its lower (respectively, upper) bound for $j \in S_l$ (respectively, $j \in S_u$) yields a basic feasible solution for X, compute this solution and update the incumbent, if necessary. In any case, given that the first two conditions of Step (ii) hold true, fathom the node subproblem.

Step (iii). For any $j \in S_l \cup S_u$, if

$$(u'_j - l'_j) \leq 2^{-L}, \tag{3.1}$$

then fix

$$x'_j = l'_j \text{ and } u'_j = l'_j \text{ if } j \in S_l, \tag{3.2}$$

$$x'_j = u'_j \text{ and } l'_j = u'_j \text{ if } j \in S_u. \tag{3.3}$$

Update S_0. If $|S_0| \geq n - m$, test whether fixing $x_j = l'_j = u'_j \ \ \forall j \in S_0$ yields a basic feasible solution, and if so, update the incumbent solution, if necessary, and fathom the node subproblem. Furthermore, if $S_0 = S_l \cup S_u$, fathom the node subproblem. Naturally, if fixing $x_j = l'_j = u'_j \ \ \forall j \in S_0$ yields infeasibility, fathom the node subproblem.

THEOREM 3.1. *The Augmentation Routine described above is valid for the foregoing class of problems. Moreover, incorporating it within a branch-and-bound algorithm satisfying Property* $1'$ *yields a finitely convergent procedure.*

Proof. If $|S_l \cup S_u| < n - m$, then X' does not contain any vertex of X and so the node subproblem may be fathomed since we are only interested in evaluating extreme points of X in order to detect a global optimum. Similarly, if $|S_l \cup S_u| = n - m$ and $S_l \cap S_u = \emptyset$, then X' can contain at most one vertex of X as identified by Step (ii), and again this step is valid.

Next consider Step (iii) and suppose that (3.1) holds true. Suppose that $j \in S_l$. (The case of $j \in S_u$ is similar.) Note that if $\hat{x}$ is an extreme point of X and $\hat{x} \in X'$, then either $\hat{x}_j = l'_j = l_j$ or else $\hat{x}_j$ is basic *and* $\hat{x}_j > l'_j = l_j$. But any basic variable for a basic feasible solution to X which is not integral has a fractionality that is at least $1/|\det_{max}| > 2^{-L}$. Hence, since $l'_j = l_j$ is integral, we must have that x_j either is nonbasic at l_j or is basic and degenerate at the value l_j at any vertex of X that is feasible to X'. Therefore, we may set $x_j = l'_j$ and revise $u'_j = l'_j$, consequently letting j belong to S_0. The validity of the remainder of Step (iii) under the condition $S_0 \geq n - m$ then follows because of the extreme-point optimality property as above.

Finally, consider the finiteness of the algorithm. Along any branch of nested intervals in the branch-and-bound tree, if the conditions of Steps (i) and (ii) do not hold, we must finitely obtain $S_l \cap S_u = \emptyset$ and (3.1) holding true for each $j \in S_l \cup S_u$, where $|S_l \cup S_u| \geq n - m$, because of the interval bisection step that is performed finitely often. But then (3.2)–(3.3) would yield $S_0 = S_l \cup S_u$ and $|S_0| \geq n - m$, and Step (iii) would fathom this node subproblem. Hence, only a finite enumeration tree can be generated, and this completes the proof. $\square$

Remark 2. Note that the key to the finiteness argument lies in the fact that we can fathom a node subproblem that does not admit at least $n - m$ variables to lie at their original bounds, yielding a basic feasible solution, and in the strategy that any interval that contains an original bound as one of the end-points can be collapsed into this bound once the interval length satisfies condition (3.1).

Note that the focus of the foregoing discussion is on converting a *given* infinitely convergent algorithm into a finite one for the stated class of problems (which, incidentally, subsumes the one considered by Shectman and Sahinidis [18]). There exists an alternative mechanism that possibly can be used to design finitely convergent algorithms for problems that yield extreme-point optimal solutions. This is based on adopting the partitioning strategy of branching on the dichotomy that any variable x_j is either at the value l_j or at the value u_j, or is basic in the interval (l_j, u_j) at an optimal solution. The first two conditions fix a given variable in value, while the last condition can be imposed on at most m variables for any basic feasible solution. This leads to a finite total enumeration tree which can then be further curtailed by the computation of suitable bounds. Note that the lower bounding scheme should be suited to this type of a partitioning process in order to avoid (close to) a total enumeration of extreme points. Sherali and Tünçbilek [20] described such a procedure in the context of solving a nonconvex location-allocation problem using squared Euclidean distance-based separation penalties.

COROLLARY 3.2. *If A is unimodular (e.g., if X represents network flow constraints, whence A is totally unimodular), then the value 2^{-L} in* (3.1) *can be replaced by ϵ for any $0 < \epsilon < 1$, and furthermore, for any $j \notin S_l \cup S_u$, if $[l'_j, u'_j]$ contains a single integer q, then we can set $l'_j = u'_j = q$, and if this interval contains no integer, then we may fathom the node subproblem.*

Proof. The proof follows that of Theorem 3.1, noting that the unimodularity of A and the integrality of the data describing X ensures that variables are integer-valued for any vertex of X. □

4. Derivation of a value for L. The practical implications of our results depend on the computability of an appropriate L, which is bounded below by a function of the largest absolute determinant of all bases of A, as in (2.2). A valid value for L can be computed by appealing to Hadamard's inequality [9].

THEOREM 4.1. *Let B be an $m \times m$ matrix. Then*

$$|\text{det}B| \leq \prod_{j=1}^{m} \left(\sum_{i=1}^{m} b_{ij}^2 \right)^{1/2} .$$

Furthermore, equality holds if and only if the columns of B are orthogonal.

Proof. The proof was stated and proved as Corollary 7.8.2 in [11]. □

Letting $B_{\cdot j}$ denote the jth column of B, Hadamard's inequality can be stated succinctly as

$$|\text{det}B| \leq \prod_{j=1}^{m} \|B_{\cdot j}\|. \tag{4.1}$$

Since $\lceil 1 + \log_2 \alpha \rceil$ is a monotonically increasing function for all $\alpha > 0$, then $L = \lceil 1 + \log_2 \beta \rceil$ satisfies (2.2) for all $\beta \geq |\det_{max}|$. Now, let $\hat{B}_{\cdot[1]}, \hat{B}_{\cdot[2]}, \ldots, \hat{B}_{\cdot[m]}$ be the columns of A having the m largest Euclidean norms, in decreasing order. That is, for $k = 2, \ldots, m$,

$$\|\hat{B}_{\cdot[k]}\| = \max\left\{\|\hat{B}_{\cdot j}\| : j \in \{1, \ldots, n\}\backslash\{[1], \ldots, [k-1]\}\right\},$$

where $[k]$ denotes the index corresponding to the kth largest norm. Note that these m largest column norms can be computed in a single pass process that (i) maintains the current m largest norm columns at any stage and (ii) performs replacement, if necessary, when the next column's norm is computed. This process is of complexity $O(mn)$. If we now define

$$\beta = \prod_{j=1}^{m} \|\hat{B}_{\cdot[j]}\|, \tag{4.2}$$

then we have from (4.1) that $\beta \geq |\det_{max}|$, so that $L = \lceil 1+\log_2 \beta \rceil$ satisfies (2.2). For the case when the columns $\hat{B}_{\cdot[1]}, \hat{B}_{\cdot[2]}, \ldots, \hat{B}_{\cdot[m]}$ are mutually orthogonal, we would have from Theorem 4.1 and the foregoing analysis that $\beta = |\det_{max}|$, and the bound (2.2) is tight for L determined as above.

Acknowledgments. The authors are grateful to Earl Barnes and to an anonymous referee for useful comments on improving this contribution.

REFERENCES

[1] F. A. Al-Khayyal, *Note on solving linear complementarity problems as jointly constrained bilinear programs,* J. Math. Anal. Appl., 158 (1991), pp. 583–589.

[2] F. A. Al-Khayyal and J. E. Falk, *Jointly constrained biconvex programming,* Math. Oper. Res., 8 (1983), pp. 272–286.

[3] F. A. Al-Khayyal and J. Kyparisis, *Finite convergence of algorithms for nonlinear programs and variational inequalities,* J. Optim. Theory Appl., 70 (1991), pp. 319–332.

[4] H. P. Benson, *A finite algorithm for concave minimization over a polyhedron,* Naval Res. Logist., 32 (1985), pp. 165–177.

[5] H. P. Benson and S. Sayin, *A finite concave minimization algorithm using branch and bound and neighbor generation,* J. Global Optim., 5 (1994), pp. 1–14.

[6] A. Charnes and K. O. Kortanek, *An Opposite Sign Algorithm for Purification to an Extreme Point Solution,* Office of Naval Research Memorandum 84, Northwestern University, Evanston, IL, 1963.

[7] A. Charnes, K. O. Kortanek, and W. Raike, *Extreme Point Solutions in Mathematical Programming: An Opposite Sign Algorithm*, Systems Memorandum 129, Northwestern University, Evanston, IL, 1965.

[8] P. Gacs and L. Lovász, *Khachian's algorithm for linear programming,* Math. Programming Stud., 14 (1981), pp. 61–68.

[9] J. Hadamard, *Résolution d'une question relative aux déterminants,* Bull. Sci. Math., 2 (1893), pp. 240–248 (in French).

[10] M. Hamami and S. E. Jacobsen, *Exhaustive nondegenerate conical process for concave minimization on convex polytopes,* Math. Oper. Res., 13 (1988), pp. 479–487.

[11] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.

[12] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to Global Optimization*, Kluwer Academic Publishers, Dordrecht, the Netherlands, 1995.

[13] R. Horst and H. Tuy, *Global Optimization: Deterministic Approaches*, 3rd ed., Springer-Verlag, Berlin, 1996.

[14] N. Karmarkar, *A new polynomial-time algorithm for linear programming,* Combinatorica, 4 (1984), pp. 373–395.

[15] A. Majthay and A. Whinston, *Quasiconcave minimization subject to linear constraints,* Discrete Math., 9 (1974), pp. 35–59.

[16] M. Nast, *Subdivision of simplices relative to a cutting plane and finite concave minimization,* J. Global Optim., 9 (1996), pp. 65–93.

[17] A. Shapiro and F. A. Al-Khayyal, *First-order conditions for isolated locally optimal solutions,* J. Optim. Theory Appl., 77 (1993), pp. 189–196.

[18] J. P. Shectman and N. V. Sahinidis, *A finite algorithm for global minimization of separable concave programs,* J. Global Optim., 12 (1998), pp. 1–36.

[19] H. D. Sherali and C. M. Shetty, *A finitely convergent algorithm for bilinear programming problems using polar cuts and disjunctive face cuts,* Math. Programming, 19 (1980), pp. 14–31.

[20] H. D. Sherali and C. H. Tünçbilek, *A squared Euclidean distance location-allocation problem,* Naval Res. Logist., 39 (1992), pp. 447–469.

[21] B. T. Tam and V. T. Bam, *Minimization of a concave function under linear constraints,* Ekonom. Mat. Metody, 21 (1985), pp. 709–714 (in Russian).

[22] H. Tuy, *Convex Analysis and Global Optimization*, Kluwer Academic Publishers, Dordrecht, the Netherlands, 1998.

SIAM J. Optim.
Vol. 10, No. 4, pp. 1058–1078

ON THE IDENTIFICATION OF ZERO VARIABLES IN AN INTERIOR-POINT FRAMEWORK[*]

FRANCISCO FACCHINEI[†], ANDREAS FISCHER[‡], AND CHRISTIAN KANZOW[§]

Abstract. We consider column sufficient linear complementarity problems and study the problem of identifying those variables that are zero at a solution. To this end we propose a new, computationally inexpensive technique that is based on growth functions. We analyze in detail the theoretical properties of the identification technique and test it numerically. The identification technique is particularly suited to interior-point methods but can be applied to a wider class of methods.

Key words. linear complementarity problem, column sufficient matrix, identification of zero variables, growth function, indicator function, interior-point method

AMS subject classifications. 90C05, 90C33, 65K05

PII. S1052623498339739

1. Introduction.

We consider the linear complementarity problem (LCP)

$$y = Mx + q, \qquad x \geq 0, \qquad y \geq 0, \qquad x^T y = 0,$$

where the matrix $M \in \mathbb{R}^{n\times n}$ and the vector $q \in \mathbb{R}^n$ are given. Throughout the paper we assume that

$$M \text{ is a column sufficient (CS) matrix}$$

(see [1, 2]); i.e., we assume that

$$x_i(Mx)_i \leq 0 \ \ \forall i \qquad \Longrightarrow \qquad x_i(Mx)_i = 0 \ \ \forall i.$$

We recall that positive semidefinite matrices and sufficient (or, equivalently, P_*-) matrices are CS, so that the class of CS LCPs includes all the classes of LCPs for which interior-point methods have been extensively studied. We denote by $\mathcal{S}$ the solution set of an LCP. This set is always closed and it is known to be convex for every q if and only if M is a CS matrix [1, Theorem 3.5.8]. We further make the blanket assumption that $\mathcal{S}$ is nonempty.

In this paper we are interested in techniques that identify the variables that are zero at a solution of an LCP. Obviously, the zero variables at a solution may be different from the zero variables at another solution. Therefore, in order to make our aim more precise, we define the following three index sets:

$$\mathcal{B} := \{i |\, x_i^* > 0 \text{ for at least one } (x^*, y^*) \in \mathcal{S}\},$$

$$\mathcal{N} := \{i |\, y_i^* > 0 \text{ for at least one } (x^*, y^*) \in \mathcal{S}\},$$

$$\mathcal{J} := \{i |\, x_i^* = y_i^* = 0 \ \forall (x^*, y^*) \in \mathcal{S}\}.$$

[*] Received by the editors June 4, 1998; accepted for publication (in revised form) August 25, 1999; published electronically June 6, 2000. This research was partially supported by NATO grant CRG 960137.

http://www.siam.org/journals/siopt/10-4/33973.html

[†] Dipartimento di Informatica e Sistemistica, Università di Roma "La Sapienza," Via Buonarroti 12, 00185 Roma, Italy (soler@dis.uniroma1.it).

[‡] Department of Mathematics, University of Dortmund, Vogelpothsweg 87, 44221 Dortmund, Germany (andreas.fischer@math.uni-dortmund.de).

[§] Institute of Applied Mathematics, University of Hamburg, Bundesstrasse 55, 20146 Hamburg, Germany (kanzow@math.uni-hamburg.de). The research of this author was supported by the Deutsche Forschungsgemeinschaft.

The following proposition describes two properties of these index sets which are well known in the case of a positive semidefinite matrix M.

PROPOSITION 1.1.

(i) *The index sets* $\mathcal{B}$*,* $\mathcal{N}$*, and* $\mathcal{J}$ *form a partition of* $\{1,\dots,n\}$*.*

(ii) *A point* $z^* = (x^*, y^*) \in \mathcal{S}$ *belongs to the relative interior* $\mathrm{ri}\mathcal{S}$ *of the solution set* $\mathcal{S}$ *if and only if*

$$x^*_{\mathcal{B}} > 0, \quad x^*_{\mathcal{N}} = 0, \quad x^*_{\mathcal{J}} = 0, \qquad y^*_{\mathcal{B}} = 0, \quad y^*_{\mathcal{N}} > 0, \quad y^*_{\mathcal{J}} = 0. \tag{1.1}$$

Proof. (i) It is obvious that $\mathcal{B}\cup\mathcal{N}\cup\mathcal{J} = \{1,\dots,n\}$. So we only have to show that $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$ are pairwise disjoint. In turn, since it is easy to see by the definition of these sets that $\mathcal{B}\cap\mathcal{J} = \emptyset = \mathcal{N}\cap\mathcal{J}$, we only have to show that $\mathcal{B}\cap\mathcal{N} = \emptyset$. Suppose by contradiction that an index i belongs to both $\mathcal{B}$ and $\mathcal{N}$. Then there exist two points $(\bar{x}, \bar{y})$ and $(\hat{x}, \hat{y})$, both belonging to the solution set $\mathcal{S}$, such that $\bar{x}_i > 0$ and $\hat{y}_i > 0$. Consequently we have $\bar{y}_i = 0$ and $\hat{x}_i = 0$. Since M is CS, $\mathcal{S}$ is convex. Therefore, the point $(x(t), y(t)) = t(\bar{x}, \bar{y}) + (1-t)(\hat{x}, \hat{y})$ belongs to $\mathcal{S}$ for every $t \in (0, 1)$. But by the relations established above we have $x_i(t) > 0$ and $y_i(t) > 0$, thus contradicting the fact that $(x(t), y(t))$ belongs to $\mathcal{S}$.

(ii) The proof is identical to the one given in [3, Theorem 2.2] for monotone complementarity problems. A closer look at that proof shows that the monotonicity is used there only to establish the convexity of the solution set. Since the convexity of $\mathcal{S}$ holds under the assumption that M is CS, the proof goes through. □

Point (ii) of the above proposition shows that, in the relative interior of the set $\mathcal{S}$, the set of zero variables is invariant with respect to the solution. We recall that, under very mild assumptions, interior-point methods generate sequences of points such that every accumulation point is in the relative interior of $\mathcal{S}$ and so these solutions share the same zero-nonzero structure; see, e.g., [14, 16].

Our aim is to identify this structure or, equivalently, the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. The correct identification of these sets is important from both theoretical and computational points of view. In fact, the knowledge of the zero-nonzero structure may allow us, on the one hand, to easily recover an exact solution from the approximated one provided by an interior-point method and, on the other hand, to improve the efficiency of interior-point methods and column generation techniques [3].

The identification of the zero variables in interior-point methods for linear programs has been the subject of intense research in the past 10 years, and we refer the reader to [3] for an exhaustive review. It is now accepted that the technique originally proposed by Tapia [18] for nonlinear programs enjoys the most interesting properties in the context of interior-point methods for linear programming (LP) [3].

This technique has also been extended to the case of LCPs [3, 4, 9, 14]. Then, however, a further difficulty can occur. In contrast to linear programs, where we always have $\mathcal{J} = \emptyset$, this is no longer true for LCPs. Problems with $\mathcal{J} = \emptyset$ are called *nondegenerate*, while those for which $\mathcal{J} \neq \emptyset$ are termed *degenerate*. Degeneracy makes the identification of the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$ more difficult [4, 9, 14].

In this paper we present a new technique for identifying the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. We show that, given a point $z = (x, y)$ belonging to a certain set $\mathcal{Z}_\varepsilon$, we are able to correctly identify $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. The set $\mathcal{Z}_\varepsilon$ is defined in such a way that virtually all interior-point methods will generate, under standard assumptions, a sequence whose points eventually belong to this set, thus ensuring finite identification. We want to stress from the outset two peculiarities which, in our view, are significant. First, the class of problems we are able to deal with is considerably broader than the ones considered in previous works. Second, unlike other works on the same subject, we do not make reference to a specific (although general) algorithmic scheme, so that the results obtained can be applied to a class of methods wider than the interior-point one.

The approach we use in this paper is reminiscent of the one proposed in [5] for general nonlinear programs. However, there is a major difference: one of the key assumptions in [5] is that the solution of interest is an isolated solution. This assumption is not sensible in the LCP case, and we therefore drop it by fully exploiting the structure of the problem. Furthermore, we are able to obtain particularly simple expressions for the growth functions (see section 3) and convergence rates estimates (see sections 4 and 5) that have no parallel in [5].

The paper is organized as follows. First, we introduce some further notation. In the next section we present the basic identification results of the paper. Sections 3 and 4 address some more technical points related to the identification technique. In section 5 we specialize some of the results to an interior-point framework, while numerical experiments are reported in section 6. In section 7 we make some final comments.

Throughout the paper $\|\cdot\|$ denotes the Euclidean norm and

$$\operatorname{dist}(z|\mathcal{S}) := \inf\{\|w - z\| \mid w \in \mathcal{S}\}$$

the Euclidean distance of the point z from the set $\mathcal{S}$. We define the set $\mathcal{Z}$ by

$$\mathcal{Z} := \{z = (x, y) \in \mathbb{R}^{n+n} \mid z \text{ satisfies conditions (C1)–(C3)}\},$$

where

(C1) $x_{\mathcal{B}} \geq \delta,\ y_{\mathcal{N}} \geq \delta$,

(C2) $\|z\| \leq C$,

(C3) $\|r\| \leq \eta \|Xy\|$ with $X := \operatorname{diag}(x_1, \ldots, x_n)$ and $r := r(z) := y - (Mx + q)$,

and where $\delta > 0$, $C > 0$, and $\eta \geq 0$ are constants such that the intersection of $\mathcal{Z}$ and the solution set $\mathcal{S}$ is nonempty. Given a positive constant ε, we shall also consider the following set $\mathcal{Z}_\varepsilon$:

$$\mathcal{Z}_\varepsilon := \mathcal{Z} \cap \{z | \operatorname{dist}(z|\mathcal{S}) \leq \varepsilon\}.$$

In this paper we show that, given a point z in $\mathcal{Z}_\varepsilon$, with ε sufficiently small, we can correctly identify the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. The set $\mathcal{Z}$ comprises those points belonging to a compact set (condition (C2)) that are neither "too close" to the boundary of $\mathcal{S}$ (condition (C1)) nor "too much infeasible" in the terminology of interior-point methods (condition (C3)). The set $\mathcal{Z}_\varepsilon$ is just the part of $\mathcal{Z}$ that is not "too distant" from the solution set. We note that under standard, mild assumptions the vast majority of existing interior-point methods for LCPs will produce a sequence of points which belongs to $\mathcal{Z}$ (for suitable δ, C, and η) and to $\mathcal{Z}_\varepsilon$ (for any fixed positive ε) eventually. To see this, we may refer to [14], where a general framework is introduced that covers a large number of interior-point methods for monotone LCPs. It can easily be seen that within this framework the conditions (C1)–(C3) are satisfied. In particular, condition (C2) is explicitly stated in property (a) of that framework, whereas condition (C3) can be directly obtained from property (d). Moreover, condition (C1) follows from [14, Lemma 2.2]. For LCPs with P_*-matrices an infeasible interior-point method is considered in [16]. Using Theorems 2.3 and 4.1 of [16], one can verify that any sequence generated by this infeasible interior-point method eventually satisfies conditions (C1)–(C3) for suitable δ, C, and η.

2. Identification results. This section contains the basic identification results of the paper. We shall show that, given any point z in $\mathcal{Z}_\varepsilon$, if ε is sufficiently small, we can correctly identify the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. To this end we need some preliminary results and definitions.

PROPOSITION 2.1. *For any $z = (x, y) \in \mathbb{R}^n \times \mathbb{R}^n$, it holds that*

$$|x_i| \leq \mathrm{dist}(z|\mathcal{S}) \quad \forall i \in \mathcal{N} \cup \mathcal{J}, \qquad |y_i| \leq \mathrm{dist}(z|\mathcal{S}) \quad \forall i \in \mathcal{B} \cup \mathcal{J}.$$

Proof. Let $z^\perp = (x^\perp, y^\perp)$ denote the orthogonal projection of $z = (x, y)$ onto $\mathcal{S}$. (We recall that $\mathcal{S}$ is a nonempty, closed, and convex set, so that the orthogonal projection onto this set is uniquely defined.) Since (1.1) holds for all $z^* \in \mathrm{ri}\mathcal{S}$ it follows that $x^*_{\mathcal{N}\cup\mathcal{J}} = 0$ and $y^*_{\mathcal{B}\cup\mathcal{J}} = 0 \; \forall \; z^* \in \mathcal{S}$, so that

$$x^\perp_{\mathcal{N}\cup\mathcal{J}} = 0, \qquad y^\perp_{\mathcal{B}\cup\mathcal{J}} = 0.$$

Thus, we get for $i \in \mathcal{N} \cup \mathcal{J}$

$$|x_i| = |x_i - 0| = |x_i - x_i^\perp| \leq \|x - x^\perp\| \leq \|z - z^\perp\| = \mathrm{dist}(z|\mathcal{S}).$$

Similar reasonings can be repeated for y_i, $i \in \mathcal{B} \cup \mathcal{J}$. This completes the proof. □

The following two definitions are fundamental for our subsequent considerations.

DEFINITION 2.2. *A function $\rho : \mathbb{R}^{n+n} \to [0, \infty)$ is called a* growth function *on $\mathcal{Z}$ if there is a constant $c_1 \geq 1$ such that*

$$\frac{1}{c_1}\mathrm{dist}(z|\mathcal{S}) \leq \rho(z) \leq c_1 \mathrm{dist}(z|\mathcal{S}) \tag{2.1}$$

$\forall z \in \mathcal{Z}$.

Note that Definition 2.2 implies that $\rho(z)$ is equal to 0 if and only if z is a solution of the LCP. Growth functions are also known as residual functions and have a wide use in mathematical programming. The inequalities in (2.1) show that ρ can be used as a surrogate for the distance function, and it should therefore be expected to be easier to calculate than the distance function itself. Growth functions can be used, for example, to define stopping rules for algorithms or to study their convergence rates; they also play a fundamental role in the study of penalty functions. The interested reader can find a detailed survey of this topic in [15]. In the next section we show that in the case of CS LCPs, it is always possible, by using the conditions (C1)–(C3), to obtain very simple growth functions.

Our interest in growth functions is due to their role in the definition of indicator functions as defined below.

DEFINITION 2.3. *Let $\rho : \mathbb{R}^{n+n} \to [0, \infty)$ be a growth function on $\mathcal{Z}$ and $\alpha \in (0, 1)$ be fixed. Then the function $S : \mathbb{R} \times \mathcal{Z} \to \mathbb{R}$ defined by*

$$S(\xi, z; \alpha) := \begin{cases} \dfrac{\xi}{\xi - \rho(z)^\alpha} & \textit{if } \xi \neq \rho(z)^\alpha, \\ 0 & \textit{otherwise} \end{cases}$$

is called an indicator function.

The following proposition justifies the name *indicator function* and motivates our interest in indicator functions.

Proposition 2.4. *For any $\alpha \in (0,1)$ it holds that*

$$\lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} S(x_i, z; \alpha) = 1 \qquad \forall i \in \mathcal{B}, \tag{2.2}$$

$$\lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} S(x_i, z; \alpha) = 0 \qquad \forall i \in \mathcal{N} \cup \mathcal{J}, \tag{2.3}$$

$$\lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} S(y_i, z; \alpha) = 1 \qquad \forall i \in \mathcal{N}, \tag{2.4}$$

$$\lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} S(y_i, z; \alpha) = 0 \qquad \forall i \in \mathcal{B} \cup \mathcal{J}. \tag{2.5}$$

Proof. The fact that $0 \le \operatorname{dist}(z|\mathcal{S}) \le \varepsilon \to 0$ and the right inequality in (2.1) imply $\rho(z) \to 0$. This and condition (C1) yield (2.2).

Suppose now that $i \in \mathcal{N} \cup \mathcal{J}$. We need to consider only those (x_i, z) with $S(x_i, z; \alpha) \neq 0$. The very definition of the indicator function S then implies that $x_i \neq 0$. Using the left inequality of (2.1) and Proposition 2.1, we therefore have

$$\left|\frac{x_i - \rho(z)^\alpha}{x_i}\right| \ge \frac{\rho(z)^\alpha}{|x_i|} - 1 \ge \frac{\operatorname{dist}(z|\mathcal{S})^\alpha}{c_1^\alpha |x_i|} - 1 \ge \frac{|x_i|^{\alpha-1}}{c_1^\alpha} - 1. \tag{2.6}$$

Proposition 2.1 and $\operatorname{dist}(z|\mathcal{S}) \le \varepsilon \to 0$ imply $x_i \to 0$. Thus, by (2.6), it follows that

$$\lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} \frac{1}{|S(x_i, z, \alpha)|} = \lim_{\varepsilon\to 0, z\in\mathcal{Z}_\varepsilon} \left|\frac{x_i - \rho(z)^\alpha}{x_i}\right| = \infty;$$

i.e., (2.3) is valid.

The limits (2.4) and (2.5) can be proved similarly. □

The above result suggests that we introduce the following approximations to the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. Let $\theta \in (0, 1/2)$ and $\alpha \in (0,1)$ be fixed and ρ be a given growth function on $\mathcal{Z}$; define

$$\begin{aligned}
\mathcal{B}(z;\alpha) &:= \{i \,|\, S(x_i, z; \alpha) \ge 1 - \theta\},\\
\mathcal{N}(z;\alpha) &:= \{i \,|\, S(y_i, z; \alpha) \ge 1 - \theta\},\\
\mathcal{J}(z;\alpha) &:= \{i \,|\, \max\{S(x_i, z; \alpha), S(y_i, z; \alpha)\} \le \theta\}.
\end{aligned}$$

Note that these three sets are pairwise disjoint, but they do not necessarily form a partition of $\{1, \dots, n\}$. The following result is the principal result of this section and shows that the sets just defined are indeed reasonable estimates of the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$.

Theorem 2.5. *Let $\alpha \in (0,1)$ and $\theta \in (0, 1/2)$ be given. Then there is an $\varepsilon > 0$ such that*

$$\mathcal{B}(z;\alpha) = \mathcal{B}, \qquad \mathcal{N}(z;\alpha) = \mathcal{N}, \qquad \mathcal{J}(z;\alpha) = \mathcal{J} \tag{2.7}$$

$\forall z \in \mathcal{Z}_\varepsilon$.

Proof. Assume the contrary. Then sequences $\{\varepsilon^k\} \to 0$ and $\{z^k\}$ exist such that, for every k, $z^k \in \mathcal{Z}_{\varepsilon^k}$ and at least one of the equalities in (2.7) is violated.

Since ε^k converges to 0, we have that $\operatorname{dist}(z^k|\mathcal{S})$ also converges to 0, so that (2.2)–(2.5) hold. This obviously implies that all the equalities in (2.7) hold eventually. Therefore, we obtain a contradiction and the proof is complete. □

Remark 2.6. Using the indicator function and its properties we can easily define different approximations to the sets $\mathcal{B}$, $\mathcal{N}$, and $\mathcal{J}$. For example, in section 6 we shall use the following approximations in the numerical tests:

$$\begin{aligned}\mathcal{B}'(z;\alpha) &:= \{i|\ \min\{S(x_i,z;\alpha), 1-S(y_i,z;\alpha)\} \geq 1-\theta\},\\ \mathcal{N}'(z;\alpha) &:= \{i|\ \min\{S(y_i,z;\alpha), 1-S(x_i,z;\alpha)\} \geq 1-\theta\},\\ \mathcal{J}'(z;\alpha) &:= \mathcal{J}(z;\alpha).\end{aligned}$$

It is easy to see that these approximations enjoy the same properties established in Theorem 2.5 and that $\mathcal{B}'(z;\alpha) \subseteq \mathcal{B}(z;\alpha)$ and $\mathcal{N}'(z;\alpha) \subseteq \mathcal{N}(z;\alpha)$, so that $\mathcal{B}'(z;\alpha)$ and $\mathcal{N}'(z;\alpha)$ may be seen as more restrictive versions of the approximations $\mathcal{B}(z;\alpha)$ and $\mathcal{N}(z;\alpha)$.

3. Growth functions. We saw in the previous section that a key role in the identification of the zero-nonzero pattern of the solutions is played by growth functions. In particular growth functions enter in the definition of indicator functions that, in turn, are a crucial ingredient in the definition of the estimates $\mathcal{B}(z;\alpha)$, $\mathcal{N}(z;\alpha)$, and $\mathcal{J}(z;\alpha)$. We can say that our approach hinges on the possibility of defining an easily computable growth function.

Before presenting a first example of a growth function, we need some preliminary results. Consider the projection of the solution set $\mathcal{S}$ on the space of x-variables and indicate it by $\mathcal{S}_x$:

$$\mathcal{S}_x := \{x \in \mathbb{R}^n \,|\, \exists y \in \mathbb{R}^n : (x,y) \in \mathcal{S}\}. \tag{3.1}$$

Since, in view of our general assumptions, the solution set $\mathcal{S}$ is nonempty, closed, and convex, $\mathcal{S}_x$ is also nonempty, closed, and convex. The following lemma gives an error bound result for the set

$$\mathcal{Z}_x := \{x \in \mathbb{R}^n \,|\, \exists y \in \mathbb{R}^n : (x,y) \in \mathcal{Z}\},$$

which, by condition (C2), is bounded.

LEMMA 3.1. *There is a constant $c_2 > 0$ such that*

$$\operatorname{dist}(x|\mathcal{S}_x) \leq c_2 \|\min\{x, Mx+q\}\|$$

$\forall x \in \mathcal{Z}_x$.

Proof. It can be easily derived from [11] that, given a point $\bar{x} \in \mathcal{S}_x$, there exist a constant $\kappa_1 > 0$ and a neighborhood Ω of $\bar{x}$ such that

$$\operatorname{dist}(x|\mathcal{S}_x) \leq \kappa_1 \|\min\{x, Mx+q\}\| \qquad \forall x \in \Omega. \tag{3.2}$$

Suppose now that the lemma is false. Then a sequence $\{x^k\}$ contained in $\mathcal{Z}_x$ exists such that

$$\operatorname{dist}(x^k|\mathcal{S}_x) > k \|\min\{x^k, Mx^k+q\}\| \qquad \forall k \in \mathbb{N}. \tag{3.3}$$

Since $\mathcal{Z}_x$ is bounded, we can assume without loss of generality that $\{x^k\}$ converges to a point $\bar{x}$. It is also easy to see that $\bar{x}$ belongs to $\mathcal{S}_x$ for, if this were not true, (3.3) would imply $\operatorname{dist}(x^k|\mathcal{S}_x) \to \infty$, which, in view of the boundedness of $\mathcal{Z}_x$, is impossible.

But if $\bar{x}$ belongs to $\mathcal{S}_x$, we have that eventually (3.3) contradicts (3.2). □

Using Lemma 3.1 we can now give an error bound result for the solution set $\mathcal{S}$.

LEMMA 3.2. *There is a constant $c_3 > 0$ such that*

$$\operatorname{dist}(z|\mathcal{S}) \leq c_3 \left(\|\min\{x,y\}\| + \eta\|Xy\|\right) \tag{3.4}$$

$\forall z \in \mathcal{Z}$, where $\eta \geq 0$ denotes the constant from condition (C3).

Proof. Let $z \in \mathcal{Z}$ with $z = (x, y)$ be given. Since, as noted before, $\mathcal{S}_x$ is nonempty, closed, and convex, there exists an orthogonal projection $x^\perp$ of $x \in \mathbb{R}^n$ on the set $\mathcal{S}_x$. By the definition of $\mathcal{S}_x$, there is a vector $y^\perp$ such that $z^\perp = (x^\perp, y^\perp) \in \mathcal{S}$. Thus, we get

$$\begin{array}{rcl} \operatorname{dist}(z|\mathcal{S}) & \leq & \|z - z^\perp\| \\ & \leq & \|x - x^\perp\| + \|y - y^\perp\| \\ & = & \|x - x^\perp\| + \|M(x - x^\perp) + r\| \\ & \leq & (1 + \|M\|)\|x - x^\perp\| + \|r\|. \end{array} \tag{3.5}$$

Using Lemma 3.1, we have

$$\operatorname{dist}(x|\mathcal{S}_x) \leq c_2\| \min\{x, Mx + q\}\| = c_2\| \min\{x, y - r\}\|, \tag{3.6}$$

where the equality follows directly from the definition of the vector r in condition (C3).

Now, taking into account the easily verified relation

$$|\min\{a, b + c\}| \leq |\min\{a, b\}| + |c| \qquad \forall a,\, b,\, c \in \mathbb{R}$$

and the fact that all norms are equivalent in $\mathbb{R}^n$, it follows that there is a constant $\kappa_2 > 0$ such that

$$\| \min\{x, y - r\}\| \leq \kappa_2 \left(\| \min\{x, y\}\| + \|r\|\right). \tag{3.7}$$

Combining the inequalities (3.5)–(3.7) and using (C3), we therefore get

$$\begin{aligned} \operatorname{dist}(z|\mathcal{S}) &\leq (1 + \|M\|)\operatorname{dist}(x|\mathcal{S}_x) + \|r\| \\ &\leq (1 + \|M\|)c_2\| \min\{x, y - r\}\| + \|r\| \\ &\leq (1 + \|M\|)\kappa_2 c_2\| \min\{x, y\}\| + (1 + \|M\|)\kappa_2 c_2 \eta\|Xy\| + \eta\|Xy\| \\ &\leq c_3(\| \min\{x, y\}\| + \eta\|Xy\|), \end{aligned}$$

where

$$c_3 := (1 + \|M\|)\kappa_2 c_2 + 1. \qquad \square$$

We are now in the position to present a first example of a growth function.

Proposition 3.3. *The function* $\rho_1 : \mathbb{R}^{n+n} \to [0, \infty)$*, defined by*

$$\rho_1(z) := \| \min\{x, y\}\|,$$

is a growth function on $\mathcal{Z}$*.*

Proof. Taking into account condition (C2) and that $|ab| = |\max\{a, b\}||\min\{a, b\}|$ is valid for arbitrary $a, b \in \mathbb{R}$, we obtain

$$\|Xy\| = \sqrt{\sum_{i=1}^n (x_i y_i)^2} \leq \sum_{i=1}^n |x_i y_i| \leq C \sum_{i=1}^n |\min\{x_i, y_i\}| \leq C\sqrt{n}\rho_1(z), \tag{3.8}$$

where $C > 0$ denotes the constant from condition (C2). Using Lemma 3.2 we therefore have, $\forall z \in \mathcal{Z}$,

$$\operatorname{dist}(z|\mathcal{S}) \leq c_3\rho_1(z) + c_3\eta\|Xy\| \leq \kappa_3\rho_1(z),$$

where $\kappa_3 = c_3(1 + C\eta\sqrt{n})$. On the other hand, the function ρ_1 is globally Lipschitz continuous on $\mathbb{R}^{n+n}$ (see [10]); let L be its Lipschitz constant. Then, denoting by $z^\perp$ the orthogonal projection of z onto $\mathcal{S}$, we get

$$\rho_1(z) = |\rho_1(z) - \rho_1(z^\perp)| \le L\|z - z^\perp\| = L\mathrm{dist}(z|\mathcal{S})$$

for each $z \in \mathcal{Z}$. Hence, ρ_1 satisfies Definition 2.2 with $c_1 := \max\{\kappa_3, L\}$. □

Using the previous proposition it is now easy to build other growth functions. In the next corollary we give two more examples.

COROLLARY 3.4. *The functions $\rho_2, \rho_3 : \mathbb{R}^{n+n} \to [0, \infty)$ defined by*

$$\rho_2(z) := \left\| \left(\sqrt{x_1^2 + y_1^2} - x_1 - y_1, \ldots, \sqrt{x_n^2 + y_n^2} - x_n - y_n \right) \right\|$$

and

$$\rho_3(z) := \|\min\{x, y\}\| + \|Xy\|$$

are growth functions on $\mathcal{Z}$.

Proof. It is known (see [19]) that a positive constant κ_4 exists such that

$$\frac{1}{\kappa_4}\rho_1(z) \le \rho_2(z) \le \kappa_4\rho_1(z) \qquad \forall z \in \mathbb{R}^{2n}.$$

From these relations and from Proposition 3.3 it then easily follows that ρ_2 is a growth function on $\mathcal{Z}$.

We next examine ρ_3. Because of Proposition 3.3 and (3.8), it follows immediately from the definitions of ρ_1 and ρ_3 that

$$\frac{1}{c_1}\mathrm{dist}(z|\mathcal{S}) \le \rho_1(z) \le \rho_3(z) \le (1 + C\sqrt{n})\rho_1(z) \le c_1(1 + C\sqrt{n})\mathrm{dist}(z|\mathcal{S})$$

$\forall z \in \mathcal{Z}$; i.e., ρ_3 is a growth function. □

4. Rates of convergence.

The main point to consider when assessing the quality of estimates $\mathcal{B}(z, \alpha)$, $\mathcal{N}(z, \alpha)$, and $\mathcal{J}(z, \alpha)$ is: How large is the region where these estimates coincide with the sets they approximate? Unfortunately, it seems difficult to give theoretical results in this direction, and the only way we know to treat this point is through numerical experiments. However, in an effort to get some theoretical insight into this problem, some researchers turned to the study of the convergence rates of the indicator function values when z tends to the solution set $\mathcal{S}$. In this section we consider this issue. On the other hand, we think that the importance of these results should not be overestimated since the connection between convergence rates and the wideness of the region of correct identification is, from the theoretical point of view, loose.

We first state a technical lemma.

LEMMA 4.1. *The inequality*

$$0 \le \frac{\xi}{\xi - r} - 1 \le \frac{4r}{\xi}$$

holds $\forall \xi, r \in \mathbb{R}$ with $\xi > 0$ and $0 \le r \le 0.75\xi$.

Proof. The left inequality is obvious. On the other hand, the inequality on the right-hand side is equivalent to

$$\frac{r}{\xi - r} \leq \frac{4r}{\xi},$$

which, in turn, is equivalent to

$$r\xi \leq 4r(\xi - r) = 4r\xi - 4r^2$$

since $\xi > 0$ and $\xi - r > 0$. Now, this inequality is satisfied if and only if

$$0 \leq 3\xi - 4r,$$

and this is true because $r \leq 0.75\xi$ by assumption. □

The following result relates the convergence rate of the indicator functions to the convergence rate of the distance of the point z to the solution set $\mathcal{S}$.

THEOREM 4.2. *Let $\alpha \in (0,1)$ be given. Then, for $z \in \mathcal{Z}$ sufficiently close to $\mathcal{S}$, it holds that*

$$|S(x_i, z; \alpha) - 1| = O(\text{dist}(z|\mathcal{S})^\alpha) \qquad \forall i \in \mathcal{B}, \tag{4.1}$$

$$|S(y_i, z; \alpha) - 1| = O(\text{dist}(z|\mathcal{S})^\alpha) \qquad \forall i \in \mathcal{N}, \tag{4.2}$$

$$|\max\{S(x_i, z; \alpha), S(y_i, z; \alpha)\}| = O(\text{dist}(z|\mathcal{S})^{1-\alpha}) \qquad \forall i \in \mathcal{J}. \tag{4.3}$$

Proof. We prove (4.1) by applying Lemma 4.1 with $\xi := x_i$ and $r := \rho(z)^\alpha$. So let $i \in \mathcal{B}$ be an arbitrary but fixed index. Since we need to consider only $z \in \mathcal{Z}_\varepsilon$ with $\text{dist}(z|\mathcal{S}) \leq \varepsilon$ sufficiently small, $r = \rho(z)^\alpha \leq 0.75x_i = 0.75\xi$ follows for these z because of condition (C1) and (2.1). Moreover, $\xi = x_i > 0$ is obvious. Therefore, Lemma 4.1 can be applied and yields, having condition (C1) and (2.1) in mind,

$$|S(x_i, z; \alpha) - 1| = \frac{x_i}{x_i - \rho(z)^\alpha} - 1 \leq \frac{4}{x_i}\rho(z)^\alpha \leq \frac{4}{\delta}c_1^\alpha \text{dist}(z|\mathcal{S})^\alpha \quad \forall i \in \mathcal{B}. \tag{4.4}$$

The proof of (4.2) is similar and we omit it.

Now, consider an arbitrary but fixed $i \in \mathcal{J}$. To prove (4.3) we first show that $|S(x_i, z; \alpha)| = O(\text{dist}(z|\mathcal{S})^{1-\alpha})$. Since only those $z \in \mathcal{Z}_\varepsilon$ with $S(x_i, z; \alpha) \neq 0$ need to be considered, the definition of the indicator function S immediately implies that $x_i \neq 0$. Since $i \in \mathcal{J}$, this means that z is not a solution of the LCP, so that $\rho(z) > 0$. Using

$$\rho(z) \leq c_1 \text{dist}(z|\mathcal{S}) \leq c_1\varepsilon$$

$\forall z \in \mathcal{Z}_\varepsilon$, we therefore obtain the existence of a sufficiently small $\varepsilon > 0$ such that

$$c_1\rho(z) = [c_1\rho(z)^{1-\alpha}]\rho(z)^\alpha < \rho(z)^\alpha$$

holds $\forall z \in \mathcal{Z}_\varepsilon$. Now Proposition 2.1 and (2.1) imply that

$$|x_i| \leq \text{dist}(z|\mathcal{S}) \leq c_1\rho(z) < \rho(z)^\alpha$$

$\forall z \in \mathcal{Z}_\varepsilon$ with $z = (x, y)$. Thus, we can introduce $a(z) := \rho(z)^\alpha / |x_i|$ and observe that, by Proposition 2.1, (2.1), and $\rho(z) > 0$,

$$0 < \frac{1}{a(z)} \leq c_1^\alpha \text{dist}(z|\mathcal{S})^{1-\alpha}. \tag{4.5}$$

This yields $a(z) \to \infty$ for $\operatorname{dist}(z|\mathcal{S}) \to 0$. Therefore, we have, for $z \in \mathcal{Z}_\varepsilon$ with ε sufficiently small,

$$|S(x_i, z; \alpha)| = \left| \frac{x_i}{x_i - \rho(z)^\alpha} \right| = \frac{1}{|1 - \rho(z)^\alpha / x_i|} \leq \frac{1}{|a(z)| - 1} \leq \frac{2}{a(z)}.$$

Together with (4.5) this gives

$$|S(x_i, z; \alpha)| = O(\operatorname{dist}(z|\mathcal{S})^{1-\alpha}).$$

The same result can be shown for $|S(y_i, z; \alpha)|$ in a similar way, so that (4.3) follows. □

5. Rates of convergence and complementarity gap. The result in the previous section is geometrically very appealing, since it relates the convergence rates of the indicator functions to the Euclidean distance to the solution set. However, in connection with interior-point methods, it is also important to relate this distance to the normalized complementarity gap

$$\mu := \mu(z) := \frac{x^T y}{n}.$$

In fact, in interior-point methods a convergence rate is often established for μ (and not for the distance); see, for example, the recent books [17, 20, 21] for a general background on interior-point methods.

Instead of the set $\mathcal{Z}$ we will now make use of its nonnegative part

$$\mathcal{Z}^+ := \{z \in \mathcal{Z} \mid z \geq 0\}.$$

Note that virtually every interior-point method will generate sequences $\{z^k\}$ belonging to $\mathcal{Z}^+$ eventually.

Before giving the main result of this section we relate the distance $\operatorname{dist}(z|\mathcal{S})$ to the complementarity gap.

PROPOSITION 5.1. *If $\mathcal{J} \neq \emptyset$, there is a constant $c_4 > 0$ such that*

$$\operatorname{dist}(z|\mathcal{S}) \leq c_4 \sqrt{\mu} \tag{5.1}$$

$\forall z \in \mathcal{Z}^+$. If, instead, $\mathcal{J} = \emptyset$, then there is a constant $c_5 > 0$ such that

$$\operatorname{dist}(z|\mathcal{S}) \leq c_5 \mu \tag{5.2}$$

$\forall z \in \mathcal{Z}^+$ sufficiently close to $\mathcal{S}$.

Proof. From Lemma 3.2, we have

$$\operatorname{dist}(z|\mathcal{S}) \leq c_3 \left(\| \min\{x, y\} \| + n\eta\mu \right). \tag{5.3}$$

Since $\min\{a, b\} \leq \sqrt{ab}$ is valid for arbitrary $a, b \geq 0$ and since $z \geq 0$, we obtain that

$$\| \min\{x, y\} \|^2 = \sum_{i=1}^n \min{}^2\{x_i, y_i\} \leq \sum_{i=1}^n x_i y_i = x^T y = n\mu.$$

This and (5.3) gives

$$\operatorname{dist}(z|\mathcal{S}) \leq c_3 \left(\sqrt{n} + n\eta\sqrt{\mu} \right) \sqrt{\mu}.$$

In view of condition (C2), there is a constant $\kappa_5 > 0$ such that

$$\sqrt{\mu} = \sqrt{\frac{x^T y}{n}} \leq \kappa_5$$

$\forall z = (x, y) \in \mathcal{Z}^+$. Hence, it follows that

$$\operatorname{dist}(z|\mathcal{S}) \leq c_4 \sqrt{\mu}$$

for $c_4 := c_3(\sqrt{n} + n\eta\kappa_5)$.

If $\mathcal{J} = \emptyset$, we have $\mathcal{B} \cup \mathcal{N} = \{1, \ldots, n\}$ by Proposition 1.1. Thus, condition (C1) gives

$$\min\{x_i, y_i\} \leq \frac{x_i y_i}{\delta}$$

for every i and for every $z \in \mathcal{Z}^+$ sufficiently close to $\mathcal{S}$. Hence, we get from condition (C1):

$$\|\min\{x, y\}\| \leq \frac{1}{\delta}\left(\sum_{i=1}^{n} x_i^2 y_i^2\right)^{1/2} \leq \frac{1}{\delta}\sum_{i=1}^{n} x_i y_i = \frac{n}{\delta}\mu.$$

Inequality (5.2) now follows from (5.3) by taking $c_5 := c_3(n/\delta + n\eta)$. □

Note that Proposition 5.1 depends on the column sufficiency of the matrix M because we use both Lemma 3.2 (which presupposes convexity of $\mathcal{S}$) and Proposition 1.1.

If the matrix M is assumed to be positive semidefinite, Proposition 5.1 can be derived from known error bound results. We refer the reader to [13, 14] for the case $\mathcal{J} \neq \emptyset$ and to [12] for $\mathcal{J} = \emptyset$. Here we have proved Proposition 5.1 under the mere conditions that $z \in \mathcal{Z}^+$ and that M is CS.

In the next theorem we give convergence rates with respect to μ. These convergence rates easily follow from Theorem 4.2 and Proposition 5.1.

THEOREM 5.2. *Let $\alpha \in (0, 1)$ be given. If $\mathcal{J} \neq \emptyset$, then, for $z \in \mathcal{Z}^+$ and $\mu \to 0$, it holds that*

$$\begin{aligned}
|S(x_i, z; \alpha) - 1| &= O(\mu^{\alpha/2}) &\quad& \forall i \in \mathcal{B},\\
|S(y_i, z; \alpha) - 1| &= O(\mu^{\alpha/2}) && \forall i \in \mathcal{N},\\
|\max\{S(x_i, z; \alpha), S(y_i, z; \alpha)\}| &= O(\mu^{(1-\alpha)/2}) && \forall i \in \mathcal{J}.
\end{aligned}$$

If, instead, $\mathcal{J} = \emptyset$, then, for $z \in \mathcal{Z}^+$ and $\mu \to 0$, it holds that

$$\begin{aligned}
|S(x_i, z; \alpha) - 1| &= O(\mu^{\alpha}) &\quad& \forall i \in \mathcal{B},\\
|S(y_i, z; \alpha) - 1| &= O(\mu^{\alpha}) && \forall i \in \mathcal{N}.
\end{aligned}$$

Theorems 4.2 and 5.2 clearly show that the convergence rates of the indicator functions depend on α. In general if we want to maximize the slower convergence rate, the best value for α is 0.5. On problems which are known to be nondegenerate, for example in the LP case, a value of α close to 1 may be preferred instead. The different way in which α influences the convergence rate of nondegenerate and degenerate indices also suggests the idea of using two different values of α: a value close to 1 in the definition of $\mathcal{B}(z; \alpha)$ and $\mathcal{N}(z; \alpha)$, and a value close to 0 in the definition of $\mathcal{J}(z; \beta)$ (where we used the symbol β to point out that this value is different from the one used in the approximation of nondegenerate indices). It is not difficult to see that all the results we proved go through after this minor modification. However, in this case the sets $\mathcal{B}(z;\alpha)$, $\mathcal{N}(z;\alpha)$, and $\mathcal{J}(z;\beta)$ need not be everywhere pairwise disjoint, even if this will always be the case eventually, and this may require the definition of additional rules to decide to which set to assign an index that belongs to more than one set among $\mathcal{B}(z;\alpha)$, $\mathcal{N}(z;\alpha)$, and $\mathcal{J}(z;\beta)$.

6. Numerical results. In order to get a feeling for the practical results that can be obtained with the new identification technique, in this section we present a summary of the results of an extensive numerical testing [6]. We report the results obtained by using

(i) the Tapia indicator [3, 9, 14], probably the best indicator available to date for linear programs [3];

(ii) the new indicator; and

(iii) the intersection indicator, that is, a combination of the Tapia indicator and the new indicator.

The Tapia indicator and its characteristics are studied in detail in references [3, 9, 14]. Here we only recall some essential facts:

(a) The Tapia indicator can be applied only to a specific (although broad) class of interior-point methods for LCPs.

(b) Given a sequence of points $\{z^k\}$ generated by a suitable interior-point method and converging to the solution set of an LCP, quantities T_x^k and T_y^k are associated with each $z^k = (x^k, y^k)$ so that, under assumptions which are similar to but stronger than conditions (C1)–(C3) used in this paper,

$$\lim_{k\to\infty} T_x^k = \begin{cases} 1 & \text{if } i \in \mathcal{B}, \\ 0.5 & \text{if } i \in \mathcal{J}, \\ 0 & \text{if } i \in \mathcal{N}, \end{cases} \qquad \lim_{k\to\infty} T_y^k = \begin{cases} 0 & \text{if } i \in \mathcal{B}, \\ 0.5 & \text{if } i \in \mathcal{J}, \\ 1 & \text{if } i \in \mathcal{N}. \end{cases}$$

We tested the three identification strategies mainly on the `netlib` collection of LP problems. Although our identification technique can be applied to a much broader class of problems, we believe that LP represents the major field of application of the techniques described in this paper; furthermore, no collection of (CS) linear complementarity test problems comparable to `netlib` exists to date. Therefore, we decided, in this first stage of our numerical experience, to deal almost exclusively with LP. We stress, however, that these tests cover only a special case of the theory developed in the previous sections. In fact, the Goldman–Tucker theorem (see [7] or [20] for a more recent reference) shows that any linear program is nondegenerate, so that in the LP case we never encounter the case $\mathcal{J} \neq \emptyset$. For that reason, we will also include a short discussion with some numerical results obtained for LCPs.

For each LP problem, we used the LIPSOL program by Zhang [22, 23] to generate a sequence of points converging to the solution set of the linear program. LIPSOL is a MATLAB/FORTRAN implementation of a predictor-corrector infeasible interior-point method. We added some lines in this code in order to calculate, at each iteration, approximations of the index sets $\mathcal{B}$ and $\mathcal{N}$ (recall that $\mathcal{J} = \emptyset$). More precisely, within each iteration, we calculate the values $S(x_i^k, z^k; \alpha)$ and $S(y_i^k, z^k; \alpha)$ after each corrector step and T_x^k and T_y^k after each predictor step. Based on these values, we approximate the index sets $\mathcal{B}$ and $\mathcal{N}$ at iteration k as follows.

(i) For the Tapia indicator we set

$$\mathcal{B}_T^k := \{i|\ \min\{T_x^k, 1 - T_y^k\} \geq 1 - \theta\},$$
$$\mathcal{N}_T^k := \{i|\ \min\{T_y^k, 1 - T_x^k\} \geq 1 - \theta\}.$$

(ii) For the new indicator we set, in a similar way,

$$\mathcal{B}_S^k := \{i \mid \min\{S(x_i^k, z^k; \alpha), 1 - S(y_i^k, z^k; \alpha)\} \geq 1 - \theta\},$$
$$\mathcal{N}_S^k := \{i \mid \min\{S(y_i^k, z^k; \alpha), 1 - S(x_i^k, z^k; \alpha)\} \geq 1 - \theta\}.$$

(iii) Finally, for the intersection indicator, we calculate approximations $\mathcal{B}_{ST}^k$ and $\mathcal{N}_{ST}^k$ by intersecting the previous estimates:

$$\mathcal{B}_{ST}^k := \mathcal{B}_S^k \cap \mathcal{B}_T^k,$$
$$\mathcal{N}_{ST}^k := \mathcal{N}_S^k \cap \mathcal{N}_T^k.$$

The rationale behind this last estimate is simply that our new indicator and the Tapia indicator are based on a totally different approach, so that if an index is estimated to be active (or nonactive) by both indicators then, and only then, we expect this prediction to be true. Note that the use of two indicators to confirm the information obtained from each one of them is also advocated in [3].

For all test runs we chose $\theta = 0.1$. Moreover, we set $\alpha = 0.5$ at the beginning of each test run and updated α after each step by

$$\alpha = \max\{\alpha, 1 - 100 * \text{TRERROR}\},$$

where TRERROR denotes a certain residual calculated within the LIPSOL program which, basically, measures the violation of the optimality conditions at the current iterate. Furthermore, we used ρ_1 as a growth function in order to compute $S(x_i^k, z^k; \alpha)$ and $S(y_i^k, z^k; \alpha)$. In view of our preliminary experience, however, the results do not change dramatically by using another growth function.

The first problem we have to tackle when analyzing the results is how to assess the quality of an indicator. While it is intuitively clear that an indicator is good if it can accurately estimate, at an early stage, the sets $\mathcal{B}$ and $\mathcal{N}$, it is not entirely clear the exact way we should measure this accuracy. In our experiments we chose to consider the following three quality indices. For simplicity we describe them making reference only to the new estimates $\mathcal{B}_S^k$ and $\mathcal{N}_S^k$, but it is obvious that analogous considerations can be made with reference to the Tapia indicator and to the intersection indicator.

(1) *Percentage of misclassified indices at iteration k.* At each iteration a variable, x_i for example, can be classified as either active ($i \in \mathcal{N}_S^k$) or nonactive ($i \in \mathcal{B}_S^k$), or it can be not classified at all ($i \notin \mathcal{N}_S^k$ and $i \notin \mathcal{B}_S^k$). The percentage of misclassified indices at iteration k is the number of indices estimated to belong to $\mathcal{B}$ ($\mathcal{N}$) at that iteration and that instead, at a solution belonging to the relative interior of the solution set, belong to $\mathcal{N}$ ($\mathcal{B}$). In formulas this corresponds to

$$100 \frac{|\mathcal{B}_S^k \setminus \mathcal{B}| + |\mathcal{N}_S^k \setminus \mathcal{N}|}{n}.$$

(2) *Percentage of correctly classified indices at iteration k.* This is easily understood to be

$$100 \frac{|\mathcal{B}_S^k \cap \mathcal{B}| + |\mathcal{N}_S^k \cap \mathcal{N}|}{n}.$$

(3) *Percentage of globally correctly classified indices at iteration k.* We say that a certain index is globally correctly identified at iteration k if its identification status is correct at iteration k and does not change from that iteration on.

Roughly speaking, the first quality index described above measures the excess of $\mathcal{B}_S^k$ and $\mathcal{N}_S^k$ over $\mathcal{B}$ and $\mathcal{N}$, respectively, while the second index measures the excess of $\mathcal{B}$ and $\mathcal{N}$ over $\mathcal{B}_S^k$ and $\mathcal{N}_S^k$, respectively. All the indices are correctly classified at iteration k if the percentage of misclassified indices is 0 and that of correctly classified ones is 100. However, neither of the two quality indices alone allows us to assess the quality of the current guessing. The third quality index is similar to the second one but with a greater emphasis on stability of the indicators. According to one's purposes one of the three quality indices above may be more important than the others, and other indices may be of interest too. However, we think that these three quality indices, considered together, give a fairly reasonable picture of the behavior of the indicators.

There is another difficulty we must mention. The evaluation of the above quality indices assumes the knowledge of $\mathcal{B}$ and $\mathcal{N}$, but this is not the case, in general, for the `netlib` problems we used. Therefore, we assumed that if in the final iteration the estimates obtained using the new indicator and the Tapia indicator coincide, i.e., if at the last iteration $\mathcal{B}_S^k = \mathcal{B}_T^k$ and $\mathcal{N}_S^k = \mathcal{N}_T^k$ hold, then these estimates coincide with $\mathcal{B}$ and $\mathcal{N}$. We ran LIPSOL on all the problems using the default parameters, but it turns out that on a considerably high percentage of problems the new indicator and the Tapia one do not coincide at the last iteration. Hence we changed the main stopping criterion (TOL) of LIPSOL from 10^{-8} to 10^{-11}. The satisfaction of this more stringent termination criterion usually required only one or two additional iterations and increased the number of problems on which the two indicators coincide at the last iteration.

Unfortunately, it is not always possible to reach this higher accuracy and we were therefore forced to consider only the 73 problems that were successfully solved with TOL=10^{-11}. For 9 of these 73 problems we do not have coincidence of the indications obtained by the new and the Tapia indicator. Since the resulting set of 64 test problems appears to be significant we have not tried to enlarge this set of test problems. In the next three subsections we summarize the behavior of the indicators on the test problem set. Because of lack of space, it is impossible to report here the complete numerical results. We tried to give a fair representation of these results by reporting some summary tables that highlight the main features of the indicators. However, it should always be kept in mind that our comments are based on the complete set of numerical results. The interested reader can find the complete and detailed numerical results in the companion report [6].

6.1. Misclassified indices.

We recall that the percentage of misclassified indices appears to be particularly important in those cases in which one wants to reduce the dimension of the problem by fixing variables to 0. In this case a high number of misclassified indices can adversely affect the efficiency of the procedure (see [3]). More in general, we tend to view this index as an important one because it tells us how much we can trust the guessing. It is useless to have a high percentage of correctly identified indices (something assessed by the indices analyzed in the next two sections) if these indices are mixed with too many misclassified ones. We also recall that the misclassified indices should not be confused with the unclassified ones.

We summarize the results in two tables. In Table 6.1 we report the number of test examples for which we have less than 1% of misclassified variables during the last

Table 6.1
Number of problems with less than 1% misclassified variables.

Iteration	Tapia indicator	New indicator	Intersection
k_f	64	64	64
$k_f - 1$	64	61	64
$k_f - 2$	64	49	64
$k_f - 3$	62	34	62
$k_f - 4$	57	24	58
$k_f - 5$	40	20	52
$k_f - 6$	28	18	47
$k_f - 7$	17	15	40
$k_f - 8$	10	10	31
$k_f - 9$	5	9	28
$k_f - 10$	3	5	27
$k_f - 11$	1	4	22
$k_f - 12$	1	3	17
$k_f - 13$	1	3	13
$k_f - 14$	1	3	11
$k_f - 15$	1	2	8
$k_f - 16$	0	2	7
$k_f - 17$	0	1	5
$k_f - 18$	0	1	3
$k_f - 19$	0	0	3
$k_f - 20$	0	0	2
$k_f - 21$	0	0	2
$k_f - 22$	0	0	1
$k_f - 23$	0	0	1

Table 6.2
Number of problems with no misclassified variables (intersection indicator).

Iteration	k_f	$k_f - 1$	$k_f - 2$	$k_f - 3$	$k_f - 4$	$k_f - 5$	$k_f - 6$	$k_f - 7$	$k_f - 8$	$k_f - 9$
	64	64	62	51	40	20	12	6	4	3

24 iterations. In this table, and in the following ones, k_f denotes the final iteration, and so $k_f - 1$ is the last but one iteration, and so on.

We see that the Tapia indicator has a better behavior than the new indicator. Indeed, for the majority of test examples the Tapia indicator has less than 1% of misclassified variables in the last five iterations, whereas for the new indicator there is a considerable number of problems with more than 1% misclassified variables even three or four steps before the final iteration.

The most interesting conclusion one can draw from Table 6.1, however, is the superior behavior of the intersection indicator. In view of its very definition, it is clear that this indicator has fewer misclassified variables than the other two indicators; however, it is not clear, a priori, that such a good behavior could be obtained. Actually, the analysis of the complete numerical results [6] shows that the number of misclassified variables by the intersection indicator is very small at almost all iterations and for almost all test examples. To further illustrate the behavior of the intersection indicator, in Table 6.2, we report how many test problems have no misclassified variables in the last 10 iterations when using the intersection indicator.

Table 6.3
Number of problems with 50% *correct identification.*

Iteration	Tapia indicator	New indicator	Intersection
k_f	64	64	64
$k_f - 1$	64	64	64
$k_f - 2$	64	64	64
$k_f - 3$	64	64	64
$k_f - 4$	64	61	58
$k_f - 5$	63	58	52
$k_f - 6$	61	57	46
$k_f - 7$	57	54	37
$k_f - 8$	48	51	32
$k_f - 9$	46	48	27
$k_f - 10$	41	44	23
$k_f - 11$	30	37	20
$k_f - 12$	25	33	18
$k_f - 13$	24	27	15
$k_f - 14$	22	27	14

Table 6.4
Number of problems with 100% *correct identification.*

Iteration	Tapia indicator	New indicator	Intersection
k_f	64	64	64
$k_f - 1$	51	46	42
$k_f - 2$	18	14	13
$k_f - 3$	2	2	0

The numbers provided by this table are still very encouraging and show that a suitable combination of the new and the Tapia indicator provides useful information.

6.2. Correctly identified indices. As we already observed, this is the second index essential to assessing the quality of an indicator. Table 6.3 shows for how many test examples we have at least 50% correctly identified indices in the last 15 iterations. We do not consider iterations before $k_f - 14$ because, by the results reported in Table 6.1, before this iteration for most problems the number of misclassified indices is higher than 1% so that the information provided by the indicators is not reliable. Table 6.4 is analogous to Table 6.3, but in this case we consider problems for which all classified indices are correctly classified.

From Tables 6.3 and 6.4 we see that the new indicator and the Tapia one have a similar behavior, although the new indicator seems able to better classify indices in early stages, while the Tapia indicator behaves better when close to a solution. By its very definition the intersection indicator is expected to have the worst behavior with respect to the percentage of correctly classified indices. However, the performance of this percentage is still more than acceptable, and furthermore the results of this section should always be read in the light of the results of the previous section showing that the intersection indicator is “slower” than the other two but more reliable.

Looking at the complete results we may also note that there is a surprisingly high number of problems where more than 50% of indices are correctly classified already in the very first iterations.

Table 6.5

Globally correctly identified variables at first iteration.

%	Tapia indicator	New indicator	Intersection
0–10	37	9	42
10–20	4	7	1
20–30	4	13	3
30–40	6	7	7
40–50	10	7	8
50–60	3	8	3
60–70	0	6	0
70–80	0	1	0
80–90	0	6	0
90–100	0	0	0

Table 6.6

Globally correctly identified variables at iteration $k_f - 4$.

%	Tapia indicator	New indicator	Intersection
0–10	0	0	0
10–20	0	0	0
20–30	0	0	0
30–40	0	1	2
40–50	0	2	4
50–60	1	6	6
60–70	2	6	7
70–80	9	11	14
80–90	10	22	18
90–100	42	16	13

6.3. Globally correctly identified indices. Also for this quality index we report two tables to summarize the results. This quality index is similar to the previous one with a greater emphasis on stability of the indicators. To give the reader a different point of view, however, the tables we report have a different structure than those of section 6.2. In Table 6.5 we report, for each indicator, the number of problems for which the percentage of globally correctly identified indices at the first iteration is between 0% and 10%, 10% and 20%, and so on. The same kind of data is reported in Table 6.6 for the iteration $k_f - 4$.

Obviously the *globally* correct classification of indices is more difficult than the simple correct identification of a certain number of indices. However, the qualitative behavior that emerges from the two tables and also from the analysis of the complete numerical results is very similar to the one described in the previous subsection: The new indicator behaves (considerably) better in early stages than the Tapia indicator, which, however, is superior eventually. The intersection indicator is obviously worse than the other two indicators, even if not drastically so, but the information it provides should be regarded as more reliable.

6.4. LCPs. In addition to our numerical results obtained for linear programs based on a suitable modification of the LIPSOL solver, we also wanted to see the behavior of the indicators when applied to LCPs, mainly because here we may have $\mathcal{J} \neq \emptyset$. To this end, we implemented an infeasible interior-point method in MATLAB using the framework from [4]. As test problems, we used some convex optimization problems from [8] as well as several randomly generated problems. The overall behavior of the different indicators seems to be very similar for most of these test problems; however, we also observed that $\mathcal{J} = \emptyset$ for almost all these test problems.

In the following, we therefore report some more details on only one particular example which has a nonempty index set $\mathcal{J}$. This example is of dimension $n = 30$ and is constructed as follows. Let

$$x^* := (\underbrace{1, \ldots, 1}_{15\times}, \underbrace{0, \ldots, 0}_{15\times})^T, \qquad y^* := (\underbrace{0, \ldots, 0}_{20\times}, \underbrace{1, \ldots, 1}_{10\times})^T$$

be a given solution of the LCP; let D be the positive semidefinite diagonal matrix

$$D := \operatorname{diag}(\underbrace{0, \ldots, 0}_{5\times}, \underbrace{1, \ldots, 1}_{25\times});$$

let A be an $n \times n$ matrix with randomly distributed entries $a_{ij} \in [0, 10]$; and define

$$X := A^T A + I.$$

Then X is nonsingular. Hence

$$M := X^T D X$$

is a positive semidefinite matrix with the same number of zero and positive eigenvalues as D (by Sylvester's law of inertia); i.e., M has 5 zero and 25 positive eigenvalues. Finally, let us define

$$q := y^* - M x^*.$$

This guarantees that (x^*, y^*) is indeed a solution of the LCP which violates the strict complementarity condition $x_i^* + y_i^* > 0$ for $i = 16, \ldots, 20$.

We illustrate the behavior of the Tapia indicator and the new indicator for one particular instance of this example in Figures 6.1 and 6.2, respectively. These figures give the values of T_x^k and $S(x_i^k, z^k; \alpha)$.

From these figures, it is obvious that the Tapia indicator has somewhat unpredictable behavior even in a small neighborhood of a solution, whereas the new indicator behaves much more smoothly and seems to provide considerably more reliable information than the Tapia indicator. The reason for this significant difference is not totally clear to us. Maybe it is because $\mathcal{J} \neq \emptyset$ for this example. However, it might also have to do with the fact that the Tapia indicator depends on an accurate solution of a linear system which typically becomes almost singular close to the solution set, and that the MATLAB linear system solver is less robust than the FORTRAN solver called within the LIPSOL program for the LP test problems.

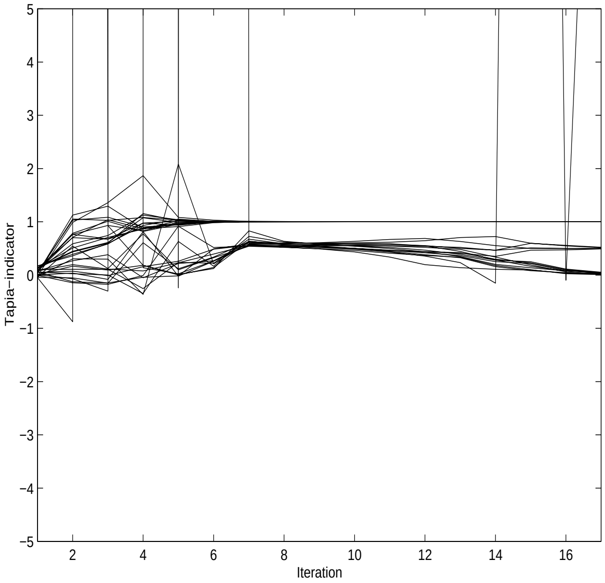


FIG. 6.1. *Behavior of the Tapia indicator for a degenerate LCP.*

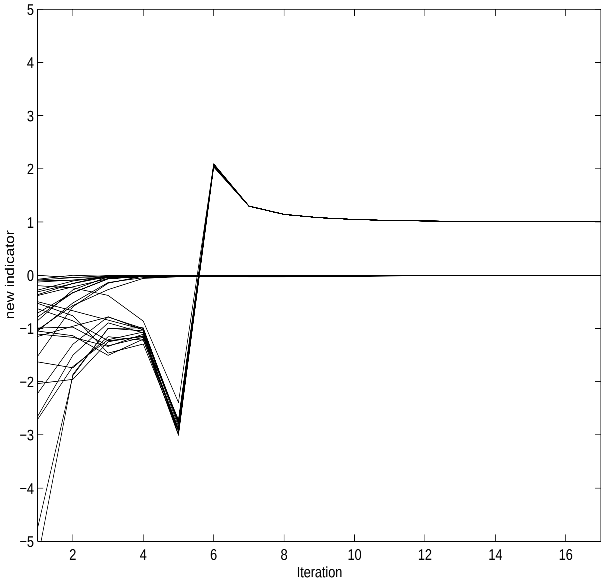


FIG. 6.2. *Behavior of the new indicator for a degenerate LCP.*

7. Summary and conclusions. In this paper we introduced a new technique for identifying the status of variables in the relative interior of the solution set of a CS LCP by using the information available in "nearby" points. The theoretical properties of the new indicator appear to be interesting. The technique we propose may be the only available option for some classes of problems (CS LCPs which are not monotone, for example) or algorithms (smoothing techniques, for example).

We tested the technique on LP problems in an interior-point framework and compared its behavior to the Tapia indicator. The results are encouraging and, in our opinion, indicate the practical viability of our approach. The combination of the new indicator with the Tapia one appears to be particularly promising. Since the computational cost of our technique is very low, this combination certainly deserves further study, at least in the LP case. However, the numerical results we report should be regarded as preliminary. In fact, on the one hand, the behavior of the new technique can probably be improved by considering different choices for the parameters α, θ (for example, a different α can be used for each index or in the definition of the sets $\mathcal{B}^k$, $\mathcal{N}^k$, and $\mathcal{J}^k$) and for the indicator function; on the other hand, the behavior of the identification technique on wider classes of problems should also be investigated.

Finally, the use of identification techniques to actually facilitate the solution of LCPs is an issue that certainly deserves careful examination and that we intend to study in the near future. We refer the interested reader to [3] for a good introduction to this kind of problem.

Acknowledgment. The authors would like to thank Jun Ji for some very helpful discussions on interior-point methods for P_*-matrix LCPs.

REFERENCES

[1] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.

[2] R. W. Cottle, J.-S. Pang, and V. Venkateswaran, *Sufficient matrices and the linear complementarity problem*, Linear Algebra Appl., 114/115 (1989), pp. 231–249.

[3] A. S. El-Bakry, R. A. Tapia, and Y. Zhang, *A study of indicators for identifying zero variables in interior-point methods*, SIAM Rev., 36 (1994), pp. 45–72.

[4] A. S. El-Bakry, R. A. Tapia, and Y. Zhang, *On the convergence rate of Newton interior-point methods in the absence of strict complementarity*, Comput. Optim. Appl., 6 (1996), pp. 157–167.

[5] F. Facchinei, A. Fischer, and C. Kanzow, *On the accurate identification of active constraints*, SIAM J. Optim., 9 (1998), pp. 14–32.

[6] F. Facchinei, A. Fischer, and C. Kanzow, *On the Identification of Zero Variables in an Interior-Point Framework: Complete Numerical Results*, Mathematical Programming Technical Report 98–05–results, Computer Sciences Department, University of Wisconsin–Madison, Madison, WI, 1998; also available online from http://www.math.uni-hamburg.de/home/kanzow, http://www-lsx.mathematik.uni-dortmund.de/user/lsx/fischer, and http://dis.uniroma1.it/~facchinei.

[7] A. J. Goldman and A. W. Tucker, *Theory of linear programming,* in Linear Inequalities and Related Systems, H. W. Kuhn and A. W. Tucker, eds., Princeton University Press, Princeton, NJ, 1966, pp. 53–97.

[8] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econom. and Math. Systems 187, Springer-Verlag, Berlin, Germany, 1981.

[9] J. Ji and F. Potra, *Tapia indicators and finite termination of infeasible-interior-point methods for degenerate LCP*, in Mathematics of Numerical Analysis, Lectures in Appl. Math. 32, J. Renegar, M. Shub, and S. Smale, eds., AMS, Providence, RI, 1996, pp. 443–454.

[10] C. Kanzow and M. Fukushima, *Equivalence of the generalized complementarity problem to differentiable unconstrained minimization*, J. Optim. Theory Appl., 90 (1996), pp. 581–603.

[11] Z.-Q. Luo and P. Tseng, *Error bound and convergence analysis of matrix splitting algorithms for the affine variational inequality problem*, SIAM J. Optim., 2 (1992), pp. 43–54.

[12] O. L. Mangasarian, *Error bounds for nondegenerate monotone linear complementarity problems*, Math. Programming, 48 (1990), pp. 437–445.

[13] O. L. Mangasarian and T.-H. Shiau, *Error bounds for monotone linear complementarity problems*, Math. Programming, 36 (1986), pp. 81–89.

[14] R. D. C. Monteiro and S. J. Wright, *Local convergence of interior-point algorithms for degenerate monotone LCP*, Comput. Optim. Appl., 3 (1994), pp. 131–155.

[15] J.-S. Pang, *Error bounds in mathematical programming*, Math. Programming, 79 (1997), pp. 299–332.

[16] F. A. Potra and R. Sheng, *A superlinearly convergent infeasible-interior-point algorithm for degenerate LCP*, J. Optim. Theory Appl., 97 (1998), pp. 249–269.

[17] C. Roos, T. Terlaky, and J.-Ph. Vial, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley, Chichester, UK, 1997.

[18] R. A. Tapia, *On the role of slack variables in quasi-Newton methods for constrained optimization*, in Numerical Optimization of Dynamic Systems, L. W. C. Dixon and G. P. Szegő, eds., North-Holland, Amsterdam, 1980, pp. 235–246.

[19] P. Tseng, *Growth behaviour of a class of merit functions for the nonlinear complementarity problem*, J. Optim. Theory Appl., 89 (1996), pp. 17–37.

[20] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1996.

[21] Y. Ye, *Interior Point Algorithms: Theory and Analysis*, John Wiley, New York, 1997.

[22] Y. Zhang, *User's guide to LIPSOL: Linear programming interior point solvers, version* 0.4, Optim. Methods Softw., 12 (1999), pp. 385–396.

[23] Y. Zhang, *Solving large-scale linear programs by interior-point methods under the MATLAB environment*, Optim. Methods Softw., 10 (1998), pp. 1–31.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1079–1096

AUTOMATIC PRECONDITIONING BY LIMITED MEMORY QUASI-NEWTON UPDATING*

JOSÉ LUIS MORALES† AND JORGE NOCEDAL‡

Abstract. This paper proposes a preconditioner for the conjugate gradient method (CG) that is designed for solving systems of equations $Ax = b_i$ with different right-hand-side vectors or for solving a sequence of slowly varying systems $A_k x = b_k$. The preconditioner has the form of a limited memory quasi-Newton matrix and is generated using information from the CG iteration. The automatic preconditioner does not require explicit knowledge of the coefficient matrix A and is therefore suitable for problems where only products of A times a vector can be computed. Numerical experiments indicate that the preconditioner has most to offer when these matrix-vector products are expensive to compute and when low accuracy in the solution is required. The effectiveness of the preconditioner is tested within a Hessian-free Newton method for optimization and by solving certain linear systems arising in finite element models.

Key words. preconditioning, conjugate gradient method, quasi-Newton method, Hessian-free Newton method, limited memory method

AMS subject classifications. 49M37, 65K05, 90C30

PII. S1052623497327854

1. Introduction.

We describe a technique for automatically generating preconditioners for the conjugate gradient (CG) method. It is designed either for solving a sequence of linear systems

$$Ax = b_i, \qquad i = 1, \dots, t, \tag{1.1}$$

in which the coefficient matrix is constant but the right-hand side varies, or for solving a sequence of systems

$$A_k x = b_k, \qquad k = 1, \dots, t, \tag{1.2}$$

where the matrices A_k vary slowly and the right-hand sides b_k are arbitrary. We assume in both cases that the coefficient matrices are symmetric and positive definite.

The automatic preconditioner makes use of quasi-Newton updating techniques. It requires that the first problem in (1.1) or (1.2) be solved by the unpreconditioned CG method and, based on the information generated during this run, generates a preconditioner for solving the next linear system in the sequence. More precisely, if $\{x_i\}$ and $\{r_i\}$ denote the sequence of iterates and residuals generated by the CG method when applied to the first of the systems in (1.1) or (1.2), we compute and store the vectors

$$s_i = x_{i+1} - x_i, \qquad y_i = r_{i+1} - r_i, \qquad i = l_1, \dots, l_m, \tag{1.3}$$

*Received by the editors September 25, 1997; accepted for publication (in revised form) September 6, 1999; published electronically June 6, 2000.
http://www.siam.org/journals/siopt/10-4/32785.html

†Departamento de Matemáticas, Instituto Tecnológico Autónomo de México, Río Hondo 1, Col Tizapán San Angel, México D.F. CP 01000, México (jmorales@gauss.rhon.itam.mx). Current address: ECE Department, Northwestern University, Evanston, IL 60208 (morales@ece.nwu.edu). This author was supported by CONACyT grant 25710-A and by Asociación Mexicana de Cultura, A.C.

‡ECE Department, Northwestern University, Evanston, IL 60208 (nocedal@ece.nwu.edu). This author was supported by National Science Foundation grants CCR-9625613 and INT-9416004 and by Department of Energy grant DE-FG02-87ER25047-A004.

corresponding to m iterates of the CG process, where m is an integer selected by the user. We then use these vectors to define a limited memory BFGS matrix H, which we call the *quasi-Newton preconditioner* and which will be used to precondition the CG method when applied to the next problem in the sequence (1.1) or (1.2). The parameter m determines the amount of memory in the preconditioner and is normally chosen to be much smaller than the number of variables so that the cost of applying the preconditioner is not too large.

The first question is how to select the m vectors (1.3) to be used in the definition of the quasi-Newton matrix. The two strategies that have performed best in our tests are to select the *last m* vectors generated during the CG iteration or to take a *uniform* sample of them. In this paper we will concentrate on the second strategy: we will save m vectors that are approximately evenly distributed throughout the CG run. A detailed description of the quasi-Newton preconditioner will be given in the next section, after we have reviewed the main ideas of limited memory BFGS updating.

Our main interest is in accelerating the CG iteration used in Hessian-free Newton methods for nonlinear optimization. There one needs to solve systems of the form (1.2), where A_k is the Hessian of the objective function at the current iterate. Hessian-free Newton methods assume that the Hessian of the objective function is not known explicitly but that products of A_k with a vector can be approximated by finite differences of gradients or by means of automatic differentiation. In either case these products can be very expensive to compute. After showing that the automatic preconditioner appears to be quite useful in a Hessian-free Newton method, we explore its behavior in a different context by testing it in the solution of linear systems arising in finite element models. In these tests we consider problems of both the forms (1.1) and (1.2).

The idea of saving information from the CG iteration in the form of a quasi-Newton matrix is not new. Nash [15] constructs a limited memory matrix with memory $m = 2$, which is different from the one proposed here, to precondition the linear system of equations arising in the Hessian-free Newton method for optimization. O'Leary and Yeremin [21] explore the use of (full-memory) quasi-Newton matrices as preconditioners for the solution of closely related linear systems. Byrd, Nocedal, and Zhu [6] propose an optimization algorithm in which information corresponding to the last m iterations of the CG method is used to update a limited memory matrix. However, in that algorithm the limited memory matrix is used only to compute a search direction and not as a preconditioner for the CG method. The motivation for the automatic preconditioner proposed in this paper arose while performing numerical tests with a Hessian-free Newton method for large scale optimization. We observed that the two-step preconditioner of Nash was effective only in a few test problems but that the technique proposed here gave improvements over a wide range of problems. The objective of this paper is to suggest that the automatic preconditioner is well suited not only to optimization but also within a wider context. Therefore we present our discussion in the framework of the general problems (1.1)–(1.2).

2. The quasi-Newton preconditioner.

In the BFGS updating formula for minimizing a function f (see, e.g., [7, 8, 10]) we are given a symmetric and positive definite $n \times n$ matrix H_k that approximates the inverse of the Hessian of f and a pair of n-vectors $s_k = x_{k+1} - x_k$ and $y_k = \nabla f(x_{k+1}) - \nabla f(x_k)$ satisfying the condition $s_k^T y_k > 0$. Using this we compute a new inverse Hessian approximation H_{k+1} by means of the updating formula

$$H_{k+1} = V_k^T H_k V_k + \rho_k s_k s_k^T, \tag{2.1}$$

where

$$\rho_k = 1/y_k^T s_k, \qquad V_k = I - \rho_k y_k s_k^T. \tag{2.2}$$

We say that the matrix H_{k+1} is obtained by updating H_k once using the *correction pair* $\{s_k, y_k\}$.

Even if H_k is sparse, the new BFGS matrix H_{k+1} will generally be dense so that storing and manipulating it is prohibitive when the number of variables is large. To circumvent this problem, the limited memory approach makes use of an alternative representation of the updating process in which the quasi-Newton matrices are not explicitly formed.

It follows from (2.1)–(2.2) that if an initial matrix $\bar{H}$ is updated m times using the BFGS formula and the m pairs $\{s_i, y_i\}$, $i = k-1, \ldots, k-m$, then the resulting matrix $H(m)$ can be written as

$$\begin{aligned} H(m) = {} & \left(V_{k-1}^T \cdots V_{k-m}^T\right) \bar{H} \left(V_{k-m} \cdots V_{k-1}\right) \\ & + \rho_{k-m} \left(V_{k-1}^T \cdots V_{k-m+1}^T\right) s_{k-m} s_{k-m}^T \left(V_{k-m+1} \cdots V_{k-1}\right) \\ & + \rho_{k-m+1} \left(V_{k-1}^T \cdots V_{k-m+2}^T\right) s_{k-m+1} s_{k-m+1}^T \left(V_{k-m+2} \cdots V_{k-1}\right) \\ & \vdots \\ & + \rho_{k-1} s_{k-1} s_{k-1}^T. \end{aligned} \tag{2.3}$$

Thus instead of forming $H(m)$ we can store the scalars ρ_i and the vectors $\{s_i, y_i\}$, $i = k-1, \ldots, k-m$, which determine the matrices V_i. A recursive formula described in [13, 19] takes advantage of the symmetry in (2.3) to compute the product $H(m)v$ for any vector v with only $4mn$ floating point operations.

The so-called L-BFGS method described in [19, 12, 9] updates Hessian approximations as follows. We first choose a sparse (usually diagonal) initial Hessian approximation $\bar{H}$ and define the first m approximations through (2.3) as $H(1), \ldots, H(m)$. At this stage the storage is full, and to construct the new Hessian approximation, we first delete the oldest correction pair from the set $\{s_i, y_i\}$ to make room for the newest one, $\{s_k, y_k\}$. The new Hessian approximation $H(m+1)$ is defined by (2.3), using the new set of pairs $\{s_i, y_i\}$. This process is repeated during all subsequent iterations: the oldest correction pair is removed to make space for the newest one.

In this paper we are interested in solving positive definite linear systems $Ax = b$, and therefore the function to be minimized is the quadratic $\frac{1}{2}x^T Ax - b^T x$, whose gradient is equal to the residual $r(x) = Ax - b$. Therefore when using the BFGS updating formula to minimize this quadratic, it is appropriate to define s_k and y_k by (1.3). To find a preconditioner for solving a sequence of problems of the form (1.1) with a constant coefficient matrix but different right-hand sides, we proceed as follows. We solve the first of the systems using the unpreconditioned CG method. We save m correction pairs $\{s_i, y_i\}$ generated during this CG iteration and use (2.3) to define the preconditioner to be $H(m)$. We solve the rest of the linear systems in (1.1) using the preconditioned CG method with this fixed preconditioner.

A similar approach can be used for solving the sequence of slowly varying linear systems (1.2). An alternative, in this case, is to generate a new preconditioner during the solution of every linear system so that the preconditioner is always based on the most recently solved system in the sequence (1.2). We will report results using both approaches.

We have experimented with various strategies for selecting the correction pairs to be saved. In analogy with nonlinear optimization we can simply save the last m pairs. But a strategy that is more effective in some cases is to save the correction pairs at regular intervals. Suppose that $m > 1$ and that ncg denotes the number of CG iterations performed during the solution of the first linear system. If we define $\nu = \lfloor ncg/(m-1) \rfloor$, then we would like to save the pairs $\{s_k, y_k\}$ for $k = 0, \nu, 2\nu, \ldots, (m-1)\nu$. Even though this cannot be done in practice since the number ncg of CG iterations is not known beforehand, in the appendix (section 6) we describe an algorithm that dynamically stores the correction pairs so that they are as evenly distributed as possible. This algorithm requires no extra storage or computation and in our tests gives essentially the same results as saving the correction pairs at exactly uniform intervals.

Following the L-BFGS algorithm, we will always choose the initial matrix $\bar{H}$ in (2.3) to be

$$\bar{H} = \frac{s_l^T y_l}{y_l^T y_l} I, \tag{2.4}$$

where l denotes the last correction pair generated in the CG cycle.

We conclude this section by noting that limited memory updating is flexible enough to accommodate information generated at any stage during the solution of the sequence of problems (1.1) or (1.2). In particular the preconditioner could contain correction pairs corresponding to different linear systems, but we will not explore this possibility here.

3. Application to the Hessian-free Newton method. In this section we investigate the effectiveness of the automatic preconditioner within a Hessian-free Newton method for solving the unconstrained optimization problem

$$\text{minimize} \quad f(x). \tag{3.1}$$

Here f is a twice continuously differentiable function of n variables. Our experiments will be performed using Nash's implementation [16, 17] of the Hessian-free Newton algorithm, which we now briefly review.

Given the current estimate x_k of the optimal solution of (3.1), we generate a search direction p_k by approximately minimizing the quadratic model

$$Q_k(p_k) = \nabla f(x_k)^T p_k + \frac{1}{2} p_k^T \nabla^2 f(x_k) p_k. \tag{3.2}$$

The new iterate is then defined to be $x_{k+1} = x_k + \alpha_k p_k$, where the step size α_k is computed by means of a line search procedure; in our code we used the line search routine developed by Moré and Thuente [14].

The approximate solution of (3.2) is obtained by applying the CG method to the system

$$\nabla^2 f(x_k) p = -\nabla f(x_k), \tag{3.3}$$

starting from the initial guess $p_k^{(0)} = 0$ and terminating if a direction of negative curvature is detected or if the following stopping test is satisfied:

$$i \left(1 - \frac{Q_k(p_k^{(i-1)})}{Q_k(p_k^{(i)})} \right) \leq 0.5, \tag{3.4}$$

Table 1

Performance of the Hessian-free Newton method for various values of the memory parameter m in the preconditioner.

	$m = 0$			$m = 4$			$m = 8$			$m = 16$		
Problem	iter	fg	cg	iter	fg	cg	iter	fg	cg	iter	fg	cg
Cvar-2	51	52	871	51	52	760	48	49	578	41	42	610
Penalty-3	25	29	142	21	25	71	20	24	72	20	24	72
Tridiag	24	25	166	24	25	119	20	21	83	20	21	83
QOR	13	14	52	13	14	32	13	14	32	13	14	32
SQRT(2)	37	43	640	34	43	416	31	37	380	36	42	359
Total	150	163	1871	143	159	1398	132	145	1145	130	143	1156

where $\{p^{(i)}\}$ denotes the sequence of CG iterates. This test aims to terminate the CG iteration when the reduction in the quadratic model is judged to be so small that the improvement in the quality of the search direction is not likely to offset the cost of computing it.

In the Hessian-free Newton method [20, 17] it is assumed that the elements of the Hessian matrix $\nabla^2 f$ are not available. One must therefore compute the matrix-vector products required by the CG iteration by automatic differentiation or, as will be done in our tests, approximate them by finite differences,

$$\nabla^2 f(x_k)v \approx \frac{\nabla f(x_k + hv) - \nabla f(x_k)}{h}, \tag{3.5}$$

where $h = (1 + ||x_k||_2)\sqrt{\epsilon_M}$ and ϵ_M denotes unit roundoff. The computational cost of a matrix-vector product in the CG iteration therefore equals the cost of a gradient evaluation. (Current software for automatic differentiation will normally be at least as expensive as finite differences.)

We make use of the automatic preconditioner as follows. During the first iteration of the Hessian-free Newton method we apply the unpreconditioned CG method to compute the first search direction and build a quasi-Newton preconditioner $H(m)$, as discussed in section 2 and using the uniform sampling strategy described in the appendix. This preconditioner is used to compute the next search direction, and during this second iteration we construct a new preconditioner $H(m)$. This process is repeated during every iteration of the Hessian-free Newton method: the search direction is always computed by means of the preconditioned CG method using the preconditioner constructed at the previous iteration. The starting point for every CG run is $p_k^{(0)} = 0$.

3.1. Experiments with selected problems. We begin by focusing on the five problems listed in Table 1 whose Hessian matrices possess five distinct classes of eigenvalue distributions. Liu, Marazzi, and Nocedal [11] describe these eigenvalue distributions and how they evolve as the iterates approach the solution. Other characteristics of the five problems are discussed in Nash and Nocedal [18]. The number of variables in all these test problems is $n = 100$. All the numerical results reported in this paper were performed on a DEC ALPHA2100 workstation with 128 Mb of main memory and using double precision FORTRAN; machine accuracy is approximately 10^{-16}.

The optimization iteration was terminated when

$$||\nabla f(x_k)||_2 \leq 10^{-5} \max\{1, ||x_k||_2\}. \tag{3.6}$$

Table 2
Average number of CG iterations per Newton step for the results of Table 1.

Problem	$m = 0$	$m = 4$	$m = 8$	$m = 16$
Cvar-2	17.0	14.9	12.0	14.9
Penalty-3	5.7	3.4	3.6	3.6
Tridiag	6.9	5.0	4.2	4.2
QOR	4.0	2.5	2.5	2.5
SQRT(2)	17.3	12.2	12.3	10.0

The results are summarized in Table 1 for various values of the memory parameter m in the preconditioner. We report the number of iterations (iter) of the Hessian-free Newton method, the number of function and gradient evaluations (fg) performed during the line search, and the number of CG iterations (cg). Recall that every iteration of the CG method requires one gradient evaluation.

Our main interest in these results lies in the number of CG iterations; the number of function or gradient evaluations in the line search and the number of iterations of the Hessian-free Newton method vary somewhat randomly due to the nonlinearities in the problem and due to the inner termination test (3.4). We observe from Table 1 that a substantial reduction in the number of CG iterations was obtained, in all problems, for $m = 8$.

No further gains were achieved by increasing m to 16 (or beyond). The reason for this is partly explained by Table 2, which reports the average number of CG iterations per Newton iteration. Note that since the preconditioner makes use of the correction pairs generated by the CG method, and since Table 2 shows that the average number of CG iterations is small, increasing the storage beyond 10 corrections will have no effect most of the time. This explains, in particular, why for several problems the results for $m = 8$ and $m = 16$ are identical.

Table 1 suggests that the preconditioner is successful. To quantify its effectiveness in a more controlled setting, we performed the following tests using problems cvar-2 and penalty-3 (similar results are obtained with the other test problems). For each function we selected an intermediate iterate generated by the Hessian-free Newton method and at that point computed the Hessian matrix using finite differences. This iterate was selected so that the Hessians were positive definite at that point. For each of the two problems, we solved the 51 linear systems

$$Ax = b_i, \quad i = 0, \ldots, 50, \tag{3.7}$$

where A denotes the Hessian matrix and where the right-hand-side vectors b_i were randomly generated with components in the interval $[0, 1]$. We solved the first system $Ax = b_0$ using unpreconditioned CG and constructed preconditioners $H(m)$ for various values of m. We then solved the remaining systems $Ax = b_i,\ i = 1, \ldots, 50$, using the preconditioned CG method. In all cases, the starting point was $x_0 = 0$ and the CG iteration was terminated by means of the residual test recommended in [2]:

$$||r_k||_\infty \leq (||A||_\infty ||x_k||_\infty + ||b||_\infty)\mathrm{TOL}. \tag{3.8}$$

In Table 3 we report the results for two values of the parameter TOL.

We observe that for a tight tolerance, $\mathrm{TOL}_1 = 10^{-7}$, the benefit of the preconditioner can be modest, as in the problem cvar-2, but that for the relaxed tolerance, $\mathrm{TOL}_2 = 10^{-3}$, the savings in the number of CG iterations are substantial. These

TABLE 3

Solving systems with a fixed coefficient matrix and multiple right-hand sides. Number of CG iterations for two tolerances, $TOL_1 = 10^{-7}$ and $TOL_2 = 10^{-3}$.

	Cvar-2		Penalty-3	
m	TOL_1	TOL_2	TOL_1	TOL_2
0	61	37	26	12
4	71	7	22	5
8	54	6	15	2
12	52	3	15	2
18	49	3	15	2
20	46	1	15	2

results are typical of what we have observed using other coefficient matrices and right-hand-side vectors. They suggest that the quasi-Newton preconditioner is well suited in settings similar to that of the Hessian-free Newton method, where the stopping test for the CG iteration often demands low accuracy.

In all these tests we have reported only the number of CG iterations, not computing times. This is because our objective in introducing the automatic preconditioner is to reduce the number of gradient evaluations that often render Hessian-free Newton methods impractical. We should mention, however, that the cost of applying the preconditioner, which is $4mn$ floating point operations, may constitute a substantial portion of the optimization process if the evaluation of the gradient is inexpensive. We will return to this point in the next section.

As mentioned in section 2, the preconditioner saves the correction pairs at uniform intervals throughout the CG run. If instead we build the preconditioner by using the last m pairs of the CG iteration, the results described in this section would not be quite as good as, but would overall be similar to, the ones obtained with the uniform sampling technique. In the next section, however, we will report experiments in which saving the last m correction pairs is a significantly inferior strategy.

3.2. Extensive tests. We now test the efficacy of the automatic preconditioner by solving a set of unconstrained problems from the CUTE collection [4]. We will use this experiment to report on one of the many variants of the sampling techniques we have tried. In addition to collecting m correction pairs during the CG cycle using the sampling technique, we will also store the correction pair produced by the outer iteration of the optimization algorithm,

$$s_k = x_{k+1} - x_k, \qquad y_k = \nabla f_{k+1} - \nabla f_k.$$

The results obtained with this strategy are shown in Table 4. Without storing the outer correction pair the results are slightly less successful and will not be reported here.

In these experiments the preconditioner is successful not only in reducing the total number of CG iterations but also in improving the reliability of our optimization method.

We conclude our numerical study in the optimization setting by considering two problems from the MINPACK-2 collection [1]. The preconditioner was the same as the one used to generate the results in Table 4. We now also report CPU time to illustrate the effect of the preconditioner on the Newton iteration. The results are presented in Table 5.

Table 4

Performance of the Hessian-free Newton method on a set of problems from the CUTE *collection. The code -2 indicates that more than* 3000 *CG iterations were performed. The code -3 indicates that the line search routine performed more than* 20 *iterations without decreasing the objective function.*

		$m = 0$			$m = 4$			$m = 8$		
Problem	n	iter	fg	cg	iter	fg	cg	iter	fg	cg
ARWHEAD	1000	6	7	15	6	7	12	6	7	12
BDQRTIC	100	16	17	69	15	16	52	17	18	52
BROYDN7D	1000	114	267	1061	105	273	634	106	273	712
CRAGGLVY	1000	20	20	98	22	22	68	23	23	74
DIXMAANA	1500	6	6	14	6	6	13	6	6	13
DIXMAANE	1500	22	23	266	22	23	198	23	24	186
DIXMAANG	1500	21	21	209	27	29	200	29	31	182
DIXMAANH	1500	21	21	207	26	26	184	24	24	158
DIXMAANI	1500	-2	-2	-2	64	65	2616	63	64	2572
DIXMAANL	1500	-2	-2	-2	227	229	2749	213	215	2586
DQDRTIC	1000	6	6	16	6	6	13	6	6	13
DQRTIC	500	-2	-2	-2	22	24	45	22	24	45
EIGENALS	110	38	39	233	36	39	218	29	32	147
EIGENBLS	110	-2	-2	-2	124	193	1020	135	198	1039
EIGENCLS	462	-2	-2	-2	128	177	1491	121	172	1528
ENGVAL1	1000	11	11	25	9	9	18	9	9	18
FREUROTH	1000	11	17	28	11	14	22	11	14	22
GENROSE	500	-2	-2	-2	366	669	1982	365	646	2000
MOREBV	1000	5	6	70	5	6	67	5	6	68
NONDQUAR	100	56	67	323	62	101	392	56	91	320
PENALTY1	1000	-3	-3	-3	41	46	84	41	46	84
PENALTY3	100	-3	-3	-3	23	31	59	23	31	59
QUARTC	1000	-3	-3	-3	24	27	49	24	27	49
SINQUAD	1000	67	84	248	22	35	58	22	35	58
SROSENBR	1000	9	10	22	9	10	19	9	10	19
TQUARTIC	1000	3	3	8	3	3	7	3	3	7
TRIDIA	1000	46	46	1306	35	35	570	34	34	575

Table 5

Performance of the Hessian-free Newton method on two problems from the MINPACK-2 *collection of problems. CPU time is reported in seconds.*

		$m = 0$				$m = 4$				$m = 8$			
Problem	n	iter	fg	cg	cpu	iter	fg	cg	cpu	iter	fg	cg	cpu
MinSurA	2500	16	19	178	19	15	19	101	14	15	18	103	16
G-L 2D	400	136	149	2948	83	76	93	1716	55	58	65	1308	46

4. Experiments with finite element matrices. Our numerical experiments with nonlinear optimization test problems suggest that the quasi-Newton preconditioner holds much promise. To continue our evaluation of its performance, we would like to test it on matrices that have different eigenvalue distributions from the ones studied so far and that are representative of an important class of applications. To this end we have selected several linear systems arising in the finite element models of Belytschko et al. [3]. The first two matrices used in our experiments, A_{1_0}, A_{1_1}, were obtained from a one-dimensional model consisting of a line of two-node elements with support conditions at both ends and a linearly varying body force. A_{1_0} has dimension $n = 50$ and A_{1_1} has dimension $n = 451$. The right-hand-side vector in these systems, which we denote by c_0, is defined by

$$c_0^1 = c_0^n = 0, \quad c_0^i = i/(n-1) \times 10^2, \quad i = 2, \ldots, n-1, \tag{4.1}$$

TABLE 6

Characteristics of the finite element test problems. 1D: *one-dimensional;* 2D: *two-dimensional.*

Problem	Origin	n	λ_{min}	λ_{max}
$A1_0$	1D	50	1.0	$.20 \times 10^{10}$
$A1_1$	1D	451	1.0	$.18 \times 10^{11}$
$A2_0$	2D	170	1.0	$.13 \times 10^9$
$A2_1$	2D pert	170	1.0	$.13 \times 10^9$
$A2_2$	2D pert	170	1.0	$.14 \times 10^9$
$A2_3$	2D pert	170	1.0	$.14 \times 10^9$
$A2_4$	2D pert	170	1.0	$.14 \times 10^9$
$A2_5$	2D pert	170	1.0	$.15 \times 10^9$

where superscripts indicate components of a vector.

The third matrix used in our tests, $A2_0$, is the stiffness matrix from a two-dimensional finite element model of a cantilever beam. The beam is fixed at one end, and a shear load is applied at the other end. The finite element mesh consists of an even array of elements in the x- and y-coordinates [3]. The right-hand-side vector for this two-dimensional model will be denoted by d_0; it has zeros in all positions except that

$$d_0^{34} = d_0^{68} = d_0^{102} = d_0^{136} = d_0^{170} = -8000. \tag{4.2}$$

We also generated five matrices $A2_1, \ldots, A2_5$ by perturbing the mesh for the cantilever model $A2_0$ along one of the coordinate directions. The size of the perturbation increases linearly with every new matrix in the sequence: it is 1% in $A2_1$ and 10% in $A2_5$.

The characteristics of the matrices are shown in Table 6, where λ_{min} and λ_{max} denote their extreme eigenvalues.

The first matrix, $A1_0$, has one eigenvalue of size $\lambda = 1$ and one of size $\lambda = .2 \times 10^7$; the rest are distributed in a wide gap and cluster near the largest eigenvalue, $\lambda = .2 \times 10^{10}$. The second matrix $A1_1$ has a similar eigenvalue distribution, except that the smallest eigenvalue $\lambda = 1$ has multiplicity 2. The matrix $A2_0$ from the two-dimensional model has a cluster of 10 eigenvalues at $\lambda = 1$; the next eigenvalue is located at $\lambda = .54 \times 10^4$, and the rest form several clusters between $\lambda = 10^7$ and $\lambda = .13 \times 10^9$. We illustrate the eigenvalue distributions of these test matrices in Figure 1.

In the experiments with finite element matrices reported next, the CG iteration was terminated using the residual test (3.8), where the value of the parameter TOL will be specified later on. The preconditioner was constructed using the uniform sampling strategy described in the appendix.

4.1. Multiple right-hand sides.

We first tested the efficiency of the quasi-Newton preconditioner in solving a sequence of problems (1.1), in which the coefficient matrix is constant but the right-hand side varies. To do so, we applied the unpreconditioned CG method to the first system $Ax = b_0$. The information generated during this run was used to construct five quasi-Newton preconditioners $H(m)$ for $m = 4, 8, 12, 16, 20$, as described in section 2. For each preconditioner $H(m)$, we solved the remaining systems $Ax = b_i$, $i = 1, \ldots, 50$, using the preconditioned CG method. The first right-hand-side vector b_0 was defined as c_0 or d_0, depending on

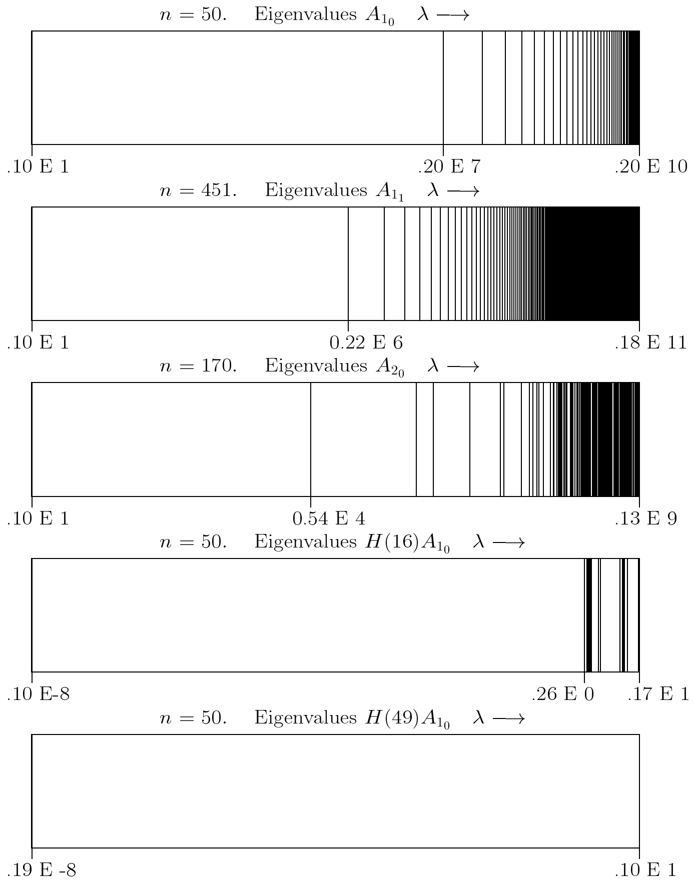


FIG. 1. *Eigenvalue distributions of three finite element matrices and the effect of preconditioning on* A_{1_0}.

whether the matrices correspond to the one- or two-dimensional models (see (4.1)–(4.2)). The other 50 right-hand sides, $b_1, \ldots, b_{50}$, were obtained according to the following recursion, which starts with $j = 0$. Using b_j as a "seed," we obtain b_{j+1} by introducing perturbations of size ±5%, with random signs, to each of the nonzero components of b_j.

The results are presented in Table 7. We report the average number of CG iterations (rounded to the nearest integer) needed to meet the stopping test (3.8) with TOL $= 10^{-7}$. We present results for two initial points, $x_0 = 0$ and $x_0 = 10^2 e$, where $e = (1, 1, \ldots, 1)^T$.

We observe that the preconditioner is successful in reducing the number of CG iterations in both the one-dimensional and two-dimensional models. To illustrate how the preconditioner transforms the spectrum of the coefficient matrix A_{1_0}, we plot in Figure 1 the eigenvalues of $H(m)A_{1_0}$ for $m = 16, 49$, as well as the spectrum of the original matrix A_{1_0}. We note that even though $H(16)A_{1_0}$ is only slightly better conditioned than A_{1_0}, its eigenvalues are more tightly clustered. We also observe that

Table 7

Results for finite element matrices using multiple right-hand sides. The table reports average numbers of CG iterations for 50 *runs, using different values of the memory parameter m, and for two different initial points.*

	A_{1_0}		A_{1_1}		A_{2_0}	
	$x_0 = 0$	$x_0 = 10^2e$	$x_0 = 0$	$x_0 = 10^2e$	$x_0 = 0$	$x_0 = 10^2e$
m	iter	iter	iter	iter	iter	iter
0	49	25	447	225	56	93
4	43	22	291	185	48	68
8	23	12	126	89	26	56
12	16	6	125	80	28	36
16	12	4	62	41	27	30
20	12	5	63	41	21	24

the condition number of $H(49)A_{1_0}$ is $.52\times 10^7$ with just one eigenvalue $\lambda = 0.19\times 10^{-8}$ and a cluster of 49 eigenvalues $\lambda = 1$; this is expected, given the properties of quasi-Newton updating and the fact that A_{1_0} is of dimension 50.

It is natural to ask whether the preconditioner provides a reduction in cpu time—and not just in CG iterations—in these finite element test problems. It turns out that since our test matrices are very sparse, the cost of applying the preconditioner is too high to offset the reduction in the number of CG iterations in these experiments. More specifically, the product Av, which is the most computationally expensive part of the unpreconditioned CG method, requires approximately $3n$ multiplications for the one-dimensional model and $14n$ multiplications for the two-dimensional model. In contrast, the product of the preconditioner $H(m)$ and a vector requires $4mn$ multiplications independently of the matrix structure. As a result, one is not able to obtain reductions in cpu time for any of the values of m listed in Table 7. Nevertheless, these results indicate that for matrices having the same eigenvalue distribution as our test matrices, but with a substantial number of nonzero elements, significant reductions in computing time can be achieved with the quasi-Newton preconditioner. For the rest of the paper we will continue to assume that the cost of computing Av is much higher than the cost of applying the quasi-Newton preconditioner, and we will report only the number of CG iterations.

4.2. Slowly varying systems. Next we consider the family of problems $A_{2_k}x = b_k$ for $k = 0, 1, \ldots, 5$, using the perturbations of the two-dimensional finite element matrix A_{2_0}. We solve the first system $A_{2_0}x = b_0$ using unpreconditioned CG, and we construct five quasi-Newton preconditioners $H(m)$ for $m = 4, 8, 12, 16, 20$. Each of these is used to solve the five remaining systems using the preconditioned CG method. The CG iteration was stopped by (3.8) with TOL $= 10^{-7}$. The first right-hand-side vector is defined to be d_0 (see (4.2)), and the remaining right-hand sides $d_1, \ldots, d_5$ were constructed as before by adding, every time, perturbations of ±5% to the nonzero elements in each of the vectors in the sequence $\{d_j\}$.

In the first set of experiments, reported in Table 8, the same starting point x_0 was used for all the systems $A_{2_k}x = b_k$. We experimented with two choices for this starting point, $x_0 = 0$ and $x_0 = 10^2e$. In the second set of experiments, reported in Table 9, the initial point for solving each system $A_{2_k}x = b_k$ was chosen to be the solution of the previous system, $A_{2_{k-1}}x = b_{k-1}$. Recall that the system A_{2_0} is always solved by unpreconditioned CG.

Table 8 shows that the preconditioner is effective. The fact that the number of iterations increases slightly as we move along a row of the table is not surprising. Since the preconditioner was generated from the first matrix A_{2_0}, and the matrices A_{2_k} differ

Table 8

Results on a sequence of slowly varying linear systems arising from the two-dimensional finite element model. The table presents the number of iterations of the preconditioned CG method needed to solve each of the systems.The starting point for solving all the systems is given by x_0.

	$x_0 = 0$						$x_0 = 10^2 e$					
m	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$
4	56	50	51	51	52	53	93	70	71	72	73	75
8	56	27	28	29	31	31	93	58	59	61	62	63
12	56	20	22	22	26	27	93	37	37	38	39	40
16	56	19	21	22	23	24	93	32	32	32	34	36
20	56	18	19	20	26	28	93	25	26	26	28	30

Table 9

A variation of the results in Table 8. The initial point for solving system $A2_k x = b_k$ is now taken as the solution of the previous system, $A2_{k-1} x = b_{k-1}$. The initial point for the first system $A2_0 x = b_0$ is given by x_0.

	$x_0 = 0$						$x_0 = 10^2 e$					
m	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$
4	56	41	41	46	47	46	93	4	5	4	4	5
8	56	20	24	25	26	26	93	4	4	7	4	5
12	56	17	18	19	19	20	93	5	6	8	9	9
16	56	17	18	15	21	18	93	5	7	8	9	10
20	56	17	18	15	19	17	93	5	5	7	8	8

more and more from it as the subscript k increases, the preconditioner becomes "older" for each new system. Table 9 indicates that using the solution of $A2_{k-1} x = b_{k-1}$ as the initial point for the new system $A2_k x = b_k$ has been advantageous.

We repeated the tests of Tables 8 and 9, refreshing the preconditioner after every solution. To be more precise, during the solution of each system $A2_k x = b_k$ we constructed a preconditioner and used it to solve the next system $A2_{k+1} x = b_{k+1}$ (as in the Hessian-free Newton method). We made an exception to this strategy when the CG method required only one or two iterations to meet the stopping test, since building a preconditioner with $m = 1, 2$ is not useful. In this case we used the preconditioner most recently generated. The results are given in Tables 10 and 11.

The results of Tables 10 and 11 are better than those of Tables 8 and 9, particularly in that there is no longer a trend for the number of CG iterations to increase as we move along a row of the table. Nevertheless the gains are less significant than one would expect. We should note that when the preconditioner is built during an unpreconditioned CG run, the number of CG iterations is larger and the pairs $\{s_k, y_k\}$ represent a better sample than that obtained during a preconditioned CG run. Indeed, if the preconditioner is so effective that the number of CG iterations is very small, then collecting information from this run may not be advantageous, as we mentioned above. Our conclusion is that the decision of when to refresh the preconditioner is not simple, and dynamic strategies that balance the currency of the information with the amount of information available could be quite effective. We will, however, not pursue this question here.

Tables 8–11 indicate that using the previous solution as the starting point for a new run (a "hot start") sometimes, but not always, leads to a substantial reduction of CG iterations. We should also point out that the results for $x_0 = 0$ in Table 11 show that the hot start benefits from preconditioning, as can be seen by reading the results one column at a time. But for $x_0 = 10^2 e$ in Table 9, preconditioning does not help the hot start strategy.

Table 10

A variation of the results given in Table 8*. A new preconditioner is now computed after every solution. The right-hand-side vectors were the vectors b_i. The starting point for solving all the systems is given by x_0.*

	$x_0 = 0$						$x_0 = 10^2 e$					
m	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$
4	56	50	44	51	55	49	93	70	73	67	73	68
8	56	27	34	41	33	33	93	58	49	52	47	51
12	56	20	31	37	24	31	93	37	41	42	43	37
16	56	19	31	21	37	27	93	32	32	32	34	36
20	56	18	26	23	32	22	93	25	43	30	31	32

Table 11

A variation of the results given in Table 9*. A new preconditioner is now computed after every solution. The initial point for solving system $A2_k x = b_k$ is taken as the solution of system $A2_{k-1} x = b_{k-1}$. The initial point for the first system $A2_0 x = b_0$ is given by x_0.*

	$x_0 = 0$						$x_0 = 10^2 e$					
m	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$	$A2_0$	$A2_1$	$A2_2$	$A2_3$	$A2_4$	$A2_5$
4	56	41	38	48	40	59	93	4	3	1	1	2
8	56	20	37	32	34	33	93	4	2	1	2	1
12	56	17	26	32	27	39	93	5	5	5	4	2
16	56	17	24	24	28	26	93	5	4	6	3	2
20	56	17	19	14	19	16	93	5	4	4	4	2

4.3. Comparing sampling strategies.

We will now perform some tests to compare the strategy of saving correction pairs at uniform intervals during the CG run with that of saving the last m pairs. In the first experiment we use the matrix $A1_0$ from the one-dimensional finite element model and solve systems of the form (1.1), where the matrix is fixed and the right-hand sides vary. The initial point was $x_0 = 0$, and the right-hand sides were chosen to have random components in the interval $[0, 1]$. The preconditioner is first constructed using the last m iterations of the CG method. The results are presented in the second and third columns of Table 12 for two values of the tolerance TOL in (3.8). It is remarkable that the preconditioner is extremely effective when TOL $= 10^{-7}$, which is a fairly tight accuracy, but that it gives only modest gains when TOL $= 10^{-9}$. We then modified the right-hand sides by setting their first and last components to zero. The results, which are markedly different, are given in the last two columns of Table 12.

We can explain these results by considering the properties of the matrix $A1_0$, which is given by

$$A1_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & a & -a/2 & 0 & \cdots & 0 & 0 \\ 0 & -a/2 & a & -a/2 & \cdots & 0 & 0 \\ 0 & 0 & -a/2 & a & \cdots & 0 & 0 \\ & & & & \vdots & & \\ 0 & 0 & 0 & 0 & \cdots & -a/2 & a \end{bmatrix},$$

where $a = 10^{-9}$. Since the first row is e_1^T, the first component of the solution x equals the first component b^1 of the right-hand-side vector. It is not difficult to show that since all the entries in b are not greater than 1, all other components of x are of order

Table 12

Constructing the preconditioner using the last m correction pairs. Number of CG iterations for two types of right-hand-side vectors b and for two levels of accuracy TOL.

	b random		b random, $b^1 = b^n = 0$	
m	TOL $= 10^{-7}$	TOL $= 10^{-9}$	TOL $= 10^{-7}$	TOL $= 10^{-9}$
0	50	50	48	49
4	7	39	45	46
8	6	34	41	42
12	6	29	37	38
16	6	27	34	34
20	5	25	29	30

Table 13

Constructing the preconditioner sampling m correction pairs. Number of CG iterations for two types of right-hand-side vectors b and for two levels of accuracy TOL.

	b random		b random, $b^1 = b^n = 0$	
m	TOL $= 10^{-7}$	TOL $= 10^{-9}$	TOL $= 10^{-7}$	TOL $= 10^{-9}$
0	50	50	48	49
4	10	34	36	44
8	25	35	37	33
12	17	22	19	24
16	12	18	14	19
20	15	18	13	16

10^{-6}. Therefore, for these random right-hand-side vectors we can expect the solutions to be closely aligned with the first coordinate direction e_1. Since the preconditioner is able to incorporate the curvature along e_1, it forces the CG iteration to immediately point towards the solution. As a result the CG iteration will terminate quickly if the required accuracy is not too high. These are the most favorable conditions for the automatic preconditioner. But if the tolerance is set to be TOL $= 10^{-9}$, the components of the solution along the other coordinate directions will need to be estimated well, and the limited memory preconditioner is only able to provide some of the needed information.

The solution will no longer be closely aligned with e_1 if the first component of the right-hand-side vector is set to zero. One can show that in this case the solution will have significant components along *all* the coordinate directions, except for the first component, which is zero. The problem thus becomes particularly difficult for limited memory preconditioning. This is confirmed by the last two columns of Table 12, which show very modest gains in performance. Note also that the performance is now insensitive to the stopping tolerance.

In Table 13 we repeat the tests reported in Table 12 but using a uniform sampling strategy. The latter clearly performs better than saving the last m pairs, except for the first case (b random TOL $= 10^{-7}$), which, as we have explained, represents a special case.

To continue our comparison of sampling strategies, we repeat in Table 14 the experiments of Table 7 with the two-dimensional finite element matrix A_{2_0}, using two different starting points. We compare the strategy of saving the last m pairs ("last") with that of uniform sampling. It is clear that the latter performs much better in this experiment.

Our computational experience, both in the optimization setting and in finite element calculations, is that saving the last m corrections usually gives comparable

Table 14

Average number of iterations of the CG method for the matrix A_{2_0} and multiple right-hand sides. Comparison of two sampling strategies in the formation of the preconditioner: saving the last m iterations and sampling at uniform intervals. Results for two starting points are given.

	$x_0 = 0$		$x_0 = 10^2 e$	
m	Last	Uniform	Last	Uniform
0	56	56	93	93
4	74	48	90	68
8	71	26	87	56
12	56	28	83	36
16	44	27	77	30
20	43	21	72	24

Table 15

Results for test matrix A_{1_0} using multiple right-hand sides. The table reports the number of iterations to achieve convergence for 50 runs, using different values of the memory parameter m and of the CG iteration limit `maxCG`. *Initial point $x^{(0)} = 0$.*

`maxCG`	$m = 4$	$m = 8$	$m = 16$
10	42	41	41
20	38	33	32
30	31	26	22
40	31	22	16
50	43	24	13

performance to the uniform sampling technique. But as we have just shown, there are cases when uniform sampling is superior. It is difficult to provide theoretical arguments in favor of either strategy, but we now report the results of controlled tests that further support the uniform sampling technique.

4.4. On the sample size. When the number of correction pairs available to form the preconditioner is small, the two strategies (uniform sampling and using the last m corrections) will clearly give similar results. Therefore, in the following tests we will force the CG algorithm to perform an increasingly large number of iterations and observe the effect that this has on the quality of the preconditioner.

More specifically we study whether the preconditioner benefits from having a larger sample of corrections to choose from for a given amount of memory m. In the tests described next, we will consider the solution of a sequence of finite element systems with multiple right-hand sides. We will fix the value of m, apply the unpreconditioned CG for a fixed number `maxCG` of CG iterations to the first system in the sequence, and build the preconditioner using the sampling technique. We then solve the rest of the linear systems using this preconditioner, terminating the CG iteration by means of (3.8). To study the benefit of a larger sample size, we repeat this test for various values of `maxCG`.

The results are given in Tables 15–17. Note that, for a given value of m, the preconditioners differ in that they use an increasingly wide sample of CG iterations. We observe that if the amount of memory is small ($m = 4$) the quality of the preconditioner appears to be independent of the sample size `maxCG`. But for larger values of m the sample size has a beneficial effect.

5. Final remarks. We have presented a quasi-Newton preconditioner for accelerating the conjugate gradient method when this is applied to a sequence of linear systems with positive definite coefficient matrices. Our numerical experiments indi-

Table 16
The experiment reported in Table 15 using the test matrix A_{1_1}.

maxCG	$m = 4$	$m = 8$	$m = 16$
250	245	209	198
300	254	192	159
350	246	160	113
400	275	114	69
450	287	125	60

Table 17
The experiment reported in Table 15 using the test matrix A_{2_0}.

maxCG	$m = 4$	$m = 8$	$m = 16$
20	57	52	50
30	49	43	41
40	32	25	22
50	48	26	19
60	48	26	19

cate that the preconditioner may be useful when the coefficient matrices A are not very sparse or when A is not explicitly available and products of A times vectors are expensive to compute. The motivation for this work arose from the desire to accelerate the CG iteration used in a Hessian-free Newton method for nonlinear optimization, and in that context the new preconditioner appears to provide substantial savings. Our experiments with finite element models suggest that the preconditioner may prove to be useful in other areas of application, but more research is required to establish this firmly.

We have experimented with several other strategies for selecting the correction pairs. One idea that deserves to be mentioned is to use the m pairs with the smallest Rayleigh quotient,

$$\frac{s_i^T y_i}{\|s_i\|^2}.$$

Even though this strategy has not proved to be more successful in our tests than the other selection schemes described in the paper, it may be effective in some areas of application.

6. Appendix. We now present a formal description of the sampling algorithm (mentioned in section 2) that collects the pairs $\{s_k, y_k\}$ as uniformly as possible, with the restriction that at most m pairs be stored at any stage. We denote the set of correction pairs that have been stored as $\mathcal{P}$. We will assume that m is an even number since this simplifies the algorithm and is not restrictive in practice.

The sampling algorithm runs parallel to the CG method. Once a pair $\{s_k, y_k\}$ has been computed by the CG method, the sampling algorithm examines the iteration index k and decides if the pair should be included in $\mathcal{P}$. When a new pair is accepted, the algorithm checks the available space, and if the number of pairs in $\mathcal{P}$ is m, then a pair is chosen to leave $\mathcal{P}$. The algorithm is started by inserting into $\mathcal{P}$ the first m pairs generated by the CG process. After this, the entering and leaving pairs are chosen to keep an *almost* uniform distribution at any time.

```
Algorithm SAMPLE.
Choose an even number $m$; set $k \leftarrow 0$ and $cycle \leftarrow 1$.
REPEAT:
  Starting
   while $k < m$,
       • get  $\{s_k, y_k\}$
       • Add $\{s_k, y_k\}$ to $\mathcal{P}$
       • $k \leftarrow k + 1$
   end while
  Deletion/Insertion.
   if  $k$ can be expressed as $k = (\frac{m}{2} + l - 1)2^{cycle}$  for an integer $l$ of the form
      $l = 1, 2, \ldots, \frac{m}{2}$,  then
       • Store $l$
       • Compute the subscript of the leaving pair as $k' = (2l - 1)2^{cycle-1}$
       • Delete $\{s_{k'}, y_{k'}\}$ from $\mathcal{P}$
       • Add $\{s_k, y_k\}$ to $\mathcal{P}$
       • if  $l = \frac{m}{2}$,  then set $cycle \leftarrow cycle + 1$
   end if
   $k \leftarrow k + 1$
END REPEAT
```

Note that the first pair ($k = 0$) generated in the CG iteration always remains in $\mathcal{P}$. This has no particular significance, and it is easy to change the algorithm so that this is not the case.

We now discuss some properties of the sampling algorithm. After the initialization, in which the first m pairs are stored, the algorithm performs deletion and insertion operations controlled by the variable *cycle*. For a given value of *cycle*, the algorithm stores $\frac{m}{2}$ new pairs spaced by a distance of 2^{cycle} and deletes the same number of pairs. Deletion takes place in such a way that the space created between two consecutive pairs is 2^{cycle}. Therefore when *cycle* attains a new value, the distribution ceases to be uniform and there is a transition period during which a new uniform distribution is generated; this is achieved at the end of the second loop. It follows that the larger m is, the longer it will take to move from one uniform distribution to the next.

Acknowledgements. We would like to thank Ted Belytschko and Mark Flemming for providing a Matlab code that generates the finite element models used in section 4, Eric Schwabe for a very useful discussion on the algorithm appearing in the appendix, and Richard Waltz for running the tests reported in Table 5. We would also like to thank a referee for suggesting the experiments reported in Tables 15–17.

REFERENCES

[1] B. Averick, R.G. Carter, and J.J. Moré, *The MINPACK-2 Test Problem Collection*, Preprint MCS-P153-0692, Argonne National Laboratory, Argonne, IL, 1991.

[2] R. Barrett, M. W. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994.

[3] T. Belytschko, A. Bayliss, C. Brinson, S. Carr, W. Kath, S. Krishnaswamy, B. Moran, J. Nocedal, and M. Peshkin, *Mechanics in the engineering first curriculum at Northwestern University*, Int. J. Engrg. Education, 13 (1997), pp. 457–472.

[4] I. Bongartz, A.R. Conn, N.I.M. Gould, and Ph.L. Toint, *CUTE: Constrained and unconstrained testing environment*, ACM Trans. Math. Software, 21 (1995), pp. 123–160.

[5] R.H. Byrd, P. Lu, J. Nocedal, and C. Y. Zhu, *A limited memory algorithm for bound constrained optimization*, SIAM J. Sci. Comput., 16 (1995), pp. 1190–1208.

[6] R.H. Byrd, J. Nocedal, and C. Zhu, *Towards a discrete Newton method with memory for large scale optimization*, in Nonlinear Optimization and Applications, G. Di Pillo and F. Giannessi, eds., Plenum, New York, 1996, pp. 1–12.

[7] J.E. Dennis, Jr., and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.

[8] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley, New York, 1987.

[9] J.C. Gilbert and C. Lemaréchal, *Some numerical experiments with variable storage quasi-Newton algorithms*, Math. Programming, 45 (1989), pp. 407–436.

[10] P.E. Gill, W. Murray, and M.H. Wright, *Practical Optimization*, Academic Press, London, 1981.

[11] G. Liu, M. Marazzi, and J. Nocedal, *Incorporating Eigenvalue Information in Limited Memory Methods for Unconstrained Optimization*, manuscript.

[12] D.C. Liu and J. Nocedal, *On the limited memory BFGS method for large scale optimization*, Math. Programming, 45 (1989), pp. 503–528.

[13] H. Matthies and G. Strang, *The solution of nonlinear finite element equations*, Int. J. Numer. Methods Engrg., 14 (1979), pp. 1613–1626.

[14] J.J. Moré and D.J. Thuente, *Line search algorithms with guaranteed sufficient decrease*, ACM Trans. Math. Software, 20 (1994), pp. 286–307.

[15] S.G. Nash, *Newton-type minimization via the Lánczos method*, SIAM J. Numer. Anal., 21 (1984), pp. 770–778.

[16] S.G. Nash, *User's Guide for TN/TNBC: FORTRAN Routines for Nonlinear Optimization*, Report 397, Mathematical Sciences Dept., The Johns Hopkins University, Baltimore, 1984.

[17] S.G. Nash, *Preconditioning of truncated-Newton methods*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 599–616.

[18] S.G. Nash and J. Nocedal, *A numerical study of the limited memory BFGS method and the truncated-Newton method for large scale optimization*, SIAM J. Optim., 1 (1991), pp. 358–372.

[19] J. Nocedal, *Updating quasi-Newton matrices with limited storage*, Math. Comput., 35 (1980), pp. 773–782.

[20] D.P. O'Leary, *A discrete Newton algorithm for minimizing a function of many variables*, Math. Programming, 23 (1982), pp. 20–33.

[21] D.P. O'Leary and A. Yeremin, *The linear algebra of block quasi-Newton algorithms*, Linear Algebra Appl., 212/213 (1994), pp. 153–168.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1097–1115

LAGRANGIAN DUALITY AND RELATED MULTIPLIER METHODS FOR VARIATIONAL INEQUALITY PROBLEMS*

ALFRED AUSLENDER† AND MARC TEBOULLE‡

Abstract. We consider a new class of multiplier interior point methods for solving variational inequality problems with maximal monotone operators and explicit convex constraint inequalities. Developing a simple Lagrangian duality scheme which is combined with the recent logarithmic-quadratic proximal (LQP) theory introduced by the authors, we derive three algorithms for solving the variational inequality (VI) problem. This provides a natural extension of the methods of multipliers used in convex optimization and leads to smooth interior point multiplier algorithms. We prove primal, dual, and primal-dual convergence under very mild assumptions, eliminating all the usual assumptions used until now in the literature for related algorithms. Applications to complementarity problems are also discussed.

Key words. Lagrangian duality, variational inequalities, interior proximal methods, complementarity problems, global convergence, Lagrangian multiplier methods

AMS subject classifications. 49A29, 49A55, 90C33, 65K10

PII. S1052623499352656

1. Introduction.

Given an operator T, point to set in general, and a closed convex subset C of $\mathbb{R}^n$, the variational inequality (VI) problem consists of finding a pair $x^* \in C$ and $g^* \in T(x^*)$ such that

$$\langle x - x^*, g^* \rangle \geq 0 \quad \forall x \in C, \tag{1.1}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual inner product of $\mathbb{R}^n$. Our analysis will focus on the case where T is a maximal monotone mapping from $\mathbb{R}^n$ into itself (see section 2 for definitions and properties) and the constraints set C is explicitly defined by

$$C := \{x \in \mathbb{R}^n : \quad F(x) \leq 0\}, \tag{1.2}$$

where $F(x) := (f_1(x), \dots, f_m(x))^T$, with $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, $i = 1, \dots, m$, given proper closed convex functions. The analysis developed in this paper can also handle additional affine equality constraints, but for simplicity of exposition this will not be discussed here.

It is well known (cf. section 2) that the VI problem (1.1) can be alternatively formulated as finding the zero of an appropriately defined maximal monotone operator Π, namely, find x^* such that $0 \in \Pi(x^*)$. One method to find the zero of a maximal monotone operator Π is the proximal point algorithm; see, e.g., [22], [29], [19]. It generates a sequence $\{x^k\}$ via the iteration

*Received by the editors February 22, 1999; accepted for publication (in revised form) December 1, 1999; published electronically June 6, 2000.
http://www.siam.org/journals/siopt/10-4/35265.html

†Laboratoire d'Econometrie de L'Ecole Polytechnique, 1 Rue Descartes, Paris 75005, France (auslen@poly.polytechnique.fr).

‡School of Mathematical Sciences, Tel-Aviv University, Ramat-Aviv 69978, Israel (teboulle@math.tau.ac.il). This author was partially supported by the Israeli Ministry of Science under grant 9636-1-96.

$$x^0 \in \mathbb{R}^n, \ \ 0 \in \Pi(x^k) + c_k^{-1}(x^k - x^{k-1}), \tag{1.3}$$

where $c_k \geq c > 0$. Recently, several works have concentrated on the generalization of the proximal algorithm based on "entropic proximal terms," and they have led to *interior proximal* point methods for variational inequality problems; see, e.g., [6], [14], [15], [16], [33]. Roughly speaking, the motivation for using these generalized proximal methods is that they allow for elimination of the constraints in a natural way within the use of an appropriate proximal-like term, and this allows for developing easier to implement and more efficient algorithms than the one obtained within the classical quadratic proximal framework. In all previously mentioned works, convergence of the resulting interior proximal methods was proved under restrictive assumptions either on the problem's data (e.g., requiring pseudomonotonicity and paramonotonicity of the operator) or on the entropic proximal term and/or on both. Very recently we proposed in [7] a new type of proximal interior method which was proven to be globally convergent to a solution of VI with linear constraints, under the *sole* assumption that the set of solutions of VI is nonempty. This method is based on a *logarithmic-quadratic proximal* (LQP) term which enjoys several useful properties and is not in the class of proximal terms used in the works previously alluded to.

In all the above mentioned works the corresponding proximal methods were developed to solve the primal formulation of the variational inequality problem (1.1). The purpose of this work is twofold. First we develop general new theoretical results for VI problems which are needed in our analysis but are also of independent interest. In particular, a new general tool based on recession analysis for maximal monotone operators is derived to prove existence results and a complete and transparent Lagrangian duality scheme for VI problems is developed under minimal hypothesis; see sections 2 and 3. Building on these theoretical results, we analyze proximal interior methods based on the LQP term, when applied to the *dual* and *primal-dual* formulations of VI. Rockafellar [29], [30] was among the first to realize that the classical quadratic proximal framework can be usefully applied to the primal-dual formulation of VI problems to generate multiplier type algorithms. Further results in that direction were developed by Gabay [18]. More recently, Eckstein and Ferris [17] have suggested using a Bregman type proximal term to produce new types of multiplier methods for the special case of monotone complementarity problems with box constraints. A key feature of these methods, which differs from the ones derived with the classical quadratic proximal scheme, is that one obtains algorithms with smooth Lagrangians (given enough smoothness in the original problems data), where the main iterative step can be solved via Newton type algorithms. Thus, the proposed methods in [17] are akin to similar smooth Lagrangian methods developed for solving convex programming problems; see, e.g., [11], [10], [16], [20], [21], [25], [26], [32]. In this paper we consider the more general VI problem given in (1.1) but within the framework of the recent LQP theory developed by the authors to appropriate dual formulations of VI. This leads to new methods of multipliers (also called augmented Lagrangians) for solving VI problems. We also emphasize that the use of the LQP theory allows us to derive in fact C^∞ Lagrangians, as opposed to the ones obtained in [17] via the Bregman proximal theory and multiplier methods with stronger convergence results under very mild assumptions. Thus, the current paper may be viewed as a natural continuation and extension of our recent works [7], [8]. The former work deals with the LQP method for solving the primal formulation (1.1) but with C being a polyhedral set, i.e., when F is an affine map, and the latter deals with a more general class of proximal terms and related algorithms, including as a special case the LQP method for solving convex optimization problems.

To construct our algorithms, we first need to develop an appropriate duality framework for VIs with convex constraints. We will thus introduce a Lagrangian duality scheme for VIs, which differs from (and for our purposes, is more appropriate than) the well-known duality framework of Mosco [24], which has also been used and improved in [18], [17]. This duality theory takes its origin in [3], [9] and is developed in section 3, after some background material and a new recession formula for maximal monotone operators given in section 2. We show that one can construct a dual and primal-dual formulation of the VI problem via appropriate operators which are shown to be maximal monotone under very mild assumptions. Using the duality framework of section 3 and combining it with the LQP theory, we produce in section 4 two new methods of multipliers with interior multipliers updates, based on the dual and primal-dual formulations of VI. Convergence of these methods is established under mild assumptions on the problem's data. In the course of our analysis, we also complement some convergence results of the LQP method as given in [7], when applied to the primal formulation (1.1) when C is a polyhedral set. We give some concluding remarks in section 5.

2. Preliminaries on maximal monotone operators.

We give in this section some important facts and results on maximal monotone operators which will be needed in our analysis. For more details on monotone operators we refer the reader to the recent monograph [31, Chapter 12].

A point to set valued map (or multifunction) $A : \mathbb{R}^n \overset{\rightarrow}{\rightarrow} \mathbb{R}^n$ is an operator which associates with each point $x \in \mathbb{R}^n$ a set (possibly empty) $A(x) \subseteq \mathbb{R}^n$. The inverse of any operator always exists and is denoted by $A^{-1}(y) := \{x \in \mathbb{R}^n | y \in A(x)\}$, and obviously we have $(A^{-1})^{-1} = A$. The domain and range of A are defined by

$$\begin{aligned} \mathrm{dom}A &:= \{x | A(x) \neq \emptyset\}, \\ \mathrm{rge}A &:= \{y | \exists x : y \in A(x)\} = \mathrm{dom}A^{-1}. \end{aligned}$$

When A is single valued (a function) we shall write $A(x) = \{y\}$ or simply $A(x) = y$.

An operator A is said to be monotone if

$$\langle x' - x, y' - y\rangle \geq 0 \quad \forall y' \in A(x'),\ \forall y \in A(x),\ \forall x, x' \in \mathrm{dom}A.$$

A monotone operator is said to be maximal if its graph is not properly contained in the graph of any other monotone operator, in other words, if

$$\langle x - x', y - y'\rangle \geq 0 \quad \forall x' \in \mathrm{dom}A\ \forall y' \in A(x') \Longrightarrow y \in A(x).$$

The normal cone operator associated with a closed convex set C is defined by

$$N_C(x) = \begin{cases} \{y : \langle y, v - x\rangle \leq 0 \quad \forall v \in C\} & \text{if} \quad x \in C, \\ \emptyset & \text{otherwise}. \end{cases}$$

Clearly, $\mathrm{dom}N_C = C$ and we always have $N_C(x) = \{0\}$ when $C \equiv \mathbb{R}^n$ or when $x \in \mathrm{int}C$, the interior of C. It is well known [27] that the normal cone operator N_C is a maximal monotone operator on $\mathbb{R}^n$; in fact $N_C = \partial\delta(\cdot|C)$, where $\delta(\cdot|C)$ is a closed proper convex function defined by $\delta(x|C) = 0$ if $x \in C$ and $+\infty$ otherwise, and ∂h denotes the subdifferential of a proper closed convex function h.

PROPOSITION 2.1.

(i) A^{-1} *is maximal monotone if and only if* A *is maximal monotone.*

(ii) *Let A_i, $i = 1,2$ be maximal monotone. Then $A_1 + A_2$ is also maximal monotone under either one of the following conditions:*

(a) $\mathrm{intdom}A_1 \cap \mathrm{dom}A_2 \neq \emptyset$,

(b) *$ri(\mathrm{dom}A_1) \cap ri(\mathrm{dom}A_2) \neq \emptyset$, where ri stands for relative interior.*

(iii) *Given any maximal monotone operator $A : \mathbb{R}^n \overset{\rightarrow}{\rightarrow} \mathbb{R}^n$, the solution set $A^{-1}(0)$ of the generalized equation $0 \in A(x)$ is closed and convex. Moreover, $A^{-1}(0)$ is nonempty and bounded if and only if* $0 \in \mathrm{int}(\mathrm{dom}A^{-1}) = \mathrm{int}(\mathrm{rge}A)$.

(iv) *Let $A : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}^n \times \mathbb{R}^d$ be maximal monotone. Fix $x \in \mathbb{R}^n$ and define $B : \mathbb{R}^d \to \mathbb{R}^d$ by*

$$B(z) := \{w | \exists v : (v, w) \in A(x, z)\}.$$

If x is such that there exists $y \in \mathbb{R}^d$ with $(x, y) \in ri(\mathrm{dom}A)$, then B is also maximal monotone.

Proof. See Chapter 12 of [31]. □

For a nonempty closed convex set C in $\mathbb{R}^n$, we denote by C_∞ the recession cone of C. For a closed and proper convex function $h : \mathbb{R}^N \to \mathbb{R} \cup +\{\infty\}$, the recession function h_∞ of h is defined by

$$\mathrm{epi}\ (h_\infty) = (\mathrm{epi}h)_\infty, \text{ where } \mathrm{epi}h = \{(x, r) \in \mathbb{R}^N \times \mathbb{R} : h(x) \leq r\},$$

and the following useful formula holds:

$$h_\infty(d) = \sup\{\langle c, d\rangle | c \in \mathrm{dom}h^*\} = \sigma_{\mathrm{dom}h^*}(d), \tag{2.1}$$

where σ_S denotes the support functional of a set S and h^* stands for the conjugate function of h; see [27]. Following [1], the recession function of a multivalued map A on $\mathbb{R}^n$ is defined by

$$f_\infty^A(d) := \sup\{\langle c, d\rangle | c \in \mathrm{rge}A\} = \sigma_{\mathrm{rge}A}(d). \tag{2.2}$$

The recession function of a multivalued map is particularly useful to establish existence of solutions for variational problems. Indeed, recall that by Minty's theorem [23], the interior of the range of a maximal monotone operator A is convex. Then using Proposition 2.1(iii) and [27, Theorem 13.1] one has that for any maximal monotone operator A on $\mathbb{R}^n$, the solution set $A^{-1}(0)$ is nonempty and bounded (hence compact) if and only if

$$\forall d \neq 0, \quad f_\infty^A(d) > 0. \tag{2.3}$$

The next proposition will be of particular importance in our analysis and extends a recent result derived in [4, Proposition 2.1].

PROPOSITION 2.2. *Let $h : \mathbb{R}^N \to \mathbb{R} \cup \{+\infty\}$ be a closed proper convex function and let A be monotone with* $\mathrm{dom}h \subset \mathrm{dom}A$ *and such that $A + \partial h$ is maximal monotone. Then*

$$f_\infty^{A+\partial h}(d) = \sup\{\langle c, d\rangle | c \in A(x),\ x \in \mathrm{dom}h\} + h_\infty(d). \tag{2.4}$$

Proof. (i) Define $\hat{A}(x) := A(x)$ if $x \in \mathrm{dom}h$ and $\emptyset$ otherwise. Then, since $\mathrm{dom}\hat{A} = \mathrm{dom}h$ and $\mathrm{dom}\partial h \subset \mathrm{dom}h$, we obtain

$$\mathrm{dom}(A + \partial h) = \mathrm{dom}(\hat{A} + \partial h) = \mathrm{dom}\partial h, \ \ \mathrm{rge}(A + \partial h) = \mathrm{rge}(\hat{A} + \partial h),$$

and hence,

$$f_\infty^{A+\partial h}(d) = f_\infty^{\hat{A}+\partial h}(d) \quad \forall d. \tag{2.5}$$

Now, since $\hat{A} + \partial h$ is also maximal monotone, then invoking [13, Theorem 4, and its variant 3, p. 176], one has

$$\overline{\mathrm{rge}(\hat{A} + \partial h)} = \overline{\mathrm{conv}(\mathrm{rge}\hat{A}) + \mathrm{dom} h^*},$$

where the upper bar denotes the closure operation and conv stands for convex hull. Using the above relation in definitions (2.2) and (2.5) we then have

$$\begin{aligned} f_\infty^{A+\partial h}(d) &= \sup\{\langle c, d\rangle | c \in \overline{\mathrm{rge}(\hat{A} + \partial h)}\} \\ &= \sup\{\langle c, d\rangle | c \in \overline{\mathrm{conv}(\mathrm{rge}\hat{A}) + \mathrm{dom} h^*}\} \\ &= \sup\{\langle c, d\rangle | c \in \mathrm{conv}(\mathrm{rge}\hat{A}) + \mathrm{dom} h^*\} \\ &= \sup\{\langle u, d\rangle | u \in \mathrm{conv}(\mathrm{rge}\hat{A})\} + \sup\{\langle v, d\rangle | v \in \mathrm{dom} h^*\} \\ &= \sup\{\langle u, d\rangle | u \in A(x),\ x \in \mathrm{dom} h\} + h_\infty(d), \end{aligned}$$

where in the third and the last equality we use the fact that for any nonempty set S the support function $\sigma_S = \sigma_{\overline{\mathrm{conv} S}} = \sigma_{\mathrm{conv} S}$, and for the second term of the last equality we use (2.1). □

An important application of the above result is to the special case when $\partial h := \partial\delta(\cdot|C) = N_C$, with $\mathrm{dom} h = C \subset \mathrm{dom} A$, as recently derived in [4]. Recalling that $\delta_\infty(\cdot|C) = \delta(\cdot|C_\infty)$, using Proposition 2.2 together with (2.3) implies that the solution set $(A + N_C)^{-1}(0)$ is nonempty and compact if and only if

$$f_\infty^{A,C}(d) > 0 \quad \forall d \neq 0, \tag{2.6}$$

with

$$f_\infty^{A,C}(d) := \sup\{\langle c, d\rangle | c \in A(x), x \in C\} \ \text{ if } d \in C_\infty, \ \ +\infty \text{ otherwise.} \tag{2.7}$$

In the rest of this paper maximal monotone operators will play a prominent role and will be used as abstract tools to reformulate equivalent formulations of VI problems.

In terms of N_C, we can rewrite the VI problem (1.1) as the one of finding the zero of the generalized equation

$$\text{(PVI)} \quad 0 \in T(x) + N_C(x). \tag{2.8}$$

Problem (PVI) will thus be considered as another equivalent *primal* formulation of the VI problem given in (1.1). The set of solutions of (PVI), namely $(T + N_C)^{-1}(0)$, will be denoted by X.

3. Duality for variational inequalities. We begin by recalling the classical and well-known duality framework for VIs as suggested by Mosco [24]. We then give a simple Lagrangian duality scheme for the VI problem which appears more appropriate for algorithmic purposes. Thus, this part of the section can be considered as a short summary of known results that do not seem to have been explicitly outlined in the literature. In the last part of this section, we then give conditions under which the dual and primal-dual operators remain maximal monotone.

3.1. Duality via Mosco's scheme. Let $c : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a closed proper convex function, A a maximal monotone multivalued map on $\mathbb{R}^n$, and consider the following general variational problem:

$$0 \in A(x) + \partial c(x). \tag{3.1}$$

Note that our variational problem (PVI) is a special case of (3.1) with the choice $A = T, c = \delta(\cdot|C)$. In [24], Mosco studied problems of the form (3.1) and shows that one can always associate a dual problem with (3.1), defined by

$$0 \in -A^{-1}(-y) + (\partial c)^{-1}(y). \tag{3.2}$$

Now, since $(\partial c)^{-1} = \partial c^*$ (see [27]), then the dual (3.2) can be equivalently rewritten as

$$0 \in -A^{-1}(-y) + \partial c^*(y). \tag{3.3}$$

It was shown in [24] that x solves (3.1) if and only if $y \in -A(x)$ solves (3.2) or (3.3).

For a more general dual framework involving the sum of two general operators, we refer the reader to the recent work [2] and also to [17].

The dual terminology for the pair of problems (3.1)–(3.2) is justified by the fact that the above scheme is akin to the Fenchel duality scheme used in convex optimization problems. Indeed, consider the special case where $A = \partial b$, the subdifferential map of some closed convex function $b : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$. Then, under appropriate regularity assumptions [27], the relations (3.1)–(3.2) are nothing else but the optimality conditions for the Fenchel primal-dual pair of convex optimization problems

$$\min\{b(x) + c(x) : x \in \mathbb{R}^n\}, \qquad \min\{b^*(-y) + c^*(y) : y \in \mathbb{R}^n\}.$$

Applying the above scheme to our VI problem (PVI), i.e., with $A = T, c(x) = \delta(x|C)$ and using the facts that $\partial c^*(y) = \partial \delta^*(y|C) = N_C^{-1}(y)$, a dual problem associated with (PVI) is then

$$0 \in -T^{-1}(-y) + N_C^{-1}(y). \tag{3.4}$$

The main difficulty with the above framework is that it requires constructing the inverse operators T^{-1} and N_C^{-1} to formulate a dual problem, a task which can be very difficult. However, we note that for the special case of box constraints, i.e., when $C = \{x \in \mathbb{R}^n : \ l \le x \le u\}$, an explicit computation of N_C^{-1} is available, as recently shown in [17].

We thus consider now another duality scheme for (PVI), which even though is in fact formally equivalent to Mosco's scheme, will be more appropriate for our algorithmic purposes.

3.2. Lagrangian duality for VI. The duality scheme given here is in the spirit of the classical Lagrangian duality framework for constrained optimization problems. It will permit us to take advantage of the particular structure of the set C described by convex inequalities and to develop explicit algorithms.

The starting point is the simple and well-known observation that x^* is a solution of VI if and only if

$$x^* \in \operatorname{argmin}\{\langle g^*, x - x^* \rangle : x \in C\}, \tag{3.5}$$

where $g^* \in T(x^*)$ and $C = \{x : f_i(x) \leq 0,\ i = 1, \ldots, m\}$.

In the rest of this paper we assume that each $f_i : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is a closed proper convex function and that $E := \cap_{i=1}^m \text{dom} f_i$ is an open set. Note that this assumption is needed to properly handle convex programs within the formalism of extended valued functions; see [27, p. 273].

Formally, we can thus associate with the *convex* optimization problem

$$\min\{\langle g^*, x - x^*\rangle : f_i(x) \leq 0,\ i = 1, \ldots, m\}, \tag{3.6}$$

a Lagrangian defined by $L : \mathbb{R}^n \times \mathbb{R}^m \to \overline{\mathbb{R}}$

$$L(x, u; x^*) := \begin{cases} \langle g*, x - x^*\rangle + \sum_{i=1}^m u_i f_i(x) & \text{if } x \in E, u \in \mathbb{R}^m_+, \\ -\infty & \text{if } u \notin \mathbb{R}^m_+, \\ +\infty & \text{otherwise,} \end{cases} \tag{3.7}$$

where $u \in \mathbb{R}^m_+$ is the dual multiplier attached to the constraints, and a dual problem defined by

$$\sup_{u \geq 0} \inf\{L(x, u, x^*) : x \in E\}. \tag{3.8}$$

By the standard saddle point optimality theorem [27] we know that $(x^*, u^*) \in E \times \mathbb{R}^m_+$ is a saddle point of L if and only if $x^* \in E$ and $u^* \geq 0$ are, respectively, optimal for the primal and dual problems (3.6)–(3.8) with no duality gap, that is, with equal optimal values. Note that here $\partial(\langle g^*, x - x^*\rangle + \delta(x|E))|_{x=x^*} = g^* + N_E(x^*)$ and since E is open $N_E(x^*) = \{0\}$. Translating this to our pair of problems (3.6)–(3.8) gives

$$0 \in g^* + \sum_{i=1}^m u_i^* \partial f_i(x^*), \tag{3.9}$$

$$0 \in -F(x^*) + N_{\mathbb{R}^m_+}(u^*), \tag{3.10}$$

with $g^* \in T(x^*)$ (recalling that the primal optimal value of (3.6) is zero). We also know (cf. [27, Corollary 28.2.1]) that under Slater's condition

$$\exists z \in \mathbb{R}^n :\ f_i(z) < 0, i = 1, \ldots, m$$

there exists a Kuhn–Tucker vector u. The relations (3.9)–(3.10) are just the KKT optimality conditions for (3.6) which are necessary and sufficient for optimality, and thus u^* can be interpreted as the solution of a *Lagrangian dual* VI which can be defined as follows. For each $u \in \mathbb{R}^m_+$, set

$$M(u) := \left\{ x \in \mathbb{R}^n : 0 \in T(x) + \sum_{i=1}^m u_i \partial f_i(x) \right\}, \tag{3.11}$$

$$G(u) := \{-F(x) : x \in M(u)\}, \tag{3.12}$$

$$T_D(u) := G(u) + N_{\mathbb{R}^m_+}(u). \tag{3.13}$$

The dual VI problem associated with VI is then

$$\text{(DVI)} \quad \text{find } u^* \in \mathbb{R}^m_+,\ d^* \in G(u^*) :\ \langle d^*, u - u^*\rangle \geq 0 \quad \forall u \in \mathbb{R}^m_+,$$

which can also be written using (3.13) as

$$\text{(DVI)} \quad 0 \in T_D(u^*). \tag{3.14}$$

Likewise, we can then associate a primal-dual formulation of VI via (3.9)–(3.10):

$$\text{(PDVI)} \qquad (0,0) \in S(x^*, u^*), \tag{3.15}$$

where the operator S is defined on $\mathbb{R}^n \times \mathbb{R}^m_+$ by

$$S(x,u) := \left\{ (y,w) \in \mathbb{R}^n \times \mathbb{R}^m_+ \,| y \in T(x) + \sum_{i=1}^m u_i \partial f_i(x), w \in -F(x) + N_{\mathbb{R}^m_+}(u) \right\}, \tag{3.16}$$

if $(x,u) \in \text{dom} S = (\text{dom} T \cap (\cap_{i=1}^m \text{dom} f_i)) \times \mathbb{R}^m_+ \neq \emptyset$, and $\emptyset$ otherwise.

From the above discussion we have thus shown that we have essentially three equivalent representations for the VI problem (1.1) or (2.8). More precisely we have proved the following.

Theorem 3.1. *Suppose that Slater's condition holds for the constraint set C. Then, $x^* \in \mathbb{R}^n$ solves* (PVI) *if and only if there exists $u^* \in \mathbb{R}^m_+$ such that (x^*, u^*) solves* (PDVI).

To be able to apply the proximal theory to the three formulations of VI, we need to guarantee that the corresponding operators are maximal monotone. The rest of this section is thus devoted to establishing conditions in terms of the problem's data, under which maximal monotonicity is preserved.

3.3. Maximal monotonicity. For convenience we will often use the following notations:

$$\begin{aligned} T_P &:= T + N_C, \\ T_D &:= G + N_{\mathbb{R}^m_+}, \\ T_S &:= S, \end{aligned}$$

and we denote by X, U, Z the set of solutions of (PVI), (DVI), and (PDVI), respectively. The primal operator poses no problems. From Proposition 2.1(ii)(a) the maximal monotonicity of T_P is preserved under the condition dom T $\cap$ int$C \neq \emptyset$.

We now turn to the dual operator T_D. First, we show the easy part, namely, that T_D is monotone on $\mathbb{R}^m_+$.

Proposition 3.1. *Let $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be monotone. Then, the dual operator $T_D = G + N_{\mathbb{R}^m_+}$ is monotone on $\mathbb{R}^m_+$.*

Proof. Since $N_{\mathbb{R}^m_+}$ is monotone, the monotonicity of T_D will follow by proving that G is monotone. Let $(u,u'), (v,v')$ be arbitrary points in G. By definition of G given in (3.12), $\exists (x,u), (y,v)$ such that

$$u' = -F(x), x \in M(u); \;\; v' = -F(y), y \in M(v), u, v \geq 0.$$

Since f_i are convex using the subgradient inequality for each f_i, and since $u, v \geq 0$ and $F(x) = (f_1(x), \dots, f_m(x))^T$, one easily obtains

$$\begin{aligned} \langle u - v, u' - v' \rangle &= \langle u - v, F(y) - F(x) \rangle \\ &\geq \langle y - x, \sum_{i=1}^m u_i \partial f_i(x) - \sum_{i=1}^m v_i \partial f_i(y) \rangle \\ &= \langle y - x, y' - x' \rangle, \; x' \in T(x), y' \in T(y) \\ &\geq 0, \end{aligned}$$

where the third equality follows by using $x \in M(u), y \in M(v)$ with $M(\cdot)$ defined in (3.11), and the last inequality is from the monotonicity of T. □

To establish the maximal monotonicity of T_D we first establish that T_S is maximal monotone. This result is in fact known for the case of T single valued continuous and f_i given finite convex and differentiable functions; see [30]. We extend this to our more general framework, and for completeness we include a proof of this slight extension.

PROPOSITION 3.2. *Let $T : \mathbb{R}^n \overset{\rightarrow}{\rightarrow} \mathbb{R}^n$ be maximal monotone such that $domT \cap (\cap_{i=1}^m domf_i) \neq \emptyset$. Then the primal-dual operator $T_S = S$ defined in* (3.16) *is maximal monotone.*

Proof. The operator T_S defined in (3.16) can be decomposed as follows. Let

$$A(x,u) = \begin{cases} T(x) \times \{0\} & \text{if } x \in \text{dom}T, \\ \emptyset & \text{otherwise} \end{cases}$$

and

$$B(x,u) = \begin{cases} \{\{\sum_{i=1}^m u_i \partial f_i(x)\} \times \{-F(x) + N_{\mathbb{R}_+^m}(u)\} & \text{if } x \in \cap_{i=1}^m \text{dom} f_i, u \in \mathbb{R}_+^m, \\ \emptyset & \text{otherwise.} \end{cases}$$

Then we have $S = A + B$. Since T is maximal monotone, it is easy to see that A is also maximal monotone. On the other hand, defining $l : \cap_{i=1}^m \text{dom} f_i \times \mathbb{R}_+^m \to \mathbb{R}$ by $l(x,u) = \sum_{i=1}^m u_i f_i(x)$, then we have $B(x,u) = \partial_x l(x,u) \times -\hat{\partial}_u l(x,u)$, which is maximal monotone [27, Corollary 34.2.2 and Corollary 37.5.2], where $\hat{\partial}$ denotes the upper subdifferential. Invoking Proposition 2.1(ii)(a) on the operators A, B then gives the desired result. □

We also need the following result establishing the boundedness of the dual solution set.

PROPOSITION 3.3. *Let $T : \mathbb{R}^n \overset{\rightarrow}{\rightarrow} \mathbb{R}^n$ be maximal monotone. Suppose that $X \neq \emptyset$, and there exists $z \in domT$ satisfying Slater's condition. Then, the solution set U of* (DVI) *is nonempty. In addition, if the solution set of X of* VI *is bounded, then the solution set U of* (DVI) *is also bounded.*

Proof. Under the given assumptions, from Theorem 3.1 we have $x^* \in X$ and $\exists u^* \geq 0$ such that (3.9)–(3.10) hold, which by (3.11) means that $x^* \in M(u^*)$. As a consequence, we have using (3.10)

$$\langle G(u^*), u - u^* \rangle = \langle -F(x^*), u - u^* \rangle \geq 0 \quad \forall u \geq 0,$$

and hence $u^* \in U$. Now, if U is not bounded, then there would exist a sequence $\{u^k, x^k, g^k, g_i^k, i = 1, \ldots, m\}$ with $u^k \geq 0, x^k \in X, g^k \in T(x^k), g_i^k \in \partial f_i(x^k), i = 1, \ldots, m$ such that

$$||u^k|| \to \infty, \quad u^k ||u^k||^{-1} \to \bar{u} \neq 0, \quad x^k \to \bar{x},$$

and (3.9)–(3.10) hold at (x^k, u^k), i.e.,

$$0 \in g^k + \sum_{i=1}^m u_i^k g_i^k, \quad u_i^k \geq 0, \quad u_i^k f_i(x^k) = 0, \quad f_i(x^k) \leq 0, \ i = 1, \ldots, m.$$

Using the subgradient inequality for the convex function f_i, multiplying by $u_i^k \geq 0$, and summing we obtain

$$\sum_{i=1}^m u_i^k f_i(z) \geq \langle z - x^k, \sum_{i=1}^m u_i^k g_i^k \rangle \quad \forall z \in \text{dom}T.$$

But since $x^k \in M(u^k)$, then the above inequality reduces to

$$\sum_{i=1}^m u_i^k f_i(z) \geq \langle z - x^k, -g^k \rangle \quad \forall z \in \mathrm{dom} T$$

$$\geq \langle x^k - z, g \rangle,\ g \in T(z), z \in \mathrm{dom} T \text{ since } T \text{ is monotone.} \tag{3.17}$$

Since we assumed that X is bounded, dividing the last inequality by $||u^k||$ and passing to the limit we obtain $\forall$ z: $\sum_{i=1}^m \bar{u}_i f_i(z) \geq 0$, and hence with z satisfying Slater's condition for C, and recalling that $\bar{u} \neq 0$, this is impossible. □

We can now establish the desired result.

Proposition 3.4. *Suppose that X is nonempty and bounded, that there exists $z \in$ dom$T$ satisfying Slater's condition, and that T_S is maximal monotone (see Proposition 3.2 for conditions). Then the dual operator T_D is maximal monotone.*

Proof. From Proposition 2.1(i), we have T_D maximal monotone if and only if T_D^{-1} is maximal monotone. Using the definition of T_D, (3.11), (3.12), and (3.16) we have

$$\begin{aligned} T_D^{-1}(w) &= \{u | w \in G(u) + N_{\mathbb{R}^m_+}(u)\} \\ &= \{u | \exists x : w \in -F(x) + N_{\mathbb{R}^m_+}(u), x \in M(u)\} \\ &= \{u | \exists x : (x, u) \in S^{-1}(0, w)\}. \end{aligned}$$

Since we assumed that $T_S := S$ is maximal monotone, invoking Proposition 2.1(iv), we thus have that T_D^{-1} is maximal monotone if $(0,0) \in \mathrm{ridom} S^{-1}$. By Proposition 2.1(iii), the latter condition will be satisfied if the solution set Z of (PDVI), i.e., $S^{-1}(0,0)$, is nonempty and bounded. But by Theorem 3.1 we have $S^{-1}(0,0) \subset X \times U$ and since we assumed that X is bounded, then by Proposition 3.3 U is also bounded and the result is proved. □

Remark 3.1. Note that a similar result was recently proved in [4], but only under the restrictive assumption that T is strongly monotone as well as some other technical assumptions on f_i.

We end this section with a result needed in the convergence analysis and which is also of independent interest. In the context of optimization problems, this result corresponds to asymptotic KKT conditions guaranteeing that a sequence is minimizing.

Recall that when C is the convex set defined in (1.2), then the recession cone of C is given by

$$C_\infty = \{d : \ (f_i)_\infty(d) \leq 0, \qquad i = 1, \ldots, m\}.$$

Proposition 3.5. *Let $T : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ be maximal monotone such that $T_P = T + N_C$ is maximal monotone. Suppose that X is nonempty and compact and that $C \subset$ dom$T$. Let u^k be a bounded sequence in $\mathbb{R}^m_+$ and consider a sequence $\{x^k, g^k, g_i^k, i = 1, \ldots, m\}$ with $g^k \in T(x^k)$, $g_i^k \in \partial f_i(x^k), i = 1, \ldots, m$ such that*

$$\varepsilon^k := g^k + \sum_i^m u_i^k g_i^k \to 0, \tag{3.18}$$

$$\limsup_{k \to \infty} f_i(x^k) \leq 0 \quad \forall i, \tag{3.19}$$

$$u_i^k f_i(x^k) \to 0 \quad \forall i. \tag{3.20}$$

Then the sequence $\{x^k\}$ is bounded, and each limit point of the sequence $\{x^k\}$ solves VI.

Proof. The proof is by contradiction. Suppose that the sequence $\{x^k\}$ is not bounded, then without loss of generality, one can assume that

$$\| x^k \| \to +\infty, \qquad \frac{x^k}{\| x^k \|} \to \bar{x} \neq 0.$$

Let $\varepsilon > 0$; then from (3.19) for k sufficiently large we have

$$f_i \left(\frac{x^k}{||x^k||} ||x^k|| \right) ||x^k||^{-1} \leq \varepsilon ||x^k||^{-1}. \tag{3.21}$$

Since by [12] for any function f we have

$$f_\infty(d) = \inf \left\{ \lim \inf_{n \to +\infty} f(t_n x_n)/t_n \ \mid \ t_n \to +\infty, x_n \to d \right\},$$

passing to the limit in (3.21), we then obtain that $(f_i)_\infty(\bar{x}) \leq 0 \ \forall i = 1, \dots, m$, which means that $\bar{x} \in C_\infty$. Now, $\forall g \in T(x), \ x \in C$, using arguments similar to the one used in Proposition 3.3, we obtain using the definition of ε_k given in (3.18)

$$\begin{aligned} \langle g, x^k \rangle &\leq \langle g, x \rangle + \langle g^k, x^k - x \rangle \\ &\leq \langle g, x \rangle + \langle \varepsilon^k, x^k - x \rangle - \left\langle \sum_{i=1}^m u_i^k f_i(x^k), x - x^k \right\rangle. \end{aligned}$$

Dividing the latter inequality by $||x^k||$, passing to the limit, and using (3.18)–(3.20) we thus obtain $\langle g, \bar{x} \rangle \leq 0$, i.e., from (2.7) that $f_\infty^{T,C}(\bar{x}) \leq 0$, which contradicts the assumption that X is compact.

Now let x^∞ be a limit point of $\{x^k\}$. Then using $\varepsilon_k = g^k + \sum_{i=1}^m u_i^k g_i^k$, the convexity of f_i and the monotonicity of T we obtain

$$\langle g_1, x - x^k \rangle \geq \sum_{i=1}^m u_i^k f_i(x^k) + \langle \varepsilon_k, x - x^k \rangle \quad \forall x \in C, \ \forall g_1 \in T(x).$$

Then, passing to the limit in the above inequality, together with (3.18)–(3.20) we thus get

$$\langle x - x^\infty, g_1 \rangle \geq 0 \quad \forall g_1 \in T(x).$$

But since $x^\infty \in C$, we also have $\langle x - x^\infty, g_2 \rangle \geq 0 \ \forall g_2 \in N_C(x), x \in C$. Therefore it follows that

$$\langle g, x - x^\infty \rangle \geq 0 \ \ \forall g \in T_P(x) = T(x) + N_C(x), \ \ \forall x \in C.$$

Since T_P is maximal monotone, this implies that $0 \in T_P(x^\infty)$, i.e., $x^\infty \in X$. □

4. Interior proximal and multiplier methods for VI. The three formulations VI via the primal, dual, and primal-dual operators reduce to the problem of finding the zero of a specific maximal monotone operator in each case. We will develop below the corresponding algorithms which are based on the LQP method recently introduced by the authors in [7]. We thus begin by recalling some results from [7] on the log-quadratic function and the corresponding interior proximal algorithm which was developed for solving the primal version of VIs over polyhedral sets.

4.1. The LQP algorithm. Let $\nu > \mu > 0$ be given fixed parameters, and define

$$\varphi(t) = \begin{cases} \frac{\nu}{2}(t-1)^2 + \mu(t - \log t - 1) & \text{if } t > 0, \\ +\infty & \text{otherwise.} \end{cases} \tag{4.1}$$

Associated with φ we define for any $v \in \mathbb{R}^p_{++}$

$$d(u,v) = \begin{cases} \sum_{i=1}^p \frac{\nu}{2}(u_i - v_i)^2 + \mu(v_i^2 \log \frac{v_i}{u_i} + u_i v_i - v_i^2) & \text{if } u \in \mathbb{R}^p_{++}, \\ +\infty & \text{otherwise.} \end{cases} \tag{4.2}$$

The functional d defined in (4.2) can be rewritten as

$$d(u,v) = \sum_{j=1}^p v_j^2 \varphi(u_j v_j^{-1}) \quad \forall u, v > 0. \tag{4.3}$$

Second-order homogeneous functionals of the form (4.3) with other choices of the kernel φ and their role in the development of interior multiplier methods for solving convex problems have been recently studied in [8]. For simplicity of exposition, in this paper we consider only the important special case (4.2). Likewise our analysis is presented only with exact versions of the algorithms. Our results, however, remain valid for many other choices of the kernel φ with d given in (4.3) and within approximate algorithms as done in [7], [8]. The next theorem recalls some important properties of φ and d; see [7], [8] for proofs.

THEOREM 4.1. *Let φ be given in* (4.1) *and d as defined in* (4.2)*. Then*

(i) *φ is a differentiable strongly convex function on $\mathbb{R}_{++}$ with modulus $\nu > 0$.*

(ii) $\lim_{t\to 0} \varphi'(t) = -\infty$.

(iii) *For any $u, v > 0$ we have $d(u,v) \geq \mu||u - v||^2$ and $d(u,v) = 0$ if and only if $u = v$.*

(iv) *The conjugate of φ is given by*

$$\varphi^*(s) = \frac{\nu}{2}t^2(s) + \mu \log t(s) - \frac{\nu}{2}, \tag{4.4}$$

$$t(s) := (2\nu)^{-1}\{(\nu - \mu) + s + \sqrt{((\nu-\mu)+s)^2 + 4\mu\nu}\} = (\varphi^*)'(s). \tag{4.5}$$

(v) dom$\varphi^* = \mathbb{R}$, *and $\varphi^* \in C^\infty(\mathbb{R})$.*

(vi)$(\varphi^*)'(s) = (\varphi')^{-1}(s)$ *is Lipschitz $\forall\ s \in \mathbb{R}$, with constant ν^{-1}.*

(vii) *φ^* is strictly convex and increasing on $\mathbb{R}$.*

(viii) $(\varphi^*)''(s) < \nu^{-1} \ \ \forall s \in \mathbb{R}$.

(ix) $(\varphi^*)_\infty(-1) = 0$ *and* $(\varphi^*)_\infty(1) = +\infty$ *where $(\varphi^*)_\infty$ is the recession function of φ^*.*

Let P be a polyhedral set on $\mathbb{R}^m$ defined by $P := \{y \in \mathbb{R}^n : Ay \leq b\}$, where A is a (p,n) matrix, $b \in \mathbb{R}^p$, $p \geq n$. We suppose that the matrix A is of maximal rank, i.e., rank$A = n$, and that the interior of P, int$P := \{y : Ay < b\}$ is nonempty. Let T be a maximal monotone set valued map such that dom$T \cap$ int$P \neq \emptyset$.

We consider the *linearly constrained* VI problem which we denote by LVI(T,P):

$$\text{Find a point } y^* \in P \text{ and } g^* \in T(y^*) \text{ satisfying } \langle g^*, y - y^* \rangle \geq 0 \quad \forall y \in P,$$

which includes as a special case the nonlinear complementarity problem by choosing $A = -I_p$ the identity matrix of order $p = n$, and $b = 0$, i.e., $P \equiv \mathbb{R}^n_+$.

Let a_i denote the rows of the matrix A, and define the following quantities:

$$\begin{array}{lcl} l_i(y) & = & b_i - \langle a_i, y\rangle,\ i = 1, \ldots, p, \\ l(y) & = & (l_1(y), l_2(y), \ldots, l_p(y))^T, \\ D(y,z) & = & d(L(y), L(z)). \end{array}$$

For each $y \in \text{int}P$, $z \in \text{int}P$, we have

$$\nabla_y D(y,z) = -\sum_{i=1}^{p} a_i l_i(z)\varphi'(l_i(y)/l_i(z)), \tag{4.6}$$

with $\varphi'(t) = \nu(t-1) + \mu(1 - t^{-1}),\ t > 0$.

To solve LVI(T,P) we consider the following method.

The LQP method.

Start with $y^0 \in \text{int}P$ and generate the sequence $\{y^k\} \subset \text{int}P$, satisfying

$$g^k + \lambda_k^{-1}\nabla_y D(y^k, y^{k-1}) = 0 \qquad \text{with } g^k \in T(y^k), \tag{4.7}$$

where $\lambda_k \geq \lambda > 0$.

We now state the main result proven in [7].

THEOREM 4.2. *Let T be a maximal monotone operator on $\mathbb{R}^n$ such that $domT \cap intP \neq \emptyset$. Then,*

(i)*Existence. For each $\lambda_k > 0$, $y^{k-1} \in intP$, there exists a unique $y^k \in intP$ satisfying* (4.7)*.*

(ii) *Convergence. If the set of solutions of* LVI(T,P)*, denoted by Y is nonempty, then the sequence $\{y^k\}$ generated by* LQP *converges to a solution $y^* \in Y$.*

We emphasize, that to the best of our knowledge, the LQP method for solving LVI is the first interior proximal method for which existence and global convergence of the sequence $\{y^k\}$ to a solution of LVI(T,P) can be established under these very mild assumptions. This is in sharp contrast with other interior proximal methods recently studied in the literature which require severe restrictions such as pseudomonotonicity and paramonotonicity of the operator T as well as some further restrictions on the proximal distance functional d; see, e.g., [14], [16], and references therein.

4.2. The primal log-quadratic method. We first complement here some of the results derived in [7] to solve LVI(T,P), for the case of polyhedral constraints, and in particular when $P = \mathbb{R}^p_+$, namely, to solve the complementarity problem. More precisely, the result below shows that LQP also solves a corresponding dual variational inequality problem (DVI) associated with the linearly constrained VI problem LVI(T,P), much like in the spirit of penalty/barrier methods, see, e.g., [4], except that here we obtain a convergence result for any fixed penalty parameter $\lambda_k \geq \lambda > 0$, in contrast with the penalty/barrier methods, where the penalty parameters must be driven to infinity to obtain convergence (compare with [4]).

Define

$$v_i^k := -l_i(y^{k-1})\varphi'(l_i(y^k)/l_i(y^{k-1})),\ \ i = 1, \ldots, p. \tag{4.8}$$

The iteration (4.7) can thus be simply rewritten using (4.6) as

$$A^T v^k = -\lambda_k g^k \qquad \text{with } g^k \in T(y^k). \tag{4.9}$$

THEOREM 4.3. *Suppose that the solution set of* LVI(T,P) *is nonempty and that T is maximal monotone with* $\text{dom}T \cap \text{int}P \neq \emptyset$*. Let $\{y^k\}$ be the sequence generated by* LQP *and set $P := \mathbb{R}^p_+$. Then*

(i) *the sequence* $\{y^k\}$ *converges to a solution of* y^* *of* LVI(T,P). *Furthermore, we have*

$$\lim \underset{k\to\infty}{inf}\, g_i^k \geq 0, \;\; \lim_{k\to\infty} y_i^k v_i^k = 0, \; i = 1,\dots,p. \tag{4.10}$$

(ii) *In addition, suppose that* $P \subset$ intdom$T$, *or that* $P \subset$ dom$T$ *with* T *single valued and continuous, and* $\lambda_k \leq \bar{\lambda}$. *Then the sequence* $\{\lambda_k^{-1}v^k\}$ *with* v^k *defined in* (4.8) *is bounded, and each limit point of the sequence* $\lambda_k^{-1}v^k$ *is a solution of the corresponding* DVI *problem.*

Proof. (i) The convergence of $\{y^k\}$ to a solution $y^* \geq 0$ of LVI(T,$\mathbb{R}^p_+$) is from Theorem 4.2. When $P = \mathbb{R}^p_+$, (4.8)–(4.9) reduces to: for each $i = 1,\dots,p$,

$$v_i^k = -y_i^{k-1}\varphi'(y_i^k/y_i^{k-1}), \tag{4.11}$$

$$v_i^k = \lambda_k g_i^k. \tag{4.12}$$

Since $\varphi'(t) = \nu(t-1) + \mu(1-t^{-1})$, and $t^{-1} - 1 \geq 1 - t \;\forall t > 0$, we obtain using (4.11)–(4.12)

$$g_i^k \geq \lambda_k^{-1}(\nu+\mu)(y_i^{k-1} - y_i^k). \tag{4.13}$$

Since $\{\lambda_k^{-1}\}$ is bounded and $\{y^k\}$ converges, it follows that $\liminf_{k\to\infty} g_i^k \geq 0 \;\forall i = 1,\dots,p$, proving the first relation in (4.10). Furthermore, we also have

$$\begin{aligned} y_i^k v_i^k &= -y_i^k y_i^{k-1}\varphi'(y_i^k/y_i^{k-1}), \\ &= (\nu y_i^k + \mu y_i^{k-1})(y_i^{k-1} - y_i^k), \end{aligned}$$

and hence $\lim_{k\to\infty} y_i^k v_i^k = 0, \; i = 1,\dots,p$.

(ii) Since y^k converges to $y^* \geq 0$ and $\mathbb{R}^p_+ \subset$ intdom T, then T is locally bounded at y^* and since $\lambda \leq \lambda_k \leq \bar{\lambda}$, it follows from (4.12) that the sequence $\{\lambda_k^{-1}v^k\}$ is bounded. The same conclusion holds under the other proposed hypothesis, i.e., when $\mathbb{R}^p_+ \subset$ dom T with T single valued and continuous. Let $\bar{v}$ be a limit point of $\{\lambda_k^{-1}v^k\}$; then passing to the limit in (4.12) together with (4.10), we have obtained

$$y^* \geq 0, \; \bar{v} \geq 0, \; \langle y^*, \bar{v}\rangle = 0, \; \bar{v} \in T(y^*). \qquad \square$$

This primal method developed for LVI cannot apparently be extended for solving our original VI problem, namely when C is described via convex inequalities. However, using the duality framework of section 3, we provide below two methods based on LQP leading to multiplier and proximal multiplier methods for solving VI given in (1.1) and which exhibit stronger convergence properties for the resulting primal-dual sequences. We begin with the dual method.

4.3. The multiplier dual method. In the rest of this subsection we make the following standing assumptions on the problems data of VI.

Assumption A. (a) T is a maximal monotone operator with $\cap_{i=1}^m$domf_i an open subset of intdom$T$.

Assumption B. (a) The solution set of VI is nonempty and compact.

(b) Slater's condition holds for $z \in$ dom$T$.

Given φ as defined in (4.1), $\lambda > 0$, we define for $u > 0$ the multifunction

$$H(x,u,\lambda) := \begin{cases} T(x) + \sum_{i=1}^m u_i(\varphi^*)'(\lambda f_i(x)/u_i)\partial f_i(x) & \text{if } x \in \cap_{i=1}^m \mathrm{dom} f_i, \\ \emptyset & \text{otherwise.} \end{cases}$$

(4.14)

To solve VI, namely the generalized equation $0 \in T(x) + N_C(x)$, we propose the following method of multipliers.

Multipliers dual method (MDM). Given φ defined in (4.1), $u^0 \in \mathbb{R}^m_{++}$ and $\lambda_k \geq \lambda > 0$, $\forall k \geq 1$, generate the sequences $\{x^k, u^k\}$ according to

$$0 \in H(x^k, u^{k-1}, \lambda_k), \tag{4.15}$$

$$u_i^k = u_i^{k-1}(\varphi^*)'(\lambda_k f_i(x^k)/u_i^{k-1}),\ i = 1,\dots,m. \tag{4.16}$$

In order to have that MDM is well defined, we have to prove that the generalized equation (4.15) has a solution.

PROPOSITION 4.1. *Let φ be given in* (4.1)*, and suppose that Assumptions* A *and* B *hold. Then $\forall \lambda > 0$, $\forall u \in \mathbb{R}^m_{++}$:*

(i) *The operator $H(\cdot, u, \lambda)$ is maximal monotone on $\mathbb{R}^n$.*

(ii) *The solution set of $0 \in H(x,u,\lambda)$, namely $H^{-1}(0,u,\lambda)$, is nonempty and compact.*

Proof. Fix $\lambda > 0, u > 0$, and define $g(x) := \lambda^{-1}\sum_{i=1}^m u_i\varphi^*(\lambda f_i(x)/u_i)$. From Theorem 4.1(ix) we have $(\varphi^*)_\infty(-1) = 0, (\varphi^*)_\infty(1) = +\infty$. We can thus apply [5, Proposition 2.1], to conclude that g is a closed proper convex function with $\mathrm{dom} g = \cap_{i=1}^m \mathrm{dom} f_i \neq \emptyset$. Furthermore, it holds that

$$g_\infty(d) = \begin{cases} 0 & \text{if } (f_i)_\infty(d) \leq 0 \quad \forall i, \\ +\infty & \text{otherwise.} \end{cases} \tag{4.17}$$

Now, since $\cap_{i=1}^m \mathrm{dom} f_i$ is open, using subdifferential calculus one can verify that $H = T + \partial g$ and then by Assumption A, from Proposition 2.1(ii)(a) it follows that H is maximal monotone. Furthermore, since $\mathrm{dom} g \subset \mathrm{dom} T$ we can apply Proposition (2.2) to obtain

$$f_\infty^H(d) = \sup\{\langle c, d\rangle | c \in T(x),\ x \in \mathrm{dom} g\} + g_\infty(d).$$

To show that the solution set $H^{-1}(0,u,\lambda)$ is nonempty and compact, it suffices to show that (cf. (2.3)) $f_\infty^H(d) > 0$, for $d \neq 0$, i.e., using (4.17) it suffices to show that

$$\sup\{\langle c, d\rangle | c \in T(x),\ x \in \mathrm{dom} g\} > 0 \ \text{ when } \ (f_i)_\infty(d) \leq 0 \quad \forall i.$$

But, since $T + N_C$ is maximal monotone and we assumed that the solution set of VI is nonempty and compact and $C \subset \cap_{i=1}^m \mathrm{dom} f_i \subset \mathrm{dom} T$, we also have using (2.7)

$$\beta := \sup\{\langle c, d\rangle | c \in T(x), x \in C\} > 0,$$

and hence since $C \subset \mathrm{dom} g$,

$$\sup\{\langle c, d\rangle | c \in T(x),\ x \in \mathrm{dom} g\} \geq \beta,$$

and the proof is completed. □

In what follows, it will be convenient to use the following notation:

$$\Phi'(a,b) := (a_1\varphi'(b_1/a_1), \dots, a_m\varphi'(b_m/a_m))^T \quad \forall a, b \in \mathbb{R}^m_{++}.$$

We are now in a position to give our convergence result for the MDM given in (4.15)–(4.16).

THEOREM 4.4. *Let $\{x^k, u^k\}$ be the sequence generated by MDM and suppose that Assumptions* A *and* B *hold. Then, the dual sequence $\{u^k\}$ globally converges to a solution $u^* \in U$ of* (DVI)*, while the primal sequence $\{x^k\}$ is bounded and all its limit points are in the solution set X of* VI.

Proof. First, we show that the sequence $\{x^k, u^k\}$ generated by MDM is nothing else but the sequence produced by the LQP method (with $P = \mathbb{R}^m_+$) when applied to solve the DVI problem: $0 \in G(u) + N_{\mathbb{R}^m_+}(u)$. Indeed, from (4.15)–(4.16) we have

$$0 \in T(x^k) + \sum_{i=1}^{m} u_i^k \partial f_i(x^k), \tag{4.18}$$

$$F(x^k) = \lambda_k^{-1} \Phi'(u^{k-1}, u^k), \tag{4.19}$$

where (4.19) follows from using the relation (vi) of Theorem 4.1. The first inclusion (4.18) is equivalent to $x^k \in M(u^k)$. Since by definition $G(u^k) = \{-F(x^k) | x^k \in M(u^k)\}$, from (4.16) we have (using Theorem 4.1(vii)) that $u^k \in \mathbb{R}^m_{++}$, so that $N_{\mathbb{R}^m_+}(u^k) = \{0\}$, and it follows that

$$\gamma^k + \lambda_k^{-1} \Phi'(u^k, u^{k-1}) = 0, \ \gamma^k \in G(u^k). \tag{4.20}$$

As a consequence of Theorem 4.2(ii) applied with $P = \mathbb{R}^m_+$, we thus have that the sequence $\{u^k\}$ globally converges to a solution of DVI, and using Theorem 4.3(i), with $y^k := u^k, \ v^k := u^k, \ g^k = \gamma^k = -F(x^k)$, it follows from (4.10) that

$$\limsup_{k \to \infty} F(x^k) \le 0 \quad \text{and} \quad \lim_{k \to \infty} \langle u^k, F(x^k) \rangle = 0.$$

Therefore, invoking Proposition 3.5, we obtain that the sequence $\{x^k\}$ is bounded and all its limit points are solutions of VI. □

Remark 4.1. When $T = \partial f_0$, with f_0 closed proper convex, we recover the interior proximal method of multipliers and convergence results for solving convex programs as recently derived in [8].

4.4. The primal-dual method.

Our third method consists of solving VI via the equivalent primal-dual formulation (PDVI), namely to solve

$$0 \in S(x, u), \tag{4.21}$$

where S is defined in (3.16). In this subsection, we assume that S is maximal monotone (see Proposition 3.2 for the conditions which guarantee maximal monotonicity) and Slater's condition holds for $z \in \text{dom}T$.

To solve (4.21), we need to consider an extension of the LQP method, where we have now unrestricted variables $x \in \mathbb{R}^n$. For this purpose, following Auslender–Teboulle–Ben-Tiba [7], we consider the distance like functional

$$D((x, u), (y, w)) := \frac{1}{2} ||x - y||^2 + d(u, w),$$

where d is as defined in (4.2) (since here the polyhedral set P $= \mathbb{R}^m_+$).

Then the main iteration of LQP becomes the following:
Start with $(x^0, u^0) \in \mathbb{R}^n \times \mathbb{R}^m_{++}$ and generate $\{(x^k, u^k)\} \subset \mathbb{R}^n \times \mathbb{R}^m_{++}$ satisfying

$$0 \in S(x^k, u^k) + \lambda_k^{-1} \nabla_{(x,u)} D((x^k, u^k), (x^{k-1}, u^{k-1})), \tag{4.22}$$

where $\lambda_k \geq \lambda > 0$.

We can then easily extend Theorem 4.2 (we omit the proof which is very similar to the one given in [7]) to obtain the following theorem.

THEOREM 4.5. *Let S be the maximal monotone operator defined in* (3.16). *Then*

(i) *there exists a unique pair $(x^k, u^k) \in \mathbb{R}^n \times \mathbb{R}^m_{++}$ satisfying* (4.22) $\forall \lambda_k > 0,\ u^{k-1} > 0$.

(ii) *If the solution set of PDVI is nonempty, then the sequence $\{x^k, u^k\}$ generated by* (4.22) *converges to a solution $(x^*, u^*) \in X \times U$.*

Writing explicitly the iteration (4.22) we obtain

$$0 \in T(x^k) + \sum_{i=1}^{m} u_i^k \partial f_i(x^k) + \frac{x^k - x^{k-1}}{\lambda_k}, \tag{4.23}$$

$$0 \in F(x^k) + \lambda_k \Phi'(u^k, u^{k-1}) + N_{\mathbb{R}^m_+}(u^k), \tag{4.24}$$

which in turn is equivalent (following the same arguments as given in the first part of the proof of Theorem 4.4) to the following method.

The proximal-multiplier dual method (PMDM). Given φ defined in (4.1), $(x^0, u^0) \in \mathbb{R}^n \times \mathbb{R}^m_{++}$ and $\lambda_k \geq \lambda > 0\ \forall k \geq 1$, generate the sequences $\{x^k, u^k\}$ according to

$$0 \in H(x, u^{k-1}, \lambda_k) + \lambda_k^{-1}(x^k - x^{k-1}), \tag{4.25}$$

$$u_i^k = u_i^{k-1} (\varphi^*)'(\lambda_k f_i(x^k)/u_i^{k-1}),\ i = 1, \ldots, m. \tag{4.26}$$

This leads to an algorithm for which the new "multiplier multifunction" is now *strongly monotone*. Indeed, for fixed $u^{k-1}, \lambda_k > 0$, define

$$H_k(x) := H(x, u^{k-1}, \lambda_k) + \lambda_k^{-1}(x - x^{k-1}).$$

PROPOSITION 4.2. *Let φ be given in* (4.1). *Suppose that T is a maximal monotone map and Slater's condition is satisfied for some $z \in \mathrm{dom} T$. Then the operator H_k is maximal monotone and also strongly monotone with modulus λ_k^{-1}, i.e.,*

$$\langle x - x', y - y' \rangle \geq \lambda_k^{-1} ||x - x'||^2 \quad \forall y \in H_k(x), y' \in H_k(x').$$

Proof. Since Slater's condition holds for $z \in \mathrm{dom} T$, then $H = T + \partial g$, (with g as defined in Proposition 4.1), and H is maximal monotone. By definition $H_k = H + \nabla Q_k$, where $Q_k(x) := ||x - x^{k-1}||^2/(2\lambda_k)$, and since ∇Q_k is strongly monotone, it follows under our assumptions, that H_k shares the same property. □

Note that here since the multifunction in (4.25) is maximal monotone and strongly monotone, it automatically implies the existence and uniqueness of the sequence x^k in (4.25). We thus immediately obtain the following result.

THEOREM 4.6. *Let T be a maximal monotone operator on $\mathbb{R}^n$. Suppose that S is maximal monotone and that Slater condition holds for $z \in$* dom$T$. *Then, if the solution set of* PDVI *is nonempty, the primal-dual sequence $\{x^k, u^k\}$ generated by* PMDM *converges to a primal-dual solution $(x^*, u^*) \in X \times U$ of* VI *and* DVI, *respectively.*

Proof. Under the assumptions which guarantee that S is maximal monotone (cf. Proposition 3.2), the proof is essentially the same as the one given in Theorem 4.4. Indeed, from the above discussion, the sequence (x^k, u^k) generated by PMDM is the same as the one produced by (4.22) and thus applying here Theorem 4.5 we obtain that the sequence $\{x^k, u^k\}$ converges to a solution of PDVI. □

5. Concluding remarks. We have presented three new methods to solve VI problems. The resulting algorithms can be viewed as a natural extension of the proximal-like and related dual multiplier methods used in convex optimization. The primal-dual method appears attractive, since it guarantees the full convergence of the primal sequence. At the computational level, the main bulk of the computation in MDM or PMDM is essentially to solve a system of nonlinear equations. For example, consider the standard nonlinear complementarity problems, i.e., with $T : \mathbb{R}^n \to \mathbb{R}^n$ single valued and continuous $T(x) := (T_1(x), \dots, T_n(x))^T$ and $C := \mathbb{R}^n_+$. In that case PMDM reduces to the following: $\forall i = 1, \dots, m$,

$$T_i(x^k) - u_i^k(\varphi^*)'(-\lambda_k x^k/u^{k-1}) + \lambda_k(x_i^k - x_i^{k-1}) = 0, \tag{5.1}$$

$$u_i^k = u_i^{k-1}(\varphi^*)'(-\lambda_k x^k/u_i^{k-1}),\ i = 1, \dots, m, \tag{5.2}$$

with φ^* explicitly given in Theorem 4.1(iv). By the same theorem φ^* enjoys the useful properties (v)–(viii), and assuming that T is smooth enough, the system of equations in the variable x^k in (5.1) can thus be solved efficiently via a Newton type method. Given the current success of smooth multiplier methods in solving efficiently large scale convex optimization problems, as exhibited by recent numerical experiments (see, e.g., the recent work [10]), we believe that the methods proposed in this paper are providing new and promising alternative numerical schemes for solving VIs and related complementarity problems that are worth further investigations.

REFERENCES

[1] H. Attouch, Z. Chbani, and A. Moudafi, *Recession Operators and Solvability of Variational Problems*, Ser. Adv. Math. Appl. Sci. 18, World Scientific, River Edge, NJ, 1994.

[2] H. Attouch and M. Thera, *A general duality principle for the sum of two operators*, J. Convex Anal., 3 (1996), pp. 1–24.

[3] A. Auslender, *Optimisation: Methodes Numeriques*, Masson, Paris, 1976.

[4] A. Auslender, *Asymptotic analysis for penalty and barrier methods in variational inequalities*, SIAM J. Control Optim., 37 (1999), pp. 653–671.

[5] A. Auslender, R. Cominetti, and M. Haddou, *Asymptotic analysis of penalty and barrier methods in convex and linear programming*, Math. Oper. Res., 22 (1997), pp. 43–62.

[6] A. Auslender and M. Haddou, *An interior proximal method for convex linearly constrained problems and its extension to variational inequalities*, Math. Programming, 71 (1995), pp. 77–100.

[7] A. Auslender, M. Teboulle, and S. Ben-Tiba, *A logarithmic-quadratic proximal method for variational inequalities*, Comput. Optim. Appl., 12 (1998), pp. 31–40.

[8] A. Auslender, M. Teboulle, and S. Ben-Tiba, *Interior proximal and multiplier methods based on second order homogeneous kernels*, Math. Oper. Res., 24 (1999), pp. 645–668.

[9] A. Bensoussan, J. L. Lions, and R. Temam, *Sur les methodes de decomposition, de decentralisation, de coordinations et applications*, in Methodes Numeriques en Sciences Physiques et Economiques, J. L. Lions and G. I. Marchouk, eds., Dunod-Bordas, Paris, 1974, pp.133–257.

[10] A. Ben-Tal and M. Zibulevsky, *Penalty/barrier multiplier methods for convex programming problems*, SIAM J. Optim., 7 (1997), pp. 347–366.

[11] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[12] C. Baiochi, G. Buttazo, F. Gastaldi, and F. Tomarelli, *General existence theorems for unilateral problems in continum mechanics*, Arch. Rational Mech. Anal., 100 (1988), pp. 149–189.

[13] H. Brezis and A. Haraux, *Images d'une somme d'operateurs monotones et applications*, Israel J. Math., 23 (1976), pp. 165–186.

[14] R. S. Burachik and A. N. Iusem, *A generalized proximal point algorithm for the variational inequality problem in a Hilbert space*, SIAM J. Optim., 8 (1998), pp. 197–216.

[15] Y. Censor, A. N. Iusem, and S. A. Zenios, *An interior-point method with Bregman functions for the variational inequality problem with paramonotone operators*, Math. Programming, 81 (1998), pp. 373–400.

[16] J. Eckstein, *Non linear proximal point algorithms using Bregman functions, with applications to convex programming*, Math. Oper. Res., 18 (1993), pp. 202–226.

[17] J. Eckstein and M. Ferris, *Smooth Methods of Multipliers for Complementarity Problems*, Research report RRR 27-96, RUTCOR, Rutgers University, New Brunswick, NJ, 1997.

[18] D. Gabay, *Applications of the method of multipliers to variational inequalities*, in Augmented Lagrangian Methods: Applications to the Solution of Boundary Value Problems, M. Fortain and R. Glowinski, eds., North-Holland, Amsterdam, 1983, pp. 299–331.

[19] O. Güler, *On the convergence of the proximal point algorithm for convex minimization*, SIAM J. Control Optim., 29 (1991), pp. 403–419.

[20] A. Iusem and M. Teboulle, *Convergence rate analysis of nonquadratic proximal and augmented Lagrangian methods for convex and linear programming*, Math. Oper. Res., 20 (1995), pp. 657–677.

[21] K. C. Kiwiel, *Proximal minimization methods with generalized Bregman functions*, SIAM J. Control Optim., 35 (1997), pp. 1142–1168.

[22] B. Martinet, *Regularisation d'inéquations variationnelles par approximations successive*, Rev. Francaise Informat. Recherche Opérationnelle, 4 (1970), pp. 154–158.

[23] G. Minty, *On the maximal domain of a monotone function*, Michigan Math. J., 8 (1961), pp. 135–137.

[24] U. Mosco, *Dual variational inequalities*, J. Math. Anal. Appl., 40 (1972), pp. 202–206.

[25] R. A. Polyak, *Modified barrier functions (theory and methods)*, Math. Programming, 54 (1992), pp. 177–222.

[26] R. A. Polyak and M. Teboulle, *Nonlinear rescaling and proximal-like methods in convex optimization*, Math. Programming, 76 (1997), pp. 265–284.

[27] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[28] R. T. Rockafellar, *On the maximality of sums of nonlinear monotone operators*, Trans. Amer. Math. Soc., 149 (1970), pp. 75–88.

[29] R. T. Rockafellar, *Monotone operators and the proximal point algorithm*, SIAM J. Control Optim., 14 (1976), pp. 877–898.

[30] R. T. Rockafellar, *Monotone operators and augmented Lagrangians in nonlinear programming*, in Nonlinear Programming 3, O. L. Mangasarian and J. B. Rosen, eds., Academic Press, New York, 1978, pp. 1–25.

[31] R. T. Rockafellar and R. J. B. Wets, *Variational Analysis*, Springer-Verlag, New York, 1998.

[32] M. Teboulle, *Entropic proximal mappings with application to nonlinear programming*, Math. Oper. Res., 17 (1992), pp. 670–690.

[33] M. Teboulle, *Convergence of proximal-like algorithms*, SIAM J. Optim., 7 (1997), pp. 1069–1083.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1116–1135

DEGENERATE NONLINEAR PROGRAMMING WITH A QUADRATIC GROWTH CONDITION*

MIHAI ANITESCU†

Abstract. We show that the quadratic growth condition and the Mangasarian–Fromovitz constraint qualification (MFCQ) imply that local minima of nonlinear programs are isolated stationary points. As a result, when started sufficiently close to such points, an L_∞ exact penalty sequential quadratic programming algorithm will induce at least R-linear convergence of the iterates to such a local minimum. We construct an example of a degenerate nonlinear program with a unique local minimum satisfying the quadratic growth and the MFCQ but for which no positive semidefinite augmented Lagrangian exists. We present numerical results obtained using several nonlinear programming packages on this example and discuss its implications for some algorithms.

Key words. nonlinear programming, quadratic growth, sequential quadratic programming, degeneracy

AMS subject classifications. 65K05, 90C30

PII. S1052623499359178

1. Introduction.

Recently, there has been renewed interest in analyzing and modifying sequential quadratic programming algorithms for constrained nonlinear optimization for cases where the traditional regularity conditions do not hold [5, 15, 14, 24, 29, 30]. This research has been motivated by the fact that large-scale nonlinear programming problems tend to be almost degenerate (have large condition numbers for the Jacobian of the active constraints). It is therefore important to establish to what extent the convergence properties of the sequential quadratic programming methods are dependent on the ill-conditioning of the constraints. In this work, we term as degenerate those nonlinear programs (NLPs) for which the gradients of the active constraints are linearly dependent. In this case there may be several feasible Lagrange multipliers.

Many of the previous analysis and rate of convergence results for degenerate NLPs [5, 15, 14, 24, 29, 30] are based on the validity of some second-order conditions. These are essentially equivalent to the condition in unconstrained optimization that, for a critical point of a function $f(x)$ to be a local minimum, $f_{xx} \succeq 0$ is a necessary condition and $f_{xx} \succ 0$ is a sufficient condition. Here $\succeq$ is the positive semidefinite ordering. The place of f_{xx} in constrained optimization is taken for these conditions by L_{xx}, the Hessian of the Lagrangian, which is now required to be positive definite on the critical cone for one or all of the Lagrange multipliers [8, 25].

This work differs from previous approaches in that we assume only that

(1) At a local solution x^* of the constrained NLP, the first-order Mangasarian–Fromovitz constraint qualification (MFCQ) [19, 20] holds.

(2) The quadratic growth (QG) condition [6, 18] is satisfied,

*Received by the editors July 22, 1999; accepted for publication (in revised form) September 28, 1999; published electronically June 6, 2000. This work was completed while the author was the Wilkinson Fellow at the Mathematics and Computer Science Division, Argonne National Laboratory, 9700 South Cass Avenue, Argonne, IL 60439. This work was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing, U.S. Department of Energy, under contract W-31-109-Eng-38.

http://www.siam.org/journals/siopt/10-4/35917.html

†Thackeray 301, Department of Mathematics, University of Pittsburgh, Pittsburgh, PA 15213 (anitescu@math.pitt.edu).

$$f(x) \geq f(x^*) + \sigma||x - x^*||^2 \tag{1.1}$$

for some $\sigma > 0$ and all x feasible in a neighborhood of x^*.

(3) The data of the problem are twice continuously differentiable.

These assumptions are equivalent to a weaker form of the second-order sufficient conditions [17, 6], which does not require the positive semidefiniteness of the Hessian of the Lagrangian on the entire critical cone.

We prove that these conditions guarantee that x^* is an isolated stationary point (1.3) of the NLP. This extends a result from [25] that required some second-order sufficient conditions to be satisfied for all multipliers. In particular, our work implies that if MFCQ holds and the second-order sufficient conditions hold for one multiplier, then x^* is a strict local minimum and an isolated stationary point. This is an important issue because it will prevent the algorithms considered in this work, which use only first-order information, from stopping arbitrarily close to x^*, except at x^*, for certain types of line searches.

We also show that, under the same assumptions, the L_∞ exact penalty sequential quadratic program (SQP) induces at least Q-linear convergence [23] of the penalized objective to $f(x^*)$ and R-linear convergence of the iterates. Finally, we provide an example of an NLP that satisfies our assumptions for which it is not possible to construct an augmented Lagrangian such that x^* will be an unconstrained local minimum. This may present an adverse case to algorithms based on this assumption, such as Lagrange multiplier methods. However, we show that it is possible to construct a nondifferentiable function that has x^* as its minimum, namely, the L_∞ penalty function (which can also be inferred from the results in [6]). We describe our computational experience with several nonlinear programming packages applied to this example and discuss the expected and observed behavior of LANCELOT [7], a Lagrange multiplier algorithm.

Our convergence analysis for the L_∞ exact penalty function suggests that it is possible to construct a convergence theory with the more general second-order conditions [17]. This may result in algorithms with superior robustness, because their properties depend on significantly fewer assumptions.

1.1. Previous work, framework, and notations.

We deal with the NLP problem

$$\min_x f(x) \quad \text{subject to } g(x) \leq 0, \tag{1.2}$$

where $f : \mathsf{R}^n \to \mathsf{R}$ and $g : \mathsf{R}^n \to \mathsf{R}^m$ are twice continuously differentiable.

We call x a stationary point if the following conditions hold for some $\lambda \in \mathsf{R}^m$:

$$\mathcal{L}_x(x, \lambda) = 0, \;\; \lambda \geq 0, \;\; g(x) \leq 0, \;\; \lambda^T g(x) = 0. \tag{1.3}$$

Here $\mathcal{L}$ is the Lagrangian function

$$\mathcal{L}(x, \lambda) = f(x) + \lambda^T g(x). \tag{1.4}$$

If certain regularity conditions hold (discussed below), then a local solution x^* of (1.2) is a stationary point. In that case (1.3) are referred to as the Karush–Kuhn–Tucker (KKT) conditions.

Since our analysis will be limited to a neighborhood of a point x^* that is a strict minimum, we will assume that all constraints are active at x^*, or $g(x^*) = 0$. Such a situation can be obtained by simply dropping the constraints i for which $g_i(x^*) < 0$, since this relationship holds in an entire neighborhood of x^*. This does not reduce the generality of our results, but it simplifies the notation because now we do not have to refer separately to the active set.

The regularity condition, or constraint qualification, ensures that a linear approximation of the feasible set in the neighborhood of x^* captures the geometry of the feasible set. Often in local convergence analysis of constrained optimization algorithms, it is assumed that the constraint gradients $\nabla g_i(x^*)$, $i = 1, 2, \ldots, m$, are linearly independent, so that the Lagrange multiplier in (1.3) is unique. We assume instead the MFCQ:

$$\nabla g_i(x^*)^T p < 0 \;\; \forall i \text{ and some } p \in \mathsf{R}^n. \tag{1.5}$$

It is well known [11] that MFCQ is equivalent to boundedness of the set $\mathcal{M}(x^*)$ of Lagrange multipliers that satisfy (1.3), that is,

$$\mathcal{M}(x^*) \overset{\text{def}}{=} \{\lambda \geq 0 \,|\, (x^*, \lambda) \;\text{ satisfy (1.3)}\}. \tag{1.6}$$

Note that $\mathcal{M}(x^*)$ is certainly polyhedral in any case.

The critical cone at x^* is [8, 26]

$$\mathcal{C} = \{u \in \mathsf{R}^n | \nabla g_i(x^*)^T u \leq 0,\; i = 1, 2, \ldots, m;\; \nabla f(x^*)^T u = 0\}. \tag{1.7}$$

We briefly review some of the second-order conditions in the literature, although they are not an assumption for our analysis but only a basis for comparison. In the framework of [8], the second-order sufficient conditions for x^* to be an isolated local solution of (1.2) are

$$\exists \lambda^* \in \mathcal{M}(x^*),\; \exists \sigma > 0 \text{ such that } v^T \mathcal{L}_{xx}(x^*, \lambda^*) v \geq \sigma \|v\|_2^2 \;\; \forall v \in \mathcal{C}. \tag{1.8}$$

If these conditions hold at x^* for some λ^*, then the quadratic growth condition is satisfied, irrespective of the validity of the first-order constraint qualification [8, 9]. However, this does not imply that x^* is an isolated stationary point, as shown by a simple example [25], which may prevent an optimization algorithm that uses only first derivative information from reaching x^* even when started arbitrarily close to x^*.

In [25] it is shown that if MFCQ holds and the relation (1.8) is satisfied for all $\lambda^* \in \mathcal{M}(x^*)$, then x^* is an isolated stationary point and a minimum of (1.2). Also, with these conditions, the exact solution is Lipschitz stable with respect to perturbations. By compactness of $\mathcal{M}(x^*)$, we can choose σ independently of λ^* in this case. In [1] it is proven that, under these assumptions, the L_∞ exact penalty SQP will converge Q-linearly to $f(x^*)$, when the descent direction is computed by a quadratic program (QP) using only first-order information.

A refinement of the second-order conditions was introduced in [17]. In the presence of MFCQ, those conditions require that

$$\forall u \in \mathcal{C},\; \exists \lambda^* \in \mathcal{M}(x^*) \text{ such that } u^T \nabla_{xx} \mathcal{L}(x^*, \lambda^*) u > 0. \tag{1.9}$$

Further analysis shows that, in the presence of MFCQ, these conditions are necessary and sufficient for the QG condition to hold [6, 17, 18, 26]. Also, the exact solution is Lipschitz stable with respect to certain classes of perturbations [26], though not to any perturbation (see an example in [12, p. 308]). In this paper we assume only the QG condition and MFCQ, and thus we do not use the perturbation results.

If the condition (1.9) holds, but (1.8) does not, then there may be no augmented Lagrangian with a positive semidefinite Hessian, as we will show with an example. This is an interesting aspect since it invalidates the usual working assumption of Lagrange multiplier methods [4].

Finally, we review some of the facts concerning the L_∞ nondifferentiable exact penalty function:

$$P(x) = \max\{0, g_1(x), g_2(x), \dots, g_m(x)\}. \tag{1.10}$$

We are looking for an unconstrained minimum of the function

$$\phi(x) = f(x) + c_\phi P(x), \tag{1.11}$$

where c_ϕ is a sufficiently large constant. Descent directions d of $\phi(x)$ at the point x can be obtained by solving the following QP [4]:

$$\begin{array}{lll} \min\limits_{\Delta} & \nabla f(x)^T\Delta + \frac{1}{2}\Delta^T H\Delta + c_\phi\zeta & \\ \text{subject to} & g_j(x) + \nabla g_j(x)^T\Delta \le \zeta, & j = 0, 1, 2, \dots, m, \end{array} \tag{1.12}$$

where H is some positive definite matrix and $g_0(x) \equiv 0$ is added for a more compact notation for $\zeta \ge 0$. In this paper the analysis will be restricted to the case $H = I$, although the same results apply for any other positive definite matrix.

At the current point x^k of an iterative procedure that attempts to determine x^*, the QP (1.12) generates the descent direction d^k. The next iterate is $x^{(k+1)} = x^k + \alpha^k d^k$, where α^k is obtained by a line search procedure. Usual stepsize rules are the minimization rule, the limited minimization rule, and the Armijo rule [4]. For these rules, any limit point of $\{x^k\}$ is a stationary point of $\phi(x)$, and the descent procedure is therefore globally convergent in this sense [4].

If, in addition,

$$c_\phi > \sum_{j=1}^{m} \lambda_j^*$$

for some $\lambda^* \in \mathcal{M}(x^*)$, then x^* is a stationary point of $\phi(x)$ [3]. A suitable value for c_ϕ is not available in the early stages of the algorithm, but a good estimate can be found via an update scheme [3]. Here we assume that c_ϕ is constant and

$$c_\phi > \sum_{j=1}^{m} \lambda_j^* + 2\gamma \tag{1.13}$$

for all $\lambda^* \in \mathcal{M}(x^*)$, where γ is some prescribed safety factor.

Consider the quadratic program

$$\begin{array}{lll} \min\limits_{\Delta} & \nabla f(x)^T\Delta + \frac{1}{2}\Delta^T\Delta & \\ \text{subject to} & g_j(x) + \nabla g_j(x)^T\Delta \le 0, & j = 1, 2, \dots, m. \end{array} \tag{1.14}$$

We denote the unique solution of this program by d or $d(x)$ and the set of its multipliers by $\mathcal{M}(x)$. At x^* (1.14) has the same multiplier set as (1.2), which are both denoted by $\mathcal{M}(x^*)$. Since MFCQ is satisfied at x^*, this QP is feasible in a neighborhood of x^*. The KKT conditions for this QP require

$$\begin{gathered} d + \nabla f(x) + \nabla g(x)\lambda = 0, \\ \lambda \geq 0, \;\; g(x) + \nabla g(x)^T d \leq 0, \;\; \lambda^T(g(x) + \nabla g(x)^T d) = 0. \end{gathered} \tag{1.15}$$

With these notations, $d = 0$ at $x = x^*$.

At x^*, the QP (1.14) satisfies MFCQ and the second-order sufficient conditions from [25]. Therefore, there exists c_d such that, in a neighborhood of x^* we have [25] $||d|| \leq c_d ||x - x^*||$ and, $\forall \lambda \in \mathcal{M}(x)$, there exists $\lambda^* \in \mathcal{M}(x^*)$ such that

$$||\lambda - \lambda^*|| \leq c_d ||x - x^*||. \tag{1.16}$$

Therefore, from the definition of c_ϕ, there exists a neighborhood of x^* such that

$$c_\phi > \gamma + \sum_{i=1}^{m} \lambda_i \tag{1.17}$$

for all multipliers $\lambda \in \mathcal{M}(x)$. For such x, it can be verified by inspection that $(d, \zeta = 0)$ is a solution of (1.12) [3, p. 195]. We therefore concentrate on the QP (1.14) because, if c_ϕ is large enough and we are sufficiently close to x^*, it generates the same descent direction as (1.12), thus sharing its global convergence property.

For some function $h : \mathsf{R}^n \to \mathsf{R}^k$ we denote by c_{1h}, c_{2h} bounds depending on the first and second derivatives of h. The positive and negative parts of $h(x)$ are $h^+(x) = \max\{h(x), 0\}$, and, respectively, $h^-(x) = \max\{-h(x), 0\}$, both taken componentwise. With this notation, $h(x) = h^+(x) - h^-(x)$. Also, in our notation, $\nabla g_i(x)$, λ, and $\nabla g(x)\lambda$ are column vectors.

2. Stationary points of NLPs satisfying MFCQ.

In this section, we assume that x is in a sufficiently small neighborhood of x^*, whose size or properties are specified in each of the following results. In particular, the standing assumptions hold on all neighborhoods considered here and

$$\nabla g_i(x)^T p < -\zeta_0 \quad \forall i \text{ and } x \in W(x^*). \tag{2.1}$$

Here p with $||p|| = 1$ is one of the vectors satisfying (1.5), $\zeta_0 > 0$, and $W(x^*)$ is a suitable neighborhood of x^*.

Lemma 2.1. *There exist $\bar{\alpha}_P > 0$, $c_P > 0$, such that, $\forall x \in W(x^*)$,*

$$g(x) \leq 0,\, g_i(x) = 0 \quad \textit{for some } i,\, 1 \leq i \leq m \Rightarrow P(x - \alpha p) \geq c_P \alpha \quad \forall \alpha \in [0, \bar{\alpha}_P].$$

Here $P(x)$ is the usual L_∞ penalty function (1.10).

Proof. We have by Taylor's theorem

$$g_i(x - \alpha p) \geq -\alpha \nabla g_i(x)^T p - c_{2g}\alpha^2 \geq \alpha\zeta_0 - \alpha^2 c_{2g}.$$

We choose

$$\bar{\alpha}_P = \frac{\zeta_0}{2c_{2g}}.$$

For $0 \leq \alpha \leq \alpha_P$ we have

$$g_i(x - \alpha p) \geq \alpha\zeta_0 - c_{2g}\alpha^2 = \alpha(\zeta_0 - \alpha c_{2g}) \geq \alpha\frac{\zeta_0}{2}.$$

The claim follows after choosing $c_P = \frac{\zeta_0}{2}$. □

The proof of the following lemma can be inferred from [6]. We include it here for completeness.

LEMMA 2.2. *There exists a c_ϕ such that*

$$f(x) + c_\phi P(x) - f(x^*) \geq \frac{\sigma}{2}||x - x^*||^2 \tag{2.2}$$

for all x in a neighborhood of x^*.

Proof. Let $r > 0$ be such that $B(x^*, r) \subset W(x^*)$. We choose $r_1 < \frac{r}{2}$ such that $\alpha = \frac{P(x)}{\zeta_0} < \min\{\bar{\alpha}_P, r/2\}$ for $x \in B(x^*, r_1)$. This is always possible because $P(x^*) = 0$. We then have that, for any $x \in B(x^*, r_1)$,

$$||x + \alpha p - x^*|| \leq ||x - x^*|| + \alpha \leq \frac{r}{2} + \frac{r}{2} = r$$

and thus $x + \alpha p \in B(x^*, r)$. By the intermediate value theorem, we have that $g_i(x + \alpha p) = g_i(x) + \alpha \nabla g_i(x + \alpha^* p)^T p$, where $0 \leq \alpha^* \leq \alpha$ and thus $x + \alpha^* p \in B(x^*, r)$, implying in turn that $\nabla g_i(x + \alpha^* p)^T p \leq -\zeta_0$. Therefore $g_i(x + \alpha p) \leq g_i(x) - \alpha\zeta_0 = g_i(x) - P(x) \leq 0$. Therefore $x + \alpha p$ is feasible.

Now take

$$\alpha_1 = \min\{\hat{\alpha} \geq 0 \,|\, g(x + \hat{\alpha} p) \leq 0\}.$$

If x is infeasible, then $\alpha_1 > 0$ and there exists i such that $g_i(x + \alpha_1 p) = 0$. Since $x + \alpha_1 p$ is feasible, and $0 \leq \alpha_1 \leq \bar{\alpha}_P$, Lemma 2.1 applies (with $x + \alpha_1 p$ replacing x and α_1 replacing α) to give

$$P(x) \geq c_P \alpha_1. \tag{2.3}$$

If x is feasible, then $\alpha_1 = 0$ and $P(x) = 0$, and the previous bound still applies.

From the QG assumption (1.1) and the feasibility of $x + \alpha_1 p$, we must have that

$$f(x + \alpha_1 p) - f(x^*) \geq \sigma||x - x^* + \alpha_1 p||^2$$

or

$$f(x) - f(x^*) \geq \sigma||x - x^* + \alpha_1 p||^2 - (f(x + \alpha_1 p) - f(x)). \tag{2.4}$$

By (2.3) and Taylor's theorem we have

$$f(x + \alpha_1 p) - f(x) \leq c_{1f}\alpha_1 \leq \frac{c_{1f}}{c_P} P(x). \tag{2.5}$$

Choose

$$c_\phi = \frac{c_{1f}}{c_P} + \frac{\sigma\bar{\alpha}_P}{c_P}.$$

Then by (2.3)

$$c_\phi P(x) = \frac{c_{1f}}{c_P} P(x) + \frac{\sigma\bar{\alpha}_P}{c_P} P(x) \geq \frac{c_{1f}}{c_P} P(x) + \sigma\bar{\alpha}_P\alpha_1 \geq \frac{c_{1f}}{c_P} P(x) + \sigma\alpha_1^2. \tag{2.6}$$

Using (2.5), (2.4), and (2.6) we get

$$f(x) - f(x^*) + c_\phi P(x) \geq \sigma||x - x^* + \alpha_1 p||^2 + \sigma\alpha_1^2 = \sigma||x - x^* + \alpha_1 p||^2 + \sigma||\alpha_1 p||^2.$$

The conclusion follows, because

$$\sigma||x - x^* + \alpha_1 p||^2 + \sigma||\alpha_1 p||^2 \geq \frac{\sigma}{2}||x - x^*||^2$$

from the Cauchy–Schwartz inequality. □

We can assume that c_ϕ from the previous lemma satisfies (1.17); otherwise we replace it with the right-hand side of (1.17) and the conclusion of the lemma still holds for the new c_ϕ.

To prove the following results, we will use the results from [16] concerning sets defined by linear constraints:

$$\mathcal{P} = \{x \in \mathsf{R}^n | a_i^T x + b_i \leq 0, i = 1, 2, \ldots, m_{ne}, \tilde{a}_j^T x + \tilde{b}_j = 0, j = 1, 2, \ldots, m_{eq}\}. \tag{2.7}$$

For such a set, denote by $d(x, \mathcal{P})$ the distance from a point $x \in \mathsf{R}^n$ to the set $\mathcal{P}$. Also, denote by $d_{\mathcal{P}}(x)$ the maximum value of the infeasibility:

$$\begin{aligned} d_{\mathcal{P}}(x) = \max\{0, a_1^T x + b_1, a_2^T x + b_2, \ldots, a_{m_{ne}}^T x + b_{m_{ne}},\\ |\tilde{a}_1^T x + \tilde{b}_1|, |\tilde{a}_2^T x + \tilde{b}_2|, \ldots, |\tilde{a}_{m_{eq}}^T x + \tilde{b}_{m_{eq}}|\}. \end{aligned}$$

Then there exists a number $\mu^*(\mathcal{P}) > 0$ such that

$$\mu^*(\mathcal{P}) d(x, \mathcal{P}) \leq d_{\mathcal{P}}(x) \quad \forall x \in \mathsf{R}^n.$$

The following lemma uses the fact that $\mathcal{M}(x^*)$ is polyhedral and can thus be expressed in the form (2.7).

Lemma 2.3. *Let $\mathcal{I}$ be an index set such that there exists a multiplier $\bar{\lambda} \in \mathcal{M}(x^*)$ with $\bar{\lambda}_{\mathcal{I}} = 0$. Then there exists a constant $c_{\mathcal{I}}$ such that $\forall \lambda \in \mathcal{M}(x^*)$ there exists a $\lambda^* \in \mathcal{M}(x^*)$ with $\lambda^*_{\mathcal{I}} = 0$ and such that $||\lambda - \lambda^*|| \leq c_{\mathcal{I}} ||\lambda_{\mathcal{I}}||_\infty$.*

For a vector λ we have denoted by $\lambda_{\mathcal{I}}$ the restriction of the vector to the index set $\mathcal{I}$.

Proof. Let $\mathcal{M}_{\mathcal{I}}(x^*)$ be the set of all $\lambda^* \in \mathcal{M}(x^*)$ such that $\lambda^*_{\mathcal{I}} = 0$. Then $\nu \in \mathcal{M}_{\mathcal{I}}(x^*)$ satisfies

$$\sum_{j=1}^{m} \nabla g_j(x^*) \nu_j = -\nabla f(x^*), \tag{2.8}$$

$$\nu_{\mathcal{I}} = 0, \tag{2.9}$$

$$\nu \geq 0. \tag{2.10}$$

From our assumptions, $\mathcal{M}_{\mathcal{I}}(x^*)$ is not empty. Thus from [16] there exists a $\mu^*(\mathcal{M}_{\mathcal{I}}) > 0$ such that

$$\mu^*(\mathcal{M}_{\mathcal{I}}) d(\lambda, \mathcal{M}_{\mathcal{I}}) \leq d_{\mathcal{M}_{\mathcal{I}}}(\lambda). \tag{2.11}$$

However, since $\lambda \in \mathcal{M}(x^*)$ satisfies (1.3), we have that only the constraints $\lambda_{\mathcal{I}} = 0$, (2.9), are violated. Thus $d_{\mathcal{M}_{\mathcal{I}}}(\lambda) = ||\lambda_{\mathcal{I}}||_\infty$. The conclusion follows from (2.11) by taking $c_{\mathcal{I}} = \frac{1}{\mu^*(\mathcal{M}_{\mathcal{I}})}$. The proof is complete. □

Corollary 2.4. *There exists $c_\lambda > 0$ such that, $\forall \lambda \in \mathcal{M}(x^*)$, there exists $\lambda^* \in \mathcal{M}(x^*)$, with $\lambda^*_{\mathcal{I}} = 0$ and such that $||\lambda - \lambda^*|| \leq c_\lambda ||\lambda_{\mathcal{I}}||_\infty$, whenever*

$$\mathcal{M}_{\mathcal{I}}(x^*) = \{\bar{\lambda} | \bar{\lambda} \in \mathcal{M}(x^*), \bar{\lambda}_{\mathcal{I}} = 0\} \neq \emptyset.$$

Proof. With the notations of Lemma 2.3, we take

$$c_\lambda = \max_{\mathcal{I}\subset\{1,2,\dots,m\}} c_{\mathcal{I}} \quad \text{for feasible } \mathcal{M}_{\mathcal{I}}(x^*). \qquad \square \tag{2.12}$$

LEMMA 2.5. *There exists a neighborhood $W_1(x^*)$ such that, $\forall x \in W_1(x^*), \lambda \in \mathcal{M}(x)$, $\lambda_{\mathcal{I}} = 0$ implies that there exists a $\lambda^* \in \mathcal{M}(x^*)$ with $\lambda^*_{\mathcal{I}} = 0$.*

Proof. Assume the contrary. Then there exists a sequence $x^k \to x^*$ such that there exists $\lambda^k \in \mathcal{M}(x)$ and an index set $\mathcal{I}$ for which $\lambda_{\mathcal{I}} = 0$, but $\lambda^*_{\mathcal{I}} \neq 0 \; \forall \lambda^* \in \mathcal{M}(x^*)$. Since there is only a finite set of index sets, we can extract an infinite subsequence for which the above happens for a fixed set $\mathcal{I}$. By extracting another subsequence, we can assume that λ^k is convergent, from (1.16) and the fact that $\mathcal{M}(x^*)$ is compact.

But then $\lambda^k \to \lambda^* \in \mathcal{M}(x^*)$ and $\lambda^*_{\mathcal{I}} = 0$, a contradiction. $\square$

From here on we will use extensively the fact that, for h twice continuously differentiable, we have

$$\left\| h(x) - h(x^*) - \tfrac{(\nabla h(x) + \nabla h(x^*))^T}{2}(x - x^*) \right\| \leq \psi_{3h}(||x - x^*||)||x - x^*||^2, \tag{2.13}$$

where $\psi_{3h}(z) : \mathsf{R} \to \mathsf{R}$ is a continuous function with $\psi_{3h}(0) = 0$. Indeed by Taylor's theorem we have that there exist continuous functions $\psi^1_{3h}(z) : \mathsf{R} \to \mathsf{R}$ and $\psi^2_{3h}(z) : \mathsf{R} \to \mathsf{R}$ with $\psi^1_{3h}(0) = \psi^2_{3h}(0) = 0$ such that

$$\begin{aligned} &\left\| h(x) - h(x^*) - \nabla_x h(x^*)^T (x - x^*) - \frac{1}{2}(x - x^*)^T \nabla_{xx} h(x^*)(x - x^*) \right\| \\ &\qquad \leq \psi^1_{3h}(||x - x^*||)||x - x^*||^2, \end{aligned}$$

and

$$\begin{aligned} &\left\| \frac{(\nabla_x h(x) + \nabla_x h(x^*))^T}{2}(x - x^*) - \frac{(\nabla_x h(x^*) + \nabla_x h(x^*))^T}{2}(x - x^*) \right. \\ &\qquad \left. - \frac{1}{2}(x - x^*)^T \nabla_{xx} h(x^*)(x - x^*) \right\| \leq \psi^2_{3h}(||x - x^*||)||x - x^*||^2. \end{aligned}$$

The relation (2.13) now follows by comparing the last two inequalities and taking $\psi_{3h}(z) = \psi^1_{3h}(z) + \psi^2_{3h}(z)$. If h were three times continuously differentiable, then ψ_{3h} would be related to the third derivative of h, from the error formula of trapezoidal integration [2], which is the origin of our subscript notation.

THEOREM 2.6. *There exists a constant $c_\sigma > 0$ such that in a neighborhood of x^* we have that*

$$||d||^2 + P(x) + \lambda^T g^-(x) \geq c_\sigma ||x - x^*||^2,$$

where (d, λ) is the solution of the QP (1.14).

Proof. From (1.16), there exists a $\lambda^* \in \mathcal{M}(x^*)$ such that $||\lambda - \lambda^*|| \leq c_d ||x - x^*||$. Let $\mathcal{I}$ be the set of indices i for which $\lambda_i = 0$. We have $||\lambda^*_{\mathcal{I}}||_\infty = ||\lambda^*_{\mathcal{I}} - \lambda_{\mathcal{I}}||_\infty \leq c_d ||x - x^*||$. From Corollary 2.4 and Lemma 2.5 there exists a $\tilde{\lambda} \in \mathcal{M}(x^*)$ with $\tilde{\lambda}_{\mathcal{I}} = 0$ and $||\tilde{\lambda} - \lambda^*|| \leq c_\lambda ||\lambda^*_{\mathcal{I}}||_\infty \leq c_\lambda c_d ||x - x^*||$. As a result

$$||\lambda - \tilde{\lambda}|| \leq ||\lambda - \lambda^*|| + ||\lambda^* - \tilde{\lambda}|| \leq (c_d + c_d c_\lambda)||x - x^*|| \tag{2.14}$$

and $\lambda_i = 0 \Rightarrow \tilde{\lambda}_i = 0$. The important consequence of this fact, using the complementarity relations from (1.15), is that

$$\begin{aligned} \left(\lambda_i + \tilde{\lambda}_i\right) g_i(x) &= 2\lambda_i g_i(x) + \left(\tilde{\lambda}_i - \lambda_i\right) g_i(x) \\ = \quad &- \left(\tilde{\lambda}_i - \lambda_i\right) \nabla g_i(x)^T d + 2\lambda_i g_i(x) \quad \forall i,\ 1 \leq i \leq m. \end{aligned} \tag{2.15}$$

Indeed, $\lambda_i > 0$ implies $g_i(x) + \nabla g_i(x)^T d = 0$ from (1.15), whereas $\lambda_i = 0$ implies $\tilde{\lambda}_i = 0$ and all the above equalities are 0.

From Lemma 2.2 we have that

$$\begin{aligned} \frac{\sigma}{2}||x - x^*||^2 &\leq f(x) - f(x^*) + c_\phi P(x) \\ \leq \psi_{3f}(||x - x^*||)||x - x^*||^2 &+ \frac{1}{2}\left(\nabla f(x) + \nabla f(x^*)\right)^T (x - x^*) + c_\phi P(x) \\ &= \psi_{3f}(||x - x^*||)||x - x^*||^2 + c_\phi P(x) \\ &+ \frac{1}{2}\left(-d - \nabla g(x)\lambda - \nabla g(x^*)\tilde{\lambda}\right)^T (x - x^*). \end{aligned}$$

Here (d, λ) is a solution of (1.15), and $\tilde{\lambda} \in \mathcal{M}(x^*)$ satisfies (1.3). We also used (2.13). We now employ the identity $ab + cd = \frac{1}{2}((a+c)(b+d) + (a-c)(b-d))$, (2.14), and $|| \left(\nabla g(x) - \nabla g(x^*)\right)^T (x - x^*)|| \leq c_{2g}||x - x^*||^2$ from Taylor's theorem to get, by continuing from the previous equation,

$$\begin{aligned} \frac{\sigma}{2}||x - x^*||^2 &\leq \psi_{3f}(||x - x^*||)||x - x^*||^2 \\ &+ \frac{1}{2}\Bigg(- d - \frac{1}{2}(\nabla g(x) + \nabla g(x^*))\left(\lambda + \tilde{\lambda}\right) \\ &- \frac{1}{2}(\nabla g(x) - \nabla g(x^*))\left(\lambda - \tilde{\lambda}\right)\Bigg)^T (x - x^*) + c_\phi P(x) \\ &\leq (\psi_{3f}(||x - x^*||) + c_{2g}c_d(1 + c_\lambda)||x - x^*||)||x - x^*||^2 \\ &+ c_\phi P(x) - \frac{1}{2}d^T(x - x^*) - \frac{1}{4}\left((\nabla g(x) + \nabla g(x^*))\left(\lambda + \tilde{\lambda}\right)\right)^T (x - x^*). \end{aligned} \tag{2.16}$$

We denote

$$\lambda_\infty^* = \max_{\lambda^* \in \mathcal{M}(x^*)} \max_{i=1,2,\dots,m} \lambda_i^*$$

and

$$\lambda_\infty = \max\{\lambda_\infty^*, 1\}. \tag{2.17}$$

From (1.16), $||\lambda + \tilde{\lambda}||_\infty \leq 4\lambda_\infty$ for x sufficiently close to x^*. By using the definition of λ_∞, (2.13), (2.15), and (2.14), we get

$$\begin{aligned} &-\frac{1}{4}\left((\nabla g(x) + \nabla g(x^*))\left(\lambda + \tilde{\lambda}\right)\right)^T (x - x^*) \\ &= -\frac{1}{4}(x - x^*)^T\left((\nabla g(x) + \nabla g(x^*))\left(\lambda + \tilde{\lambda}\right)\right) \\ &\leq 2\lambda_\infty \psi_{3g}\left(||x - x^*||\right)||x - x^*||^2 - \frac{1}{2}\left(\lambda + \tilde{\lambda}\right)^T g(x) \\ &= 2\lambda_\infty \psi_{3g}\left(||x - x^*||\right)||x - x^*||^2 + \frac{1}{2}\left(\tilde{\lambda} - \lambda\right)^T \nabla g(x)^T d - \lambda^T g(x) \\ &\leq 2\lambda_\infty \psi_{3g}\left(||x - x^*||\right)||x - x^*||^2 + c_{1g}\left(c_d + c_d c_\lambda\right)||x - x^*||||d|| + \lambda^T g^-(x), \end{aligned}$$

since $-\lambda^T g(x) = \lambda^T g^-(x) - \lambda^T g^+(x)$. Using the above bound in (2.16), together with $-\frac{1}{2}d^T(x-x^*) \leq \frac{1}{2}||d||||x-x^*||$, we get

$$\begin{aligned}\frac{\sigma}{2}||x-x^*||^2 \leq \big(\psi_{3f}\left(||x-x^*||\right) + c_{2g}c_d\left(1+c_\lambda\right)||x-x^*|| \\ +2\lambda_\infty\psi_{3g}\left(||x-x^*||\right)\big)||x-x^*||^2 \\ +c_\phi P(x) + \frac{1}{2}||d||||x-x^*|| + c_{1g}\left(c_d + c_d c_\lambda\right)||x-x^*||||d|| + \lambda^T g^-(x) \\ = c_\phi P(x) + \lambda^T g^-(x) + B||x-x^*||||d|| + \psi\left(||x-x^*||\right)||x-x^*||^2,\end{aligned}$$

where $B = \frac{1}{2} + c_{1g}(c_d + c_d c_\lambda)$ and $\psi(||x-x^*||) = (\psi_{3f}\left(||x-x^*||\right) + c_{1g}c_d\left(1+c_\lambda\right)||x-x^*|| + 2\lambda_\infty\psi_{3g}\left(||x-x^*||\right))$.

We can now choose a sufficiently small neighborhood of x^* such that $\psi(||x-x^*||) \leq \frac{\sigma}{4}$ and subtract the last term of the last relation from the lower bound $\frac{\sigma}{2}||x-x^*||^2$. We take $A = \lambda^T g^-(x) + c_\phi P(x)$, and with this new notation, we get that

$$\frac{\sigma}{4}||x-x^*||^2 \leq A + B||d||||x-x^*||.$$

We treat $||x-x^*||$ as a variable and, by using the formulas for the quadratic equation, we get that

$$||x-x^*|| \leq \frac{2}{\sigma}\left(B||d|| + \sqrt{B^2||d||^2 + A\sigma}\right).$$

By using the arithmetic-quadratic mean inequality, we get that

$$\begin{aligned}||x-x^*||^2 \leq \frac{8}{\sigma^2}\left(2B^2||d||^2 + A\sigma\right) = \frac{16}{\sigma^2}B^2||d||^2 + \frac{8}{\sigma}\left(\lambda^T g^-(x) + c_\phi P(x)\right) \\ \leq \max\left\{\frac{16}{\sigma^2}B^2, \frac{8}{\sigma}, \frac{8}{\sigma}c_\phi\right\}\left(||d||^2 + P(x) + \lambda^T g^-(x)\right).\end{aligned}$$

Choosing

$$c_\sigma = \frac{1}{\max\left\{\frac{16}{\sigma^2}B^2, \frac{8}{\sigma}, \frac{8}{\sigma}c_\phi\right\}},$$

we prove the claim. □

COROLLARY 2.7. *x^* is an isolated stationary point.*

Proof. Let x be another stationary point of the NLP in the neighborhood of x^* where the above theorem holds. Therefore there exists a $\lambda \in \mathcal{M}(x)$ satisfying (1.3). Hence $(d = 0, \lambda)$ is a solution of (1.15) and $d = 0$ is the unique solution of the strictly convex QP (1.14). Since $d = 0$, x is feasible from (1.14) and $P(x) = 0$ or $g = -g^-(x)$. Now from the complementarity conditions in (1.15) we get $\lambda^T g^- = -\lambda^T g = 0$. From the previous theorem we get $x = x^*$, which proves the claim. □

COROLLARY 2.8. *If the second-order sufficient condition* (1.8) *is satisfied for one multiplier, and if MFCQ holds at x^*, then x^* is an isolated stationary point.*

Proof. Since x^* satisfies the QG condition (1.1) under these assumptions [8, 9] and MFCQ holds, Corollary 2.7 applies. □

3. An example without a locally convex augmented Lagrangian. We construct an example of an NLP with a unique solution at which the QG condition (1.1) and MFCQ (1.5) hold but for which (1.8) does not occur for any multiplier

$\lambda^* \in \mathcal{M}(x^*)$. The example from [17, p. 270] also does not satisfy (1.8) for any multiplier $\lambda^* \in \mathcal{M}(x^*)$, but it does not have a locally unique solution ($\zeta_1 = \zeta_2$ is a solution of that example) and cannot satisfy (1.1).

Consider the matrix

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & -2 \end{pmatrix}.$$

Take $u = (\frac{\sqrt{3}}{2}, \frac{1}{2})$. We then have that $u^T Q u = \frac{1}{4}$ and $||u||^2 = 1$. Since the vector $u_0 = (1, 0)$ corresponds to the positive eigenvalue, we have that for any u at an angle of at most $\frac{\pi}{6}$ from u_0, $u^T Q u \geq \frac{1}{4}||u||^2$. Consider now the rotation matrix

$$U_k = \begin{pmatrix} \cos(\frac{k\pi}{4}) & \sin(\frac{k\pi}{4}) \\ -\sin(\frac{k\pi}{4}) & \cos(\frac{k\pi}{4}) \end{pmatrix}.$$

Define $Q_k = U_k^T Q U_k$ for $k = 0, \ldots, 3$. We then have $Q_0 + Q_2 = Q_1 + Q_3 = -I_2$, since Q_0 and Q_2 have the same axes of symmetry, but with the eigenvalues switched. Also, for any $u \in \mathsf{R}^2$, there exists a k such that $u^T Q_k u \geq \frac{1}{4}||u||^2$, since the $\frac{\pi}{3}$ wide cones centered at the axis of the positive eigenvalues of Q_k now sweep the entire R^2.

Consider now the optimization problem

$$\min_{(x,y,z)} z \quad \text{subject to } z \geq (x\,y)Q_k(x\,y)^T, \quad k = 0, \ldots, 3. \tag{3.1}$$

By the previous observation, we have that $z \geq \frac{1}{4}(x^2 + y^2)$ on the feasible set; thus $z \geq 0$. Clearly, the only solution of the problem is $(0, 0, 0)$. Since $z \geq \frac{z^2}{4}$, if $0 \leq z \leq 4$, we have that $z \geq \frac{1}{8}(x^2 + y^2 + z^2)$ $\forall$ x, y, z feasible and $0 < z < 4$.

Therefore at $x^* = (0, 0, 0)$ the QC condition is satisfied for the above NLP, with constant $\frac{1}{8}$. Obviously, MFCQ holds at $(0, 0, 0)$, and a simple calculation following (1.3) shows that $\sum_{k=0\ldots3} \lambda_k = 1$, for λ_k a multiplier of (3.1). In particular, at least one multiplier has to be positive. Also, at $(0, 0, 0)$, all constraints are active and their gradients are $(0, 0, -1)$ for any of them. As a result, the linear constraints in (1.8) now become either $z \geq 0$ or $z = 0$, with at least one being $z = 0$. Therefore the critical cone at x^* is $\mathcal{C} = \{(x, y, z) \in \mathsf{R}^3 | z = 0\}$.

Assume that there is a choice $\lambda \in \mathcal{M}(x^*)$ such that L_{xx}, the Hessian of the Lagrangian, is positive semidefinite on the critical cone:

$$(x\,y\,z) \begin{pmatrix} \sum_{k=0\ldots3} \lambda_k Q_k & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \geq 0 \quad \forall (x, y, z) \text{ such that } z = 0. \tag{3.2}$$

This is equivalent to

$$\sum_{k=0\ldots3} \lambda_k Q_k \succeq 0. \tag{3.3}$$

Since our construction is invariant to rotations with $\frac{\pi}{4}$ ($U_1^T Q_3 U_1 = Q_0$), it follows that the positive semidefiniteness holds for any circular permutation σ of this multiplier set:

$$\sum_{k=0\ldots3} \lambda_{\sigma(k)} Q_k \succeq 0. \tag{3.4}$$

We denote by $\mathcal{A}_c(4)$ the set of circular permutations of four elements. Since the set of positive definite matrices is a convex cone, and

$$\sum_{\sigma \in \mathcal{A}_c(4)} \lambda_{\sigma(k)} = 1,$$

we must have

$$0 \preceq \frac{1}{4} \sum_{\sigma \in \mathcal{A}_c(4)} \sum_{k=0\ldots3} \lambda_{\sigma(k)} Q_k = \frac{1}{4} \sum_{k=0\ldots3} Q_k \sum_{\sigma \in \mathcal{A}_c(4)} \lambda_{\sigma(k)} = \frac{1}{4} \sum Q_k = -\frac{1}{2} I,$$

which is impossible. Therefore L_{xx} cannot be positive semidefinite on the critical cone for any choice $\lambda \in \mathcal{M}(x^*)$. Hence the second-order conditions from [8, 25] will not hold for any choice of the multipliers.

3.1. The augmented Lagrangian approach of LANCELOT. Here we discuss the expected behavior of LANCELOT [7], a Lagrange multiplier algorithm, when applied to this example. For this method, the inequalities of the NLP (1.2) are converted into equalities [4, 7]. The feasible set can be represented as [7]

$$g_i(x) + t_i = 0, \ t_i \geq 0 \quad \text{for } i = 1, \ldots, m.$$

The NLP is replaced by a bound-constrained optimization problem. The equality constraints are incorporated in the objective function based on an estimate λ of the multipliers and a penalty term,

$$\begin{array}{lll} & \min\limits_{x} & f(x) + \sum_{i=1}^4 [\lambda_i (g_i(x) + t_i) + \frac{1}{\mu}(g_i(x) + t_i)^2] \\ (3.5) & & \\ & \text{subject to} & t_i \geq 0, \quad i = 1, \ldots, m. \end{array}$$

Here μ is the barrier parameter. The objective function in (3.5) is the augmented Lagrangian. The problem is subjected to an additional trust-region constraint [7] to enforce global convergence.

The desired outcome is to have μ bounded away from zero and the trust region inactive as λ approaches $\mathcal{M}(x^*)$ and the solution of the above problem approaches x^*.

If that happens for our example, then, by a continuity argument following the lower boundedness of μ, $(x^*, t = 0)$ should be a solution of (3.5) for an appropriate choice of λ, μ. Since (3.5) has linearly independent gradients of the constraints, both the first- and second-order necessary conditions of the type (1.8) must hold [9]. The first-order necessary condition results in

$$\nabla f(x^*) + \nabla g(x^*) \lambda = 0, \quad \lambda + \nu = 0, \quad \nu \leq 0,$$

where ν, with components $\nu_i \leq 0$, are the multipliers associated with the variables t_i. As a result $\lambda \in \mathcal{M}(x^*)$. The second-order necessary conditions require that

$$\nabla_{(x,t)(x,t)} L|_{(x^*,0)} = \begin{pmatrix} F_{xx} + \sum_{i=1}^4 (\lambda_i G_{xx} + \frac{2}{\mu} \nabla g_i(x^*) \nabla g_i(x^*)^T) & \frac{2}{\mu} \nabla g(x^*) \\ \frac{2}{\mu} \nabla g(x^*)^T & \frac{2}{\mu} I_4 \end{pmatrix}$$

be positive semidefinite on the critical cone $\mathcal{C}$ of the NLP (3.5). In this example, $\mathcal{C}$ contains the subspace $(\delta x, \delta t)$ with $\delta t = 0$. This results in

$$0 \preceq F_{xx} + \sum_{i=1}^4 \left(\lambda_i G_{xx} + \frac{2}{\mu} \nabla g_i(x^*) \nabla g_i(x^*)^T \right) = \begin{pmatrix} \sum_{i=1}^4 \lambda_i Q_i & 0 \\ 0 & \frac{8}{\mu} \end{pmatrix}$$

since $\nabla g_i(x^*)^T = (0, 0, -1)$ $\forall i = 1, \ldots, m$ or

$$0 \preceq \begin{pmatrix} \sum_{i=1}^4 \lambda_i Q_i & 0 \\ 0 & \frac{8}{\mu} \end{pmatrix}.$$

We proved that the last matrix cannot be positive semidefinite for our example and we thus get a contradiction. This shows that either the trust region will be active arbitrarily close to x^* or $\mu \to 0$.

This also shows that the Hessian of the augmented Lagrangian of the equality constrained problem

$$F_{xx} + \sum_{i=1}^{4} \left[\lambda_i G_{xx} + \frac{2}{\mu} \nabla g_i(x^*) \nabla g_i(x^*)^T \right]$$

is not positive semidefinite and thus the augmented Lagrangian of the equality constrained problem cannot be locally convex.

4. Linear convergence of the SQP with L_∞ penalty $P(x)$. In this section we analyze the rate of convergence of an SQP algorithm that uses (1.14) to determine a descent direction for the merit function $\phi(x)$ (1.11). The key result is Lemma 4.2, which bounds below the decrease in $\phi(x)$ proportionally to the quantity in Theorem 2.6.

The points x considered in this subsection are assumed to be sufficiently close to x^*. The notation d and $\lambda \in \mathcal{M}(x)$ will refer to the solutions of (1.14) and (1.15). Also, $P(x)$ is the L_∞ penalty function (1.10) and $\phi(x) = f(x) + c_\phi P(x)$.

4.1. Proof of the technical results.

LEMMA 4.1. *There exists $c_{2g} > 0$ such that*

$$P(x + \alpha d) \leq (1 - \alpha) P(x) + c_{2g} \alpha^2 ||d||^2 \quad \forall \alpha \in [0, 1].$$

Proof. Since d is a feasible point of (1.14), we have that $\nabla g_i(x)^T d \leq -g_i(x)$ $\forall i \in \{1, \ldots, m\}$. By Taylor's remainder theorem

$$g_i(x + \alpha d) \leq (1 - \alpha) g_i(x) + c_{2g} \alpha^2 ||d||^2 \quad \forall \alpha \in [0, 1] \quad \forall i = 1, \ldots, m.$$

Hence

$$\begin{aligned} \max_{1 \leq i \leq m} \{g_i(x + \alpha d)\} &\leq (1 - \alpha) \max_{1 \leq i \leq m} \{g_i(x)\} + c_{2g} \alpha^2 ||d||^2 \\ &\leq (1 - \alpha) P(x) + c_{2g} \alpha^2 ||d||^2 \quad \forall \alpha \in [0, 1]. \end{aligned}$$

This completes the proof. □

LEMMA 4.2. *There exist $\bar{\alpha}$, $0 < \bar{\alpha} \leq 1$, and $c_2 > 0$ such that, for some $(\lambda) \in \mathcal{M}(x)$*

$$\begin{aligned} \phi(x + \alpha d) - \phi(x) &\leq -\alpha \frac{1}{2} \left((d)^T d + \gamma P(x) + \lambda^T g^-(x) \right) \\ &\leq -c_2 \alpha (||d||^2 + P(x) + \lambda^T g^-(x)) \quad \forall \alpha \in [0, \bar{\alpha}]. \end{aligned}$$

Proof. Writing the KKT conditions for (1.14), we obtain

$$d + \nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x) = 0$$

and, hence,

$$(d)^T d + \nabla f(x)^T d + \sum_{i=1}^m \lambda_i \nabla g_i(x)^T d = 0,$$
$$(d)^T d + \nabla f(x)^T d - \sum_{i=1}^m \lambda_i g_i(x) = 0,$$

since, by the complementarity conditions satisfied by the solution of (1.14), $\lambda^T \nabla g(x)^T d = -\lambda^T g(x) \;\; \forall i = 1, m$. Therefore, since $g_i(x) = g_i^+(x) - g_i^-(x)$,

$$\begin{aligned} \nabla f(x)^T d &= -(d)^T d + \sum_{i=1}^m \lambda_i (g_i^+(x) - g_i^-(x)) \\ &\le -(d)^T d + P(x) \left(\sum_{i=1}^m \lambda_i \right) - \lambda^T g^-(x) \qquad (4.1) \\ &\le -(d)^T d + (c_\phi - \gamma) P(x) - \lambda^T g^-(x) \end{aligned}$$

by (1.10), (1.17). By Taylor's remainder theorem,

$$f(x + \alpha d) \le f(x) + \alpha \nabla f(x)^T d + c_{2f} \alpha^2 ||d||^2.$$

Hence, for $\alpha \in [0, 1]$, we have that

$$\begin{aligned} f(x + \alpha d) + c_\phi P(x + \alpha d) &\le f(x) + \alpha \nabla f(x)^T d + c_{2f} \alpha^2 ||d||^2 \\ +(1-\alpha) c_\phi P(x) + c_\phi c_{2g} \alpha^2 ||d||^2 &\le f(x) + (1 - \alpha) c_\phi P(x) \\ +\alpha \left(-(d)^T d + (c_\phi - \gamma) P(x) - \lambda^T g^-(x) \right) &+ (c_\phi c_{2g} + c_{2f}) \alpha^2 ||d||^2 \\ = f(x) + c_\phi P(x) - \alpha \left((d)^T d + \gamma P(x) + \lambda^T g^-(x) \right) &+ (c_\phi c_{2g} + c_{2f}) \alpha^2 ||d||^2 \end{aligned}$$

from (4.1) and Lemma 4.1. Therefore, for $\alpha \in [0, 1]$,

$$\phi(x + \alpha d) - \phi(x) \le -\alpha \left((d)^T d + \gamma P(x) + \lambda^T g^-(x) \right) + (c_\phi c_{2g} + c_{2f}) \alpha^2 ||d||^2.$$

The result of the statement follows by choosing $\bar{\alpha} = \min\{1, \frac{1}{2(c_\phi c_{2g} + c_{2f})}\}$ and $c_2 = \frac{1}{2} \min\{\gamma, 1\}$. □

LEMMA 4.3. *There exists a constant c_5 such that, $\forall \lambda \in \mathcal{M}(x)$,*

$$\phi(x) - \phi(x^*) \le c_5 \left(P(x) + ||x - x^*||^2 + \lambda^T g^-(x) + ||d||^2 \right).$$

Proof. From (1.15) and the definition of the Lagrangian (1.4) it follows, using Taylor's theorem, that for a sufficiently small neighborhood of x, we have

$$\mathcal{L}(x, \lambda^*) - \mathcal{L}(x^*, \lambda^*) - \Sigma ||x - x^*||^2 \le 0 \quad \forall \lambda^* \in \mathcal{M}(x^*), \qquad (4.2)$$

where $\Sigma = \max\{c_{2f}, c_\phi c_{2g}\}$. Also, by (1.16), we can choose $\lambda^* \in \mathcal{M}(x^*)$ such that

$$|g(x)^T (\lambda^* - \lambda)| \le c_{1g} c_d ||x - x^*||^2.$$

Since $\mathcal{L}(x^*, \lambda^*) = f(x^*)$ and thus $\mathcal{L}(x, \lambda^*) - \mathcal{L}(x^*, \lambda^*) = f(x) - f(x^*) + (\lambda^*)^T g(x)$, (4.2) results in

$$\begin{aligned} & f(x) - f(x^*) - \Sigma ||x - x^*||^2 + (\lambda)^T g(x) \\ &= f(x) - f(x^*) - \Sigma ||x - x^*||^2 + (\lambda^*)^T g(x) + (\lambda - \lambda^*)^T g(x) \\ &\le (\lambda - \lambda^*)^T g(x) \le c_{1g} c_d (||x - x^*||^2), \end{aligned}$$

and thus

$$\begin{aligned}f(x) - f(x^*) &\leq (\Sigma + c_{1g}c_d)||x - x^*||^2 - (\lambda)^T g(x)\\ &\leq (\Sigma + c_{1g}c_d)||x - x^*||^2 + \lambda^T g^-(x).\end{aligned}$$

Therefore

$$f(x) + c_\phi P(x) - f(x^*) \leq (\Sigma + c_{1g}c_d)||x - x^*||^2 + c_\phi P(x) + \lambda^T g^-(x).$$

The conclusion of the lemma follows by choosing $c_5 = \max\{\Sigma + c_{1g}c_d, c_\phi, 1\}$. □

4.2. Nondifferentiable exact penalty algorithms and the linear convergence theorem. The linearization algorithm [4, p. 372] has the following form:

(1) Set $k = 0$ and choose x^0.
(2) Compute d^k from (1.12).
(3) Choose α^k from a line search procedure, and set $x^{(k+1)} = x^k + \alpha^k d^k$.
(4) Set $k = k + 1$ and return to step 2.

The stepsize α^k is chosen by one of the following procedures [4, p. 372].

(a) *Minimization rule.* Here α^k is chosen such that

$$\phi(x^k + \alpha^k d^k) = \min_{\alpha \geq 0}\{\phi(x^k + \alpha d^k)\}.$$

(b) *Limited minimization rule.* Here a fixed scalar $s > 0$ is selected, and α^k is chosen such that

$$\phi(x^k + \alpha^k d^k) = \min_{\alpha \in [0,\, s]}\{\phi(x^k + \alpha d^k)\}.$$

(c) *Armijo rule.* Here fixed scalars s, τ, and σ with $s > 0$, $\tau \in (0, 1)$, and $\sigma \in (0, \frac{1}{2})$ are chosen and we set $\alpha^k = \tau^{m_k} s$, where m_k is the first nonnegative integer m for which

$$\phi(x^k) - \phi(x^k + \tau^m s d^k) \geq \sigma \tau^m s (d^k)^T d^k.$$

It can be shown that the Armijo rule yields a stepsize after a finite number of iterations.

The following theorem establishes the convergence properties of the linearization algorithm. The global convergence properties, established in [3, Prop. 4.3.3], are also stated here for completeness.

THEOREM 4.4. *Let x^k be a sequence generated by the linearization algorithm, where the stepsize α^k is chosen by the minimization rule, limited minimization rule, or Armijo rule. Then any accumulation point of the sequence x^k is a stationary point of $\phi(x) = f(x) + c_\phi P(x)$. If $x^k \to x^*$, where x^* is a strict local minimum of the problem* (1.2) *satisfying the local quadratic growth* (1.1)*, the MFCQ* (1.5)*, and with c_ϕ satisfying* (1.17)*, then $\phi(x^k) \to \phi(x^*)$ Q-linearly and $x^k \to x^*$ R-linearly.*

Proof. The global convergence properties are proved in [3, Prop. 4.3.3]. For the rate of convergence, we use the argument from section 1.1 that near x^*, (1.12) produces the same direction as (1.14), for which we proved our estimates. We prove the linear convergence statement only for the Armijo rule, the proof being similar for the other stepsize selection mechanisms. By Lemma 4.2

$$\begin{aligned}\phi(x^k) - \phi(x^k + \alpha d^k) &\geq \alpha \tfrac{1}{2}\left((d^k)^T d^k + \tfrac{\gamma}{2} P(x^k) + (\lambda^k)^T g^-(x^k)\right)\\ &\geq \alpha \tfrac{1}{2}(d^k)^T d^k > \sigma \alpha (d^k)^T d^k\end{aligned}$$

$\forall\ \alpha \in [0, \bar{\alpha}]$. Since m_k is the smallest integer m for which

$$\phi(x^k) - \phi(x^k + \tau^m s d^k) \geq \sigma \tau^m s (d^k)^T d^k,$$

it follows that $\tau^m s \geq \tau \bar{\alpha}$. This therefore ensures that the stepsize is at least $\tau\bar{\alpha}$ for k sufficiently large. As a result of Lemma 4.2, we have that

$$\phi(x^k) - \phi(x^{(k+1)}) \geq c_2 \tau \bar{\alpha} \left(||d^k||^2 + P(x^k) + (\lambda^k)^T g^-(x^k) \right). \tag{4.3}$$

On the other hand, by Lemma 4.3 we have that

$$\phi(x^k) - \phi(x^*) \leq c_5 \left(P(x^k) + ||x^k - x^*||^2 + ||d^k||^2 + (\lambda^k)^T g^-(x^k) \right).$$

By Theorem 2.6 and the previous relation it follows that there exists $c_6 = c_5(1 + \frac{1}{c_\sigma})$ such that

$$\begin{aligned} \phi(x^k) - \phi(x^*) &\leq c_6((\lambda^k)^T g^-(x^k) + P(x^k) + ||d^k||^2) \\ &\leq \frac{c_6}{\tau\bar{\alpha}c_2}(\phi(x^k) - \phi(x^{k+1})) = \delta(\phi(x^k) - \phi(x^{k+1})) \\ &= \delta(\phi(x^k) - \phi(x^*)) - \delta(\phi(x^{(k+1)}) - \phi(x^*)) \end{aligned}$$

by using (4.3) and where $\delta = \frac{c_6}{\tau\bar{\alpha}c_2}$. After some obvious manipulation, it follows that

$$\delta(\phi(x^{(k+1)}) - \phi(x^*)) \leq (\delta - 1)(\phi(x^k) - \phi(x^*)),$$

which proves the Q-linear convergence [23] of the sequence $\phi(x^k)$ to $\phi(x^*)$ with a linear rate of at most $\frac{\delta-1}{\delta}$. Therefore

$$\limsup_{k\to\infty} \sqrt[k]{\phi(x^k) - \phi(x^*)} \leq \frac{\delta - 1}{\delta}.$$

From Lemma 2.2

$$\phi(x^k) - \phi(x^*) \geq \frac{\sigma}{2}||x^k - x^*||^2.$$

Therefore

$$\limsup_{k\to\infty} \sqrt[k]{||x^k - x^*||} \leq \left(\frac{\delta - 1}{\delta}\right)^{\frac{1}{2}},$$

which proves the R-linear convergence [23] to 0 of the sequence $x^k - x^*$. The proof is complete. □

5. Numerical experiments with degenerate NLP.

We experimented with several nonlinear programming packages on the example from section 3. Certainly, comparing the behavior of nonlinear programming algorithms on a unique degenerate example cannot result in a complete characterization of the relative performance. Nevertheless, it may be of interest to determine whether methods using augmented Lagrangians will really encounter problems when solving an example without a positive semidefinite augmented Lagrangian. We also desire to validate the theoretical conclusions of the preceding sections.

Table 5.1
Rates of convergence for the L_∞ penalty algorithm.

Iteration	$\frac{\phi(x^k)-\phi(x^*)}{\phi(x^{k+1})-\phi(x^*)}$
4	4.00
9	4.00
14	3.99
19	3.99
24	4.00
27	4.00

We have shifted the origin for our example to avoid one-step convergence of algorithms that start at $(0,0,0)$ by default. The algebraic form of the example is

$$\begin{array}{lll} & \min_{(x,y,z)} z & \\ \text{subject to} & g_0(x,y,z) = (x-1)^2 - 2(y-1)^2 - z & \leq 0, \\ & g_1(x,y,z) = -\frac{1}{2}((x-1)^2 + (y-1)^2) + 3(x-1)(y-1) - z & \leq 0, \\ & g_2(x,y,z) = -2(x-1)^2 + (y-1)^2 - z & \leq 0, \\ & g_3(x,y,z) = -\frac{1}{2}((x-1)^2 + (y-1)^2) - 3(x-1)(y-1) - z & \leq 0. \end{array} \tag{5.1}$$

From our analysis, we have that $w^* = (1,1,0)$ is a minimum satisfying the QG condition (1.1) with $z - 0 \geq \frac{1}{8}((x-1)^2 + (y-1)^2 + z^2)$ for feasible (x,y,z) near w^*.

Among the solvers we used, MINOS [21] and SNOPT [13] use quasi-Newton methods that do not require second-order derivatives of the constraints. They also use an augmented Lagrangian as a merit function. DONLP2 [27] solves a linear system instead of a QP at each iteration and uses an L_1 penalty function. LANCELOT [7] uses an augmented Lagrangian technique in conjunction with a trust region. FilterSQP [10] also uses a trust region approach but with a special classification of the relative merits of the iterates instead of a penalty or merit function. LOQO [28] is an interior-point approach. Finally, LINF is an ad hoc Matlab implementation of the L_∞ exact penalty function described in the preceding section, with an Armijo rule. The latter algorithm is started at $(0,0,0)$. All runs, except for the L_∞ penalty and FilterSQP algorithms, were done on the NEOS server [22], where additional documentation can be found for all of the above solvers.

For such a small example the time of execution is not relevant in comparing the behavior of the solvers. Since the solution of the problem is known, we chose as a criterion for comparison the best achievable solution. We set all relevant tolerances to $1\text{e}-16$, via the AMPL interface of NEOS. Smaller tolerances may interfere with the machine precision, though most of the solvers gave comparable answers even when the tolerances were set to $1\text{e}-20$. Larger tolerances ($1\text{e}-12$ to $1\text{e}-15$) again resulted in very similar results. We also changed the iteration limit for LOQO and both runs are reported. DONLP2 converged to all digits in the mantissa with the default settings, and no change was made.

Table 5.1 shows the ratios $(\phi(x^k) - \phi(x^*))/(\phi(x^{k+1}) - \phi(x^*))$ at various iterations for our implementation LINF. All are close to 4.00, which is consistent with the Q-linear convergence claim for $\phi(x)$ from Theorem 4.4.

The selected μ updates from Table 5.2 show that LANCELOT decreases successively the value of the penalty parameter (by 16 orders of magnitude), until it stops with the message "Step size too small." This was indeed one of the alternatives allowed by our analysis in section 3.1 ($\mu \to 0$). This is an undesirable outcome since the

TABLE 5.2
Reduction of the penalty parameter μ for LANCELOT.

Iteration	Penalty parameter μ	Trust region radius $\|\|\|\|_\infty$
16	$1e-2$	3.81 e $-$ 02
43	$1e-4$	1.1 e $-$ 02
85	$1e-6$	1.35 e $-$ 03
141	$1e-8$	4.22 e $-$ 05
203	$1e-10$	5.28 e $-$ 06
241	$1e-12$	1.70 e $-$ 06
268	$1e-14$	1.93
283	$1e-16$	4.41 e02
323	$1e-18$	2.19 e04
336	STOP	

TABLE 5.3
Best achievable solution for various nonlinear solvers on the problem (5.1).

Solver	$\|\|x^{final}-x^*\|\|_2$	Iterations	Final message
DONLP2	$1.45e-16$	4	Success
FilterSQP	$5.26e-09$	28	Convergence
LANCELOT	$8.65e-07$	336	Step size too small
LINF	$1.05e-08$	28	Step size too small
LOQO	$1.60e-07$	200	Iteration limit
LOQO	$5.50e-07$	1000	Iteration limit
MINOS	$4.76e-06$	27	Current point cannot be improved
SNOPT	$3.37e-07$	3	Optimal Solution Found

subproblems (3.5) may become harder to solve. The values of all parameters except μ were read before a penalty update.

The results for all runs are illustrated in Table 5.3. It can be seen that the solvers that use augmented Lagrangians MINOS, SNOPT, and LANCELOT exhibit an error of at least one order of magnitude larger compared to all other algorithms. However, one would expect that SNOPT and MINOS would have at least as good a behavior as LINF if they use a different merit function, since the nature of the QP solved is very similar to (1.14). Increasing the iteration limit in LOQO did not result in a better outcome. It is interesting to note that the outcome in FilterSQP and LINF differ by only a factor of 2 in the same number of iterations, though FilterSQP uses second-order information whereas LINF does not. Both LINF and FilterSQP solve QPs at each iteration. DONLP2 has a remarkable behavior, though further investigation is necessary to determine whether this has some general implications.

We cannot draw a definitive conclusion from one example. However, based on this experiment and our theoretical developments, there seems to be an adverse bias for methods using augmented Lagrangians on degenerate NLPs, like the one above. We are not advocating the use of LINF on general NLP, since its similarity to steepest descent makes it very sensitive to ill-conditioning. But the fact that it gives an outcome comparable to the one of solvers using second-order information shows that, for better results, a different way of incorporating second-order derivatives may be necessary.

6. Conclusions. In this work we analyze the behavior of nonlinear programs in the presence of constraint degeneracy: linear dependence of the gradients of the active constraints. The problems of interest exhibit minima with a QG property that satisfy the MFCQ. The novelty of our approach is that, while studying the SQP convergence properties, we do not assume the positive semidefiniteness of the Hessian of the Lagrangian on the critical cone for any of the feasible Lagrange multipliers. Our conditions are equivalent to a weak second-order sufficient condition [17, 26].

We prove that, under these assumptions, if the data of the problem are twice continuously differentiable, the target minimum will be an isolated stationary point of the NLP. We also show that, when started sufficiently close to the minimum, the L_∞ exact penalty SQPs induce Q-linear convergence of the values of the penalized objective $\phi(x) = f(x) + c_\phi P(x)$ and R-linear convergence of the iterates. This shows that such methods are robust with respect to constraint degeneracy.

We give an example of an NLP with a unique minimum that satisfies our conditions for which the Hessian of the Lagrangian is not positive semidefinite on the critical cone for any feasible choice of the multipliers. The direct consequence of this fact is that there is no augmented Lagrangian that will be positive semidefinite at the solution. Therefore, Lagrange multiplier algorithms will have to drive the penalty parameter to zero for such examples unless the trust region is active even at convergence.

We provide our computational experience with this small nonlinear program. As a criteria for comparison we used the best achievable solution, which was obtained after tuning the parameters of the algorithms. We observed that, for this example, algorithms that use augmented Lagrangians resulted in errors of one order of magnitude or larger when compared to the other approaches. The Lagrange multiplier package that we used (LANCELOT [7]) was confined to decrease substantially the value of the penalty parameter (16 orders of magnitude), which is one of the outcomes allowed by our analysis. The linear convergence results concerning the L_∞ penalty function were also validated by our experiments.

We believe that attempting to develop a convergence theory in the absence of the usual second-order conditions is interesting because it may result in algorithms that are more robust by virtue of the fact that their properties depend on fewer assumptions. However, how to improve on the current results, and especially how to define reliable variants of the Newton method (if possible) for this case, is a subject of future research.

Acknowledgments. Thanks to Stephen Wright, Jorge Moré, and Danny Ralph for the many discussions on the subject. David Gay, Sven Leyffer, and Chi-Jen Lin have kindly provided me information and support for the numerical examples. I am grateful to the referees for their suggestions and especially for pointing out reference [16].

REFERENCES

[1] M. Anitescu, *On the Rate of Convergence of Sequential Quadratic Programming with Nondifferentiable Exact Penalty Function in the Presence of Constraint Degeneracy*, Preprint ANL/MCS-P760-0699, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1999.

[2] K. Atkinson, *An Introduction to Numerical Analysis*, John Wiley & Sons, New York, 1988.

[3] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[4] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[5] J. F. Bonnans, *Local analysis of Newton-type methods for variational inequalities and nonlinear programming*, Appl. Math. Optim., 29 (1994), pp. 161–186.

[6] J. F. Bonnans and A. Ioffe, *Second-order sufficiency and quadratic growth for nonisolated minima*, Math. Oper. Res., 20 (1995), pp. 801–819.

[7] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *LANCELOT: A Fortran Package for Large-Scale Nonlinear Optimization*, Springer-Verlag, Berlin, 1992.

[8] A. V. Fiacco, *Introduction to Sensitivity and Stability Analysis in Nonlinear Programming*, Academic Press, New York, 1983.

[9] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons, Chichester, UK, 1987.

[10] R. Fletcher and S. Leyffer, *Nonlinear Programming without a Penalty Function*, Numerical Analysis Report NA/171, Department of Mathematics, University of Dundee, UK, 1997.

[11] J. Gauvin, *A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming*, Math. Programming, 12 (1977), pp. 136–138.

[12] J. Gauvin and J. W. Tolle, *Differential stability in nonlinear programming*, SIAM J. Control Optim., 15 (1977), pp. 294–311.

[13] P. E. Gill, W. Murray, and M. A. Saunders, *User's Guide for SNOPT* 5.3*: A Fortran Package for Large-Scale Nonlinear Programming*, Report NA 97-5, Department of Mathematics, University of California, San Diego, 1997.

[14] W. W. Hager, *Stabilized sequential quadratic programming*, Comput. Optim. Appl., 12 (1999), pp. 253–273.

[15] W. W Hager and M. S. Gowda, *Stability in the presence of degeneracy and error estimation*, Math. Programming, 85 (1999), pp. 181–192.

[16] A. J. Hoffman, *On approximate solutions of systems of linear inequalities*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 263–265.

[17] A. D. Ioffe, *Necessary and sufficient conditions for a local minimum.* III*: Second order conditions and augmented duality*, SIAM J. Control Optim., 17 (1979), pp. 266–288.

[18] A. Ioffe, *On sensitivity analysis of nonlinear programs in Banach spaces: The approach via composite unconstrained optimization*, SIAM J. Optim., 4 (1994), pp. 1–43.

[19] O. L. Mangasarian and S. Fromovitz, *The Fritz John necessary optimality conditions in the presence of equality constraints*, J. Math. Anal. Appl., 17 (1967), pp. 34–47.

[20] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[21] B. Murtagh and M. A. Saunders, *MINOS* 5.0 *User's Guide*, Technical Report SOL 83-20, Stanford University, Stanford, CA, 1983.

[22] *The NEOS Guide*, http://www.mcs.anl.gov/otc/Guide.

[23] J. Ortega and W. Rheinboldt, *Iterative Solutions of Nonlinear Equations in Several Variables*, Academic Press, New York, 1972.

[24] D. Ralph and S. J. Wright, *Superlinear Convergence of an Interior-Point Method Despite Dependent Constraints*, Preprint ANL/MCS-P622-1196, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1996.

[25] S. M. Robinson, *Generalized equations and their solutions, Part* II*: Applications to nonlinear programming*, Math. Programming Stud., 19 (1980), pp. 200–221.

[26] A. Shapiro, *Sensitivity analysis of nonlinear programs and differentiability properties of metric projections*, SIAM J. Control Optim., 26 (1988), pp. 628–645.

[27] P. Spelucci, *An SQP Method for General Nonlinear Programs Using Only Equality Constrained Subproblems*, http://www.mathematik.tu-darmstadt.de/ags/ag8/spellucci/.

[28] R. J. Vanderbei, *LOQO: An interior-point code for quadratic programming*, Optim. Methods Softw., 12 (1999), pp. 451–484.

[29] S. J. Wright, *Superlinear convergence of a stabilized SQP method to a degenerate solution*, Comput. Optim. Appl., 11 (1998), pp. 253–275.

[30] S. J. Wright, *Modifying SQP for Degenerate Problems*, Preprint ANL/MCS-P699-1097, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1997.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1136–1148

A SHARP LAGRANGE MULTIPLIER RULE FOR NONSMOOTH MATHEMATICAL PROGRAMMING PROBLEMS INVOLVING EQUALITY CONSTRAINTS*

X. WANG† AND V. JEYAKUMAR†

Abstract. It is shown that a Lagrange multiplier rule that uses approximate Jacobians holds for mathematical programming problems involving Lipschitzian functions, finitely many equality constraints, and convex set constraints. It is sharper than the corresponding Lagrange multiplier rules for the convex-valued subdifferentials such as those of Clarke [*Optimization and Nonsmooth Analysis*, 2nd ed., SIAM, 1990] and Michel and Penot [*Differential Integral Equations*, 5 (1992), pp. 433–454]. The Lagrange multiplier result is obtained by means of a controllability criterion and the theory of fans developed by A. D. Ioffe [*Math. Oper. Res.*, 9 (1984), pp. 159–189, *Math. Programming*, 58 (1993), pp. 137–145]. As an application, necessary optimality conditions are derived for a class of constrained minimax problems. An example is discussed to illustrate the nature of the multiplier rule.

Key words. generalized Jacobians, nonsmooth analysis, sharp Lagrange mutipliers, equality constraints, minimax problems

AMS subject classifications. 49A52, 90C30, 26A24

PII. S1052623499354540

1. Introduction.

In this paper we study mathematical programming problems of the form

$$\text{(PE)}\qquad \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \le 0, \quad i = 1, \dots, p, \\ & f_i(x) = 0, \quad i = p+1, \dots, m, \\ & x \in Q, \end{array}$$

where $f_0, \dots, f_m : \mathbb{R}^n \to \mathbb{R}$ are (not necessarily differentiable) locally Lipschitzian functions and Q is a closed convex subset of $\mathbb{R}^n$. We present a Lagrange multiplier rule which includes many smooth and nonsmooth multiplier rules as corollaries. It uses the approximate Jacobian of Jeyakumar and Luc [13] which has recently been shown to enjoy rich (and often exact) calculus for continuous maps, produce sharp conditions for Lipschitzian maps, and offer a flexible approach for certain analysis and applications as it can suitably be chosen for specific applications (see [2, 12, 13, 14, 15, 16]). This approximate Jacobian is connected to the Gâteaux derivative in the sense that a map between finite dimensional spaces is Gâteaux differentiable at a point if and only if it admits an approximate Jacobian which is single-valued at the point.

Studies of nonsmooth optimization had led in recent years to the development of various generalized gradients and associated Lagrange multiplier rules for mathematical programming problems. In particular, the generalized gradients of Clarke [1], Ioffe [8], Michel and Penot [17], and Mordukhovich [18] have proved to be potent and powerful tools in mathematical programming. On the other hand, it has long been

*Received by the editors March 31, 1999; accepted for publication (in revised form) December 17, 1999; published electronically June 13, 2000.

http://www.siam.org/journals/siopt/10-4/35454.html

†Department of Applied Mathematics, University of New South Wales, UNSW Sydney NSW2052, Australia (xwang@maths.unsw.edu.au, jeya@maths.unsw.edu.au). The research of the second author was partially supported by a grant from the Australian Research Council.

recognized that a sharp Lagrange multiplier rule for a nonsmooth optimization problem is vital for obtaining accurate and more selective first-order necessary conditions. It is also one of the chief reasons behind the development of smaller convex-valued subdifferentials such as those introduced by Michel and Penot [17] and Treiman [21], and nonconvex subdifferentials such as those studied by Ioffe [9], Mordukhovich [18], and Treiman [22].

There have been many contributors to the extension of the classical Lagrange multiplier rules to nonsmooth mathematical programming problems involving finitely many inequalities and no equality constraints (see [1, 3, 4, 11, 17] and other references therein). However, such results for problems with equality and set constraints have so far been limited. Recently, Ioffe [10] has made a valuable contribution to the subject by establishing a Lagrange multiplier rule with small convex-valued subdifferentials for problems involving both finitely many equality and convex set constraints. In fact, Ioffe's approach to the Lagrange multiplier theory, which uses controllability criteria and the theory of fans, provides the inspiration for the present work. It is shown first that an approximate Jacobian of a locally Lipschitzian map produces a fan which is a prederivative of the map [7, 8]. The multiplier result is then obtained by means of the controllability criterion employed by Ioffe [8].

Our discussion proceeds as follows. In section 2 we review the generalized calculus of approximate Jacobians and present a sharp generalized chain rule formula for differentiation of composite functions. In section 3 we relate approximate Jacobians to the ideas of fans and prederivatives, establish a Lagrange multiplier rule for (PE) and compare with other corresponding multiplier rules. Section 4 provides an application of the multiplier rule to a class of nonsmooth minimax problems.

2. Sharp nonsmooth calculus.

In this section we present generalized calculus of approximate Jacobians. We assume throughout the paper that F is a map from $\mathbb{R}^n$ into $\mathbb{R}^m$ with the components $(f_1, \ldots, f_m)$. For each $v \in \mathbb{R}^m$ the composite function $(vF) : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$(vF)(x) = \langle v,\ F(x)\rangle = \sum_{i=1}^{m} v_i f_i(x).$$

The upper and lower Dini directional derivatives of $f : \mathbb{R}^n \to \mathbb{R}$ at x in the direction $u \in \mathbb{R}^n$ are defined by

$$f^+(x,u) := \limsup_{t \downarrow 0} \frac{f(x+tu) - f(x)}{t}$$

and

$$f^-(x,u) := \liminf_{t \downarrow 0} \frac{f(x+tu) - f(x)}{t}.$$

Note, for instance, that if $f : \mathbb{R}^n \to \mathbb{R}$ is Lipschitzian with Lipschitz constant k', then the upper Dini directional derivative $f^+(x,\cdot)$ is also Lipschitzian with the same Lipschitz constant k'. We denote by $\mathbb{R}^{m\times n}$ the space of all $(m \times n)$ matrices. The *convex hull* and the *closed convex hull* of a set A are denoted by $co(A)$ and $\overline{\text{co}}\,(A)$, respectively.

Definition 2.1 (see [16, 13]). *The map $F : \mathbb{R}^n \to \mathbb{R}^m$ admits an* approximate Jacobian *$\partial^* F(x)$ at $x \in \mathbb{R}^n$ if $\partial^* F(x) \subseteq \mathbb{R}^{m\times n}$ is closed and for each $v \in \mathbb{R}^m$,*

$$(vF)^+(x,u) \le \sup_{M \in \partial^* F(x)} \langle v, Mu\rangle \ \ \forall u \in \mathbb{R}^n. \tag{2.1}$$

We allow infinite values on both sides of the inequalities in (2.1) so that the Dini directional derivatives may attain infinite values. A matrix M of $\partial^* F(x)$ is called an *approximate Jacobian matrix* of F at x. When $m = 1$, the condition (2.1) is equivalent to the conditions that

$$f^+(x,u) \leq \sup_{\xi \in \partial^* f(x)} \langle \xi, u \rangle \quad \& \quad f^-(x,u) \geq \inf_{\xi \in \partial^* f(x)} \langle \xi, u \rangle.$$

In this case the set $\partial^* f(x)$ is called the *Jeyakumar–Luc (J–L) subdifferential* of f at x.

Note that the Clarke subdifferential $\partial_C f(x)$ [1], Michel–Penot subdifferential $\partial^\diamond f(x)$ [17], Mordukhovich subdifferential $\partial_M f(x)$ [19], and Treiman subdifferential $\partial_T f(x)$ [22] are examples of the J–L subdifferential for a locally Lipschitzian function (see [14]). However, the following example shows that the J–L subdifferential may be "smaller" than these subdifferentials for locally Lipschitzian functions.

Example 2.1. Let $f(x_1, x_2) = |x_1| - |x_2|$. Then it is easy to verify that

$$\partial^* f(0,0) = \left\{ \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ 1 \end{pmatrix}^T \right\},$$

$$\partial_T f(0,0) = \partial_M f(0,0) = \left\{ \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ 1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ 1 \end{pmatrix}^T \right\},$$

$$\partial^\diamond f(0,0) = \partial_C f(0,0) = co \left\{ \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ 1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ 1 \end{pmatrix}^T \right\}.$$

Note that $co\partial^* f(0,0) \subseteq \partial^\diamond f(0,0) = \partial_C f(0,0) = co\partial_T f(x) = co\partial_M f(x)$.

It was shown in [12] that a (not necessarily Lipschitzian) function $f : \mathbb{R}^n \to \mathbb{R}$ is Gâteaux differentiable at x with the Gâteaux derivative $\nabla f(x)$ if and only if f admits a J–L subdifferential $\partial^* f$ which is single-valued at x with $\partial^* f(x) = \{\nabla f(x)\}$. The J–L subdifferentials enjoy useful calculus rules as illustrated in the following propositions.

Proposition 2.2 (see [16, 14]). *Let $f : \mathbb{R}^n \to \mathbb{R}$ and let $0 \neq \alpha \in \mathbb{R}$. If $\partial^* f(x)$ is a J–L subdifferential of f at $x \in \mathbb{R}^n$, then $\alpha \partial^* f(x)$ is a J–L subdifferential of αf at x.*

Proposition 2.3 (see [16, 14]). *Let $f_i : \mathbb{R}^n \to \mathbb{R}$ for $i = 1, 2$. Suppose that for each $i = 1, 2$, $\partial^* f_i(x)$ is a J–L subdifferential of f_i at x. Then the set $\partial^* f_1(x) + \partial^* f_2(x)$ is a J–L subdifferential of $f := f_1 + f_2$ at x.* For further details, see [13, 14, 16]. The following example shows that the J–L subdifferential $\partial^* f_1(x) + \partial^* f_2(x)$ may contain another J–L subdifferential of $f_1 + f_2$.

Example 2.2. Let $f_1(x,y) = |x| - |y|, \quad f_2(x,y) = |y| - |x|$. Then $f_1(x,y) + f_2(x,y) = 0$. Then it is easy to verify that

$$\partial^* f_1(0,0) = \partial^* f_2(0,0) = \left\{ \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ 1 \end{pmatrix}^T \right\};$$

$$\partial^* (f_1 + f_2)(0,0) = \left\{ \begin{pmatrix} 0 \\ 0 \end{pmatrix}^T \right\}.$$

Hence both $\partial^*(f_1+f_2)(0,0)$ and $\partial^* f_1(0,0)+\partial^* f_2(0,0)$ are J–L subdifferentials of f_1+f_2 at $(0,0)$ with $\partial^*(f_1+f_2)(0,0)\subset \partial^* f_1(0,0)+\partial^* f_2(0,0)$.

The following proposition provides a slightly stronger form of the mean value theorem of Jeyakumar and Luc [13].

PROPOSITION 2.4 (mean value inequality). *Let $a, b\in\mathbb{R}^n$ and let $f:\mathbb{R}^n\to\mathbb{R}$ be continuous on $\mathbb{R}^n$. Assume that for each $x\in[a,b]$, f admits a J–L subdifferential $\partial^* f(x)$ at x. Then for each $v\in\mathbb{R}^m$ there exists $t_0\in(0,1)$ such that*

$$\inf_{\xi\in\partial^* f(a+t_0(b-a))}\langle\xi, b-a\rangle \le f(b)-f(a)\le \sup_{\xi\in\partial^* f(a+t_0(b-a))}\langle\xi, b-a\rangle.$$

Recall that the set-valued map $\partial^* F: x \Longrightarrow \partial^* F(x)$ is *locally bounded* at x_0 if there exist positive constants α and δ such that

$$\sup\{\|M\| \ :\ M\in\partial^* F(x), x\in B_\delta(x_0)\}\le\alpha,$$

where $\|M\|$ is a matrix norm. Note that a continuous map F admits a locally bounded approximate Jacobian at a point if and only if F is locally Lipschitzian at the point [13]. Moreover, if $f:\mathbb{R}^n\to\mathbb{R}$ and $\partial^* f(x)$ is a J–L subdifferential of f at $x\in\mathbb{R}^n$, and if f attains its extremum at x, then

$$0\in\overline{\text{co}}\,(\partial^* f(x)). \tag{2.2}$$

For details see [14]. We now see that J–L subdifferentials enjoy a general chain rule involving Gâteaux differentiable maps.

THEOREM 2.5 (generalized chain rule). *Let $f:\mathbb{R}^m\to\mathbb{R}$ be locally Lipschitzian and let $F:\mathbb{R}^n\to\mathbb{R}^m$ be Gâteaux differentiable at $x\in\mathbb{R}^n$. If $\partial^* f(x)$ is the J–L subdifferential of f at x, then $\partial^* f(F(x))\nabla F(x)$ is a J–L subdifferential of the composite function $f\circ F$ at x.*

Proof. Since F is Gâteaux differentiable at x, for all sufficiently small $t>0$,

$$F(x+th)=F(x)+t\nabla F(x)h+o(t),$$

where

$$\frac{\|o(t)\|}{t}\to 0, \qquad \text{as } t\to 0.$$

Now from the Lipschitzian property of f at $F(x)$, we get

$$t^{-1}\|f(F(x)+t\nabla F(x)h+o(t))-f(F(x)+t\nabla F(x)h)\|\le t^{-1}K\|o(t)\|,$$

where K is a Lipschitzian constant for f near $F(x)$. So,

$$\lim_{t\to 0}\|f(F(x)+t\nabla F(x)h+o(t))-f(F(x)+t\nabla F(x)h)\|=0.$$

Hence we deduce that for each $\alpha\in\mathbb{R}$

$$\begin{aligned}
&\limsup_{t\to 0^+} t^{-1}[\alpha\, f(F(x+th))-\alpha\, f(F(x))]\\
&\quad=\limsup_{t\to 0^+} t^{-1}[\alpha\, f(F(x)+t\nabla F(x)h+o(t))-\alpha\, f(F(x)+t\nabla F(x)h)]\\
&\qquad+\limsup_{t\to 0^+} t^{-1}[\alpha\, f(F(x)+t\nabla F(x)h)-\alpha\, f(F(x))]\\
&\quad=\limsup_{t\to 0^+} t^{-1}[\alpha\, f(F(x)+t\nabla F(x)h)-\alpha\, f(F(x))]\\
&\quad\le \sup_{\xi\in\partial^* f(F(x))}\langle\alpha\xi, \nabla F(x)h\rangle.
\end{aligned}$$

This inequality ensures that $\partial^* f(F(x))\nabla F(x)$ is a J–L subdifferential of $f \circ F$ at x. □

Note that $\overline{\mathrm{co}}\,(\partial^* f(F(x)))\nabla F(x)$ is also a J–L subdifferential for $(f \circ F)$ at x since

$$\partial^* f(F(x))\nabla F(x) \subseteq \overline{\mathrm{co}}\,(\partial^* f(F(x))\nabla F(x)) = \overline{\mathrm{co}}\,(\partial^* f(F(x)))\nabla F(x).$$

So our chain rule yields the corresponding result in [20].

3. Generalized Lagrange multiplier rules. We begin this section by examining the relationship between fans and approximate Jacobians. Recall that a set-valued map A from $\mathbb{R}^n$ into $\mathbb{R}^m$ is called a *bounded fan* if

(i) $A(x)$ is a nonempty closed convex set for each $x \in \mathbb{R}^n$;
(ii) $A(\lambda x) = \lambda A(x)$ for any $x \in \mathbb{R}^n$ and any $\lambda > 0$;
(iii) $A(0) = \{0\}$;
(iv) there is a $k > 0$ such that $\|y\| \leq k\|x\|$ if $y \in A(x)$;
(v) $A(x' + x'') = A(x') + A(x'')\ \forall\ x', x'' \in \mathbb{R}^n$.

A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be *sublinear* if it is convex and positively homogeneous. A function $q : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ is said to be *bounded bisublinear* if it is sublinear in each argument, and there exists $K_0 > 0$ such that for each $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$, $q(y^*, x) \leq K_0\|y^*\|\|x\|$. For the fan A from $\mathbb{R}^n$ to $\mathbb{R}^m$, define $s : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}$ by

$$s(y^*, x) = \sup_{y \in A(x)} \langle y^*, y\rangle, \qquad y^* \in \mathbb{R}^m.$$

The function $s(.,.)$ is called the support function of the fan A. It is easy to see that $s(.,.)$ is a bounded bisublinear function. Conversely, if $s(.,.)$ is a bounded bisublinear function, then it is the support function of some fan A which is defined by

$$A(x) = \{y \in \mathbb{R}^m : \langle y^*, y\rangle \leq s(y^*, x)\ \ \forall y^* \in \mathbb{R}^m\}.$$

We refer to [7, 8] for further details about fans.

Let $F(\cdot)$ be a map from a neighborhood of $\mathbb{R}^n$ into $\mathbb{R}^m$. A fan A from $\mathbb{R}^n$ into $\mathbb{R}^m$ is called a *weak prederivative* [7, 9] of F at x if for any subspace $L \subset \mathbb{R}^n$ and any $\epsilon > 0$ there is a $\delta > 0$ such that

$$F(x + h) \subset F(x) + A(h) + \epsilon\|h\|B_m \qquad \text{for } \|h\| < \delta,\ h \in L.$$

Here B_m is the unit ball in $\mathbb{R}^m$. Let $s(.,.)$ be the support function of the bounded fan A and let $K \subset \mathbb{R}^n$ be a closed convex cone. Then the quantity

$$C(A, K) = -\sup_{\|y^*\|=1} \inf_{\substack{\|h\|\leq 1 \\ h \in K}} s(y^*, h)$$

is called the *Banach constant* with respect to K. For a closed convex set S containing x, the *radial tangent cone* to S at x is defined as

$$T(S, x) = \bigcup_{\lambda > 0} \lambda(S - x),$$

and the *interior* of S is denoted by *int* S.

Lemma 3.1 (a controllability theorem [10]). *Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a continuous map and let $S \subset \mathbb{R}^n$ be a closed convex set containing x. Let A be a bounded fan from $\mathbb{R}^n$ into $\mathbb{R}^m$ which is a weak prederivative of F at x. If*

$$C(A, T(S, x)) = c > 0,$$

then for any $\delta > 0$

$$F(x) \in int\, F\left(S \bigcap (x+\delta B)\right),$$

where B is the unit ball in $\mathbb{R}^n$.

LEMMA 3.2. *Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be locally Lipschitzian which admits a locally bounded approximate Jacobian $\partial^* F$. Then the function*

$$s(v,u) = \max_{M \in \overline{\text{co}}\,(\partial^* F(x))} \langle v, Mu \rangle, \quad v \in \mathbb{R}^m,\ u \in \mathbb{R}^n,$$

is the support function of a fan which is a weak prederivative of F at x.

Proof. It is easy to see that $s(v,u) \leq k\|v\|\|u\|$ for some $k > 0$. It is also obvious that $s(v,u)$ is sublinear as a function of v and u. Let $A(h) = \overline{\text{co}}\,(\partial^* F(x))h = \{Ah | A \in \overline{\text{co}}\,(\partial^* F(x))\}$. Then it follows that the set-valued map A is a fan and that $s(v,u)$ defined above is the support function of the fan A. The map A is a weak prederivative of F at x if we can show that for every $\epsilon > 0$ there is a $\delta > 0$ such that

$$\langle v, F(x+u) - F(x) \rangle \leq (v \circ F)^+(x;u) + \epsilon\|u\|, \tag{3.1}$$

for each $\|v\| = 1, \|u\| \leq \delta$. This will be proven later. First we show that A is a weak prederivative of F at x using (3.1). This follows from the fact that

$$(v \circ F)^+(x;u) \leq \max_{M \in \overline{\text{co}}\,(\partial^* F(x))} \langle v, Mu \rangle = s(v,u).$$

In fact, for any fixed u, (3.1) gives us that for each v,

$$\langle v, F(x+u) - F(x) \rangle \leq s(v,u) + \epsilon\|v\|\|u\|, \tag{3.2}$$

which by the separation theorem yields the required inclusion that

$$F(x+u) - F(x) \in A(u) + \epsilon\|u\|B.$$

We now establish (3.1). Assume to the contrary that there exist an $\epsilon > 0$ and sequences $\{v_n\}$ and $\{u_n\}$ such that $\|v_n\| = 1, u_n \to 0$ and

$$\langle v_n, F(x+u_n) - F(x) \rangle > (v_n \circ F)^+(x;u_n) + \epsilon\|u_n\|. \tag{3.3}$$

Set $t_n = \|u_n\|, e_n = u_n/t_n$. We can assume that both e_n and v_n converge to certain $e \in \mathbb{R}^n, v \in \mathbb{R}^m$ with $\|e\| = \|v\| = 1$. Let k' be a Lipschitz constant of F. Then it follows from (3.3) that

$$\begin{aligned}
\langle v, F(x+t_n e) - F(x) \rangle &= \langle v_n, F(x+u_n) - F(x) \rangle + \langle v - v_n, F(x+u_n) - F(x) \rangle \\
&\quad + \langle v, F(x+t_n e) - F(x+u_n) \rangle \\
&\geq \langle v_n, F(x+u_n) - F(x) \rangle - |\langle v - v_n, F(x+u_n) - F(x) \rangle| \\
&\quad - |\langle v, F(x+t_n e) - F(x+u_n) \rangle| \\
&\geq \langle v_n, F(x+u_n) - F(x) \rangle - t_n k'(\|v - v_n\| + \|e - e_n\|) \\
&> (v_n \circ F)^+(x;u_n) + \epsilon\|u_n\| - t_n k'(\|v - v_n\| + \|e - e_n\|) \\
&= t_n[(v_n \circ F)^+(x;e_n) + \epsilon - k'(\|v - v_n\| + \|e - e_n\|)] \\
&\geq t_n[(v_n \circ F)^+(x;e) + \epsilon - 2k'(\|v - v_n\| + \|e - e_n\|)],
\end{aligned}$$

which is a contradiction in view of the definition of the upper Dini derivative. □

THEOREM 3.3 (generalized Lagrange multiplier rule). *For the problem* (PE), *let* $F(x) = (f_0(x), \ldots, f_m(x))$. *Assume that* F *admits a locally bounded approximate Jacobian at* $\overline{x} \in \mathbb{R}^n$. *If* $\overline{x}$ *is a minimizer of* (PE), *then there exist Lagrange multipliers* $\lambda_0 \geq 0, \ldots, \lambda_p \geq 0, \lambda_{p+1}, \ldots, \lambda_m$, *not all zero, such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \ldots, m,$$

$$0 \in \sum_{i=0}^{m} \lambda_i \overline{\mathrm{co}}\, \partial^* F(\overline{x})^T e_{i+1} + N(Q, \overline{x}), \tag{3.4}$$

where $e_i = [0, \ldots, 0, 1, 0, \ldots, 0]^T \in \mathbb{R}^{m+1}$ *is a unit vector.*

Note that the meaning of $\overline{\mathrm{co}}\, \partial^* F(\overline{x})^T e_{i+1}$, $i = 0, 1, \ldots, m$, in (3.4) is that every vector $z \in \overline{\mathrm{co}}\, \partial^* F(\overline{x})^T e_{i+1}$ can be represented as $z = M^T e_{i+1}$ for some $M \in \overline{\mathrm{co}}\, \partial^* F(\overline{x})$.

Proof. Assume for simplicity that $f_i(\overline{x}) = 0$, $i = 1, \ldots, p$. Let $Z = \mathbb{R}^n \times \mathbb{R}^{p+1}$ and

$$S = Q \times \mathbb{R}^{p+1}_+ = \{z = (x, a) \in Z : x \in Q, a_i \geq 0, i = 0, \ldots, p\} \qquad (a = (a_0, \ldots, a_p)).$$

Clearly, S is a closed convex set and the tangent cone to S at $\overline{z} = (\overline{x}, 0)$ is

$$T(S, \overline{z}) = T(Q, \overline{x}) \times \mathbb{R}^{p+1}_+,$$

where $T(Q, \overline{x})$ is the tangent cone to Q at $\overline{x}$ and $\mathbb{R}^{p+1}_+$ is the nonnegative orthant in $\mathbb{R}^{p+1}$. Let $Y = \mathbb{R}^{m+1}$ and let G be the mapping from Z into Y defined as follows:

$$(G(x, a))_i = \begin{cases} f_i(x) + a_i, & i = 0, \ldots, p, \\ f_i(x), & i = p+1, \ldots, m. \end{cases}$$

Then G is Lipschitzian and the set

$$\partial^* G(z) = \{(M, I) | M \in \partial^* F(x)\}$$

is an approximate Jacobian of G at z, where $I \in \mathbb{R}^{(m+1)\times(p+1)}$ is defined by

$$I = [e_1, \ldots, e_{p+1}],$$

where $e_i = [0, \ldots, 0, 1, 0, \ldots, 0]^T$.

Since $\overline{x}$ is a minimizer of (PE), $G(\overline{z}) = (f_0(\overline{x}), \ldots, f_m(\overline{x}))$ cannot be in the interior of $G(S \bigcap (\overline{z} + \lambda B_Z))$ for any $\lambda > 0$ because otherwise there would exit some point $y \in S \bigcap (\overline{z} + \lambda_0 B_Z)$ for some $\lambda_0 > 0$ such that

$$\begin{aligned} &f_0(y) < f_0(\overline{x}), \\ &f_i(y) = f_i(\overline{x}), \quad i = 1, \ldots, m, \end{aligned}$$

which implies that y is a feasible point and hence contradicts the hypothesis that $\overline{x}$ is a minimizer. By Lemma 3.1, we must have

$$C(A, T(S, \overline{z})) = 0,$$

for any weak prederivative A of G at $\overline{z}$. So, there is a v, $\|v\| = 1$ such that

$$s(v, u) \geq 0 \quad \forall u \in T(S, \overline{z}).$$

By Lemma 3.2, $s(v,u)$ is the support function of a weak prederivative of G at $\overline{z}$. We set $v=(\lambda_0,\dots,\lambda_m), u=(h,b)$. Then

$$\begin{aligned} s(v,u) &= \max\langle v, \partial^* G(\overline{z})u\rangle \\ &= \max\langle \partial^* G(\overline{z})^T v, u\rangle \\ &= \max\left\langle \sum_{i=0}^{m} \lambda_i \partial^* F(\overline{x})^T\, e_{i+1}, h\right\rangle + \sum_{i=0}^{p} \lambda_i b_i. \end{aligned}$$

Since $s(v,u)\geq 0$,

$$\max\left\langle \sum_{i=0}^{m} \partial^* F(\overline{x})^T\, e_{i+1}, h\right\rangle + \sum_{i=0}^{p} \lambda_i b_i \geq 0$$

$\forall\ h\in T(Q,\overline{x}), b=(b_0,\dots,b_p)\geq 0$. This can happen only if $\lambda_i\geq 0, i=1,\dots,p$.

Setting $b_i=0$, we have

$$\max\left\langle \sum_{i=0}^{m} \lambda_i \partial^* F(\overline{x})^T\, e_{i+1}, h\right\rangle \geq 0 \qquad \text{for } h\in T(Q,\overline{x}).$$

This gives us the inclusion

$$0\in \overline{\mathrm{co}}\left(\sum_{i=0}^{m} \lambda_i \partial^* F(\overline{x})^T\, e_{i+1}\right) + N(Q,\overline{x}).$$

Since $\partial^* F(\overline{x})$ is closed and bounded, we obtain the required inclusion (3.4)

$$0\in \sum_{i=0}^{m} \lambda_i \overline{\mathrm{co}}\, \partial^* F(\overline{x})^T\, e_{i+1} + N(Q,\overline{x}). \qquad \square$$

Let $F_1=(f_1,\dots,f_m)$ and $I\in\mathbb{R}^{(m+1)\times(p+1)}$ is defined as before. Then the set

$$\partial^* G(\overline{z}) = (\partial^* f_0(\overline{x}), \partial^* F_1(\overline{x}), I)$$

is an approximate Jacobian of G at z. Then we have the following corollary.

COROLLARY 3.4. *Let $\overline{x}$ be a solution to* (PE). *Assume that the functions $f_0,\dots,f_m$ admit locally bounded J–L subdifferentials $\partial^* f_i(\overline{x})$ at $\overline{x}$. Then there exist Lagrange multipliers $\lambda_0\geq 0,\dots,\lambda_p\geq 0,\lambda_{p+1},\dots,\lambda_m$, not all zero, such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i=1,\dots,m,$$

$$0\in \lambda_0 \overline{co}\,\partial^* f_0(\overline{x})^T + \sum_{i=1}^{m} \lambda_i \overline{co}\,\partial^* f_i(\overline{x})^T + N(Q,\overline{x}).$$

Proof. Since $\partial^* F(x) = \partial^* f_0(x)\times\cdots\times\partial^* f_m(x)$ is a locally bounded approximate Jacobian of F at x (see [16]), the conclusion follows from Theorem 3.3. $\square$

The standard form of the Lagrange multiplier rule for the Michel–Penot subdifferentials follows easily from Corollary 3.4.

COROLLARY 3.5. *If $\overline{x}$ is a solution to* (PE)*, then there exist multipliers $\lambda_0 \geq 0, \ldots, \lambda_p \geq 0, \lambda_{p+1}, \ldots, \lambda_m$, not all zero, such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \ldots, m,$$

$$0 \in \sum_{i=0}^{m} \lambda_i \partial^\diamond f_i(\overline{x})^T + N(Q, \overline{x}). \tag{3.5}$$

Proof. Choose $\partial^\diamond f_i(\overline{x})$ as the J–L subdifferential of f_i at $\overline{x}$. Then the conclusion follows easily from Corollary 3.4. □

In passing, observe that a slightly strong form of condition (3.5) was obtained by Ioffe [10] for the Michel–Penot subdifferentials. A version of the Lagrange multiplier rule for the Clarke subdifferentials also follows from Corollary 3.4.

COROLLARY 3.6. *For the problem* (PE)*, let $F = (f_0, \ldots, f_m)$. If $\overline{x}$ is a solution to* (PE)*, then there exist multipliers $\lambda_0 \geq 0, \ldots, \lambda_p \geq 0, \lambda_{p+1}, \ldots, \lambda_m$, not all zero, such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \ldots, m,$$

$$0 \in \sum_{i=0}^{m} \lambda_i \partial_C F(\overline{x})^T \, e_{i+1} + N(Q, \overline{x}). \tag{3.6}$$

Proof. Let $\partial^* F(x) = \partial_C F(x)$. Then the conclusion follows directly from Theorem 3.3. □

It is worth noting that, in the case where Q is closed and convex, the separated multiplier rule of Clarke [1], $0 \in \sum_{i=0}^{m} \lambda_i \partial_C f_i(\overline{x})^T + N(Q, \overline{x})$, follows from (3.6) since $\partial_C F(\overline{x})^T \, e_{i+1} \subseteq \partial_C f_i(\overline{x})^T$.

The following example illustrates that our multiplier rule (3.4) is sharper than (3.5).

Example 3.1. Consider the problem

$$\begin{array}{c} \text{minimize} \quad (x_1+1)^2 + x_2^2 \\ \text{subject to } 2x_1 + |x_1| - |x_2| = 0. \end{array}$$

Clearly, $(0,0)^T$ is the minimum point of the above problem. Let $f_0(x_1, x_2) = (x_1+1)^2 + x_2^2$ and let $f_1(x_1, x_2) = 2x_1 + |x_1| - |x_2|$. Then

$$\overline{\text{co}}\, \partial^* f_0(0,0) = \partial^\diamond f_0(0,0) = \partial_C f_0(0,0) = \left\{ \begin{pmatrix} 2 \\ 0 \end{pmatrix}^T \right\};$$

$$\partial^* f_1(0,0) = \left\{ \begin{pmatrix} 3 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ 1 \end{pmatrix}^T \right\};$$

$$\partial^\diamond f_1(0,0) = \partial_C f_1(0,0) = \overline{\text{co}} \left\{ \begin{pmatrix} 3 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ 1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 3 \\ 1 \end{pmatrix}^T \right\}.$$

It is easy to verify that for $\lambda_0 = 1$ and $\lambda_1 = -1$,

$$\begin{pmatrix} 0 \\ 0 \end{pmatrix}^T \in \lambda_0 \overline{\text{co}}\, \partial^* f_0(0,0) + \lambda_1 \overline{\text{co}}\, \partial^* f_1(0,0) \subset \partial^\diamond(\lambda_0 f_0 + \lambda_1 f_1)(0,0),$$

where

$$\partial^\diamond(\lambda_0 f_1 + \lambda_1 f_1)(0,0) = \overline{\text{co}} \left\{ \begin{pmatrix} 1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ 1 \end{pmatrix}^T, \begin{pmatrix} -1 \\ -1 \end{pmatrix}^T, \begin{pmatrix} 1 \\ 1 \end{pmatrix}^T \right\}.$$

A Kuhn–Tucker type necessary optimality condition follow from Theorem 3.3 under a constraint qualification [1, 4, 5, 6]. Consider first the problem

$$\text{(PE1)} \qquad \begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) = 0, \quad i = 1, \dots, p, \end{array}$$

where $x \in \mathbb{R}^n$, $f_0, \dots, f_p : \mathbb{R}^n \to \mathbb{R}$ are locally Lipschitzian functions. Let $F_2 = (f_1, \dots, f_p)$.

THEOREM 3.7. *Assume that $F_2 = (f_1, \dots, f_p)$ admits a locally bounded approximate Jacobian $\partial^* F_2(\overline{x})$ at $\overline{x}$. If $\overline{x} \in \mathbb{R}^n$ is a solution to* (PE1) *and $\overline{\text{co}}\, \partial^* F_2(\overline{x})$ is of maximal rank, then there exist Lagrange multipliers $\lambda_1, \dots, \lambda_p$ such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \dots, p,$$
$$0 \in \overline{\text{co}}\, \partial^* f_0(\overline{x})^T + \sum_{i=1}^p \lambda_i \overline{\text{co}}\, \partial^* F_2(\overline{x})^T e_i.$$

Remark. Of course, $\overline{\text{co}}\, \partial^* F_2(\overline{x})$ is of maximal rank means here that each matrix $M \in \overline{\text{co}}\, \partial^* F_2(\overline{x})$ is of maximal rank.

Proof. Corollary 3.4 gives us that there exist multipliers $\lambda_0, \lambda_1, \dots, \lambda_p$, not all zero, such that

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \dots, p,$$

$$0 \in \lambda_0 \overline{\text{co}}\, \partial^* f_0(\overline{x})^T + \sum_{i=1}^p \lambda_i \overline{\text{co}}\, \partial^* F_2(\overline{x})^T e_i. \tag{3.7}$$

Suppose that $\lambda_0 = 0$. Then we have

$$0 \in \sum_{i=1}^p \lambda_i \overline{\text{co}}\, \partial^* F_2(\overline{x})^T e_i.$$

This contradicts the hypothesis that $\overline{\text{co}}\, \partial^* F_2(\overline{x})$ is of maximal rank. Hence $\lambda_0 = 1$ may be assumed and so the conclusion holds. □

It follows from Theorem 3.7 that if $F_2 = (f_1, \dots, f_p)$, and $\overline{x} \in \mathbb{R}^n$ is a solution to (PE1) and if $\partial_C F_2(\overline{x})$ is of maximal rank, then there exist multipliers $\lambda_1, \dots, \lambda_p$ such that

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \dots, p,$$

$$0 \in \overline{\text{co}}\, \partial^* f_0(\overline{x})^T + \sum_{i=1}^p \lambda_i \partial_C F_2(\overline{x})^T e_i.$$

Similarly, for the general problem (PE), the Kuhn–Tucker conditions of the form

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \ldots, m,$$

$$0 \in \overline{\mathrm{co}}\, \partial^* f_0(\overline{x})^T + \sum_{i=1}^{m} \lambda_i \overline{\mathrm{co}}\, \partial^* F_1(\overline{x})^T\, e_i + N(Q, \overline{x}),$$

where $F_1 = (f_1, \ldots, f_p, f_{p+1}, \ldots, f_m)$, follow from Corollary 3.4 under the following constraint qualification: $(\overline{\mathrm{co}}\, \partial^* F_1(\overline{x})^T\, e_{p+1}, \ldots, \overline{\mathrm{co}}\, \partial^* F_1(\overline{x})^T\, e_m)$ is of maximal rank and there exists a vector $v \in T(Q, \overline{x})$ such that

$$\begin{array}{ll} \langle \overline{\mathrm{co}}\, \partial^* F_1(\overline{x})^T\, e_i, v\rangle < 0 & \text{if } f_i(\overline{x}) = 0, i = 1, \ldots, p, \\ \langle \overline{\mathrm{co}}\, \partial^* F_1(\overline{x})^T\, e_i, v\rangle = 0, & i = p+1, \ldots, m. \end{array}$$

If we choose $\partial^* f_0(\overline{x}) = \partial^\diamond f_0(\overline{x})$ and $\partial^* F_1(\overline{x}) = \partial^\diamond f_1(\overline{x}) \times \cdots \times \partial^\diamond f_m(\overline{x})$, then we obtain directly the following corollary from Theorem 3.3, extending the corresponding result of Hiriart-Urruty [4, Theorem 6]. Notice now the constraint qualification for (PE) becomes

> $(\partial^\diamond f_{p+1}(\overline{x})^T, \ldots, \partial^\diamond f_m(\overline{x})^T)$ is of maximal rank and there exists a vector $v \in T(Q, \overline{x})$ such that

$$\begin{array}{ll} \langle \partial^\diamond f_i(\overline{x})^T, v\rangle < 0 & \text{if } f_i(\overline{x}) = 0, i = 1, \ldots, p, \\ \langle \partial^\diamond f_i(\overline{x})^T, v\rangle = 0, & i = p+1, \ldots, m. \end{array}$$

COROLLARY 3.8. *If $\overline{x} \in \mathbb{R}^n$ is a solution to* (PE) *and the above constraint qualification for problem* (PE) *is satisfied at $\overline{x}$, then there exist multipliers $\lambda_1, \ldots, \lambda_m$ such that*

$$\lambda_i f_i(\overline{x}) = 0, \qquad i = 1, \ldots, m,$$

$$0 \in \partial^\diamond f_0(\overline{x})^T + \sum_{i=1}^{m} \lambda_i \partial^\diamond f_i(\overline{x})^T + N(Q, \overline{x}).$$

Proof. The conclusion follows from Theorem 3.3 by standard arguments and so is omitted. □

4. Minimax problems.

Consider the following minimax problem:

$$\text{(CP)} \qquad \begin{array}{lll} \min\limits_{x \in \mathbb{R}^n} \max\limits_{1 \le k \le s} f_0^k(x) & & \\ \text{subject to} & f_i(x) \le 0, & i = 1, \ldots, p, \\ & f_i(x) = 0, & i = p+1, \ldots, m, \\ & x \in Q, & \end{array}$$

where $f_0^1, \ldots, f_0^s, f_1, \ldots, f_m : \mathbb{R}^n \to \mathbb{R}$ are locally Lipschitzian functions and Q is a closed convex subset of $\mathbb{R}^n$ containing $\overline{x}$. The function f_0, defined by

$$f_0(x) = \max\{f_0^k : k = 1, \ldots, s\},$$

is easily seen to be Lipschitz near $\overline{x}$. For any x, $I(x)$ denotes the set of indices j for which $f_0^j(x) = f_0(x)$.

In the following we use our generalized multiplier rule to deduce the optimality conditions for the above minimax problem.

THEOREM 4.1. *Assume that $f_0^1,\dots,f_0^s,f_1,\dots,f_m$ are locally Lipschitzian. Suppose that $F_1=(f_1,\dots,f_m)$ admits a locally bounded approximate Jacobian $\partial^* F_1(\overline{x})$ at $\overline{x}$. If $\overline{x}\in\mathbb{R}^n$ is a minimizer to* (CP)*, then there exist multipliers $\lambda_0\geq 0,\dots,\lambda_p\geq 0,\lambda_{p+1},\dots,\lambda_m$, not all zero, such that*

$$\lambda_i f_i(\overline{x})=0, \qquad i=1,\dots,m,$$

$$0\in\lambda_0\overline{\text{co}}\left(\bigcup_{j\in I(\overline{x})}\partial^* f_0^j(\overline{x})\right)+\sum_{i=1}^m\lambda_i\overline{\text{co}}\,\partial^* F_1(\overline{x})^T e_i+N(Q,\overline{x}).$$

Proof. By Corollary 3.4 there exist multipliers $\lambda_0\geq 0,\dots,\lambda_p\geq 0,\lambda_{p+1},\dots,\lambda_m$, not all zero, such that

$$\lambda_i f_i(\overline{x})=0, \qquad i=1,\dots,m,$$

$$0\in\lambda_0\overline{\text{co}}\,\partial^* f_0(\overline{x})^T+\sum_{i=1}^m\lambda_i\overline{\text{co}}\,\partial^* F_1(\overline{x})^T e_i+N(Q,\overline{x}).$$

The direct calculation of $\partial^* f_0(\overline{x})$ shows that $\partial^* f_0(\overline{x}):=\bigcup_{j\in I(\overline{x})}\partial^* f_0^j(\overline{x})$ is a J–L subdifferential of f_0 at $\overline{x}$. Indeed, for each $h\in\mathbb{R}^n$,

$$f_0^+(\overline{x},h)=\max_{j\in I(\overline{x})}(f_0^j)^+(\overline{x},h)\leq\max_{j\in I(\overline{x})}\max_{\xi^j\in\partial^* f_0^j(\overline{x})}\langle\xi^j,h\rangle=\max_{\xi\in\bigcup_{j\in I(\overline{x})}\partial^* f_0^j(\overline{x})}\langle\xi,h\rangle$$

and

$$f_0^-(\overline{x},h)\geq\max_{j\in I(\overline{x})}(f_0^j)^-(\overline{x},h)\geq\max_{j\in I(\overline{x})}\min_{\xi^j\in\partial^* f_0^j(\overline{x})}\langle\xi^j,h\rangle\geq\min_{\xi\in\bigcup_{j\in I(\overline{x})}\partial^* f_0^j(\overline{x})}\langle\xi,h\rangle.$$

Hence the condition holds. □

We conclude by noting that in Theorem 4.1 if we further assume that $f_0^k, k=1,\dots,s$, are also Gâteaux differentiable at $\overline{x}$, then there exist multipliers $\lambda_0\geq 0,\dots,\lambda_p\geq 0,\lambda_{p+1},\dots,\lambda_m$, not all zero, such that

$$\lambda_i f_i(\overline{x})=0, \qquad i=1,\dots,m,$$

$$0\in\lambda_0\overline{\text{co}}\,\{\nabla f_0^j(\overline{x}):j\in I(\overline{x})\}+\sum_{i=1}^m\lambda_i\overline{\text{co}}\,\partial^* F_1(\overline{x})^T e_i+N(Q,\overline{x}).$$

Acknowledgment. The authors are extremely grateful to the referee for his valuable suggestions and helpful comments, which have contributed to the final preparation of this paper.

REFERENCES

[1] F. H. Clarke, *Optimization and Nonsmooth Analysis*, 2nd ed., Classics in Applied Mathematics 5, SIAM, Philadelphia, 1990.

[2] V. F. Demyanov and V. Jeyakumar, *Hunting for a smaller convex subdifferential*, J. Global Optim., 10 (1997), pp. 305–326.

[3] V. F. Demyanov and A. M. Rubinov, *Constructive Nonsmooth Analysis*, Verlag Peter Lang, 1995.

[4] J.-B. Hiriart-Urruty, *On necessary optimality conditions in nondifferentiable programming*, Math. Programming, 14 (1978), pp. 73–86.

[5] J.-B. Hiriart-Urruty, *Tangent cones, generalized gradients and mathematical programming in Banach spaces*, Math. Oper. Res., 4 (1979), pp. 79–97.

[6] J.-B. Hiriart-Urruty, *Refinements of necessary optimality conditions in nondifferentiable programming*, Appl. Math. Optim., 5 (1979), pp. 63–82.

[7] A. D. Ioffe, *Nonsmooth analysis: Differential calculus of non-differentiable mappings*, Trans. Amer. Math. Soc., 266 (1981), pp. 1–56.

[8] A. D. Ioffe, *Necessary conditions in nonsmooth optimization*, Math. Oper. Res., 9 (1984), pp. 159–189.

[9] A. D. Ioffe, *Approximate subdifferentials and applications* 1*, The finite dimensional theory*, Trans. Amer. Math. Soc., 281 (1984), pp. 289–316.

[10] A. D. Ioffe, *A Lagrange multiplier rule with small convex-valued subdifferentials for nonsmooth problems of mathematical programming involving equality and nonfunctional constraints*, Math. Programming, 58 (1993), pp. 137–145.

[11] V. Jeyakumar, *On optimality conditions in nonsmooth inequality constrained minimization*, Numer. Funct. Anal. Optim., 9 (1987), pp. 535–546.

[12] V. Jeyakumar, *Simple Characterizations of Superlinear Convergence for Semismooth Equations via Approximate Jacobians*, Applied Mathematical Report AMR98/28, University of New South Wales, Sydney, Australia, 1998.

[13] V. Jeyakumar and D. T. Luc, *Approximate Jacobian matrices for nonsmooth continuous maps and C^1-Optimization*, SIAM J. Control Optim., 36 (1998), pp. 1815–1832.

[14] V. Jeyakumar and D. T. Luc, *Nonsmooth calculus, minimality and monotonicity of convexificators*, J. Optim. Theory Appl., 101 (1999), pp. 599–621.

[15] V. Jeyakumar, D. T. Luc, and S. Schaible, *Characterizations of generalized monotone nonsmooth continuous maps using approximate Jacobians*, J. Convex Anal., 5 (1998), pp. 119–132.

[16] V. Jeyakumar and X. Wang, *Approximate Hessian matrices and second-order optimality conditions for nonlinear programming problem with C^1-data*, J. Austral. Math. Soc. Ser. B, 40 (1999), pp. 403–420.

[17] P. Michel and J. P. Penot, *A generalized derivative for calm and stable functions*, Differential Integral Equations, 5 (1992), pp. 433–454.

[18] B. S. Mordukhovich, *Generalized differential calculus for nonsmooth and set-valued mappings*, J. Math. Anal. Appl., 183 (1994), pp. 250–288.

[19] B. S. Mordukhovich and Y. Shao, *On nonconvex subdifferential calculus in Banach spaces*, J. Convex Anal., 2 (1995), pp. 211–228.

[20] M. Studniarski and V. Jeyakumar, *A generalized mean-value theorem and optimality conditions in composite nonsmooth minimization*, Nonlinear Anal., 24 (1995), pp. 883–894.

[21] J. S. Treiman, *Shrinking generalized gradients*, Nonlinear Anal., 12 (1988), pp. 1429–1449.

[22] J. S. Treiman, *The linear nonconvex generalized gradient and Lagrange multipliers*, SIAM J. Optim., 5 (1995), pp. 670–680.

SIAM J. Optim.
Vol. 10, No. 4, pp. 1149–1162

SUPERLINEAR CONVERGENCE AND IMPLICIT FILTERING*

T. D. CHOI† AND C. T. KELLEY‡

Abstract. In this paper we show how the implicit filtering algorithm can be coupled with the BFGS quasi-Newton update to obtain a superlinearly convergent iteration if the noise in the objective function decays sufficiently rapidly as the optimal point is approached. In this way we give insight into the observations of good performance in practice of quasi-Newton methods when they are coupled with implicit filtering. We also report on numerical experiments that show how an implementation of implicit filtering that exploits these new results can improve the performance of the algorithm.

Key words. noisy optimization, implicit filtering, BFGS algorithm, superlinear convergence

AMS subject classifications. 65K05, 65K10, 90C30

PII. S1052623499354096

1. Introduction.

The implicit filtering algorithm is a backtracking line search quasi-Newton method which uses difference gradients, reducing the difference increment as the optimization progresses. Through this variation of the difference increment, one hopes to "filter out" high-frequency low-amplitude contributions to the function and thereby avoid local minima.

Implicit filtering has been successfully applied to problems in semiconductor design [47, 52, 48, 51], design of high-field magnets [29, 44, 43, 8], automotive engineering [18, 17, 13], and geosciences [28, 41, 2, 1]. In many of these applications, replacing the identity with a quasi-Newton model Hessian, either SR1 [5, 24] or BFGS [6, 31, 25, 46], measurably improves the performance of the implicit filtering algorithm.

This paper is a first step toward an analysis of these observations. We combine the local theory for convergence of BFGS [7, 23] and the convergence theory for implicit filtering [36, 4, 27] to formulate and prove a local superlinear convergence result. We then show how implicit filtering can be implemented in a way that is consistent with the new theory and we report on computational experiments that illustrate the results.

Implicit filtering, like the Nelder–Mead [42], multidirectional search (MDS) [49], Hooke–Jeeves [32], and DIRECT [33] algorithms, is a sampling method. This means that the algorithm uses only evaluations of the function to be minimized in the optimization. The implicit filtering, Hooke–Jeeves, and MDS methods all sample the function on a stencil and reduce the size of the stencil as the optimization progresses. The rules for shrinking the size of the stencil in these three methods are similar enough to allow for a common framework for their analysis [36, 4]. Implicit filtering is unique among these methods in that the use of a quasi-Newton model Hessian can improve its performance.

Sampling methods converge slowly and, when gradient information is available,

*Received by the editors March 14, 1999; accepted for publication (in revised form) January 10, 2000; published electronically June 13, 2000. This research was supported in part by National Science Foundation grant DMS-9700569 and a Department of Education GAANN fellowship.

http://www.siam.org/journals/siopt/10-4/35409.html

†Intelligent Information Systems, 4915 Prospectus Drive, Suite C2, Durham, NC 27713 (tonyc@renewal-iis.com).

‡Center for Research in Scientific Computation and Department of Mathematics, North Carolina State University, Box 8205, Raleigh, NC 27695-8205 (Tim_Kelley@ncsu.edu).

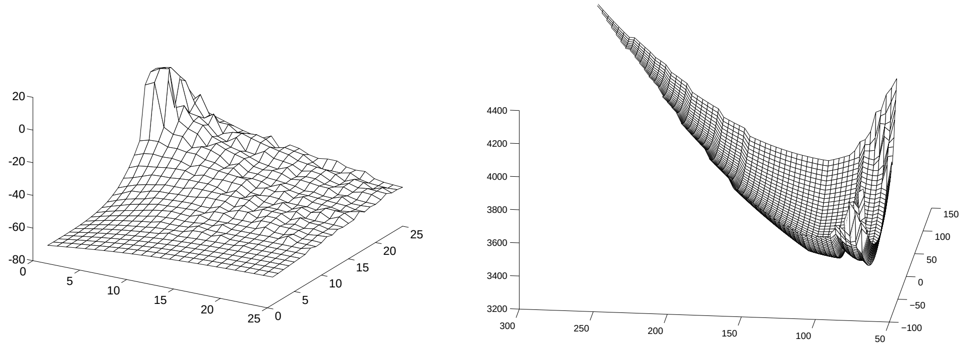


FIG. 1.1. *Optimization landscapes.*

conventional methods work far better. Therefore, sampling methods are usually applied to difficult problems with complex optimization landscapes [47, 26, 29, 30, 12]. The landscapes in Figure 1.1 are from semiconductor modeling [47, 27] (left) and the gas pipeline industry [11, 12] (right).

The objective functions for these problems can be nonsmooth or discontinuous and are rarely given by simple formulae. One common approach to understanding sampling algorithms is to analyze their performance in contexts simpler than those in which the algorithms are applied in practice. Sampling methods have been analyzed for smooth objective functions [38, 37, 22, 50, 14, 15] and objective functions that are perturbations of smooth functions by low-amplitude noise [39, 27, 35, 4, 36, 53]. In this paper we take the latter, more general approach. Convergence results for sampling methods that account for noise in this way must make assumptions on either the size of the noise or its rate of decay near an optimal point.

We have seen decay of noise near optimality in practice [47, 18, 51, 52, 48]. Such decay may happen because numerical models may be more accurate near optimality, internal iterations may terminate in a more uniform way, and table lookup may be more accurate.

We consider an objective function f defined on R^N that can be decomposed into the sum

$$f(x) = f_s(x) + \phi(x) \tag{1.1}$$

of a smooth and easy-to-minimize function f_s and a high-frequency and low-amplitude perturbation ϕ. We assume that the perturbation ϕ, which we refer to as noise, is uniformly bounded and is small relative to f_s. High-frequency oscillations in ϕ could cause f to have several local minima which would trap a conventional, gradient-based algorithm. The perturbation can be discontinuous, reflecting sensitivity of computations internal to f (such as temporal integrations with error control parameters) to their input, or random, reflecting, for example, the error in an experiment [30] or a probabilistic simulation internal to f [47].

Implicit filtering differs from other methods in the literature that explicitly use approximate gradients and Hessians. The convergence theory requires assumptions on the decay of the noise near optimality. However, we do not assume that we can control the errors in the function evaluation directly, and our results differ from those of [9] and [10], where it was assumed that control of the errors in function and gradient evaluations was possible and where global convergence of a trust region algorithm that managed these errors separately was proved. The superlinearly convergent algorithm in [40], which combines a coordinate search with a difference Hessian, is intended for noise-free function evaluations and is not applicable here. Implicit filtering does not attempt to model Hessians with interpolation, as does the trust region/interpolation method of [16, 14, 15]. We believe that the quasi-Newton approach has an advantage for noisy problems, where the errors in a Hessian formed by differences or interpolation can be large.

In section 2 we describe an implementation of implicit filtering that uses BFGS model Hessians, state the precise assumptions that we make on the decay of the perturbation ϕ near optimality, and show how that decay can be exploited in an implementation. We state and prove the local convergence result in section 3 and report on some computational experiments in section 4.

2. Implicit filtering.

Implicit filtering was designed for problems with optimization landscapes, like the ones shown in Figure 1.1, and objective functions that satisfy (1.1) in mind. The theoretical convergence results require assumptions on the size of the perturbation ϕ and its rate of decay near an optimizer of f_s. In section 2.1 we will formally describe an implementation of implicit filtering and in section 2.2 state a convergence result that makes minimal assumptions on ϕ.

Sharper results that give linear or superlinear convergence rates require stronger assumptions on the decay of the noise near optimality. We discuss that issue in section 2.3, where we review the assumptions needed for the linear convergence results from [27] and state the assumptions required for the superlinear convergence results in this paper.

2.1. Implicit filtering with BFGS model Hessians.

Throughout this paper $\|\cdot\|$ will denote the ℓ^2 norm on R^N. We will discuss only central difference gradients because implicit filtering performs far better [47, 36, 17] if central rather than one-sided differences are used. For $x \in R^N$ and $h \neq 0$ the central difference gradient of f with *scale* h at x is given by

$$(\nabla_h f(x))_i = \frac{f(x+he_i) - f(x-he_i)}{2h},$$

where e_i is the unit vector in the ith coordinate direction and $(\nabla_h f(x))_i$ denotes the ith component of the difference gradient.

The inner iteration in implicit filtering is a quasi-Newton optimization using a difference gradient and a backtracking line search. The iteration taking current approximation x_c to a new one x_+ for a fixed h is

$$x_+ = x_c + \lambda d_c, \text{ where } d_c = -H_c^{-1}\nabla_h f(x_c),$$

and H_c is the current model Hessian. λ is a step length determined by a backtracking line search and the sufficient decrease condition

$$f(x_c + \lambda d_c) - f(x_c) < \alpha\lambda\nabla_h f(x_c)^T d_c. \tag{2.1}$$

In (2.1) α is a small parameter (10^{-4} is a typical value [21, 34]). If (2.1) fails to hold, the step length λ is reduced. Typical methods [36, 21] for doing this include reducing λ by a predetermined factor and constructing a polynomial model of $f(x + \lambda d)$.

Unlike the noise-free case, $-\nabla_h f(x_c)$ need not be a direction of descent for f and, therefore, there is no guarantee that (2.1) can be satisfied for any value of λ. Hence, one must limit the number of reductions in λ.

We terminate the central difference quasi-Newton algorithm `fdquasi` when

$$\|\nabla_h f(x)\| \le \tau h \tag{2.2}$$

for some $\tau > 0$, when more than *pmax* iterations have been taken, if

$$f(x) \le \min_j \{f(x \pm h e_j)\}, \tag{2.3}$$

or when the line search fails by taking more than *amax* backtracks. Even the failures of `fdquasi` can be used to advantage [27, 36] by triggering a reduction in h. The parameter τ in the termination criterion (2.2) does not affect the convergence result that we state here, but can affect performance.

At this point we will elaborate on the use of (2.3) (*stencil failure*) as a termination criterion for `fdquasi`. The *stencil failure theorem* from [4, 36] states that (2.3) implies that $\|\nabla f_s(x)\| = O(h)$. Stencil failure indicates that a reduction in h (i.e., termination of `fdquasi`) is appropriate. Stencil failure is also the criterion for reduction of the size of the stencil in the Hooke–Jeeves and MDS methods.

ALGORITHM 1 (`fdquasi`$(x, f, pmax, \tau, h, amax)$).

```
$p = 1$
**while** $p \le pmax$ and $\|\nabla_h f(x)\| \ge \tau h$ **do**
  compute $f$ and $\nabla_h f$
  **if** (2.3) holds **then**
    terminate and report **stencil failure**
  **end if**
  update $H$ if appropriate; solve $Hd = -\nabla_h f(x)$
  use a backtracking line search, with at most *amax* backtracks, to find a step
  length $\lambda$
  **if** *amax* backtracks have been taken **then**
    terminate and report **line search failure**
  **end if**
  $x \leftarrow x + \lambda d$
  $p \leftarrow p + 1$
**end while**
if $p > pmax$ report **iteration count failure**
```

Algorithm `fdquasi` will terminate after finitely many iterations because of the limits on the number of iterations and the number of backtracks. Implicit filtering calls `fdquasi` repeatedly, reducing h after the return of `fdquasi`. The input to implicit filtering, in addition to the data for `fdquasi`, is the sequence of difference increments $\{h_k\}$. That sequence is infinite in the formulation we present here to allow us to state asymptotic convergence results.

ALGORITHM 2 (`imfilter`$(x, f, pmax, \tau, \{h_k\}, amax)$).

```
**for** $k = 0, \ldots,$ **do**
  fdquasi$(x, f, pmax, \tau, h_k, amax)$
**end for**
```

The results in this paper will lead to a different formulation of implicit filtering in which the scales are computed within the algorithm to improve the speed of convergence.

2.2. Basic convergence result. Let x_k be an iteration of algorithm `imfilter` and $S^k = \{x_k \pm h_k e_j\}_{j=1}^N$ be the difference stencil. We measure the local size of the noise by

$$\|\phi\|_{S^k} = \max\left(|\phi(x)|, \max_{r \in S^k} |\phi(r)| \right).$$

Theorem 2.1 is a global convergence result whose key assumption, (2.4), requires only that the size of the noise decay more rapidly than the difference increment.

THEOREM 2.1. *Let f satisfy* (1.1) *and let ∇f_s be Lipschitz continuous. Assume that the set $\{x \,|\, f(x) \le f(x_0)\}$ is bounded and the model Hessians and their inverses are uniformly bounded. Let $\{x_k\}$ be the implicit filtering sequence. Assume that* `fdquasi` *terminates infinitely often with either* (2.2) *or* (2.3) *holding. Then if*

$$\lim_{k\to\infty} (h_k + h_k^{-1}\|\phi\|_{S^k}) = 0, \tag{2.4}$$

then any limit point of the sequence $\{x_k\}$ is a critical point of f_s.

The proof follows from an estimate from [36],

$$\|\nabla_h f(x) - \nabla f_s(x)\| = O(h^2 + h^{-1}\|\phi\|_S), \tag{2.5}$$

the stencil failure theorem, and (2.4), which also implies that $\|\nabla f_s(x_k)\| = O(h_k)$.

In this paper we focus on the BFGS update

$$H_+ = H_c + \frac{yy^T}{y^Ts} - \frac{(H_cs)(H_cs)^T}{s^TH_cs}, \tag{2.6}$$

where $s = x_+ - x_c$ and $y = \nabla_h f(x_+) - \nabla_h f(x_c)$.

2.3. Rates of convergence and decay of ϕ near optimality. In order to obtain rates of convergence, stronger assumptions on the decay of the noise must be made and the difference increments must be adjusted to reflect that decay. Our proof of local superlinear convergence will require that h and ϕ satisfy

$$\|\nabla_h \phi(x)\| = O(\|x - x^*\|^{1+p}) \tag{2.7}$$

for some $p > 0$. In (2.7), x^* is the local minimizer to which the BFGS iteration for minimizing f_s would converge. In [27] a weaker rate of decay, $\|\nabla_h\phi(x)\| \le \epsilon\|x - x^*\|$ for a small ϵ, was used to prove linear convergence.

The assumption below is a combination of the standard assumptions for local convergence of Newton's method and a rate-of-decay rate on ϕ. We will show how this will imply that implicit filtering can be implemented so that (2.7) holds.

ASSUMPTION 2.1. *f_s has a local minimizer x^*, $\nabla^2 f_s$ is Lipschitz continuous in a neighborhood of x^*, $\nabla^2 f_s(x^*)$ is positive definite, and for x sufficiently near x^*,*

$$|\phi(x)| = O(\|x - x^*\|^{2+2p}), \tag{2.8}$$

for some $p > 0$.

Equations (2.5) and (2.8) imply that, for x sufficiently near x^*,

$$|\nabla_h \phi(x)| = O(h^{-1}\|x - x^*\|^{2+2p}) = O(h^{-1}\|\nabla f_s(x)\|^{2+2p}).$$

Hence,

$$\nabla_h f(x) = \nabla f_s(x) + O(h^2 + h^{-1}\|\nabla f_s(x)\|^{2+2p}). \tag{2.9}$$

So if we can control h and x simultaneously so that

$$C_1^{-1}\|x - x^*\|^{1+p} \leq h \leq C_1\|x - x^*\|^{(1+p)/2}, \tag{2.10}$$

for some $C_1 > 0$, then (2.9) will imply (2.7), for x sufficiently near x^*.

We now show how (2.10) can be enforced during an iteration. Let $x_k \to x^*$ at least q-linearly and no faster than q-quadratically. This means that there are $r \in (0,1)$ and $C_2 > 0$ such that

$$\|x_{k+1} - x^*\| \leq r\|x_k - x^*\| \leq C_2\|x_{k+1} - x^*\|^{1/2}. \tag{2.11}$$

Lemma 2.2. *Let Assumption* 2.1 *hold and let* $\{x_k\}$ *satisfy* (2.11). *Then there are* K *and* C_1 *such that if* $k \geq K$, h_k *and* x_k *satisfy* (2.10), *and*

$$h_{k+1} = \|\nabla_{h_k} f(x_k)\|^{1+p}, \tag{2.12}$$

then h_{k+1} *and* x_{k+1} *also satisfy* (2.10).

Proof. Assumption 2.1 implies that there is K_1 such that for all x sufficiently near x^*

$$K_1^{-1}\|x - x^*\| \leq \|\nabla f_s(x)\| \leq K_1\|x - x^*\|.$$

Hence (2.12) imples that for k sufficiently large

$$\nabla_{h_k} f(x_k) = \nabla f_s(x_k) + E_k. \tag{2.13}$$

Here E_k is the sum of the $O(h_k^2)$ difference error and $\nabla_{h_k}\phi(x_k)$. Hence, there is $C_D > 0$ such that

$$\|E_k\| \leq C_D(h_k^2 + h_k^{-1}\|x_k - x^*\|^{2+2p}).$$

Since h_k and x_k satisfy (2.10) by assumption, we have

$$h_k^2 + h_k^{-1}\|x_k - x^*\|^{2+2p} \leq (C_1^2 + C_1)\|x_k - x^*\|^{1+p}. \tag{2.14}$$

Hence, if $C_3 = C_D(C_1^2 + C_1)$, then

$$\begin{aligned}(K_1^{-1} - C_3\|x_k - x^*\|^p)\|x_k - x^*\| &\leq \|\nabla_{h_k} f(x_k)\| \\ &\leq (K_1 + C_3\|x_k - x^*\|^p)\|x_k - x^*\|.\end{aligned} \tag{2.15}$$

Let k be large enough so that

$$K_1^{-1} - C_3\|x_k - x^*\|^p > K_1^{-1}/2 \text{ and } K_1 + C_3\|x_k - x^*\|^p < 2K_1.$$

Equation (2.15) implies that

$$(2K_1)^{-p-1}\|x_k - x^*\|^{p+1} \leq h_{k+1} \leq (2K_1)^{p+1}\|x_k - x^*\|^{p+1}.$$

We complete the proof by using (2.11) to conclude that

$$(2K_1)^{-(p+1)}\|x_{k+1} - x^*\|^{p+1} \leq h_{k+1} \leq (2K_1)^{p+1}C_2^{p+1}\|x_{k+1} - x^*\|^{p+1/2},$$

which completes the proof if $C_1 \geq (2K_1)^{p+1}C_2^{p+1}$. □

We show in section 4 how an implementation of implicit filtering that uses (2.12) to compute the difference increments performs for some example problems. We close this section with a simple corollary of Lemma 2.2 which states that if Assumption 2.1 holds and the difference increments are managed so that (2.11) and (2.10) are valid, then the error in the gradient can be estimated in a way that will allow us to prove the superlinear convergence result in section 3.

Corollary 2.3. *Let Assumption* 2.1 *hold and let* $\{x_k\}$ *and* $\{h_k\}$ *satisfy* (2.11) *and* (2.10). *Let* $\rho > 1$. *If* x *satisfies*

$$\rho\|x_{k+1} - x^*\| \leq \|x - x^*\| \leq \rho^{-1}\|x_k - x^*\|, \tag{2.16}$$

then

$$\nabla_{h_{k+1}}\phi(x) = O(\|x - x^*\|^{1+p}). \tag{2.17}$$

3. Local convergence.

Throughout this section we assume Assumption 2.1 holds. This implies that the standard assumptions [21, 36] for the convergence of Newton's method hold at x^*.

We also assume that (2.8) holds and that the scales have been managed so that the consequences of Corollary 2.3 hold. We use the paradigm of [23] and [34] to simplify notation and avoid having to explicitly discuss the iteration index and the scales in the analysis. We let $g = \nabla_h f(x) \approx \nabla f_s(x)$ be the approximate gradient and let

$$N(x) = g(x) - \nabla f_s(x) = \nabla_h \phi(x)$$

denote the gradient error. If (2.10) holds, then there are $C_\epsilon, p > 0$ such that

$$\|N(x)\| \leq C_\epsilon \|x - x^*\|^{1+p} \tag{3.1}$$

for x sufficiently near x^*.

The quasi-Newton implementation uses g instead of ∇f_s in both the computation of the BFGS step (we take full steps in a local theory)

$$s = -H_c^{-1} g(x_c) \tag{3.2}$$

and in the difference in gradients

$$y = g(x_+) - g(x_c), \tag{3.3}$$

both of which are used in the BFGS update (2.6) of the model Hessian H.

We begin with a simple result that is a restatement of Lemma (2.4) in [23] and Theorem (5.4.1) of [34].

Theorem 3.1. *Let Assumption* 2.1 *and* (2.10) *hold. There are* $C_L > 0$ *and* $\delta_0 > 0$ *so that if* $\delta \in [0, \delta_0]$, $\|x - x^*\| < \delta$, *and* $\|H_c - \nabla^2 f_s(x^*)\| < \delta$, *then*

$$\|x_+ - x^*\| \leq r\|x_c - x^*\|, \tag{3.4}$$

where

$$r \leq C_L(\delta + \|x_c - x^*\|^p). \tag{3.5}$$

We will need estimates of the difference between H_+ and an idealized update

$$\bar{H}_+ = H_c + \frac{\bar{y}\bar{y}^T}{\bar{y}^T\bar{s}} - \frac{(H_c\bar{s})(H_c\bar{s})^T}{\bar{s}^T H_c \bar{s}} \tag{3.6}$$

that uses data only from f_s. In (3.6) $\bar{s} = -H_c^{-1}\nabla f_s(x_c)$ and $\bar{y} = \nabla f_s(x_c+\bar{s}) - \nabla f_s(x_c)$. We define

$$M_c = H_+ - \bar{H}_+. \tag{3.7}$$

LEMMA 3.2. *Let the assumptions of Theorem* 3.1 *hold. Then there are C_M and $\delta > 0$ such that if $\|x_c - x^*\| < \delta$, and $\|H_c - \nabla^2 f_s(x^*)\| < \delta$, then*

$$\|M_c\| \le C_M \|x_c - x^*\|^p. \tag{3.8}$$

Proof. We write $M_c = M_1 + M_2$, where

$$M_1 = \frac{yy^T}{y^T s} - \frac{\bar{y}\bar{y}^T}{\bar{y}^T\bar{s}} \text{ and } M_2 = \frac{(H_c\bar{s})(H_c\bar{s})^T}{\bar{s}^T H_c\bar{s}} - \frac{(H_c s)(H_c s)^T}{s^T H_c s}.$$

We will show that $M_1 = O(\|x_c - x^*\|^p)$. The bound on M_2 can be obtained in a similar fashion.

Let δ be small enough so that the hypotheses of Theorem 3.1 hold with

$$r \le C_L(\delta + \delta^p) < 1/2.$$

Since

$$y = g(x_+) - g(x_c) = \bar{y} + N(x_+) - N(x_c),$$

and $\|x_+ - x^*\| \le \|x_c - x^*\|/2$, we have

$$\|y - \bar{y}\| \le 2C_\epsilon \|x_c - x^*\|^{p+1}. \tag{3.9}$$

Hence,

$$yy^T = \bar{y}\bar{y}^T + O(\|x_c - x^*\|^{p+2}). \tag{3.10}$$

Similarly,

$$y^T s = \bar{y}^T\bar{s} + O(\|x_c - x^*\|^{p+2}). \tag{3.11}$$

The standard assumptions imply (reducing δ if necessary) that there is c_y

$$\|\bar{y}\| \ge c_y\|x_c - x^*\| \text{ and } |\bar{y}^T\bar{s}| \ge c_y\|x_c - x^*\|^2;$$

hence $M_1 = O(\|x_c - x^*\|^p)$, as asserted. □

We use Lemma 3.2 to obtain a q-linear convergence from Theorem 3.1 via a bounded deterioration result. Then q-superlinear convergence will follow from the classic arguments [7, 20, 36]. We let $\|\cdot\|_F$ be the Frobenius norm.

Lemma 3.2 and known results for the BFGS update imply Corollary 3.3. This result, with $p = 1$, is used in the classical analysis of BFGS convergence.

COROLLARY 3.3. *Let the assumptions of Theorem* 3.1 *hold. Then there are $C_H, \delta > 0$ such that if $\|x_c - x^*\| < \delta$ and $\|H_c - \nabla^2 f_s(x^*)\| < \delta$, then*

$$\|H_+ - \nabla^2 f_s(x^*)\| \le \|H_c - \nabla^2 f_s(x^*)\| + C_H(\|x_c - x^*\|^p + \|x_+ - x^*\|^p). \tag{3.12}$$

From this point the methods from [7] can be applied directly to prove superlinear convergence, and we give only a sketch of that argument. Corollary 3.3 implies a q-linear convergence result, which is exactly the same as the one for the BFGS algorithm itself. The proofs differ only in that $p = 1$ in the classic proof from [7], a difference which requires no change at all in the logic of the proof.

Lemma 3.4. *Let the assumptions of Theorem* 3.1 *hold and let* $\delta_0, r \in (0, 1)$. *Then there is* δ_1 *such that if* $\delta \in [0, \delta_1)$, $\|x_0 - x^*\| < \delta$, *and* $\|H_0 - \nabla^2 f_s(x^*)\| < \delta$, *then for all* $k \geq 0$,

1. H_k *is nonsingular,*
2. $\|H_k - \nabla^2 f_s(x^*)\| \leq \delta_0$, *and*
3. $\|x_{k+1} - x^*\| \leq r\|x_k - x^*\|$.

The final step from Lemma 3.4 to the superlinear convergence result is the Frobenius norm estimate

$$\|H_{k+1}-\nabla^2 f_s(x^*)\|_F^2 \leq \|H_k-\nabla^2 f_s(x^*)\|_F^2-\|(H_k-\nabla^2 f_s(x^*))w_k\|^2+O(\|e_k\|^p), \tag{3.13}$$

where $w_k = s_k/\|s_k\|$. The q-linear convergence result implies that $\sum \|e_k\|^p < \infty$, which, together with (3.13), implies the Dennis–Moré condition [19, 20]. This is all that is needed for local superlinear convergence.

Theorem 3.5. *Let the assumptions of Lemma* 3.4 *hold. Then there is* $\delta > 0$ *such that if* $\|x_0 - x^*\| < \delta$ *and* $\|H_0 - \nabla^2 f_s(x^*)\| < \delta$, *the quasi-Newton iteration given by* (3.2) *and* (2.6) *converges q-superlinearly to* x^*.

4. Numerical results.

Methods such as implicit filtering are typically applied to complex problems whose objective functions may contain proprietary code or be difficult to detach from the larger application containing them [26, 47, 30, 18, 17]. Because of this we confine our numerical testing to two simple example problems that illustrate the types of problems we have encountered in practice and that at the same time are easy to describe and implement. Neither of the examples completely satisfies the assumptions of the theory because the noise does not decay to zero as optimality is approached. However, as the results in Figures 4.1 and 4.2 indicate, the BFGS iteration provides an improvement in performance, especially with the modification proposed in this paper.

The implementation of implicit filtering is a modification of the algorithm from [36] which is able to enforce (2.12). In each example we compare the simple implementation of implicit filtering without a quasi-Newton method, Algorithm `imfilter1`, one that uses the BFGS quasi-Newton update but does not reduce the scales using (2.12), and a BFGS implementation that enforces (2.12). The sequence of scales used in the examples is

$$h_k^{(1)} = 2^{-k-k_l},\ k = 0, \dots, k_u - k_l, \tag{4.1}$$

where the limits k_l and k_u on k depend on the problem. We enforce a scaled form of (2.12),

$$h_k^{(2)} = \max(\min(h_k^1, (\|\nabla_h f(x_k)\|_\infty / \|\nabla_{h_0} f(x_0)\|)_\infty^{1+p}), h_{min}), \tag{4.2}$$

where $h_{min} = 10^{-5}$. h_{min} is roughly the cube root of machine roundoff and is the optimal choice of h for a central difference.

In the examples the line search strategy is to reduce the step by half if the sufficient decrease condition fails. The parameters in implicit filtering for both examples were $\alpha = 10^{-4}$, $\tau = .01$, $pmax = 200$, and $amax = 10$.

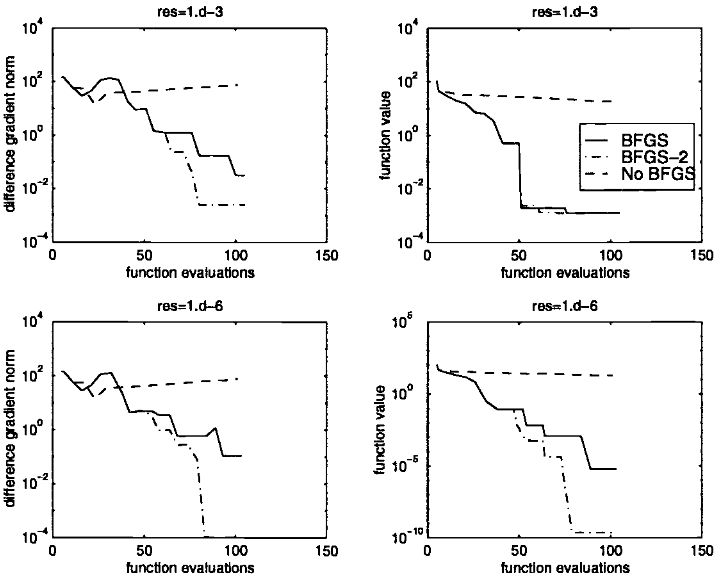


FIG. 4.1. *Parameter identification example.*

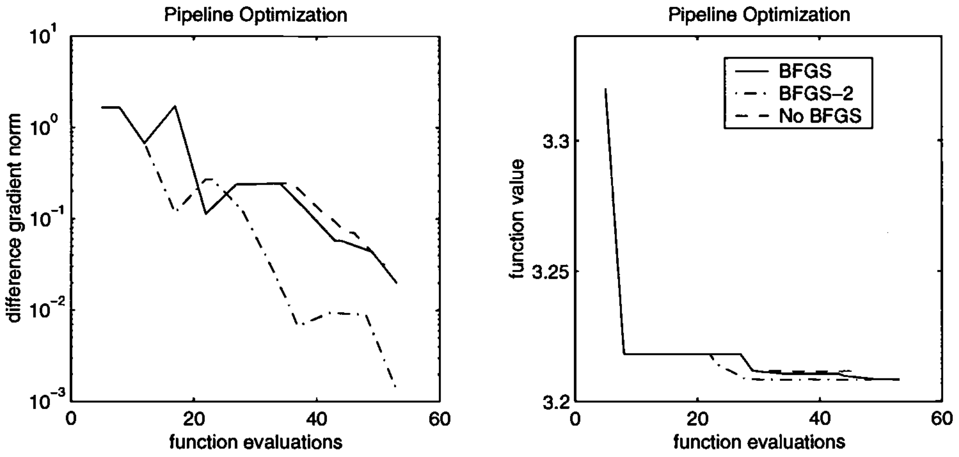


FIG. 4.2. *Pipeline network example.*

4.1. Parameter identification. This problem, taken from [3, 36], is to identify (using least squares fit to data) damping and spring constants c and k so that the numerical solution of

$$u'' + cu' + ku = 0; u(0) = u_0, u'(0) = 0$$

best fits the data in the least squares sense. Our data consists of values of the exact solution at the points $t_i = i/100$ for $1 \leq i \leq 100$. We computed the numerical solution with the code ODE15s from [45]. This is an adaptive (in step size and order) code with error control based on user supplied relative (*rtol*) and absolute (*atol*) tolerances for local truncation error. The level of noise in the objective is roughly the same size as the tolerances. We report on two computations: one with $rtol = atol = 10^{-3}$ and the other with $rtol = atol = 10^{-6}$. The noise in the problem with $rtol = atol = 10^{-6}$ is smaller and the rapid convergence of BFGS persists for more iterations than for the problem with $rtol = atol = 10^{-3}$.

In both cases the initial iterate is $(c, k) = (2, 3)$ and the optimization is limited to 100 function evaluations. In Figure 4.1 we plot the norm of the difference gradient and the least squares residual against the number of function evaluations. The dashed line corresponds to the implementation (no BFGS) without a quasi-Newton model Hessian, the solid line to the implementation (BFGS) with a BFGS model Hessian that does not enforce (2.12), and the dot–dashed line to the implementation (BFGS-2) that does enforce (2.12). In this problem the scales $\{h_k^{(1)}\}$ are given by (4.1) with $k_l = 4$ and $k_u = 17$.

The noise in this problem plays the role that floating point roundoff would play in a smooth problem, and the results for the BFGS implementation clearly show the superlinear reduction in the norm of the difference gradient by the concave shape of the curve.

4.2. Pipeline network optimization.

The problem considered in this section was provided by Richard Carter of Stoner and Associates [12]. The objective function is a piecewise linear interpolation of a function that was computed by Carter using proprietary models and data. The optimization landscape is illustrated on the right side of Figure 1.1. This example shows how the implementation of implicit filtering proposed in this paper can improve performance in an engineering application. The reader should be warned, however, that applying any algorithm of this type requires tuning of many algorithmic parameters (the initial simplex in Nelder–Mead and MDS; the scales and algorithmic parameters in implicit filtering; the rate at which the simplex size is decreased, for example) and the results in this example are only representative.

This problem is more difficult than the one in section 4.1. The function is nonsmooth and, as one can see from Figure 1.1, close to a smooth function. However, unlike the parameter identification example, there is no simple way to estimate the magnitude of the nonsmooth part. A second problem, which is not a factor in this paper, is that the function is not everywhere defined, and an attempt to evaluate the function outside the region under the surfaces graphed in Figure 1.1 will fail. We call the requirement that the objective be defined a “hidden constraint,” because there is no a priori way to tell if a point is feasible without attempting to evaluate the function. The example in this section has its optimal point in the interior of the feasible region, and the initial iterate is close enough to avoid the need to evaluate the objective at an infeasible point.

The problem is also poorly scaled. The function has a minimal value of about 3200 and a maximum of about 7800. The dependent variables have a range of $[-200, 200]$. Before applying implicit filtering we scaled the function by a factor of 10^{-3} and the dependent variables by 10^{-2}. The data we present are for the scaled variables. In this problem the scales $\{h_k^{(1)}\}$ are given by (4.1) with $k_l = 0$ and $k_u = 10$. The initial iterate is $(1, 1)^T$.

As with the previous example, in Figure 4.2 we plot the norm of the difference gradient and the least squares residual against the number of function evaluations for three different algorithmic variations.

Acknowledgments. Both authors, especially the second, gratefully acknowledge several verbal and electronic communications with Richard Byrd, John Dennis, and Jorge Nocedal that helped them formulate the ideas in this paper. The authors also thank Richard Carter from Stoner and Associates for providing them with the data used in section 4.2, and they thank two very thorough referees.

REFERENCES

[1] K. R. Bailey and B. G. Fitzpatrick, *Estimation of Groundwater Flow Parameters Using Least Squares*, Tech. Rep. CRSC-TR96-13, North Carolina State University, Center for Research in Scientific Computation, Raleigh, NC, April 1996.

[2] K. R. Bailey, B. G. Fitzpatrick, and M. A. Jeffries, *Least Squares Estimation of Hydraulic Conductivity from Field Data*, Tech. Rep. CRSC-TR95-8, North Carolina State University, Center for Research in Scientific Computation, Raleigh, NC, February 1995.

[3] H. T. Banks and H. T. Tran, *Mathematical and Experimental Modeling of Physical Processes*, unpublished lecture notes for MA 573-4, Department of Mathematics, North Carolina State University, Raleigh, NC, 1997.

[4] D. M. Bortz and C. T. Kelley, *The simplex gradient and noisy optimization problems*, in Computational Methods in Optimal Design and Control, J. T. Borggaard, J. Burns, E. Cliff, and S. Schreck, eds., Progr. Systems Control Theory 24, Birkhäuser, Boston, 1998, pp. 77–90.

[5] C. G. Broyden, *Quasi-Newton methods and their application to function minimization*, Math. Comp., 21 (1967), pp. 368–381.

[6] C. G. Broyden, *A new double-rank minimization algorithm*, AMS Notices, 16 (1969), p. 670.

[7] C. G. Broyden, J. E. Dennis, Jr., and J. J. Moré, *On the local and superlinear convergence of quasi-Newton methods*, J. Inst. Math. Appl., 12 (1973), pp. 223–245.

[8] L. J. Campbell, Y. Eyssa, P. Gilmore, P. Pernambuco-Wise, D. M. Parkin, D. G. Rickel, J. B. Schillig, and H. J. Schneider-Muntau, *The US* 100*-T magnet project*, Phys. B, 211 (1995), pp. 52–55.

[9] R. G. Carter, *On the global convergence of trust region algorithms using inexact gradient information*, SIAM J. Numer. Anal., 28 (1991), pp. 251–265.

[10] R. G. Carter, *Numerical experience with a class of algorithms for nonlinear optimization using inexact function and gradient information*, SIAM J. Sci. Comput., 14 (1993), pp. 368–388.

[11] R. G. Carter, *Private communication*, 1999.

[12] R. G. Carter, W. W. Schroeder, and T. D. Harbick, *Some causes and effect of discontinuities in modeling and optimizing gas transmission networks*, paper PSIG-9308, Proceedings of the Pipeline Simulation Interest Group, Pittsburgh, PA, 1993.

[13] T. D. Choi, O. J. Eslinger, C. T. Kelley, J. W. David, and M. Etheridge, *Optimization of Automotive Valve Train Components with Implicit Filtering*, Tech. Rep. CRSC-TR98-44, North Carolina State University, Center for Research in Scientific Computation, Raleigh, NC, December 1998. Optim. Engrg., to appear.

[14] A. R. Conn, K. Scheinberg, and P. L. Toint, *On the convergence of derivative-free methods for unconstrained optimization*, in Approximation Theory and Optimization: Tributes to M. J. D. Powell, A. Iserles and M. Buhmann, eds., Cambridge University Press, Cambridge, UK, 1997, pp. 83–108.

[15] A. R. Conn *Recent progress in unconstrained nonlinear optimization without derivatives*, Math. Programming Ser. B, 79 (1997), pp. 397–414.

[16] A. R. Conn and P. L. Toint, *An Algorithm Using Quadratic Interpolation for Unconstrained Derivative-Free Optimization*, Tech. Rep. 95/6, Facultès Universitaires de Namur, Namur, Belgium, 1995.

[17] J. W. David, C. Y. Cheng, T. D. Choi, C. T. Kelley, and J. Gablonsky, *Optimal Design of High Speed Mechanical Systems*, Tech. Rep. CRSC-TR97-18, North Carolina State University, Center for Research in Scientific Computation, Raleigh, NC, July 1997. Math. Model. Sci. Comput., to appear.

[18] J. W. David, C. T. Kelley, and C. Y. Cheng, *Use of an implicit filtering algorithm for mechanical system parameter identification*, in Modeling of CI and SI Engines, SAE Paper 960358, 1996 SAE International Congress and Exposition Conference Proceedings, Society of Automotive Engineers, Washington, DC, 1996, pp. 189–194.

[19] J. E. Dennis and J. J. Moré, *A characterization of superlinear convergence and its application to quasi-Newton methods*, Math. Comp., 28 (1974), pp. 549–560.

[20] J. .E. Dennis, *Quasi-Newton methods, motivation and theory*, SIAM Rev., 19 (1977), pp. 46–89.

[21] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Classics Appl. Math. 16, SIAM, Philadelphia, 1996.

[22] J. E. Dennis, Jr., and V. Torczon, *Direct search methods on parallel machines*, SIAM J. Optim., 1 (1991), pp. 448–474.

[23] J. E. Dennis and H. F. Walker, *Inaccuracy in quasi-Newton methods: Local improvement theorems*, in Mathematical Programming Study 22: Mathematical Programming at Oberwolfach II, North–Holland, Amsterdam, 1984, pp. 70–85.

[24] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Classics Appl. Math. 4, SIAM, Philadelphia, 1990.

[25] R. Fletcher, *A new approach to variable metric methods*, Comput. J., 13 (1970), pp. 317–322.

[26] S. J. Fortune, D. M. Gay, B. W. Kernighan, O. Landron, R. A. Valenzuela, and M. H. Wright, *WISE design of indoor wireless systems*, IEEE Comput. Sci. Engrg., 2 (1995), pp. 58–68.

[27] P. Gilmore and C. T. Kelley, *An implicit filtering algorithm for optimization of functions with many local minima*, SIAM J. Optim., 5 (1995), pp. 269–285.

[28] P. Gilmore, C. T. Kelley, C. T. Miller, and G. A. Williams, *Implicit filtering and optimal design problems*, in Proceedings of the workshop on Optimal Design and Control, Blacksburg, VA, April 8–9, 1994, J. Borggaard, J. Burkhardt, M. Gunzburger, and J. Peterson, eds., Progr. Systems Control Theory 19, Birkhäuser, Boston, 1995, pp. 159–176.

[29] P. Gilmore, P. Pernambuco-Wise, and Y. Eyssa, *An Optimization Code for Pulse Magnets*, Tech. Rep., National High Magnetic Field Laboratory, Florida State University, Tallahassee, FL, August 1994.

[30] P. A. Gilmore, S. S. Berger, R. F. Burr, and J. A. Burns, *Automated optimization techniques for phase change piezoelectric ink jet performance enhancement*, in Proceedings 1997 International Conference on Digital Printing Technologies, Society for Imaging Science and Technology, Springfield, VA, 1997, pp. 716–721.

[31] D. Goldfarb, *A family of variable metric methods derived by variational means*, Math. Comp., 24 (1970), pp. 23–26.

[32] R. Hooke and T. A. Jeeves, *'Direct search' solution of numerical and statistical problems*, J. ACM, 8 (1961), pp. 212–229.

[33] D. R. Jones, C. C. Perttunen, and B. E. Stuckman, *Lipschitzian optimization without the Lipschitz constant*, J. Optim. Theory Appl., 79 (1993), pp. 157–181.

[34] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, Frontiers Appl. Math. 16, SIAM, Philadelphia, 1995.

[35] C. T. Kelley, *Detection and remediation of stagnation in the Nelder–Mead algorithm using a sufficient decrease condition*, SIAM J. Optim., 10 (1999), pp. 43–55.

[36] C. T. Kelley, *Iterative Methods for Optimization*, Frontiers Appl. Math. 18, SIAM, Philadelphia, 1999.

[37] J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright, *Convergence properties of the Nelder–Mead simplex algorithm in low dimensions*, SIAM J. Optim., 9 (1998), pp. 112–147.

[38] S. Lucidi and M. Sciandrone, *On the Global Convergence of Derivative Free Methods for Unconstrained Optimization*, preprint, Università di Roma "La Sapienza," Dipartimento di Informatica e Sistemistica, Rome, Italy, 1997.

[39] S. Lucidi and M. Sciandrone, *A Derivative-free Algorithm for Bound Constrained Optimization*, preprint, Instituto di Analisi dei Sistemi ed Informatica, Consiglio Nazionale delle Richerche, Rome, Italy, 1999.

[40] R. Mifflin, *A superlinearly convergent algorithm for minimization without derivatives*, Math. Programming, 9 (1975), pp. 100–117.

[41] C. T. Miller, G. A. Williams, and C. T. Kelley, *Transformation Approaches for Simulating Flow in Variably Saturated Porous Media*, Tech. Rep. CRSC-TR98-01, North Carolina State University, Center for Research in Scientific Computation, Raleigh, NC, January 1998. Water Resources Research, to appear.

[42] J. A. Nelder and R. Mead, *A simplex method for function minimization*, Comput. J., 7 (1965), pp. 308–313.

[43] P. Pernambuco-Wise, P. Gilmore, and Y. Eyssa, *An optimization code for pulse magnets*, Phys. B, to appear.

[44] P. Pernambuco-Wise, P. Gilmore, B. Lesch, Y. Eyssa, and H. J. Schneider-Muntau, *Systematic failure testing of internally reinforced magnets*, IEEE Trans. Magnetics, 4 (1996), pp. 2458–2461.

[45] L. F. Shampine and M. W. Reichelt, *The MATLAB ODE suite*, SIAM J. Sci. Comput., 18 (1997), pp. 1–22.

[46] D. F. Shanno, *Conditioning of quasi-Newton methods for function minimization*, Math. Comp., 24 (1970), pp. 647–657.

[47] D. Stoneking, G. Bilbro, R. Trew, P. Gilmore, and C. T. Kelley, *Yield optimization using a GaAs process simulator coupled to a physical device model*, IEEE Trans. Microwave Theory and Techniques, 40 (1992), pp. 1353–1363.

[48] D. E. Stoneking, G. L. Bilbro, R. J. Trew, P. Gilmore, and C. T. Kelley, *Yield optimization using a GaAs process simulator coupled to a physical device model*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 374–383.

[49] V. Torczon, *Multidirectional Search*, Ph.D. thesis, Rice University, Houston, TX, 1989.

[50] V. Torczon, *On the convergence of the multidirectional search algorithm*, SIAM J. Optim., 1 (1991), pp. 123–145.

[51] T. A. Winslow, R. J. Trew, P. Gilmore, and C. T. Kelley, *Doping profiles for optimum class B performance of GaAs mesfet amplifiers*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 188–197.

[52] T. A. Winslow, R. J. Trew, P. Gilmore, and C. T. Kelley, *Simulated performance optimization of GaAs MESFET amplifiers*, in Proceedings IEEE/Cornell Conference on Advanced Concepts in High Speed Devices and Circuits, IEEE, Piscataway, NJ, 1991, pp. 393–402.

[53] S. K. Zavriev, *On the global optimization properties of finite-difference local descent algorithms*, J. Global Optim., 3 (1993), pp. 67–78.

SIAM J. Optim.
Vol. 10, No. 4, pp. 1163–1179

A FAMILY OF SCALED FACTORIZED BROYDEN-LIKE METHODS FOR NONLINEAR LEAST SQUARES PROBLEMS*

J. Z. ZHANG†, L. H. CHEN†, and N. Y. DENG‡

Abstract. In this paper, a family of scaled factorized Broyden-like methods for unconstrained nonlinear least squares problems with zero or nonzero residual is presented. This family is based on a full rank factorized form of a structured quasi-Newton update and a scaled approximation of the second order term using given information. The resulting algorithms yield a q-quadratic convergence for the zero residual case and a q-superlinear convergence for the nonzero residual case, and maintain at least locally, the positive definiteness of the approximate Hessian matrices.

Key words. nonlinear least squares problem, structured secant method, factorized Broyden-like method, q-quadratic convergence, q-superlinear convergence

AMS subject classifications. 65K05, 49D37, 90C30

PII. S1052623498345300

1. Introduction.

In this paper, we consider local algorithms to solve the problem

$$\min_{x\in R^n} f(x) = \frac{1}{2}\|R(x)\|_2^2 = \frac{1}{2}\sum_{i=1}^{l} {r_i}^2(x), \tag{1.1}$$

with zero or nonzero residual, where $R(x) = (r_1(x), \ldots, r_l(x))^T$ and the norm $\|R(x)\|$ is called the residual at the point x. Our algorithm to solve this problem is especially designed to be efficient in the two cases where the residual can vanish or not at the solution.

If we set $\bar{C}(x) = J^T(x)J(x)$, where $J(x)$ is the Jacobian matrix of $R(x)$, and $A(x) = \sum_{i=1}^{l} r_i(x) \bigtriangledown^2 r_i(x)$, then

$$\begin{aligned} J(x) &= (\frac{\partial r_i(x)}{\partial x_j}),\ i = 1, \ldots, l,\ j = 1, \ldots, n, \\ g(x) &= J^T(x)R(x), \\ \bigtriangledown^2 f(x) &= \bar{C}(x) + A(x), \end{aligned} \tag{1.2}$$

where $g(x) = \bigtriangledown f(x)$. Thus, if the Jacobian matrix $J(x)$ is available, we know the first part $\bar{C}(x)$ of the Hessian $\bigtriangledown^2 f(x)$ without requiring any second order information. The inherent structure leads to a lot of special methods for the least squares probem (1.1), such as those in [2], [4], and [5]. The Gauss–Newton method and the Levenberg–Marquadt method mentioned, respectively, in [4], [5], and [10] are two typical methods. These two methods are based on the observation that the second part $A(x)$ can be negligible when x is sufficiently close to the optimal point x^* at which $f(x^*)$ is small or equal to zero. They have a q-quadratic convergence rate for problems with zero

*Received by the editors September 25, 1998; accepted for publication (in revised form) December 20, 1999; published electronically June 13, 2000. This work was partially supported by the Hong Kong Universities Grant Council through grant CERG CITYU-9040289.

http://www.siam.org/journals/siopt/10-4/34530.html

†Department of Mathematics, City University of Hong Kong, Hong Kong (mazhang@cityu.edu.hk, clh1014@yahoo.com).

‡Division of Basic Science, China Agriculture University, Beijing, China (dengny@ihw.com.cn).

residual at the solution, but they may perform poorly when the residual is nonzero at the solution and the residual function is highly nonlinear [1]. The main reason is that in this case the deletion of the second derivative term $A(x)$ leads the methods to linear convergence. To overcome this difficulty, structured secant methods such as those proposed in [3] can be used. These methods get the next point x_{k+1} from x_k and A_k, an approximation to $A(x_k)$, as follows:

Step 1. let

$$B_k = J^T(x_k)J(x_k) + A_k; \tag{1.3}$$

Step 2. solve $B_k s = -g(x_k)$ to obtain s_k;

Step 3. let $x_{k+1} = x_k + s_k$;

Step 4. update B_k as following:

$$\begin{aligned} A_{k+1} &= \text{ Update } (A_k, s_k, y_k); \\ B_{k+1} &= J^T(x_{k+1})J(x_{k+1}) + A_{k+1}, \end{aligned} \tag{1.4}$$

where $y_k = g(x_{k+1}) - g(x_k)$. The structured secant methods can produce a q-superlinear convengence for both zero and nonzero residual cases if the update formula for A_k is chosen carefully.

We know that when a line search globalization technique is used, it is often preferred to maintain the positive definiteness of the working matrices so that the resulting directions s_k are descent directions of f. However, for structured secant methods, most update formulas for A_k do not guarantee B_k to preserve positive definite property. Yabe and Takahashi [7], [8] and Yabe and Yamaki [9] proposed a factorized quasi-Newton method to compute the search direction s_k by solving the equations

$$(J(x_k) + L_k)^T(J(x_k) + L_k)s = -g(x_k), \tag{1.5}$$

where the matrix L_k is an $l \times n$ correction of the Jacobian matrix $J(x_k)$ such that $(J(x_k)+L_k)^T(J(x_k)+L_k)$ is an approximation to $\bigtriangledown^2 f(x_k)$, which is at least positive semidefinite. In fact as shown by them, the matrix $(J(x_k) + L_k)^T(J(x_k) + L_k)$ is locally positive definite under some mild conditions. According to this idea, they proposed BFGS-like and DFP-like updates for L_k in [7] and showed their superlinear convergence properties in [8]. In [9], they proposed a Broyden-like family which includes the above two updates as special cases.

Their factorized quasi-Newton methods achieve q-superlinear convergence for both zero and nonzero residual cases, but lose q-quadratic convergence for zero residual problems, which otherwise can be obtained by the Gauss–Newton method and the Levenberg–Marquadt method. In order to obtain a q-quadratic convergence rate for the zero residual case, Huschens proposed a family of secant methods in [6]. His main idea is to rewrite $\bigtriangledown^2 f(x)$ as

$$\bigtriangledown^2 f(x) = \bar{C}(x) + ||R(x)|| \sum_{i=1}^{l} \frac{r_i(x)}{||R(x)||} \bigtriangledown^2 r_i(x),$$

and to approximate

$$\sum_{i=1}^{l} \frac{r_i(x)}{||R(x)||} \bigtriangledown^2 r_i(x)$$

by a suitable matrix. The extracted multiplier $||R(x)||$ in front of the sum plays a role of self-scaling that will, at least locally, give the right tendency for sizing the approximation to the matrix $A(x)$. The self-scaling technique brings about quadratic convergence to his secant methods for the zero residual case. However, his approximation to the Hessian $\bigtriangledown^2 f(x)$ may not be positive definite, and hence the resulting search direction is in general not a descent one.

In this paper, we are going to improve the factorized quasi-Newton methods, proposed by Yabe and his partners, by exploiting Huschens' technique. The new methods still have a factorized structure and make the updated matrices positive definite, at least locally. They maintain local q-superlinear convergence for the nonzero residual case but speed up the convergence rate from q-superlinear to q-quadratic in the zero residual case.

The paper is organized as follows. We present the family of scaled factorized Broyden-like methods in section 2. In section 3, we analyze the local convergence and the convergence rate of these methods. Finally we give conclusions and further research directions in section 4. Unless stated otherwise, the vector norm $||\cdot||$ used in this paper is the l_2 norm, and the matrix norm is consistent with the l_2 vector norm. The weighted Frobenius norm $||X||_{M_T}$ for any matrix $X \in R^{n\times n}$ is defined by $||M_T X M_T||_F$ for a symmetric, positive definite matrix $M_T \in R^{n\times n}$.

2. The algorithms.

In [9] Yabe and Yamaki compute the approximate Hessian B_k by

$$B_k = (J(x_k) + L_k)^T (J(x_k) + L_k), \tag{2.1}$$

where L_k is updated by

$$\begin{aligned} L_{k+1} = L_k &+ (1 - \sqrt{\phi_k}) \left(\frac{L_k^{\#} s_k}{s_k^T B_k^{\#} s_k} \right) (\sqrt{\lambda_k} z_k - B_k^{\#} s_k)^T \\ &+ \sqrt{\phi_k} L_k^{\#} (\sqrt{\lambda_k} (B_k^{\#})^{-1} z_k - s_k) \left(\frac{z_k}{s_k^T z_k} \right)^T \end{aligned} \tag{2.2}$$

with

$$\begin{aligned} z_k^{\#} &= [J(x_{k+1}) - J(x_k)]^T R(x_{k+1}), && (2.3)\\ z_k &= J^T(x_{k+1}) J(x_{k+1}) s_k + z_k^{\#}, && (2.4)\\ L_k^{\#} &= J(x_{k+1}) + L_k, && (2.5)\\ B_k^{\#} &= L_k^{\#T} L_k^{\#}, && (2.6)\\ \phi_k &\in [0, \phi], \\ \lambda_k &= \frac{1}{(1-\phi_k)\frac{s_k^T z_k}{s_k^T B_k^{\#} s_k} + \phi_k \frac{z_k^T (B_k^{\#})^{-1} z_k}{s_k^T z_k}}, \end{aligned}$$

where ϕ is a given upper bound for all ϕ_k. The quasi-Newton methods with the working matrices B_k generated by this set of formulas guarantee only q-superlinear convergence in the zero residual case. The main reason is that in this case when k tends to ∞, these $\{B_k\}$ may not converge to $J(x^*)^T J(x^*)$, the Hessian at x^*. To further improve their result, we approximate the Hessian $\bigtriangledown^2 f(x_k)$ by the matrix

$$B_k = (J(x_k) + ||R(x_k)||L_k)^T (J(x_k) + ||R(x_k)||L_k), \tag{2.7}$$

where matrix L_k can be updated in such a way that B_k satisfies a secant equation and inherits at least locally the positive definiteness. The role of the factor $||R(x_k)||$ in (2.7) is to yield a self-scaling property which will asymptotically give the right tendency by sizing the approximation to the second part $\sum_{i=1}^{l} r_i(x_k) \bigtriangledown^2 r_i(x_k)$ of the Hessian $\bigtriangledown^2 f(x_k)$; e.g., for the zero residual case ($||R(x^*)|| = 0$), the approximate Hessians $\{B_k\}$ calculated by (2.7) converge to the Hessian $J(x^*)^T J(x^*)$ at x^* when k tends to ∞. Furthermore, in order to make the resulting methods converge q-quadratically in the zero residual case and q-superlinearly in the nonzero residual case, we exploit Huschens' idea to introduce the multiplier $\frac{||R(x_{k+1})||}{||R(x_k)||}$ to the term $[J(x_{k+1}) - J(x_k)]^T R(x_{k+1})$ which is a usual approximation to $\sum_{i=1}^{l} r_i(x_{k+1}) \bigtriangledown^2 r_i(x_{k+1}) s_k$ (see [3] and [4]). That is, we introduce a new approximation

$$\sum_{i=1}^{l} r_i(x_{k+1}) \bigtriangledown^2 r_i(x_{k+1}) s_k \approx [J(x_{k+1}) - J(x_k)]^T \frac{R(x_{k+1})}{||R(x_k)||} ||R(x_{k+1})||, \tag{2.8}$$

where $s_k = x_{k+1} - x_k$. Since

$$\begin{aligned} \bigtriangledown^2 f(x_{k+1}) s_k &= J^T(x_{k+1}) J(x_{k+1}) s_k + \sum_{i=1}^{l} r_i(x_{k+1}) \bigtriangledown^2 r_i(x_{k+1}) s_k \\ &\approx J^T(x_{k+1}) J(x_{k+1}) s_k + \|R(x_{k+1})\| z_k^{\#}, \end{aligned}$$

where $z_k^{\#} = [J(x_{k+1}) - J(x_k)]^T \cdot \frac{R(x_{k+1})}{||R(x_k)||}$, a secant equation would be

$$B_{k+1} s_k = z_k, \tag{2.9}$$

where $z_k = J^T(x_{k+1}) J(x_{k+1}) s_k + ||R(x_{k+1})|| z_k^{\#}$. We now modify the formulas (2.2)–(2.6) to obtain a new updating formula:

$$\begin{aligned} L_{k+1} = \frac{||R(x_{k+1})||}{||R(x_k)||} L_k + \Bigg[&(1 - \sqrt{\phi_k}) \left(\frac{L_k^{\#} s_k}{s_k^T B_k^{\#} s_k} \right) \left(\sqrt{\lambda_k} z_k - B_k^{\#} s_k \right)^T \\ &+ \sqrt{\phi_k} L_k^{\#} \left(\sqrt{\lambda_k} (B_k^{\#})^{-1} z_k - s_k \right) \left(\frac{z_k}{s_k^T z_k} \right)^T \Bigg] \frac{1}{||R(x_{k+1})||}, \end{aligned} \tag{2.10}$$

where

$$z_k^{\#} = [J(x_{k+1}) - J(x_k)]^T \frac{R(x_{k+1})}{||R(x_k)||}, \tag{2.11}$$

$$z_k = J^T(x_{k+1}) J(x_{k+1}) s_k + ||R(x_{k+1})|| z_k^{\#}, \tag{2.12}$$

$$L_k^{\#} = J(x_{k+1}) + \frac{||R(x_{k+1})||^2}{||R(x_k)||} L_k, \tag{2.13}$$

$$
\begin{aligned}
(2.14)\qquad & B_k^{\#} = L_k^{\#T} L_k^{\#}, \\
& \phi_k \in [0, \phi], \\
& \lambda_k = \frac{1}{(1-\phi_k)\frac{s_k^T z_k}{s_k^T B_k^{\#} s_k} + \phi_k \frac{z_k^T (B_k^{\#})^{-1} z_k}{s_k^T z_k}},
\end{aligned}
$$

and propose our scaled factorized Broyden-like methods as follows.

A family of scaled factorized Broyden-like methods (SFB).
(Initial Step). Choose an initial approximation $x_0 \in R^n$ to the solution of problem (1.1) and an initial matrix $L_0 \in R^{l\times n}$. Compute $B_0 = (J(x_0)+||R(x_0)||L_0)^T(J(x_0)+||R(x_0)||L_0)$ and set $k=0$. Assume $g(x_0)\neq 0$ and $R(x_0)\neq 0$.
(Iterative Steps). Generally, for given $x_k \in R^n$, $L_k \in R^{l\times n}$, and $B_k \in R^{n\times n}$ the steps for getting x_{k+1}, L_{k+1}, and B_{k+1} for $k = 0, 1, 2, \ldots$, are as follows:
Step 1. Obtain s_k by solving

$$(2.15)\qquad B_k s = -g(x_k).$$

Step 2. Set $x_{k+1} = x_k + s_k$. If $g(x_{k+1}) = 0$ or $R(x_{k+1}) = 0$, stop; otherwise
Step 3. compute $z_k^{\#}$, z_k, $L_k^{\#}$, and $B_k^{\#}$ by (2.11)–(2.14).
Step 4. Update L_k and B_k by (2.10) and

$$(2.16)\qquad B_{k+1} = (J(x_{k+1}) + ||R(x_{k+1})||L_{k+1})^T(J(x_{k+1}) + ||R(x_{k+1})||L_{k+1}).$$

Step 5. Let $k \leftarrow k+1$ and go to Step 1.

Remark 1. We exploit Yabe et al.'s [7]–[9] factorized technique and Huschens' [6] idea in our family of scaled factorized Broyden-like methods. The two multipliers before L_k that we introduce in (2.10) and (2.13), respectively, aim to achieve the convergence properties which we expect.

Remark 2. By (2.10) and (2.16), it is easy to obtain

$$(2.17)\qquad B_{k+1} = B_k^{\#} - \frac{B_k^{\#} s_k s_k^T B_k^{\#}}{s_k^T B_k^{\#} s_k} + \frac{z_k z_k^T}{s_k^T z_k} + \phi_k (s_k^T B_k^{\#} s_k) v_k v_k^T,$$

where

$$v_k = \frac{B_k^{\#} s_k}{s_k^T B_k^{\#} s_k} - \frac{z_k}{s_k^T z_k}.$$

Therefore, the B_{k+1} obtained at Step 4 satisfies the secant equation (2.9). Note that (2.17) is the well-known updating formula of the Broyden family to obtain B_{k+1} from $B_k^{\#}$ (not from B_k), and the cases $\phi_k = 0$ and $\phi_k = 1$ in (2.17) are equivalent to BFGS-like update and DFP-like update, respectively, from $B_k^{\#}$ to B_{k+1}. Thus, we call (2.17) a scaled factorized Broyden-like family and our methods can be described as scaled factorized Broyden-like methods.

Remark 3. In this paper we discuss only local behavior of the new methods. Hence no line search is imposed and we take a unit step length.

3. Convergence analysis. In this section, we show that the family of scaled factorized Broyden-like methods yields q-quadratic convergence for the zero residual problems and q-superlinear convergence for the nonzero residual case. The following assumptions for problem (1.1) will be used in the rest of the paper. Let $D = \{x : ||x - x^*|| \leq \epsilon_1\}$, where $\epsilon_1 > 0$ is a sufficiently small constant.

(A1) The point $x^* \in R^n$ is a local minimizer of problem (1.1) in D.

(A2) In D, $f(x)$ is twice continuously differentiable and there exists a constant $C > 0$ such that for all $x, x_+ \in D$,

$$\begin{aligned} || \bigtriangledown^2 f(x) - \bigtriangledown^2 f(x_+)|| &\leq C||x - x_+||, \\ ||\bar{C}(x) - \bar{C}(x_+)|| &\leq C||x - x_+||, \\ ||J(x) - J(x_+)|| &\leq C||x - x_+||, \\ ||R(x) - R(x_+)|| &\leq C||x - x_+||, \\ ||g(x) - g(x_+)|| &\leq C||x - x_+||. \end{aligned}$$

(A3) The matrix $\bigtriangledown^2 f(x^*)$ is positive definite and there exist constants m and M: $0 < m < M$ and $M > 1$, such that for all $x \in D$ and $d \in R^n$,

$$m||d||^2 \leq \frac{1}{2} d^T \bigtriangledown^2 f(x) d \leq M||d||^2.$$

It follows easily from Assumption (A2) that for any x, $\hat{x} \in D$,

$$||g(x) - g(\hat{x}) - \bigtriangledown^2 f(x^*)(x - \hat{x})|| \leq C\sigma(x, \hat{x})||x - \hat{x}||, \tag{3.1}$$

where $\sigma(x, \hat{x}) = \max\{||x - x^*||, ||\hat{x} - x^*||\}$.

3.1. *Q*-quadratic convergence for zero residual problems. We divide the convergence analysis of the SFB methods into two parts: the convergence result for zero residual problems and the convergence result for nonzero residual problems. The proof of the local q-quadratic convergence in the zero residual case is based on the bounded deterioration property of L_k and the use of the product structure in factorized secant methods, whereas the proof of the local q-superlinear convergence in the nonzero residual case mainly exploits the idea which was used by Yabe and Yamaki in [9] to show the convergence properties of the factorized Broyden-like family. We start with the zero residual case.

THEOREM 3.1. *Suppose $R(x^*) = 0$. Then, there exist ϵ, $\delta > 0$ such that for any initial point x_0 with $||x_0 - x^*|| < \epsilon$ and any $L_0 \in R^{l \times n}$ with $||L_0|| < \delta$, the sequence $\{x_k\}$ generated by any SFB method is well defined and converges at a q-quadratic rate to x^*.*

Proof. Let $B^* = \bigtriangledown^2 f(x^*) = J^T(x^*)J(x^*)$. Since $J^T(x^*)J(x^*)$ is positive definite, there exist $\epsilon_1^* > 0$ and $M' > m' > 0$ such that for all $d \in R^n$ and $||x - x^*|| \leq \epsilon_1^*$,

$$m'||d||^2 \leq d^T J^T(x) J(x) d \leq M'||d||^2. \tag{3.2}$$

Choose $\epsilon > 0$ and $\delta > 0$ sufficiently small such that

$$\epsilon < \min\left\{\epsilon^*, \ \frac{m}{\theta^* K_1 K_2 + (C + \sqrt{\frac{M}{m}} C^2)}\right\}, \ \delta < \delta^*, \tag{3.3}$$

where

$$\delta^* = \min\left\{\frac{1}{M},\ 1\right\}, \tag{3.4}$$

$$\begin{aligned}\epsilon^* \le \min\Bigg\{&1,\ \epsilon_1,\ \epsilon_1^*,\ \frac{1}{C},\ \frac{m}{\gamma^*+K},\ \frac{m}{C+\sqrt{\frac{M}{m}}C^2},\\ &\frac{m'}{2C\sqrt{\frac{M}{m}}(C+||J(x^*)||)+\frac{M}{m}C^2}\Bigg\}.\end{aligned} \tag{3.5}$$

The constants $K,\ K_1,\ \gamma^*,\ \theta^*$, and K_2 in (3.3) and (3.5) are given by

$$K = C(C+||J(x^*)||)(1+2\delta^*)+C^2\delta^{*2}, \tag{3.6}$$

$$K_1 = C+\sqrt{\frac{M}{m}}C^2+2M, \tag{3.7}$$

$$\gamma^* = C+2MC^2\sqrt{\frac{M}{m}}+2CM\sqrt{\frac{M}{m}}||J(x^*)||+C^2\frac{M^3}{m},$$

$$\theta^* = \frac{1}{m-\epsilon^*\gamma^*}, \tag{3.8}$$

$$K_2 = \theta^* K\left[C^2\sqrt{\frac{M}{m}}+2C\sqrt{\frac{M}{m}}(C+||J(x^*)||)\delta^*+\frac{M\theta^*K}{m}C^2\delta^{*2}\right]. \tag{3.9}$$

Furthermore, the constants below will be used later:

$$\gamma = \epsilon\left[C+2MC^2\sqrt{\frac{M}{m}}+2CM\sqrt{\frac{M}{m}}||J(x^*)||+C^2\frac{M^3}{m}\right] = \epsilon\gamma^*,$$

$$\theta = \frac{1}{m-\gamma}, \tag{3.10}$$

$$\bar{\theta} = \theta K\epsilon, \tag{3.11}$$

$$K_3 = 2m-\left(C+\sqrt{\frac{M}{m}}C^2\right)\epsilon, \tag{3.12}$$

$$K_4 = M'+2C\sqrt{\frac{M}{m}}[C+||J(x^*)||]\delta+\frac{M}{m}C^2\delta^2, \tag{3.13}$$

$$K_5 = m'-2C\epsilon\sqrt{\frac{M}{m}}[C+||J(x^*)||]-C^2\epsilon\frac{M}{m}, \tag{3.14}$$

$$K_6 = \frac{(1+\phi+\phi\theta K_4)K_1K_2}{K_3^2}. \tag{3.15}$$

Note that as $0 < \epsilon < \epsilon^* \le \frac{m}{\gamma^*}$, θ is well defined, positive, and satisfies $\theta < \theta^*$. Moreover, from $\epsilon^* \le \frac{m}{\gamma^*+K}$ it is easy to see that $\bar{\theta} < 1$ since

$$\bar{\theta} = \theta K\epsilon < \theta^* K\epsilon^* \le \frac{K}{m-\frac{m\gamma^*}{\gamma^*+K}}\frac{m}{\gamma^*+K} = 1.$$

With the ϵ and δ satisfying (3.3), we will prove the following results by induction: there exist constants $\alpha_1 > 0$ and $\alpha_2 > 0$ such that for all k,

$$(1)\ ||x_{k+1}-x^*|| \le \theta K||x_k-x^*||^2 \le \bar{\theta}||x_k-x^*||; \tag{3.16}$$

(3.17) (2) $||L_{k+1}|| \le (1+\alpha_1\sigma(x_{k+1},x_k))||L_k|| + \alpha_2\sigma(x_{k+1},x_k)$;

(3.18) (3) $||L_{k+1}|| \le 1$;

(3.19) (4) $||B_{k+1} - B^*|| < \gamma$.

In fact (3.16) just means that $\{x_k\}$ converges to x^* q-quadratically. By the way, (3.19) ensures that all B_k are positive definite.

By (3.3), (3.4), and $||L_0|| < \delta$, we obtain $||L_0|| < 1$. Thus, by assumptions (A1)–(A2) and $R(x^*) = 0$, we have

$$\begin{aligned}
&||B_0 - B^*|| \\
&= ||(J(x_0) + ||R(x_0)||L_0)^T(J(x_0) + ||R(x_0)||L_0) - J^T(x^*)J(x^*)|| \\
&\le ||J^T(x_0)J(x_0) - J^T(x^*)J(x^*)|| + 2||R(x_0)|| \cdot ||L_0|| \cdot ||J(x_0)|| + ||R(x_0)||^2||L_0||^2 \\
&\le C||x_0 - x^*|| + 2C||x_0 - x^*|| \cdot ||L_0||(C||x_0 - x^*|| + ||J(x^*)||) \\
&\quad + C^2||L_0||^2||x_0 - x^*||^2 \\
&= \{C + 2C^2||L_0|| \cdot ||x_0 - x^*|| + 2C||L_0||||J(x^*)|| \\
&\quad + C^2||L_0||^2||x_0 - x^*||\}\ ||x_0 - x^*|| \\
&< \epsilon[C + 2MC^2 + 2CM||J(x^*)|| + C^2M^2] \le \gamma = \epsilon\gamma^* < \epsilon^*\gamma^* \le m.
\end{aligned} \tag{3.20}$$

By (A3) and (3.20), for all $d \in R^n$ we have

$$d^TB_0d \ge 2m||d||^2 - d^T(B^* - B_0)d \ge (2m - \epsilon\gamma^*)||d||^2 > 0.$$

Therefore, B_0 is positive definite. Recalling the Banach perturbation lemma, Theorem 3.1.4 of [4], we have

$$||B_0^{-1}|| \le \theta \text{ for all } B_0 \in R^{n\times n} \text{ with } ||B_0 - B^*|| < \gamma, \tag{3.21}$$

where θ is defined by (3.10). Using the mean value theorem and by (A2), (3.21), and the fact $||R(x^*)|| = 0$, we have

$$\begin{aligned}
&||x_1 - x^*|| \\
&= ||x_0 - x^* - B_0^{-1}J^T(x_0)R(x_0)|| \\
&\le ||B_0^{-1}||[||B_0(x_0 - x^*) - J^T(x_0)R(x_0)||] \\
&= ||B_0^{-1}||[||(J(x_0) + ||R(x_0)||L_0)^T(J(x_0) + ||R(x_0)||L_0)(x_0 - x^*) - J^T(x_0)R(x_0)||] \\
&\le ||B_0^{-1}||[||J^T(x_0)J(x_0)(x_0 - x^*) - J^T(x_0)R(x_0) + J^T(x_0)R(x^*)|| + 2||x_0 - x^*|| \\
&\quad \cdot||R(x_0) - R(x^*)|| \cdot ||L_0|| \cdot ||J(x_0)|| + ||R(x_0) - R(x^*)||^2||L_0||^2||x_0 - x^*||] \\
&\le \theta[C||J(x_0)|| \cdot ||x_0 - x^*||^2 + 2C\delta||J(x_0)|| \cdot ||x_0 - x^*||^2 + C^2\delta^2||x_0 - x^*||^3] \\
&\le \theta[C(C + ||J(x^*)||)(1 + 2\delta^*) + C^2\delta^{*2}]||x_0 - x^*||^2 \\
&= \theta K||x_0 - x^*||^2 \le \bar{\theta}||x_0 - x^*||.
\end{aligned} \tag{3.22}$$

Noticing that for all x close enough to x^* and $\theta(x) \in (0,1)$,

$$\begin{aligned}
2M||x - x^*||^2 \ge ||R(x)||^2 &= (x - x^*)^T \bigtriangledown^2 f(x^* + \theta(x)(x - x^*))(x - x^*) \\
&\ge 2m||x - x^*||^2,
\end{aligned} \tag{3.23}$$

and hence

$$\begin{aligned}
&||z_0 - B^* s_0|| \\
&= ||J(x_1)^T J(x_1) s_0 + \tfrac{||R(x_1)||}{||R(x_0)||}(J(x_1) - J(x_0))^T (R(x_1) - R(x^*)) - J^T(x^*) J(x^*) s_0|| \\
&\le C||x_1 - x^*|| \cdot ||s_0|| + \sqrt{\tfrac{M}{m}} \tfrac{||x_1 - x^*||}{||x_0 - x^*||} C^2 ||s_0|| \cdot ||x_1 - x^*|| \\
&\le [C + \sqrt{\tfrac{M}{m}} C^2] \sigma(x_0, x_1) ||s_0||.
\end{aligned} \tag{3.24}$$

So,

$$\begin{aligned}
s_0^T z_0 &\le ||s_0|| ||z_0 - B^* s_0|| + s_0^T B^* s_0 \\
&\le \left[C + \sqrt{\frac{M}{m}} C^2 \right] \sigma(x_0, x_1) ||s_0||^2 + 2M||s_0||^2 \\
&\le \left[C + \sqrt{\frac{M}{m}} C^2 + 2M \right] ||s_0||^2 \\
&= K_1 ||s_0||^2.
\end{aligned} \tag{3.25}$$

On the other hand, with (3.24) and (3.12) we get

$$\begin{aligned}
s_0^T z_0 &\ge 2m||s_0||^2 - \left[C + \sqrt{\frac{M}{m}} C^2 \right] \sigma(x_0, x_1) ||s_0||^2 \\
&\ge \left[2m - \left(C + \sqrt{\frac{M}{m}} C^2 \right) \epsilon \right] ||s_0||^2 = K_3 ||s_0||^2.
\end{aligned} \tag{3.26}$$

Notice that (3.22) gives

$$||x_1 - x^*|| \le \bar{\theta} ||x_0 - x^*|| < \epsilon < 1.$$

Thus, by (2.14), (3.2), (3.23), $||R(x_1)|| \le C||x_1 - x^*||$, and (3.13), for all $d \in R^n$ we have

$$\begin{aligned}
d^T B_0^{\#} d &= d^T \left(J(x_1) + \frac{||R(x_1)||^2}{||R(x_0)||} L_0 \right)^T \left(J(x_1) + \frac{||R(x_1)||^2}{||R(x_0)||} L_0 \right) d \\
&= d^T J^T(x_1) J(x_1) d + \frac{||R(x_1)||^2}{||R(x_0)||} d^T J^T(x_1) L_0 d \\
&\quad + \frac{||R(x_1)||^2}{||R(x_0)||} d^T L_0^T J(x_1) d + \frac{||R(x_1)||^4}{||R(x_0)||^2} d^T L_0^T L_0 d \\
&\le M' ||d||^2 + 2C \sqrt{\frac{M}{m}} \frac{||x_1 - x^*||^2}{||x_0 - x^*||} [C||x_1 - x^*|| + ||J(x^*)||] \delta ||d||^2 \\
&\quad + C^2 \frac{M}{m} \frac{||x_1 - x^*||^4}{||x_0 - x^*||^2} \delta^2 ||d||^2 \\
&\le \left[M' + 2C \sqrt{\frac{M}{m}} (C + ||J(x^*)||) \delta + C^2 \frac{M}{m} \delta^2 \right] ||d||^2 = K_4 ||d||^2.
\end{aligned} \tag{3.27}$$

Similar to (3.27) and by (3.14),

$$d^T B_0^{\#} d \geq \left[m' - 2C\sqrt{\frac{M}{m}}\epsilon(C + ||J(x^*)||) - \epsilon\frac{M}{m}C^2 \right] ||d||^2$$

$$= K_5||d||^2 \text{ for all } d \in R^n. \tag{3.28}$$

Thus, with (3.22), (3.23), $||R(x^*)|| = 0$, and (3.9) we get

$$\begin{aligned}
&||z_0 - B_0^{\#}s_0||\\
&= ||J^T(x_1)J(x_1)s_0 + \tfrac{||R(x_1)||}{||R(x_0)||}(J(x_1) - J(x_0))^T(R(x_1) - R(x^*)) - J^T(x_1)J(x_1)s_0\\
&-\tfrac{||R(x_1)||^2}{||R(x_0)||}J^T(x_1)L_0s_0 - \tfrac{||R(x_1)||^2}{||R(x_0)||}L_0^TJ(x_1)s_0 - \tfrac{||R(x_1)||^4}{||R(x_0)||^2}L_0^TL_0s_0||\\
&\leq C^2\sqrt{\tfrac{M}{m}}\tfrac{||x_1-x^*||}{||x_0-x^*||}||s_0||\cdot||x_1 - x^*|| + 2\tfrac{||R(x_1)||^2}{||R(x_0)||}||J(x_1)||\cdot||L_0||\cdot||s_0||\\
&+\tfrac{||R(x_1)||^4}{||R(x_0)||^2}||L_0||^2||s_0||\\
&\leq C^2\theta K\sqrt{\tfrac{M}{m}}||x_0 - x^*||\cdot||s_0||\cdot||x_1 - x^*|| + 2C\theta K\sqrt{\tfrac{M}{m}}||x_1 - x^*||\cdot||x_0 - x^*||\\
&\cdot||J(x_1)||\cdot||L_0||\cdot||s_0|| + C^2\theta^2K^2\tfrac{M}{m}||x_0 - x^*||^2||x_1 - x^*||^2||L_0||^2||s_0||\\
&\leq C^2\theta K\sqrt{\tfrac{M}{m}}||s_0||\cdot||x_1 - x^*||\sigma(x_1, x_0) + 2\theta K\sqrt{\tfrac{M}{m}}C||x_1 - x^*||\sigma(x_1, x_0)\cdot\\
&||J(x_1)||\cdot||L_0||\cdot||s_0|| + \tfrac{M}{m}C^2\theta^2K^2||x_1 - x^*||\sigma(x_1, x_0)||L_0||^2||s_0||\\
&\leq \theta K\left[C^2\sqrt{\tfrac{M}{m}} + 2\sqrt{\tfrac{M}{m}}C\delta^*(C + ||J(x^*)||) + \tfrac{M\theta K}{m}C^2\delta^{*2}\right]\\
&\cdot||x_1 - x^*||\cdot||s_0||\sigma(x_1, x_0)\\
&\leq K_2||x_1 - x^*||\cdot||s_0||\sigma(x_1, x_0).
\end{aligned} \tag{3.29}$$

By (3.24),

$$||z_0|| \leq ||z_0 - B^*s_0|| + ||B^*s_0|| \leq K_1||s_0||, \tag{3.30}$$

where K_1 is defined by (3.7). Hence, by (2.10), (3.23), (3.26), (3.28), (3.30), and

$$|\sqrt{\lambda_0} - 1| = \frac{|\lambda_0 - 1|}{|\sqrt{\lambda_0} + 1|} \leq |\lambda_0 - 1|,$$

we have

$$\begin{aligned}
||L_1|| &= \left|\left|\frac{||R(x_1)||}{||R(x_0)||}L_0 + \left[(1 - \sqrt{\phi_k})\left(\frac{L_0^{\#}s_0}{s_0^TB_0^{\#}s_0}\right)(\sqrt{\lambda_0}z_0 - B_0^{\#}s_0)^T\right.\right.\right.\\
&\quad\left.\left.\left. + \sqrt{\phi_k}L_0^{\#}\Big(\sqrt{\lambda_0}(B_0^{\#})^{-1}z_0 - s_0\Big)\left(\frac{z_0}{s_0^Tz_0}\right)^T\right]\frac{1}{||R(x_1)||}\right|\right|\\
&\leq \frac{||R(x_1)||}{||R(x_0)||}||L_0|| + \left[|1 - \sqrt{\phi_k}|\left|\left|\frac{L_0^{\#}s_0}{s_0^TB_0^{\#}s_0}\right|\right|\cdot||\sqrt{\lambda_0}z_0 - B_0^{\#}s_0||\right.\\
&\quad\left. + \sqrt{\phi_k}||L_0^{\#}(B_0^{\#})^{-1}||\cdot||\sqrt{\lambda_0}z_0 - B_0^{\#}s_0||\cdot\left|\left|\frac{z_0}{s_0^Tz_0}\right|\right|\right]\frac{1}{||R(x_1)||}\\
&\leq \theta K\sqrt{\frac{M}{m}}\sigma(x_1, x_0)||L_0|| + \left[|1 - \sqrt{\phi_k}|\frac{||L_0^{\#}||\cdot||s_0||}{K_5||s_0||^2}\right.\\
&\quad\left. + \sqrt{\phi_k}||L_0^{\#}||\cdot||(B_0^{\#})^{-1}||\frac{K_1||s_0||}{K_3||s_0||^2}\right]\frac{||\sqrt{\lambda_0}z_0 - B_0^{\#}s_0||}{||R(x_1)||}
\end{aligned}$$

$$\leq \theta K\sqrt{\frac{M}{m}}\sigma(x_1,x_0)||L_0|| + \left[|1-\sqrt{\phi_k}|\frac{||L_0^{\#}||}{K_5}\right.$$

$$\left. +\sqrt{\phi_k}||L_0^{\#}||\cdot||(B_0^{\#})^{-1}||\frac{K_1}{K_3}\right]\frac{|\lambda_0-1|||z_0||+||z_0-B_0^{\#}s_0||}{||s_0||\cdot||R(x_1)||}. \tag{3.31}$$

Notice that

$$\begin{aligned}
&||B_0^{\#}-B^*||\\
&= \left|\left|\left(J(x_1)+\frac{||R(x_1)||^2}{||R(x_0)||}L_0\right)^T\left(J(x_1)+\frac{||R(x_1)||^2}{||R(x_0)||}L_0\right)-J^T(x^*)J(x^*)\right|\right|\\
&\leq ||J^T(x_1)J(x_1)-J^T(x^*)J(x^*)||+2\frac{||R(x_1)||^2}{||R(x_0)||}||L_0||\cdot||J(x_1)||+\frac{||R(x_1)||^4}{||R(x_0)||^2}||L_0||^2\\
&\leq C||x_1-x^*||+2C\sqrt{\frac{M}{m}}||x_1-x^*||\cdot||L_0||(C||x_1-x^*||+||J(x^*)||)\\
&\quad +C^2\frac{M}{m}||L_0||^2||x_1-x^*||^2\\
&= \left[C+2C^2\sqrt{\frac{M}{m}}||L_0||||x_1-x^*||+2C\sqrt{\frac{M}{m}}||L_0||||J(x^*)||+C^2\frac{M}{m}||L_0||^2\right]||x_1-x^*||\\
&< \epsilon\left[C+2MC^2\sqrt{\frac{M}{m}}+2CM\sqrt{\frac{M}{m}}||J(x^*)||+\frac{M^3}{m}C^2\right]=\gamma.
\end{aligned}$$

(3.32)

So, by (3.21),

$$||(B_0^{\#})^{-1}||\leq\theta. \tag{3.33}$$

On the other hand,

$$\begin{aligned}
|z_0^TB_0^{\#-1}z_0| &\geq |z_0^Ts_0|-||z_0||\cdot||B_0^{\#-1}||\cdot||z_0-B_0^{\#}s_0||\\
&\geq K_3||s_0||^2-\theta K_1||s_0||K_2\sigma(x_1,x_0)||s_0||\\
&\geq (K_3-\theta K_1K_2\epsilon)||s_0||^2. \qquad (3.34)
\end{aligned}$$

Thus, (3.24)–(3.30), (3.33), and (3.34) give

$$\begin{aligned}
&|\lambda_0-1|\\
&=\left|\frac{1-(1-\phi_k)\frac{s_0^Tz_0}{s_0^TB_0^{\#}s_0}-\phi_k\frac{z_0^TB_0^{\#-1}z_0}{s_0^Tz_0}}{(1-\phi_k)\frac{s_0^Tz_0}{s_0^TB_0^{\#}s_0}+\phi_k\frac{z_0^TB_0^{\#-1}z_0}{s_0^Tz_0}}\right|\\
&=\left|\frac{(1-\phi_k)(s_0^TB_0^{\#}s_0-s_0^Tz_0)(s_0^Tz_0)+\phi_k(s_0^TB_0^{\#}s_0)(s_0^Tz_0-z_0^TB_0^{\#-1}z_0)}{(1-\phi_k)(s_0^Tz_0)^2+\phi_k(z_0^TB_0^{\#-1}z_0)(s_0^TB_0^{\#}s_0)}\right|\\
&=\left|\frac{(1-\phi_k)(s_0^TB_0^{\#}s_0-s_0^Tz_0)(s_0^Tz_0)+\phi_k(s_0^TB_0^{\#}s_0)(s_0^Tz_0-z_0^TB_0^{\#-1}z_0)}{(s_0^Tz_0)^2+\phi_k[(z_0^TB_0^{\#-1}z_0)(s_0^TB_0^{\#}s_0)-(s_0^Tz_0)^2]}\right|.
\end{aligned}$$

The Cauchy–Schwarz inequality yields

$$(z_0^TB_0^{\#-1}z_0)(s_0^TB_0^{\#}s_0)-(s_0^Tz_0)^2\geq 0.$$

Thus, by (3.26) we have

$$
\begin{aligned}
&|\lambda_0 - 1| \\
&\leq \frac{|1-\phi_k|K_1K_2\sigma(x_1,x_0)||x_1-x^*||\cdot||s_0||^4 + \phi_k\theta K_1K_4K_2\sigma(x_1,x_0)||x_1-x^*||\cdot||s_0||^4}{(s_0^Tz_0)^2} \\
&\leq \frac{(1+\phi+\phi\theta K_4)K_1K_2}{K_3^2}\sigma(x_1,x_0)||x_1-x^*|| \\
&= K_6\sigma(x_1,x_0)||x_1-x^*||,
\end{aligned}
\tag{3.35}
$$

where K_6 is defined in (3.15). Notice that

$$||L_k^{\#}|| = \left|\left|J(x_{k+1}) + \frac{||R(x_{k+1})||^2}{||R(x_k)||}L_k\right|\right|.$$

By (3.29), (3.30), (3.33), and (3.35), (3.31) becomes

$$
\begin{aligned}
||L_1|| &\leq \theta K\sqrt{\frac{M}{m}}\sigma(x_1,x_0)||L_0|| + \left[\frac{|1-\sqrt{\phi_k}|}{K_5} + \frac{\sqrt{\phi_k}\theta K_1}{K_3}\right]\left[C + ||J(x^*)||\right. \\
&\qquad \left. + CM\sqrt{\frac{M}{m}}||L_0||\right]\cdot\frac{||x_1-x^*||[K_1K_6\sigma(x_1,x_0)||s_0|| + K_2\sigma(x_1,x_0)||s_0||]}{\sqrt{m}||s_0||\cdot||x_1-x^*||} \\
&\leq \theta K\sqrt{\frac{M}{m}}\sigma(x_1,x_0)||L_0|| + \left[\frac{1+\sqrt{\phi}}{K_5} + \frac{\sqrt{\phi}\theta K_1}{K_3}\right] \\
&\qquad \cdot\left[C + ||J(x^*)|| + \sqrt{\frac{M}{m}}CM||L_0||\right]\frac{K_1K_6+K_2}{\sqrt{m}}\sigma(x_1,x_0).
\end{aligned}
\tag{3.36}
$$

Taking

$$
\begin{aligned}
\alpha_1 &= \theta K\sqrt{\frac{M}{m}} + CM\sqrt{\frac{M}{m}}\left[\frac{1+\sqrt{\phi}}{K_5} + \frac{\sqrt{\phi}\theta K_1}{K_3}\right]\frac{K_1K_6+K_2}{\sqrt{m}}, \\
\alpha_2 &= \left[\frac{1+\sqrt{\phi}}{K_5} + \frac{\sqrt{\phi}\theta K_1}{K_3}][C + ||J(x^*)||\right]\frac{K_1K_6+K_2}{\sqrt{m}},
\end{aligned}
$$

(3.36) becomes

$$||L_1|| \leq \alpha_1\sigma(x_1,x_0)||L_0|| + \alpha_2\sigma(x_1,x_0).$$

Hence, (3.17) is true for $k=0$. By this property and similar to the corresponding proof of Theorem 3.1 in [6], we can show that (3.18) holds for $k=0$:

$$||L_1|| \leq 1.$$

Since

$$
\begin{aligned}
&||B_1 - B^*|| \\
&= ||(J(x_1) + ||R(x_1)||L_1)^T(J(x_1) + ||R(x_1)||L_1) - J^T(x^*)J(x^*)|| \\
&\leq ||J^T(x_1)J(x_1) - J^T(x^*)J(x^*)|| + 2||R(x_1)||\cdot||L_1||\cdot||J(x_1)|| + ||R(x_1)||^2||L_1||^2 \\
&\leq C||x_1-x^*|| + 2C||x_1-x^*||\cdot||L_1||(C||x_1-x^*|| + ||J(x^*)||) + C^2||L_1||^2||x_1-x^*||^2 \\
&= [C + 2C^2||L_1||\cdot||x_1-x^*|| + 2C||L_1||||J(x^*)|| + C^2||L_1||^2||x_1-x^*||]||x_1-x^*|| \\
&< \epsilon[C + 2MC^2 + 2CM||J(x^*)|| + C^2M^2] \leq \gamma,
\end{aligned}
$$

(3.19) is true when $k = 0$. For $k = 1, 2, \ldots$, (3.16)–(3.19) can be proved inductively using the same arguments. □

Note that in our algorithm and the above proof, parameters ϕ_k are allowed to vary at iterations, and as long as they are upper bounded by a constant, say ϕ, the proof is valid.

3.2. *Q*-superlinear convergence for nonzero residual problems. In this subsection, we consider the superlinear convergence rate for the least squares problems with nonzero residual. In what follows, we define $M_T = \bigtriangledown^2 f(x^*)^{\frac{1}{2}}$. Note that by the equivalence of the norms, there exist positive constants η_1 and η_2 such that

$$\frac{1}{\eta_1}||\cdot||_{M_T} \leq ||\cdot|| \leq \eta_2||\cdot||_{M_T}. \tag{3.37}$$

The proof of the following theorem is based partially on the techniques used in [9].

THEOREM 3.2. *Suppose $||R(x^*)|| > 0$. Then there exist ϵ, $\delta > 0$ such that for any initial point x_0 with $||x_0 - x^*|| < \epsilon$ and any $L_0 \in R^{l\times n}$ with $||B_0^{-1} - \bigtriangledown^2 f(x^*)^{-1}||_{M_T} < \delta$, the sequence $\{x_k\}$ generated by Algorithm SFB converges to x^* q-superlinearly.*

Proof. Take $H_k = B_k^{-1}$ for $k = 0, 1, 2 \ldots$. Following the proof of Theorem 1 and Theorem 2 in [9] carefully, it is enough to show that for any $\nu \in (0, 1)$ and all $k = 0, 1, 2, \ldots$, there exist sufficiently small $\epsilon > 0$ and $\delta > 0$ such that

(1) $||x_{k+1} - x^*|| \leq \nu||x_k - x^*||$.

(2) There exists $\bar{K}_1 > 0$ such that

$$||z_k - \bigtriangledown^2 f(x^*)s_k|| \leq \bar{K}_1\sigma(x_{k+1}, x_k)||s_k||,$$

where $z_k = J^T(x_{k+1})J(x_{k+1})s_k + \frac{||R(x_{k+1})||}{||R(x_k)||}(J(x_{k+1}) - J(x_k))^T R(x_{k+1})$.

(3) There exist $\bar{K}_2 > 0$ and $\bar{K}_3 > 0$ such that

$$\bar{K}_2||s_k||^2 \leq s_k^T z_k \leq \bar{K}_3||s_k||^2.$$

(4) There exists $\bar{K}_4 > 0$ such that $||H_k|| < \bar{K}_4$, $||B_k|| < \bar{K}_4$, and $||L_k|| < \bar{K}_4$.

(5) There exists $\bar{K}_5 > 0$ such that

$$||B_k^{\#} - B_k|| \leq \bar{K}_5\sigma(x_{k+1}, x_k).$$

(6) There exist $\alpha_1 > 0$ and $\alpha_2 > 0$ such that

$$\begin{aligned}&||H_{k+1} - \bigtriangledown^2 f(x^*)^{-1}||_{M_T}\\ &\leq ||H_k - \bigtriangledown^2 f(x^*)^{-1}||_{M_T} + \alpha_1\sigma(x_{k+1}, x_k) - \alpha_2\frac{||\hat{s}_k - \hat{H}_k^{\#}\hat{s}_k||^2}{||\hat{s}_k||^2},\end{aligned}$$

where $\hat{s}_k = M_T s_k$, $\hat{H}_k^{\#} = M_T H_k^{\#} M_T$ and $H_k^{\#} = (B_k^{\#})^{-1}$, where $B_k^{\#}$ is defined in Algorithm SFB.

For (1), by the condition of the theorem, we know that $||B_0^{-1}|| \leq || \bigtriangledown^2 f(x^*)^{-1}|| + \delta$. Thus, by (3.1) and $g(x^*) = 0$, we have

$$\begin{aligned}||x_1 - x^*|| &= ||x_0 - x^* - B_0^{-1}g(x_0)||\\ &\leq ||x_0 - x^* - B_0^{-1}g(x_0) + B_0^{-1}g(x^*)\\ &\quad + B_0^{-1}\bigtriangledown^2 f(x^*)(x_0 - x^*) - B_0^{-1}\bigtriangledown^2 f(x^*)(x_0 - x^*)||\\ &\leq ||B_0^{-1}|| \cdot ||g(x_0) - g(x^*) - \bigtriangledown^2 f(x^*)(x_0 - x^*)||\\ &\quad + ||\bigtriangledown^2 f(x^*)||||B_0^{-1} - \bigtriangledown^2 f(x^*)^{-1}||||x_0 - x^*||\\ &\leq [C(||\bigtriangledown^2 f(x^*)^{-1}|| + \delta)||x_0 - x^*|| + \delta||\bigtriangledown^2 f(x^*)||]\ ||x_0 - x^*||\\ &\leq [C(||\bigtriangledown^2 f(x^*)^{-1}|| + \delta)\epsilon + ||\bigtriangledown^2 f(x^*)||\delta]\ ||x_0 - x^*||.\end{aligned}$$

Therefore, if $\epsilon \le \frac{\nu}{2C(||\bigtriangledown^2 f(x^*)^{-1}||+\delta)}$ and $\delta \le \frac{\nu}{2||\bigtriangledown^2 f(x^*)||}$, then $||x_1 - x^*|| \le \nu||x_0 - x^*||$.

We now turn to (2). Because $||R(x^*)|| > 0$, due to the continuity of $R(x)$ and $J(x)$, for any x sufficiently close to x^* there exist $M'' > 0$ and $m'' > 0$ such that

$$m'' \le ||R(x)|| \le M'', \; ||J(x)|| \le M''.$$

Now, by (A2) and (3.1), first we have

$$\begin{aligned}
&\big|\big|z_0 - \bigtriangledown^2 f(x^*)s_0\big|\big| \\
&= \left|\left| J^T(x_1)J(x_1)s_0 + \frac{||R(x_1)||}{||R(x_0)||}(J(x_1) - J(x_0))^T R(x_1) - \bigtriangledown^2 f(x^*)s_0 \right|\right| \\
&\le ||J^T(x_1)J(x_1)s_0 + (J(x_1) - J(x_0))^T R(x_1) - \bigtriangledown^2 f(x^*)s_0|| \\
&\quad + \frac{|||R(x_1)|| - ||R(x_0)|||}{||R(x_0)||}||J(x_1) - J(x_0)|| \cdot ||R(x_1)|| \\
&\le ||J^T(x_1)J(x_1)s_0 - J^T(x_0)J(x_0)s_0|| \\
&\quad + ||J^T(x_0)J(x_0)s_0 - J^T(x_0)R(x_1) + J^T(x_0)R(x_0)|| \\
&\quad + ||J^T(x_1)R(x_1) - J^T(x_0)R(x_0) - \bigtriangledown^2 f(x^*)s_0|| \\
&\quad + \frac{|||R(x_1)|| - ||R(x_0)|||}{||R(x_0)||}||J(x_1) - J(x_0)|| \cdot ||R(x_1)|| \\
&\le C||s_0||^2 + ||J(x_0)|| \cdot ||J(x_0)s_0 - (R(x_1) - R(x_0))|| + ||g(x_1) - g(x_0) - \bigtriangledown^2 f(x^*)s_0|| \\
&\quad + \frac{M''}{m''}||J(x_1) - J(x_0)|| \cdot ||R(x_1) - R(x_0)|| \\
&\le \left[3C + 2M''C + \frac{2C^2M''}{m''}\right] \sigma(x_1, x_0)||s_0||.
\end{aligned}$$

Taking $\bar{K}_1 = 3C + 2M''C + \frac{2C^2M''}{m''}$, (2) holds when $k = 0$.

For (3), noticing that by (A3) and (1),

$$\begin{aligned}
s_0^T z_0 &\ge ||s_0^T \bigtriangledown^2 f(x^*)s_0|| - ||s_0||||z_0 - \bigtriangledown^2 f(x^*)s_0|| \\
&\ge 2m||s_0||^2 - \bar{K}_1\sigma(x_1, x_0)||s_0||^2 \\
&\ge (2m - \bar{K}_1\epsilon)||s_0||^2
\end{aligned}$$

and

$$\begin{aligned}
s_0^T z_0 &\le ||s_0^T \bigtriangledown^2 f(x^*)s_0|| + ||s_0||||z_0 - \bigtriangledown^2 f(x^*)s_0|| \\
&\le 2M||s_0||^2 + \bar{K}_1\sigma(x_1, x_0)||s_0||^2 \\
&\le (2M + \bar{K}_1\epsilon)||s_0||^2.
\end{aligned}$$

We take $\bar{K}_2 = (2m - \bar{K}_1\epsilon)$ and $\bar{K}_3 = (2M + \bar{K}_1\epsilon)$. If $\epsilon < m/\bar{K}_1$, (3) holds when $k = 0$. Since by (3.37), we have

$$\begin{aligned}
||H_0|| &\le ||H_0 - \bigtriangledown^2 f(x^*)^{-1}|| + ||\bigtriangledown^2 f(x^*)^{-1}|| \\
&\le \delta\eta_2 + ||\bigtriangledown^2 f(x^*)^{-1}||.
\end{aligned}$$

The matrix H_0 is bounded. If $\delta \le \frac{\nu}{(\nu+1)\eta_2||\bigtriangledown^2 f(x^*)||}$, then

$$||\bigtriangledown^2 f(x^*)||||H_0 - \bigtriangledown^2 f(x^*)^{-1}|| \le \delta\eta_2||\bigtriangledown^2 f(x^*)|| \le \frac{\nu}{1+\nu} < 1.$$

By the Banach perturbation lemma,

$$||B_0|| = ||H_0^{-1}|| \leq (\nu+1)|| \bigtriangledown^2 f(x^*)||.$$

Thus,

$$\begin{aligned}
||L_0|| &= \left|\left| \frac{1}{||R(x_0)||}[J(x_0) + ||R(x_0)|| \cdot L_0 - J(x_0)]\right|\right| \\
&\leq \frac{1}{||R(x_0)||}[||J(x_0) + ||R(x_0)|| \cdot L_0|| + ||J(x_0)||] \\
&\leq \frac{1}{m''}[||B_0||^{\frac{1}{2}} + ||J(x_0)||] \\
&\leq \frac{1}{m''}[(\nu+1)^{\frac{1}{2}}|| \bigtriangledown^2 f(x^*)||^{\frac{1}{2}} + M''].
\end{aligned}$$

Taking $\bar{K}_4 = \max\{\delta\eta_2 + ||\bigtriangledown^2 f(x^*)^{-1}||,\ (\nu+1)||\bigtriangledown^2 f(x^*)||,\ \frac{1}{m''}[(\nu+1)^{\frac{1}{2}}||\bigtriangledown^2 f(x^*)||^{\frac{1}{2}} + M'']\}$, (4) is achieved when $k=0$.

For (5), we have

$$\begin{aligned}
&||B_0^{\#} - B_0|| \\
&\leq || \left(J(x_1) + \frac{||R(x_1)||^2}{||R(x_0)||}L_0\right)^T \left(J(x_1) + \frac{||R(x_1)||^2}{||R(x_0)||}L_0\right) \\
&\quad -(J(x_0) + ||R(x_0)||L_0)^T(J(x_0) + ||R(x_0)||L_0)|| \\
&\leq ||J^T(x_1)J(x_1) - J^T(x_0)J(x_0)|| + \left|\left| \frac{||R(x_1)||^2}{||R(x_0)||} L_0^T J(x_1) - ||R(x_0)||L_0^T J(x_0)\right|\right| \\
&\quad + \left|\left| \frac{||R(x_1)||^2}{||R(x_0)||} J^T(x_1) L_0 - ||R(x_0)||J^T(x_0)L_0\right|\right| \\
&\quad + \left|\left| \left(\frac{||R(x_1)||^2}{||R(x_0)||}\right)^2 L_0^T L_0 - ||R(x_0)||^2 L_0^T L_0\right|\right| \\
&\leq ||J^T(x_1)J(x_1) - J^T(x_0)J(x_0)|| + \frac{||||R(x_1)||^2 L_0^T J(x_1) - ||R(x_0)||^2 L_0^T J(x_0)||}{m''} \\
&\quad + \frac{||||R(x_1)||^2 J^T(x_1) L_0 - ||R(x_0)||^2 J^T(x_0) L_0||}{m''} + \frac{||||R(x_1)||^4 L_0^T L_0 - ||R(x_0)||^4 L_0^T L_0||}{m''^2}.
\end{aligned}$$

Since $R(x)$ and $J(x)$ are Lipschitz functions, so are $J^T(x)J(x)$, $||R(x)||^2 L_0^T J(x)$, $||R(x)||^2 J^T(x) L_0$, and $||R(x)||^4 L_0^T L_0$. By (1), $x_1 \in D$. Thus, there exists a constant L' such that

$$\begin{aligned}
||J^T(x_1)J(x_1) - J^T(x_0)J(x_0)|| &\leq L'||x_1 - x_0||, \\
||||R(x_1)||^2 L_0^T J(x_1) - ||R(x_0)||^2 L_0^T J(x_0)|| &\leq L'||x_1 - x_0||, \\
||||R(x_1)||^2 J^T(x_1) L_0 - ||R(x_0)||^2 J^T(x_0) L_0|| &\leq L'||x_1 - x_0||, \\
||||R(x_1)||^4 L_0^T L_0 - ||R(x_0)||^4 L_0^T L_0|| &\leq L'||x_1 - x_0||.
\end{aligned}$$

So,

$$||B_0^{\#} - B_0|| \leq L'\left(1 + \frac{2}{m''} + \frac{1}{m''^2}\right)||x_1 - x_0||.$$

Letting $\bar{K}_5 = L'(1 + \frac{2}{m''} + \frac{1}{m''^2})$, we have

$$||B_0^{\#} - B_0|| \leq \bar{K}_5 \sigma(x_1, x_0).$$

By (1)–(5) and similar to the proof of expression (E9;k) in Theorem 1 of [9], (6) holds for $k = 0$.

Following the inductive argument, we can show that for any $\nu \in (0, 1)$ and all $k = 0, 1, \ldots$, if $\epsilon > 0$ and $\delta > 0$ are sufficiently small, (1)–(6) hold.

Obviously, (1)–(6) imply that the sequence $\{x_k\}$ is well defined and converges linearly to the local minimizer x^*. Note that

$$\begin{aligned}\frac{||(B_k - \bigtriangledown^2 f(x^*))s_k||}{||s_k||} &\leq \frac{||M_T(M_T^{-1} B_k M_T^{-1} - I)M_T s_k||}{||M_T s_k||} \frac{||M_T s_k||}{||s_k||} \\ &\leq ||M_T||^2 \frac{||\hat{B}_k \hat{s}_k - \hat{s}_k||}{||\hat{s}_k||},\end{aligned}$$

where $\hat{B}_k = M_T^{-1} B_k M_T^{-1}$ and $\hat{s}_k$ is defined in (6). With (6) and similar to the proof of Theorem 2 in [9], we have

$$\lim_{k \longrightarrow \infty} \frac{||\hat{B}_k \hat{s}_k - \hat{s}_k||}{||\hat{s}_k||} = 0.$$

Thus,

$$\lim_{k \longrightarrow \infty} \frac{||(B_k - \bigtriangledown^2 f(x^*))s_k||}{||s_k||} = 0,$$

which concludes the q-superlinear convergence of our method. □

Note that conclusions (3) and (5) mean locally $s_k^T z_k > 0$ and $B_k^{\#}$ is positive definite if B_k is so. Since B_{k+1} is obtained by the Broyden's formula (2.17) from $B_k^{\#}$, by the theory of quasi-Newton methods we know that B_{k+1} remains positive definite if B_k is so.

4. Conclusions. In this paper we are concerned with structured secant methods for solving nonlinear least squares problems. We improve Yabe and his partners' factorized Broyden family of secant methods so that in the zero residual case the convergence rate of the new methods can be enhanced substantially from superlinear to quadratic; meanwhile, in the nonzero residual case, the new methods still maintain a superlinear convergence rate. Our main idea for this improvement is stimulated from Huschens' method to express the matrix $A(x)$, i.e., the part of second order information, in the Hessian matrix of the least squares problems as a product form so that a scaling factor $\|R(x)\|$ appears explicitly as a multiplier. Accordingly, we change the matrix L_k in the factorized form of the updating matrix B_k into $\|R(x_k)\|L_k$. This modification is successful in the zero residual case because L_k may not approach the zero matrix, but $\|R(x_k)\|L_k$ does. Generally speaking, the multiplier $\|R(x)\|$ plays a role of self-scaling to adjust the modification to the working matrix B_k. The quadratic convergence rate derived this way may hold even in the infinite-dimensional case, for example, in a Hilbert space with l_2 norm. Indeed, the proof of Theorem 3.1 does not depend on the dimension n of the space in which the least squares problems are discussed. With this main feature, it is suitable to call the new methods scaled factorized secant methods.

Our discussion in this paper focuses on the local behavior of the new methods. Their global behavior has not been studied. In particular the global convergence with a line search or trust region strategy and the positive definiteness of B_k independent of the initial status are two interesting and important questions for further research on this type of methods. Comprehensive numerical testing will also be helpful to justify the new methods.

Acknowledgments. The authors are grateful to Prof. J. Gilbert, the associate editor who viewed this paper, and to the referees for their valuable comments.

REFERENCES

[1] M. Al-Baali and R. Fletcher, *Variational methods for non-linear least squares*, J. Oper. Res. Soc., 36 (1985), pp. 405–421.

[2] J. E. Dennis, D. M. Gay, and R. E. Welsch, *An adaptive nonlinear least squares algorithm*, ACM Trans. Math. Software, 7 (1981), pp 348–368.

[3] J. E. Dennis, Jr., H. J. Martinez, and R. A. Tapia, *Convergence theory for the structured BFGS secant method with an application to nonlinear least squares*, J. Optim. Theory Appl., 61 (1989), pp. 161–178.

[4] J. E. Dennis, Jr. and R. B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.

[5] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley & Sons, Chichester, UK, 1987.

[6] J. Huschens, *On the use of product structure in secant methods for nonlinear least squares problems*, SIAM J. Optim., 4 (1994), pp. 108–129.

[7] H. Yabe and Takahashi, *Structured quasi-Newton methods for nonlinear least squares problems*, TRU Math., 24 (1988), pp. 195–209.

[8] H. Yabe and Takahashi, *Factorized quasi-Newton methods for nonlinear least squares problems*, Math. Programming, 51 (1991), pp. 75–100.

[9] H. Yabe and N. Yamaki, *Convergence of a factorized Broyden-like family for nonlinear least squares problems*, SIAM J. Optim., 5 (1995), pp. 770–791.

[10] J. Zhang and L. Chen, *Nonmonotone Levenberg–Marquardt algorithms and their convergence analysis*, J. Optim. Theory Appl., 92 (1997), pp. 397–422.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1180–1195

OPTIMAL CONTROL OF THE STERILIZATION OF PREPACKAGED FOOD*

D. KLEIS† AND E. W. SACHS†

Abstract. To model the process of sterilization by heating in the food industry, we derive an optimal control problem with state and control constraints governed by a nonlinear heat equation. A discretized form of the problem can then be expressed as a large-scale continuous optimization problem and solved by a special sequential quadratic programming method. The model provides useful insights—for example, when maximizing the retention of vitamins, the computed optimal control differs from the one typically used in industry—and can be generalized.

Key words. optimal control, industrial application, SQP methods, food sterilization

AMS subject classifications. 49M37, 90C55, 90C90

PII. S1052623497331208

1. Introduction.

In this paper we consider an optimal control problem resulting from a mathematical model of the heat sterilization of food.

The goal of this paper is to present the application, to show how a complex optimization problem in control can be derived, to use an optimization technique for the solution of the problem, and to interpret the results obtained with regard to their impact on the application. Details can be found in [8].

Heat sterilization of food is applied industrially to obtain products which are safe and have a long shelf life. The aim is to thermally destroy microorganisms which cause the spoilage of the food or which can pose a danger to consumers' health.

One of the most important procedures used to sterilize prepackaged foods is thermal processing in a batch retort (autoclave). Small containers like cans, jars, or pouches, which are filled with a food product, are called food containers in what follows. These are placed in a large container called an autoclave. The autoclave is filled with hot water or steam which heats the cans. Due to the heating the microorganisms in the food are destroyed. Afterward the water in the autoclave is cooled and the sterilization process is finished. In this paper we consider the development of the temperature inside the can where the control variable is the temperature in the autoclave which acts on the surface or boundary of the can. The temperature of the autoclave is influenced by a control mechanism. However, not only the microorganisms are heat sensitive. Severe heat treatment during in-container sterilization also produces substantial changes in the nutritional and sensory quality of the food. For example, a certain amount of the vitamins will be destroyed.

Both the destruction of microorganisms and the degradation of nutrients depend on the profile of the temperature over time inside the food containers, which in turn depends on the profile of the temperature over time inside the autoclave. Therefore the sterilization process can be controlled by this latter temperature and by the process

*Received by the editors December 8, 1997; accepted for publication (in revised form) September 19, 1999; published electronically June 13, 2000. This work was supported by the Forschungsfonds des Landes Rheinland-Pfalz.

http://www.siam.org/journals/siopt/10-4/33120.html

†Universität Trier, FB IV—Mathematik, D-54286 Trier, Germany (sachs@uni-trier.de). The research of the second author was partially supported by the Air Force Office of Scientific Research under grant F49620-96-1-0329 while visiting ICAM at Virginia Tech.

time. The main goal of the process is to achieve a prescribed amount of microorganism reduction. But this requirement does not totally characterize how to control the process. A control that is typically used in industry is to heat for a certain time with a constant temperature and then cool down. However, it turns out that this is not optimal in terms of nutrient retention.

Thus the question arises of how to determine a control for the sterilization process so that the required sterility is achieved and the disadvantageous effects are minimized. Other possible objectives could be to reach sterility in minimal time or with minimal energy consumption.

The paper is organized as follows. In section 2 a mathematical model of the sterilization process is derived. First it is described how the thermal effect on microorganisms and nutrients is modeled in the engineering literature. In particular, the F-value, an important tool in the food industry, is introduced. To evaluate these models it is necessary to know the profile of the temperature inside the can during the sterilization process. For foods heated by conduction this temperature can be modeled by the heat equation, a nonlinear parabolic initial boundary value problem. The optimal control problem resulting from the model of the sterilization process is then presented.

In section 3 the discretization of the parabolic boundary control problem is considered. A finite element discretization in space and an implicit time stepping scheme yield a large-scale optimization problem with control and state constraints. We demonstrate the sparsity of the Jacobian and Hessian matrices, a property that allows savings of storage in the numerical solution.

Numerical results are presented in section 4. For the solution we used an inexact sequential quadratic programming (SQP) method. One of the advantages of this method is that the underlying quadratic subproblems can be solved by an iterative procedure rather than a direct solution by matrix factorization. The test examples used are typical prototypes in industry. The results demonstrate that by use of optimization an improved temperature control can be obtained that is less destructive to vitamins while still maintaining the sterilization requirements.

The models and methods used in this paper are from a mathematical point of view more complex than those employed in the papers [3, 14, 15, 16]. Our model allows for a nonlinear heat conduction process that is often used to include effects due to thickening of the product. The SQP method, which we apply to a finite element discretization, not a finite difference discretization, can adapt to various requirements from the application such as various nonlinearities in the heat equation or constraints on the state variables (the temperature of the food product). Since the required F-value that appears as a constraint in the optimization problem is very sensitive to changes in a certain temperature regime, a good accuracy is required in the temperature computed from the nonlinear heat equation.

2. Model of the sterilization process and optimal control problem.

In this section we consider aspects of modeling the heat sterilization of food.

The destruction of both microorganisms and nutrients is usually described by a process of first order

$$\frac{\partial C(x,t)}{\partial t} = -K(\theta(x,t))C(x,t), \tag{2.1}$$

where $C(x,t)$ is the concentration of living microorganisms or nutrients and $\theta(x,t)$ is the absolute temperature, measured in degrees Kelvin, at the point x inside a food container at time t. The function K in (2.1) depends on the temperature through the Arrhenius equation [7]

$$K(\theta) = K_r \exp\left(-\frac{E}{R}\left(\frac{1}{\theta} - \frac{1}{\theta_r}\right)\right), \tag{2.2}$$

where K_r is the value of K at a reference temperature θ_r, E is the activation energy, and R is the universal gas constant.

Denote by C_v and C_m the concentration of vitamins and microorganisms and let C_{v0} and C_{m0} be the initial concentrations at time $t = 0$. Both C_v and C_m depend on the temperature of the product, which in turn depends on the temperature u of the heating medium, i.e., u is the control. The temporal development of these variables is described by a process of first order, i.e., it follows a differential equation of first order. We consider the retention of the vitamins at a point x_b on the boundary or surface of the product. The prescribed reduction of the microorganisms is monitored at the center of the food container x_c:

$$\frac{\partial C_v(x_b,\cdot)}{\partial t} = -C_v(x_b,\cdot)K_{vr}\exp\left(-\frac{E_v}{R}\left(\frac{1}{\theta} - \frac{1}{\theta_{vr}}\right)\right) \quad \text{in } (0,T), \tag{2.3}$$

$$C_v(x_b, 0) = C_{v0}(x_b), \tag{2.4}$$

$$\frac{\partial C_m(x_c,\cdot)}{\partial t} = -C_m(x_c,\cdot)K_{mr}\exp\left(-\frac{E_m}{R}\left(\frac{1}{\theta} - \frac{1}{\theta_{mr}}\right)\right) \quad \text{in } (0,T), \tag{2.5}$$

$$C_m(x_c, 0) = C_{m0}(x_c). \tag{2.6}$$

The heat transfer from the surrounding heating medium to the food product is modeled by a nonlinear heat equation. The domain Ω with boundary Γ describes the spatial region occupied by the food container, e.g., a can, and T is the process time. The following functions depend on the temperature θ: the density ρ, the heat capacity c, and the heat conductivity k of the product to be heated. The constant α denotes the heat transfer coefficient. The boundary condition is derived from the fact that the heat flux into the can is proportional to the temperature difference between the can and the water of the autoclave:

$$\rho c(\theta)\frac{\partial \theta}{\partial t} - \nabla \cdot (k(\theta)\nabla\theta) = 0 \quad \text{in } \Omega \times (0,T), \tag{2.7}$$

$$k(\theta)\frac{\partial \theta}{\partial n} = \alpha(u - \theta) \quad \text{in } \Gamma \times (0,T), \tag{2.8}$$

$$\theta(\cdot, 0) = \theta_0(\cdot) \quad \text{in } \Omega. \tag{2.9}$$

The inequality constraints include upper and lower bounds on the temperature of the autoclave due to technical restrictions. Furthermore, it is important that the temperature inside the can is not too high when the process is stopped. Otherwise the sterilization process continues after the process is terminated or a deformation of the can may occur because of differences in pressure between inside the can and outside:

$$u_{low} \le u(\cdot) \le u_{up}(\cdot) \quad \text{in } (0,T), \tag{2.10}$$

$$\theta(x_c, T) \le \theta_{end}. \tag{2.11}$$

At this point we have not imposed a constraint that guarantees a prescribed reduction of microorganisms at the center of the food container x_c. We will come back to this point later. The objective in this model problem is to optimize the retention of vitamins at a point x_b on the surface of the product. We also take into account the energy that is required for the heating and cooling of the autoclave. It is assumed to be proportional to the square of the control variable. This is added to the objective with a weighting factor. As is well known, this addition results in a convexification of the optimization problem that is essential for ill-posed problems and mathematically often mandatory; however, in our numerical results we could set this factor often to zero:

$$\min -C_v(x_b,T) + \frac{\delta}{2}\|u\|^2_{L^2(0,T)}. \tag{2.12}$$

We reconsider the requirement on the concentration of the microorganisms. Note that the differential equations (2.3) and (2.5) can be solved analytically. If the function θ is continuous we obtain for C_m

$$C_m(x_c,t) = C_{m0}(x_c)\exp\left(-\int_0^t K(\theta(x_c,\tau))d\tau\right) \tag{2.13}$$

and for C_v an analogous formula. We introduce the concept of an F-value, which is a well-known and accepted tool in the food industry. Federal regulations state that a food product is considered sterile if the concentration of the microorganisms is reduced by a factor of $10^{-\beta}$. Mathematically, this means that at the final time T the following inequality has to hold:

$$C_m(x_c,T) \le 10^{-\beta}C_m(x_c,0). \tag{2.14}$$

Then from (2.2) and (2.13), we obtain, by applying logarithms on both sides,

$$\begin{aligned}\beta\frac{\ln 10}{K_r} &\le \int_0^T \exp\left(-\frac{E_a}{R}\left(\frac{1}{\theta(x_c,\tau)} - \frac{1}{\theta_r}\right)\right)d\tau \\ &= \int_0^T 10^{E_a(\theta(x_c,\tau)-\theta_r)/(R\theta(x_c,\tau)\theta_r\ln 10)}d\tau. \qquad (2.15)\end{aligned}$$

In the food literature [2] the term $\theta(x_c,\tau)\theta_r$ is often replaced by θ_r^2. This yields the advantage that the exponent depends in a linear way on θ. Equation (2.15) is then simplified to

$$\beta\frac{\ln 10}{K_r} \le \int_0^T 10^{(\theta(x_c,\tau)-\theta_r)/z}d\tau, \tag{2.16}$$

where F and z are defined in the following lines.

Definition 2.1. *For $x \in \Omega$ the function*

$$F(\theta)(x) := \int_0^T 10^{\frac{\theta(x,\tau)-\theta_r}{z(\theta_r)}}\,d\tau \tag{2.17}$$

is called the F-value at the point x corresponding to the reference temperature θ_r and

$$z(\theta_r) := \frac{R\theta_r^2\ln 10}{E} \tag{2.18}$$

is called the z-value.

Due to regulatory agencies and the sensitivity of the public to changes in the requirement of food sterility, the formula (2.17) with the simplification is still in use today.

The lower bound on the F-value in (2.16) is often called the F_0-value. This numerical value, which is based on β and K_r, is determined by experience and the requirement of the product, e.g., its consistency and its geographical usage.

The sterility condition (2.16) can be rewritten as

$$F(\theta)(x_c) \geq F_0 \quad \text{with} \quad F_0 = \beta \ln 10 / K_r. \tag{2.19}$$

The requirement on sterility should be placed at every point in the can which would imply that this condition must hold not only at x_c but at every point $x \in \Omega$. Only x_c is considered because this is the coldest point during the heating process. Since the F-value is monotone with respect to θ, under this assumption the lowest F-value is reached at the center and it suffices to consider (2.19) only for $x = x_c$. We checked this in the numerical results that support this assumption. In [8] one can find a proof for this claim in the one-dimensional case with a linear heat equation. Note, however, that in the cooling phase this is no longer true.

Analogously to the F-value, a model for the destruction of vitamins can be derived. Optimization of the surface quality of the product can then be expressed by minimizing

$$J(\theta)(x_b) := \int_0^T 10^{(\theta(x_b,\tau) - \theta_q)/z_q} d\tau \tag{2.20}$$

(e.g., in [14], [15], and [16]).

Using the monotonicity of the functions J and F it can be seen (see [8]) that the sterility requirement $F(\theta)(x_c) \geq F_0$ is active at the optimal solution. Therefore this constraint can be handled as an equality constraint $F(\theta)(x_c) = F_0$.

In order to state the discretized problem properly we rephrase the problem in a more precise functional analytic way. In a previous paper in a different context by Burger and Pogu [4] a detailed derivation of the variational formulation of the nonlinear heat equation can be found. Define

$$\bar{c}(v) = \int_0^v \rho c(\xi) d\xi \quad \text{and} \quad \bar{k}(v) = \int_0^v k(\xi) d\xi \,. \tag{2.21}$$

Assume that the domain Ω and the coefficients in the heat equation are such that Green's formula can be applied. Then the variational formulation of the initial boundary value problem (2.7)–(2.9) is

$$\left(\frac{\partial}{\partial t}\bar{c}(\theta), v\right) + (\nabla \bar{k}(\theta), \nabla v) + \langle \theta, v \rangle = u(t)\langle 1, v \rangle \quad \forall v \in H^1(\Omega), \tag{2.22}$$

$$\theta(\cdot, 0) = \theta_0(\cdot), \tag{2.23}$$

where $\langle v, w \rangle := \alpha \int_\Gamma v(x)w(x)dx$ and $(u,v) := \int_\Omega u(x)v(x)dx$ and the Sobolev space H^1 can be found in [1].

The optimal control problem (OCP) now reads as

(OCP)

$$\min J(\theta)(x_b) + \frac{\delta}{2}\|u\|^2_{L^2(0,T)} \quad u \in L^\infty(0,T) \quad \text{s.t.}$$

$$\begin{aligned} & F(\theta)(x_c) = F_0 \\ & u_{low} \le u(t) \le u_{up}(t) \qquad \text{in } \ (0,T) \\ & \theta(x_c,T) \le \theta_{end} \end{aligned}$$

and $\theta \in L^2(0,T;H^1(\Omega))$ is a solution of the variational formulation of the heat equation (2.22)–(2.23) corresponding to the control u

for $\delta \ge 0$.

The choice of the function spaces is not trivial and is still an open problem. If $u \in L^\infty(0,T)$, then it is not clear for the solution of (2.22)–(2.23) that the state inequality constraint is well defined. In [4] some existence theorems for generalized solutions are given. For a discussion of the existence of optimal controls see [8]. Without proper assumptions like monotonicity on the nonlinearity in the partial differential equation no uniqueness of the optimal control can be expected. In this paper we do not want to address all the issues that are important from a theoretical point of view but rather to emphasize the numerical computation and the practical implications.

With respect to the feasibility of a control, note that for any positive constant control the constraint $F(\theta)(x_c) = F_c$ will be reached if T is sufficiently large. Similarly, if u is set to u_{low} at the end of the process, the temperature decays and falls below θ_{end} if T is large enough unless u_{low} is too high and θ_{end} is too large. However, these bounds on the constraints come from practical applications, so the existence of feasible controls is assured.

3. Discretization.

Although many problems remain in the theoretical part, it is helpful for the solutions of the industrial problem to discretize the optimization problem. If one considers a cylindrical can that is heated by a uniform temperature in the autoclave, by symmetry, it is sufficient to consider one half of a vertical cross section that reduces the three-dimensional problem to a two-dimensional one. In this paper we present our findings for the one-dimensional case that were obtained by the authors. The results of a recent thesis by Justen [6] show that in the three-dimensional case the optimal control exhibits a similar shape and the same conclusions can be drawn.

We start with a discretization of the heat equation using finite elements for the spatial variable and the backward Euler method for the time variable. This discretization has been derived in [4], [10], or [11], where a more detailed presentation can be found.

In the one-dimensional case we consider a straight line from a point a on the boundary of a container to the center x_c. Due to symmetry the boundary conditions (2.8) in the heat equation are replaced by

$$\begin{aligned} k(\theta)\frac{\partial\theta}{\partial x} &= \alpha(\theta - u) && \text{for } x = a,\ t \in (0,T), \\ k(\theta)\frac{\partial\theta}{\partial x} &= 0 && \text{for } x = x_c,\ t \in (0,T)\,. \end{aligned}$$

We partition the intervals (a, x_c) and $(0, T)$ into equidistant subintervals by

$$h = \frac{x_c - a}{N} \quad \text{and} \quad x_i = a + (i-1)h, \qquad i = 1, \dots, N+1\,,$$
$$\tau = \frac{T}{M} \quad \text{and} \quad t^k = (k-1)\tau, \qquad k = 1, \dots, M+1\,.$$

The state space $L^2(0, T; H^1(\Omega))$ is then approximated by

$$V^{NM} = \left\{ w(x,t) = \sum_{k=1}^{M} w^{k+1}(x)\chi_k(t)\,,\ w^k \in V^N, k = 2, \dots, M+1 \right\},$$

where

$$V^N = \operatorname{span}(S_1, \dots, S_{N+1})$$

and the functions $S_1, \dots, S_{N+1}$ are the usual basis of one-dimensional piecewise linear spline functions satisfying

$$S_i(x_j) = \delta_{ij},$$

where δ_{ij} denotes the Kronecker symbol. The characteristic function on the interval $(t_k, t_{k+1}]$ is denoted by $\chi_k(t)$. The control space $L^\infty(0, T)$ is approximated by

$$U^M = \left\{ u_M(t) = \sum_{k=1}^{M} u^{k+1}\chi_k(t), u^{k+1} \in \mathbb{R} \right\}.$$

Under appropriate assumptions one can prove the existence of a unique solution for the discretized state (cf. [4])

$$\theta_{NM}(x,t) = \sum_{k=1}^{M} \chi_k(t) \left(\sum_{i=1}^{N+1} \theta_i^{k+1} S_i(x) \right) \in V^{NM}.$$

The vector $\underline{v} = (v_1, \dots, v_{N+1})^T$ is composed of the coefficients for $v = \sum_{i=1}^{N+1} v_i S_i \in V^N$. Then for $v, w \in V^N$

$$(v, w) = \underline{v}^T B \underline{w} \quad \text{and} \quad \left(\frac{\partial}{\partial x} v, \frac{\partial}{\partial x} w \right) = \underline{v}^T D \underline{w},$$

where

$$B = (S_i, S_j) \quad \text{and} \quad D = (\nabla S_i, \nabla S_j)\,.$$

Next define $\bar{c}(\underline{v}) = (\bar{c}(v_1), \dots, \bar{c}(v_{N+1}))^T$ and $\bar{k}(\underline{v}) = (\bar{k}(v_1), \dots, \bar{k}(v_{N+1}))^T$ with $\bar{c}$ and $\bar{k}$ from (2.21). The discretized version of (2.22) yields

$$\text{(3.1)}\quad \frac{1}{\tau} B(\bar{c}(\underline{\theta}^{k+1}) - \bar{c}(\underline{\theta}^k)) + D\bar{k}(\underline{\theta}^{k+1}) + \alpha(\theta_1^{k+1} - u^{k+1})e_1 = 0, \quad k = 1, \dots, M,$$

$$\text{(3.2)}\qquad \underline{\theta}^1 = \underline{\theta}_{0N},$$

where $e_1 = (1, 0, \dots, 0)^T \in \mathbb{R}^{N+1}$, and $\underline{\theta}_{0N}$ is the solution of

$$B\underline{\theta}_{0N} = ((\theta_0, S_1), \dots, (\theta_0, S_{N+1}))^T.$$

Since we consider from here on only the discretized problem, we will omit underlines and denote the coefficients of θ_{NM} and u_M by

$$\theta = ((\theta^2)^T, \dots, (\theta^{M+1})^T)^T \in \mathbb{R}^{M(N+1)} \quad \text{and} \quad u = (u^2, \dots, u^{M+1})^T \in \mathbb{R}^M .$$

In the discretized optimization problem we treat both u and θ as variables. In order to write (3.1) and (3.2) as nonlinear equality constraints in this finite-dimensional optimization problem, we introduce the function

$$h^{NM} : \mathbb{R}^{M(N+1)} \times \mathbb{R}^M \to \mathbb{R}^{M(N+1)}$$

with components $h^{NM,j} : \mathbb{R}^{M(N+1)} \times \mathbb{R}^M \to \mathbb{R}^{N+1}$ for $k = 2, \dots, M$ defined by

$$\begin{aligned} h^{NM,1}(\theta, u) &= \frac{1}{\tau} B(\bar{c}(\theta^2) - \bar{c}(\theta_{0N})) + D\bar{k}(\theta^2) + \alpha(\theta_1^2 - u^2)e_1, \\ h^{NM,k}(\theta, u) &= \frac{1}{\tau} B(\bar{c}(\theta^{k+1}) - \bar{c}(\theta^k)) + D\bar{k}(\theta^{k+1}) + \alpha(\theta_1^{k+1} - u^{k+1})e_1. \end{aligned}$$

We discretize the objective function of the control problem (OCP) and the function $F(\theta)(x)$. From the approximation for h, F, J, u, and θ we have $\theta_1^k = \theta_{NM}(a, t)$, and $\theta_{N+1}^k = \theta_{NM}(x_c, t)$ on $(t^{k-1}, t^k]$. Evaluating the integral terms in the objective function and constraints of the control problem (OCP) then yields

$$\begin{aligned} \|u_M\|_{L^2}^2 &= \tau \sum_{k=2}^{M+1} (u^k)^2 , \\ F^{NM}(\theta) := F(\theta_{NM})(x_c) &= \tau \sum_{k=2}^{M+1} 10^{(\theta_{N+1}^k - \theta_r)/z_r}, \\ J^{NM}(\theta) := J(\theta_{NM})(a) &= \tau \sum_{k=2}^{M+1} 10^{(\theta_1^k - \theta_q)/z_q} . \end{aligned} \tag{3.3}$$

The discretized optimal control problem (DOCP) is

$$\text{(DOCP)} \quad \boxed{\begin{array}{l} \min \; J^{NM}(\theta) + \dfrac{\delta}{2}\tau \displaystyle\sum_{k=2}^{M+1} (u^k)^2 \quad (\theta, u) \in \mathbb{R}^{M(N+1)} \times \mathbb{R}^M, \\[2em] \qquad F^{NM}(\theta) = F_0, \\ \qquad h^{NM}(\theta, u) = 0, \\ \qquad u_{low} \le u^k \le u_{up} , \qquad k = 2, \dots, M+1, \\ \qquad \theta_{N+1}^{M+1} \le \theta_{end}. \end{array}}$$

The lower and the upper bound on the control are discretized in the same way as the control u.

This is a nonlinear optimization problem with nonlinear equality constraints and box constraints. SQP methods belong to the most successful optimization problem solvers. They are based on an iterative procedure where in each step a quadratic optimization problem is solved. For the numerical solution of this problem with SQP methods we can provide the first and second derivatives of the objective function and of the constraints. The derivatives of the objective function and of the discrete sterility constraint are easily computed. We mention that the Hessians of the functions J^{NM} and F^{NM} are diagonal matrices with only nonnegative entries. The Jacobian of the function $h^{NM}(\theta, u)$ with respect to θ and u is given by

$$h^{NM}_{(\theta,u)} = \begin{pmatrix} G(\theta^2) & & & & Q^2 \\ H(\theta^2) & G(\theta^3) & & & Q^3 \\ & \ddots & \ddots & & \vdots \\ & & H(\theta^M) & G(\theta^{M+1}) & Q^{M+1} \end{pmatrix} \in \mathbb{R}^{M(N+1)\times M(N+2)},$$

where for $w \in \mathbb{R}^{N+1}$

$$\begin{aligned} G(w) &= \frac{1}{\tau} B \operatorname{diag}(\rho c(w_1), \dots, \rho c(w_{N+1})) + D \operatorname{diag}(k(w_1), \dots, k(w_{N+1})) \\ &\quad + \operatorname{diag}(\alpha, 0, \dots, 0)\,, \\ H(w) &= -\frac{1}{\tau} B \operatorname{diag}(\rho c(w_1), \dots, \rho c(w_{N+1})) \end{aligned}$$

are tridiagonal matrixes in $\mathbb{R}^{(N+1)\times(N+1)}$ and

$$Q^k = -\alpha e_1 \tilde{e}_k^T \in \mathbb{R}^{(N+1)\times M}\,.$$

Here e_1 is the first unit vector in $\mathbb{R}^{N+1}$ and $\tilde{e}_k$ is the kth unit vector in $\mathbb{R}^M$. In [4] it is shown that $G(\theta^k)$ is invertible under conditions on the discretization parameters. Therefore h^{NM}_θ is invertible and $h^{NM}_{(\theta,u)}$ has full row rank. If the functions c and k are continuously differentiable, then h^{NM} is twice continuously differentiable and the Hessian of each component $h_i^{NM,k}$ is a diagonal matrix (cf. [12]).

4. Numerical results.

The finite-dimensional discretized optimization problem is a large-scale optimization problem with sparse Jacobians and Hessians. For general literature on SQP methods see, for example, [5] or [13]. There are various codes to treat these problems. In order to avoid active set strategies, we use an SQP implementation where the inequality constraints are transformed into equality constraints by slack variables; see [9]. The inexact solution of the subproblems overcomes the known deficiency for the slack variable technique of freezing the active constraints; for details see [9]. The method is implemented in Fortran, and the numerical results of this section were obtained on a Cray EL 98.

We demonstrate the advantage of the optimization approach for the food sterilization problem by comparing the solution of the optimal control problem with a conventionally used control.

We consider two examples where we optimize the sterilization process so that retention of thiamin on the surface of the product is maximized.

4.1. Examples.

The data of the first example can be found in the engineering literature [7] and correspond to a meat product. Since the coefficients of the heat equation are constants we consider the linear heat equation.

Example 1. The coefficients of the heat equation are taken from Kessler [7]:

$$\begin{aligned} c &= 3000\ J/(\mathrm{kgK})\,,\\ \rho &= 970\ \mathrm{kg/m^3}\,,\\ k &= 0.4\ W/(\mathrm{mK})\,,\\ \alpha &= 5000\ J/(\mathrm{sm^2K})\,. \end{aligned}$$

The initial temperature of the product is assumed to be $\theta_0(x) = 30 + 273.15$ K in Ω (which corresponds to 30°C). In the objective function we consider only the retention of thiamin on the surface, i.e., we choose $\delta = 0$. The microorganism under consideration is *Clostridium botulinum*. The corresponding data can be found in [3]:

$$\begin{aligned} \theta_q &= 121.11 + 273.15\ \mathrm{K}\,,\\ z_q &= 25.56\ \mathrm{K}\,,\\ \mathrm{K}_q &= 2.1510^{-4}\ 1/\mathrm{s}\,,\\ z_r &= 10\ \mathrm{K}\,,\\ \theta_r &= 121.11 + 273.15\ \mathrm{K}\,. \end{aligned}$$

We consider the one-dimensional heat equation. The domain Ω is given by a line segment of length 0.02 m, i.e., $\Omega = (0, 0.02) = (0, x_c)$. For the process time we use $T = 7200$ s, and the bounds in the box constraints are chosen as

$$\begin{aligned} u_{low} &= 10 + 273.15\ \mathrm{K}\,,\\ u_{up}^k &= \min\big(\bar{u}_{up}, 30 + t^k(\bar{u}_{up} - 30)/T_c\big) + 273.15\ \mathrm{K}\,, \qquad k = 2, \dots, M+1\,,\\ \bar{u}_{up} &= 130\,,\\ T_c &= 180\ \mathrm{s}\,,\\ \theta_{end} &= 80 + 273.15\ \mathrm{K}\,. \end{aligned}$$

In the second example we modify the heat conductivity and the heat capacity of Example 1 so that the resulting heat equation is nonlinear. The shape of the function $k(\theta)$ is chosen to be similar to data given in [7] for other food products.

Example 2. Here we use the same data as in Example 1 except

$$\begin{aligned} k(\theta) &= 0.001\theta,\\ c(\theta) &= 2960 + 0.1\theta. \end{aligned}$$

4.2. Interpretation of the results for food sterilization. We first consider Example 1. The discretized optimal control problem (DOCP) was solved for $N = 30$ and $M = 500$. As a result the number of unknowns in the SQP algorithm was 33503 with 15501 nonlinear equality, 32 nonlinear inequality constraints, and 1000 linear constraints resulting from slack variables.

For this example we compare the computed optimal control with a conventionally used control. As mentioned before, in industry a typical control is to heat a certain time with constant temperature and then cool down. Such a control for the data of Example 1 is shown in Figure 4.1. This control is chosen so that at the center x_c the required F-value is attained, and the desired temperature at the end of the process is reached, i.e., $F^{NM}(\theta) = 180$, $\theta_{N+1}^{M+1} \approx 80 + 273.15$ K.

Since we fixed the temperature of the cooling phase to be equal to u_{low}, a typical control can be described by the heating temperature and length of the heating phase.

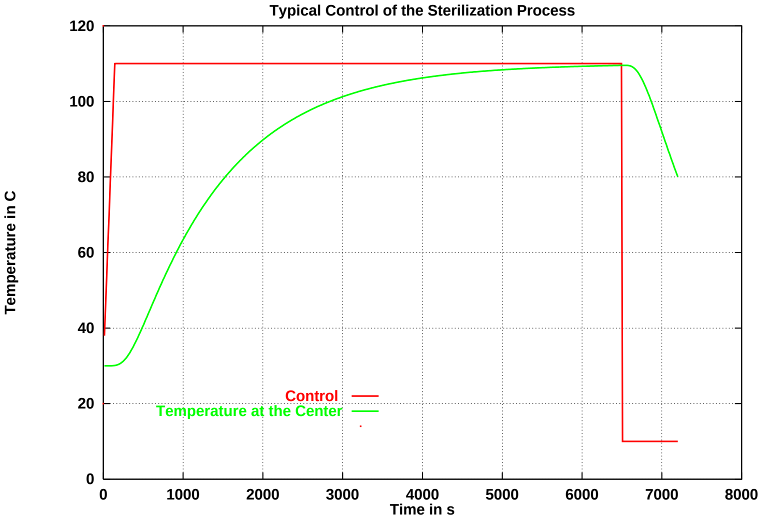


Fig. 4.1. *Typical control with the corresponding temperature at the center.*

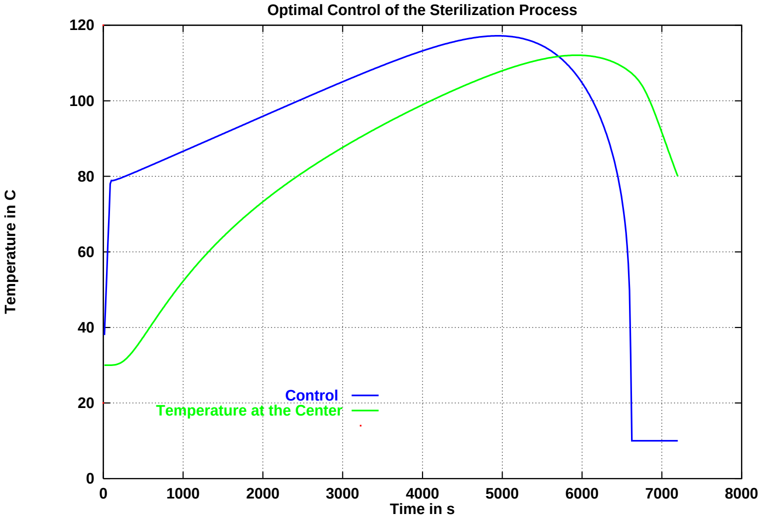


Fig. 4.2. *Optimal control with the corresponding temperature at the center.*

Thus there are only two degrees of freedom and two constraints such that this control is uniquely determined. In Figure 4.1 the corresponding temperature at the centerpoint x_c is also shown.

The computed optimal control and the corresponding temperature at the point x_c are plotted in Figure 4.2. The termination criterion for the SQP method was satisfied when the norm of the gradient of the Lagrangian and the norm of the constraint violation were less than 10^{-6}. The shape of the control at the beginning of the process is due to the upper bound on the control. After this the temperature rises more slowly but reaches a higher level than with the typical control. There is a cooling phase, and

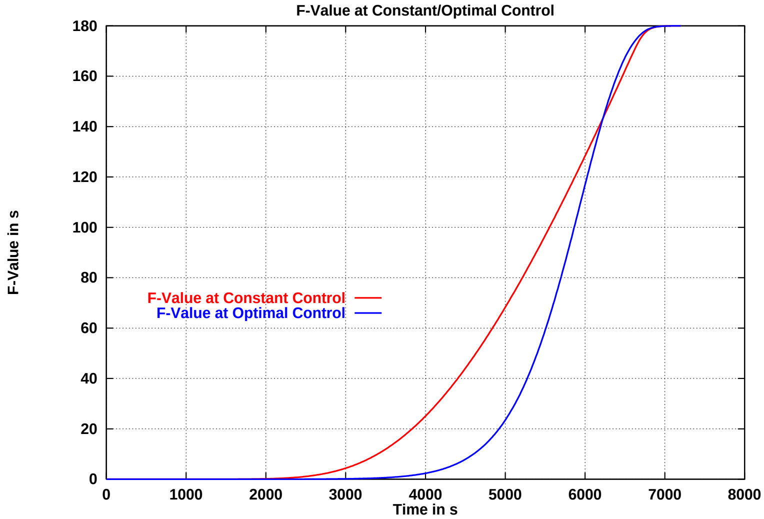


Fig. 4.3. *Profile of the F-value for the typical and optimal controls.*

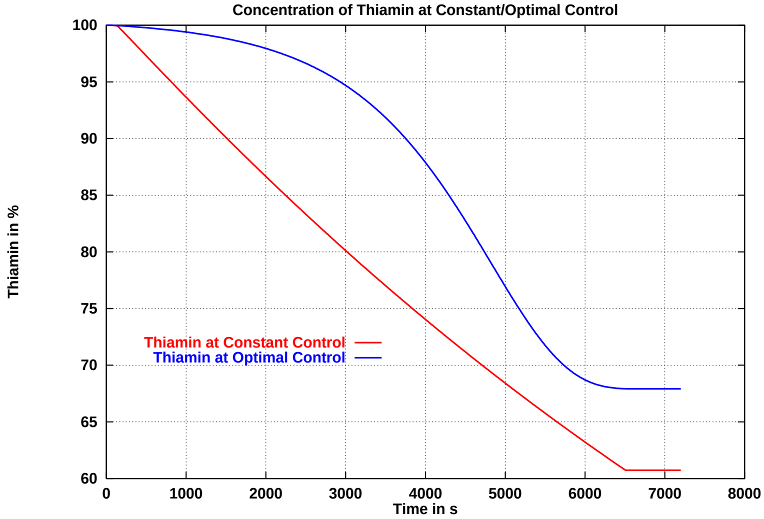


Fig. 4.4. *Profile of the concentration of thiamin for the typical and optimal controls.*

the shape of the control at the end of the process is again due to the box constraints.

In Figure 4.3 we consider the profile of the F-value over the process time for the typical control and for the optimal control. In both cases the required value of 180s is reached at the end of the process. Figure 4.4 shows the percentage of the initial concentration of thiamin on the surface of the product over process time that corresponds to each of these controls. With the typical control there is an almost linear decrease of this concentration, and at the end of the process we reach 60.7%. With the optimal control the concentration of thiamin decreases more slowly and remains on the higher level of 67.9%.

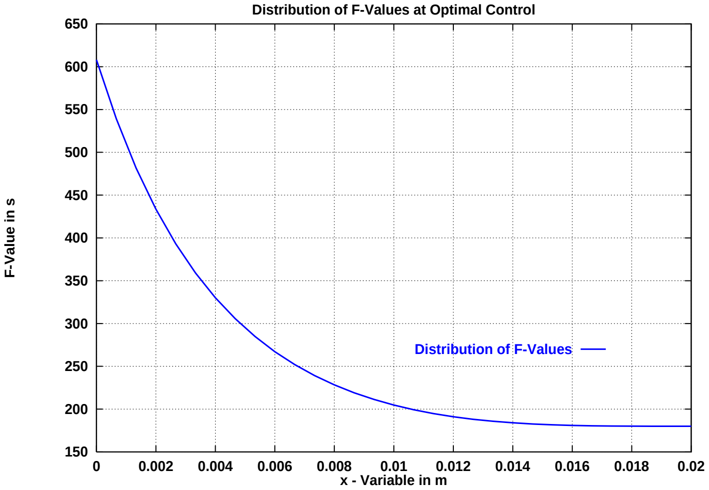


FIG. 4.5. *Spatial distribution of F-values using the optimal control.*

In some other applications like cooking ham (not sterilization), the industry often uses a different control law. For example, it is required that the difference between the temperature on the boundary and the center does not exceed a prescribed bound in order to avoid larger losses of the sensory quality. As one can see in Figure 4.2 the maximal difference between the two temperatures is much smaller than in Figure 4.1, so the results from a qualitative point are plausible in food technology.

To estimate the effect on the energy consumption we used $(\int_0^T u_M^2 dt)^{\frac{1}{2}}$. For the typical control a value of 31722 was computed and for the optimal control a value of 31159. Thus we obtain a better quality of the product in the same process time with less energy consumption if the optimal control is used.

If a higher F_0-value of 540s is required the differences are a bit wider. The concentration of nutrients on the surface corresponding to the optimal control is now 54.9% and 46.1% for the typical control. For the energy consumption we computed values of 31483 and 32030 for the optimal and typical control, respectively.

Appropriate modeling of the sterility requirement was one of the questions we discussed in the previous sections. Figure 4.5 shows the F-value $F(\theta_{NM})(x)$ as a function of $x \in (0, x_c)$ for the temperature θ corresponding to the optimal control. We can see that it is appropriate at least for this example to consider the sterility requirement only at the point x_c. In Figure 4.6 a plot of the temperature over space and time is provided. It shows that the boundary regions heat up faster and that the center is the coldest point during the heating phase. At the end of the heating phase the temperature is almost homogeneous. In the cooling phase the boundary regions cool down faster, and the center point is the hottest one. This justifies that the requirement on the temperature of the product at the end of the process is formulated only for the point x_c.

As we have seen, the centerpoint is not the coldest one during the whole process but only for the heating phase. In our numerical tests, after computing an optimal control, we checked if $F(\theta_{NM})(x_c) \leq F(\theta_{NM})(x_i)$ for $i = 1, \dots, N$. For the examples we have considered this was always fulfilled. Thus the model of the sterility requirement was appropriate in these cases.

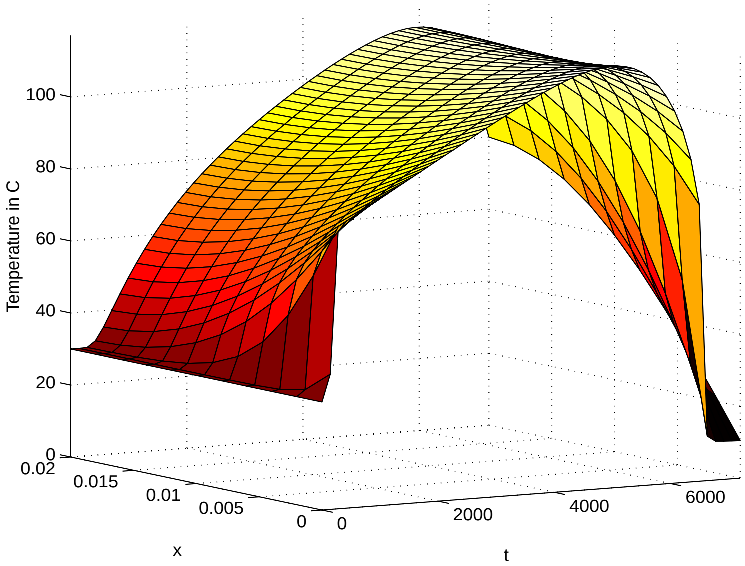


FIG. 4.6. *Temperature profile.*

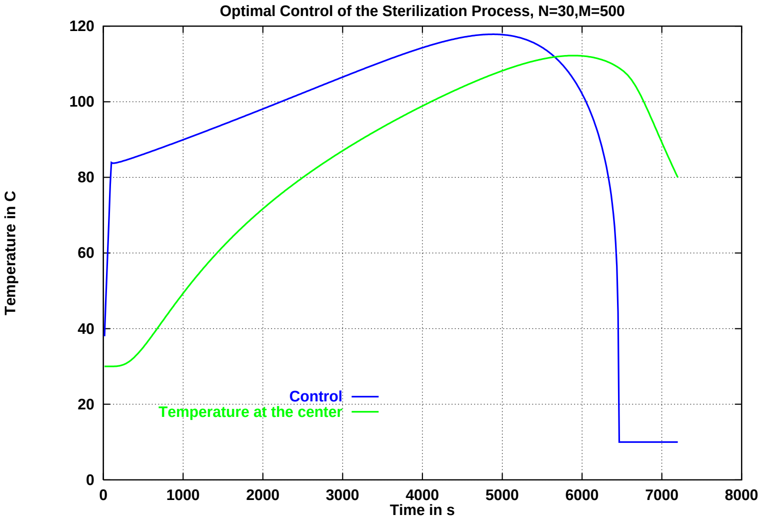


FIG. 4.7. *Optimal control with the corresponding temperature at the center for Example* 2.

Now we consider Example 2. For the discretization we chose again $N = 30$ and $M = 500$. The optimal control and the corresponding temperature at the center are given in Figure 4.7. The retention of thiamin is 66.2%.

If stronger restrictions are imposed on the control, then we expect that the retention of thiamin will be lower. We solved the optimal control problem for Example 2 but with $\bar{u}_{up} = 115$ K as the upper bound for the control. Now the retention of thiamin is 65.9%. The optimal control is shown in Figure 4.8. The second curve in this figure shows the starting data for the control for the SQP algorithm.

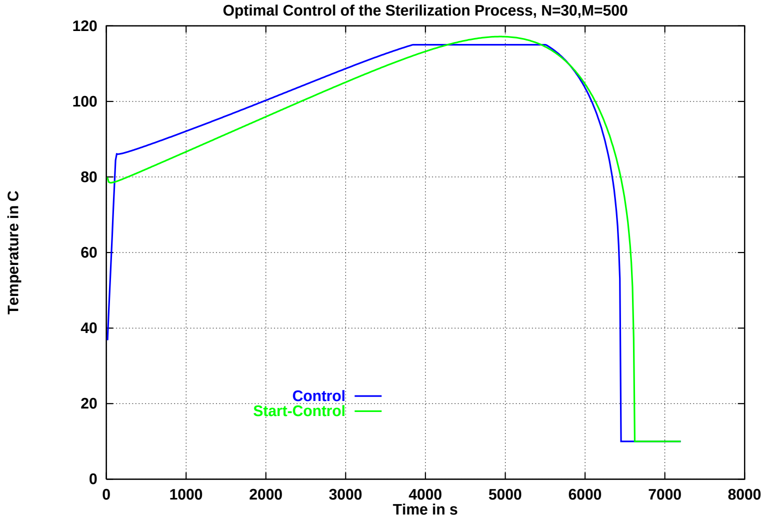


FIG. 4.8. *Optimal control and start control for Example* 2 *with* $\bar{u}_{up} = 115$ K.

5. Conclusion. We considered an application from the food industry that is concerned with the sterilization of packaged food. The process is formulated as an optimization problem where the objective is to retain a maximum of vitamins subject to the constraint that a certain level of sterility is obtained. Mathematically, this is an optimal control problem with a nonlinear parabolic boundary value problem and boundary controls. There are constraints on the control and the state. A proper discretization leads to a large-scale optimization problem that is numerically solved by an SQP method. The result yields a control strategy that is quite different from that used in industry. While the sterility level is the same, it yields a better treatment of the vitamins. Other applications to be considered in future work include models in higher space dimensions, the influence of convective terms, and other objective functions such as the energy consumption.

Acknowledgment. The authors are thankful for many helpful comments by the referees.

REFERENCES

[1] R. A. ADAMS, *Sobolev Spaces*, Academic Press, New York, 1975.

[2] C. O. BALL AND F. C. W. OLSON, *Sterilization in Food Technology*, McGraw-Hill, New York, Toronto, London, 1957.

[3] J. R. BANGA, R. I. PEREZ-MARTIN, J. M. GALLARDO, AND J. J. CASARES, *Optimization of the thermal processing of conduction-heated canned foods: Study of several objective functions*, J. Food Engineering, 14 (1991), pp. 25–51.

[4] J. BURGER AND M. POGU, *Functional and numerical solution of a control problem originating from heat transfer*, J. Optim. Theory Appl., 68 (1991), pp. 49–73.

[5] P. E. GILL, W. MURRAY, AND M. H. WRIGHT, *Practical Optimization*, Academic Press, London, 1981.

[6] P. JUSTEN, *Optimal Control of Thermally Coupled Navier-Stokes Equations in Food Industry*, doctoral thesis, Universität Trier, Germany, 1999.

[7] H. G. KESSLER, *Lebensmittel- und Bioverfahrenstechnik*, Verlag A. Kessler, München, 1988.

[8] D. KLEIS, *Augmented Lagrange SQP Methods and Application to the Sterilization of Prepackaged Food*, doctoral thesis, Universität Trier, Germany, 1997.

[9] D. KLEIS AND E. W. SACHS, *A Modification of SQP Methods with Slack Variables*, Technical report, Universität Trier, Germany, 1997.

[10] F.-S. KUPFER, *Reduced Successive Quadratic Programming in Hilbert Space with Applications to Optimal Control*, doctoral thesis, Universität Trier, Germany, 1992.

[11] F.-S. KUPFER AND E. W. SACHS, *Numerical solution of a nonlinear parabolic control problem by a reduced SQP method*, Comput. Optim. Appl., 1 (1992), pp. 113–135.

[12] F. LEIBFRITZ AND E. W. SACHS, *Numerical solution of parabolic state constrained control problems using SQP- and interior-point-methods*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer, Dordrecht, The Netherlands, 1994, pp. 251–264.

[13] S. G. NASH AND A. SOFER, *Linear and Nonlinear Programming*, McGraw-Hill, New York, 1996.

[14] C. SILVA, M. HENDRICKX, F. OLIVEIRA, AND P. TOBBACK, *Critical evaluation of commonly used objective functions to optimize overall quality and nutrient retention of heat-preserved foods*, J. Food Engineering, 17 (1992), pp. 241–258.

[15] C. SILVA, M. HENDRICKX, F. OLIVEIRA, AND P. TOBBACK, *Optimal sterilization temperatures for conduction heating foods considering finite surface heat transfer coefficients*, J. Food Science, 57 (1992), pp. 743–748.

[16] C. L. M. SILVA, F. A. R. OLIVEIRA, AND M. HENDRICKX, *Modelling optimum processing conditions for the sterilization of prepackaged foods*, Food Control, 4 (1993), pp. 67–78.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1196-1211

NONMONOTONE SPECTRAL PROJECTED GRADIENT METHODS ON CONVEX SETS*

ERNESTO G. BIRGIN†, JOSÉ MARIO MARTÍNEZ†, AND MARCOS RAYDAN‡

Abstract. Nonmonotone projected gradient techniques are considered for the minimization of differentiable functions on closed convex sets. The classical projected gradient schemes are extended to include a nonmonotone steplength strategy that is based on the Grippo–Lampariello–Lucidi nonmonotone line search. In particular, the nonmonotone strategy is combined with the spectral gradient choice of steplength to accelerate the convergence process. In addition to the classical projected gradient nonlinear path, the feasible spectral projected gradient is used as a search direction to avoid additional trial projections during the one-dimensional search process. Convergence properties and extensive numerical results are presented.

Key words. projected gradients, nonmonotone line search, large-scale problems, bound constrained problems, spectral gradient method

AMS subject classifications. 49M07, 49M10, 65K, 90C06, 90C20

PII. S1052623497330963

1. Introduction.

We consider the projected gradient method for the minimization of differentiable functions on nonempty closed and convex sets. Over the last few decades, there have been many different variations of the projected gradient method that can be viewed as the constrained extensions of the optimal gradient method for unconstrained minimization. They all have the common property of maintaining feasibility of the iterates by frequently projecting trial steps on the feasible convex set. This process is in general the most expensive part of any projected gradient method. Moreover, even if projecting is inexpensive, as in the box-constrained case, the method is considered to be very slow, as is its analogue, the optimal gradient method (also known as steepest descent), for unconstrained optimization. On the positive side, the projected gradient method is quite simple to implement and very effective for large-scale problems.

This state of affairs motivates us to combine the projected gradient method with two recently developed ingredients in optimization. First we extend the typical globalization strategies associated with these methods to the nonmonotone line search schemes developed by Grippo, Lampariello, and Lucidi [17] for Newton's method. Second, we propose to associate the spectral steplength, introduced by Barzilai and Borwein [1] and analyzed by Raydan [26]. This choice of steplength requires little computational work and greatly speeds up the convergence of gradient methods. In fact, while the spectral gradient method appears to be a generalized steepest descent method, it is clear from its derivation that it is related to the quasi-Newton family of methods through an approximated secant equation. The fundamental difference is

*Received by the editors December 3, 1997; accepted for publication (in revised form) June 15, 1999; published electronically June 13, 2000.
http://www.siam.org/journals/siopt/10-4/33096.html

†Departamento de Matemática Aplicada, IMECC-UNICAMP, CP 6065, 13081-970 Campinas SP, Brazil (ernesto@ime.unicamp.br, martinez@ime.unicamp.br). The research of the first author was sponsored by FAPESP (grant 95/2452-6). The research of the second author was sponsored by FAPESP (grant 90/3724-6), CNPq, and FAEP-UNICAMP.

‡Departamento de Computación, Facultad de Ciencias, Universidad Central de Venezuela, Ap. 47002, Caracas 1041-A, Venezuela (mraydan@reacciun.ve). This author's research was sponsored by FAPESP, FAEP-UNICAMP, and CDCH-UCV.

that it is a two-point method while the steepest descent method is not. The main idea behind the spectral choice of steplength is that the steepest descent method is very slow but it can be accelerated by taking, instead of the stepsize that comes from the minimization of the function along the gradient of the current iteration, the one that comes from the one-dimensional minimization at the previous step. See Glunt, Hayden, and Raydan [15] for a relationship with the shifted power method to approximate eigenvalues and eigenvectors and also for an interesting chemistry application. See also Raydan [27] for a combination of the spectral choice of steplength with nonmonotone line search techniques to solve unconstrained minimization problems. A successful application of this technique can be found in [5].

Therefore, it is natural and rather easy to transport the spectral gradient idea with a nonmonotone line search to the projected gradient case in order to speed up the convergence of the projected gradient method. In particular, in this work we extend the practical version of Bertsekas [2] that enforces an Armijo-type condition along the curvilinear projection path. This practical version is based on the original version proposed by Goldstein [16] and Levitin and Polyak [19]. We also apply the new ingredients to the feasible continuous projected path that will be properly defined in section 2.

The convergence properties of the projected gradient method for different choices of stepsize have been extensively studied. See, e.g., [2, 3, 7, 11, 16, 19, 22, 30]. For an interesting review of the different convergence results that have been obtained under different assumptions, see Calamai and Moré [7]. For a complete survey see Dunn [12].

In section 2 of this paper we define the spectral projected gradient algorithms and prove global convergence results. In section 3 we present numerical experiments. This set of experiments shows that, in fact, the spectral choice of the steplength represents considerable progress in relation to constant choices and that the nonmonotone framework is useful. Some final remarks are presented in section 4. In particular, we elaborate on the relationship between the spectral gradient method and the quasi-Newton family of methods.

2. Nonmonotone gradient-projection algorithms.

The nonmonotone spectral gradient-projection algorithms introduced in this section apply to problems of the form

$$\text{minimize } f(x) \quad \text{subject to} \quad x \in \Omega,$$

where Ω is a closed convex set in $\mathbb{R}^n$. Throughout this paper we assume that f is defined and has continuous partial derivatives on an open set that contains Ω. Throughout this work $\|\cdot\|$ denotes the 2-norm of vectors and matrices, although in some cases it can be replaced by an arbitrary norm.

Given $z \in \mathbb{R}^n$ we define $P(z)$ as the orthogonal projection on Ω. We denote $g(x) = \nabla f(x)$. The algorithms start with $x_0 \in \Omega$ and use an integer $M \geq 1$, a small parameter $\alpha_{\min} > 0$, a large parameter $\alpha_{\max} > \alpha_{\min}$, a sufficient decrease parameter $\gamma \in (0, 1)$, and safeguarding parameters $0 < \sigma_1 < \sigma_2 < 1$. Initially, $\alpha_0 \in [\alpha_{\min}, \alpha_{\max}]$ is arbitrary. Given $x_k \in \Omega$ and $\alpha_k \in [\alpha_{\min}, \alpha_{\max}]$, Algorithms 2.1 and 2.2 describe how to obtain x_{k+1} and α_{k+1} and when to terminate the process.

ALGORITHM 2.1.

```
Step 1. Detect whether the current point is stationary
    If $\|P(x_k - g(x_k)) - x_k\| = 0$, stop, declaring that $x_k$ is stationary.
Step 2. Backtracking
Step 2.1. Set $\lambda \leftarrow \alpha_k$.
```

Step 2.2. Set $x_+ = P(x_k - \lambda g(x_k))$.
Step 2.3. If

$$f(x_+) \leq \max_{0 \leq j \leq \min\,\{k, M-1\}} f(x_{k-j}) + \gamma \langle x_+ - x_k, g(x_k) \rangle, \tag{1}$$

then define $\lambda_k = \lambda$, $x_{k+1} = x_+$, $s_k = x_{k+1} - x_k$, $y_k = g(x_{k+1}) - g(x_k)$, and go to Step 3.

If (1) does not hold, define

$$\lambda_{new} \in [\sigma_1 \lambda, \sigma_2 \lambda], \tag{2}$$

set $\lambda \leftarrow \lambda_{new}$, and go to Step 2.2.
Step 3.

Compute $b_k = \langle s_k, y_k \rangle$.

If $b_k \leq 0$, set $\alpha_{k+1} = \alpha_{\max}$; else, compute $a_k = \langle s_k, s_k \rangle$ and

$$\alpha_{k+1} = \min\,\{\alpha_{\max}, \max\,\{\alpha_{\min}, a_k / b_k\}\}.$$

The one-dimensional search procedure of Algorithm 2.1 (called SPG1 from now on) takes into account points of the form $P(x_k - \lambda g(x_k))$ for $\lambda \in (0, \alpha_k]$, which, in general, form a curvilinear path (piecewise linear if Ω is a polyhedral set). For this reason, the scalar product $\langle x_+ - x_k, g(x_k) \rangle$ in the nonmonotone Armijo condition (1) must be computed for each trial point x_+. Moreover, in the SPG1 formulation the distance between two consecutive trial points could be very small or even null in the vicinity of corner points of the set Ω. In fact the distance between the projections of two points on the feasible set can be small, even if the points are distant from each other. Clearly, to evaluate the objective function on two close points represents a bad use of available information. Of course, proximity of two consecutive trial points can be computationally detected at the expense of $O(n)$ operations or comparisons.

These observations motivated us to define Algorithm 2.2. This algorithm is also based on the spectral projected gradient direction $P(x_k - \alpha_k g(x_k)) - x_k$, with α_k as the safeguarded "inverse Rayleigh quotient" $\frac{\langle s_{k-1}, s_{k-1} \rangle}{\langle s_{k-1}, y_{k-1} \rangle}$. (Observe that $\frac{\langle s_{k-1}, y_{k-1} \rangle}{\langle s_{k-1}, s_{k-1} \rangle}$ is in fact a Rayleigh quotient corresponding to the average Hessian matrix $\int_0^1 \nabla^2 f(x_{k-1} + t s_{k-1}) dt$.) However, in the case of rejection of the first trial point, the next ones are computed along the same direction. As a consequence, $\langle x_+ - x_k, g(x_k) \rangle$ must be computed only at the first trial and the projection operation must be performed only once per iteration. So, Algorithm 2.2, which will be called SPG2 in the rest of the paper, coincides with SPG1 except at the backtracking step, whose description is given below.

Algorithm 2.2.
Step 2 (*Backtracking*)
Step 2.1. Compute $d_k = P(x_k - \alpha_k g(x_k)) - x_k$. Set $\lambda \leftarrow 1$.
Step 2.2. Set $x_+ = x_k + \lambda d_k$.
Step 2.3. If

$$f(x_+) \leq \max_{0 \leq j \leq \min\,\{k, M-1\}} f(x_{k-j}) + \gamma \lambda \langle d_k, g(x_k) \rangle, \tag{3}$$

then define $\lambda_k = \lambda$, $x_{k+1} = x_+$, $s_k = x_{k+1} - x_k$, $y_k = g(x_{k+1}) - g(x_k)$, and go to Step 3.

If (3) does not hold, define λ_{new} as in (2), set $\lambda \leftarrow \lambda_{new}$, and go to Step 2.2.

In both algorithms the computation of λ_{new} uses one-dimensional quadratic interpolation and it is safeguarded taking $\lambda \leftarrow \lambda/2$ when the minimum of the one-dimensional quadratic lies outside $[0.1, 0.9\lambda]$. Notice also that the line search conditions (1) and (3) guarantee that the sequence $\{x_k\}$ remains in $\Omega_0 \equiv \{x \in \Omega : f(x) \leq f(x_0)\}$.

It will be useful in our theoretical analysis to define the *scaled projected gradient* $g_t(x)$ as

$$g_t(x) = [P(x - tg(x)) - x]$$

for all $x \in \Omega$, $t > 0$. If x is an iterate of SPG1 or SPG2 and $t = \alpha_k$ the scaled projected gradient is the *spectral projected gradient* (SPG) that gives the name to our methods. If $t = 1$, the scaled projected gradient is the *continuous projected gradient* whose ∞-norm $\|g_1(x)\|_\infty$ is used for the termination criterion of the algorithms. In fact, the annihilation of $g_t(x)$ is equivalent to the satisfaction of first-order stationary conditions. This property is stated in the following lemma, whose proof is a straightforward consequence of the convexity of Ω.

LEMMA 2.1. *For all $x \in \Omega$, $t \in (0, \alpha_{\max}]$,*

(i) $\langle g(x), g_t(x)\rangle \leq -\frac{1}{t}\,\|g_t(x)\|_2^2 \leq -\frac{1}{\alpha_{\max}}\,\|g_t(x)\|_2^2$.

(ii) *The vector $g_t(\bar{x})$ vanishes if and only if $\bar{x}$ is a constrained stationary point.*

Now, let us prove that both algorithms are well defined and have the property that every accumulation point $\bar{x}$ is a constrained stationary point, *i.e.*, that

$$\langle g(\bar{x}), x - \bar{x}\rangle \geq 0 \quad \text{for all} \quad x \in \Omega.$$

The proof of our first theorem relies on Proposition 2.3.3 in Bertsekas [3], which is related to the Armijo condition along the projection arc. This proposition was originally shown in [14]. For completeness we include in the next lemma some technical results from [3] that will be used in our proof.

LEMMA 2.2. (i) *For all $x \in \Omega$ and $z \in \mathbb{R}^n$, the function $h : [0, \infty) \to \mathbb{R}$ given by*

$$h(s) = \frac{\|P(x + sz) - x\|}{s} \quad \textit{for all } s > 0$$

is monotonically nonincreasing.

(ii) *For all $x \in \Omega$ there exists $s_x > 0$ such that for all $t \in [0, s_x]$ it holds that*

$$f(P(x - tg(x))) - f(x) \leq \gamma\langle g(x), g_t(x)\rangle.$$

Proof. See Lemma 2.3.1 and Theorem 2.3.3 (part (a)) in [3].

THEOREM 2.3. *Algorithm SPG1 is well defined, and any accumulation point of the sequence $\{x_k\}$ that it generates is a constrained stationary point.*

Proof. From Lemma 2.2(ii), we have for all $\lambda \in [0, \min\{s_{x_k}, \alpha_{\min}\}]$ that

$$\begin{aligned} f(P(x_k - \lambda g(x_k))) - \max_{0 \leq j \leq M-1} f(x_{k-j}) &\leq f(P(x_k - \lambda g(x_k))) - f(x_k) \\ &\leq \gamma\langle g(x_k), g_\lambda(x_k)\rangle. \end{aligned}$$

Therefore, a stepsize satisfying (1) will be found after a finite number of trials, and Algorithm SPG1 is well defined.

Let $\bar{x} \in \Omega$ be an accumulation point of $\{x_k\}$, and relabel $\{x_k\}$ a subsequence converging to $\bar{x}$. We consider two cases.

Case 1. If $\inf \lambda_k = 0$, then there exists a subsequence $\{x_k\}_K$ such that

$$\lim_{k \in K} \lambda_k = 0.$$

In that case, from the way λ_k is chosen in (1), there exists an index $\bar{k}$ sufficiently large such that for all $k \geq \bar{k}$, $k \in K$, there exists ρ_k, $0 < \sigma_1 \leq \rho_k \leq \sigma_2$, for which $\psi_k \equiv \lambda_k / \rho_k > 0$ fails to satisfy condition (1), i.e.,

$$f(P(x_k - \psi_k g(x_k))) > \max_{0 \leq j \leq M-1} f(x_{k-j}) + \gamma \langle g(x_k), P(x_k - \psi_k g(x_k)) - x_k \rangle$$

$$\geq f(x_k) + \gamma \langle g(x_k), P(x_k - \psi_k g(x_k)) - x_k \rangle.$$

Therefore, it follows that

$$f(P(x_k - \psi_k g(x_k))) - f(x_k) > \gamma \langle g(x_k), g_{\psi_k}(x_k) \rangle. \tag{4}$$

By the mean value theorem we obtain

$$f(P(x_k - \psi_k g(x_k))) - f(x_k) = \langle g(x_k), g_{\psi_k}(x_k) \rangle + \langle g(\xi_k) - g(x_k), g_{\psi_k}(x_k) \rangle, \tag{5}$$

where ξ_k lies along the line segment connecting x_k and $P(x_k - \psi_k g(x_k))$.

Combining (4) and (5) we obtain for all $k \in K$ sufficiently large that

$$(1 - \gamma) \langle g(x_k), g_{\psi_k}(x_k) \rangle > \langle g(x_k) - g(\xi_k), g_{\psi_k}(x_k) \rangle. \tag{6}$$

Using Lemmas 2.1 and 2.2, we have

$$\langle g(x_k), g_{\psi_k}(x_k) \rangle \leq -\frac{1}{\psi_k} \|g_{\psi_k}(x_k)\|_2^2 \leq -\frac{1}{\alpha_k} \|g_{\alpha_k}(x_k)\|_2 \, \|g_{\psi_k}(x_k)\|_2, \tag{7}$$

where α_k is the initial stepsize at iteration k. Combining (6) and (7) and using the Schwartz inequality, we obtain for $k \in K$ sufficiently large

$$\frac{(1 - \gamma)}{\alpha_k} \|g_{\alpha_k}(x_k)\|_2 \, \|g_{\psi_k}(x_k)\|_2 < \langle g(\xi_k) - g(x_k), g_{\psi_k}(x_k) \rangle$$

$$\leq \|g(\xi_k) - g(x_k)\|_2 \, \|g_{\psi_k}(x_k)\|_2.$$

Using that $\|g_{\psi_k}(x_k)\|_2 \neq 0$, we have

$$\frac{(1 - \gamma)}{\alpha_k} \|g_{\alpha_k}(x_k)\|_2 < \|g(\xi_k) - g(x_k)\|_2. \tag{8}$$

Since $\psi_k \to 0$ and $x_k \to \bar{x}$ as $k \to \infty$, $k \in K$, then $\xi_k \to \bar{x}$ as $k \to \infty$, $k \in K$. Taking a convenient subsequence $\bar{K} \subseteq K$ such that $\{\alpha_k\}$ is convergent to $\bar{\alpha} \in [\alpha_{\min}, \alpha_{\max}]$, and taking limits in (8) as $k \to \infty$, $k \in \bar{K}$, we deduce that

$$\|g_{\bar{\alpha}}(\bar{x})\|_2 \leq 0.$$

Therefore, $g_{\bar{\alpha}}(\bar{x}) = 0$, and $\bar{x}$ is a constrained stationary point.

Case 2. Assume that $\inf \lambda_k \geq \rho > 0$. Let us suppose by way of contradiction that $\bar{x}$ is not a constrained stationary point. Therefore $\|g_\lambda(\bar{x})\| > 0$ for all $\lambda \in (0, \alpha_{\max}]$.

By continuity and compactness, there exists $\delta > 0$ such that $\|g_\lambda(\bar{x})\| \geq \delta > 0$ for all $\lambda \in [\rho, \alpha_{\max}]$. Using the first part of the proof of the theorem in [17, p. 709], we obtain a monotonically nonincreasing sequence $\{f(x_{l(k)})\}$. Indeed, let $l(k)$ be an integer such that $k - \min\{k, M-1\} \leq l(k) \leq k$ and

$$f(x_{l(k)}) = \max_{0 \leq j \leq \min\{k, M-1\}} f(x_{k-j}).$$

From (1) it follows that, for $k > M-1$ (see [17] for details),

$$f(x_{l(k)}) \leq f(x_{l(l(k)-1)}) + \gamma \langle g(x_{l(k)-1}), g_{\lambda_{l(k)-1}}(x_{l(k)-1}) \rangle.$$

By continuity, for $k \geq \bar{k}$ sufficiently large, $\|g_\lambda(\bar{x}_k)\| \geq \delta/2$. Hence, using Lemma 2.1, we obtain

$$f(x_{l(k)}) \leq f(x_{l(l(k)-1)}) - \frac{\gamma}{\alpha_{\max}} \|g_{\lambda_{l(k)-1}}(x_{l(k)-1})\|_2^2 \leq f(x_{l(l(k)-1)}) - \frac{\gamma\delta^2}{4\alpha_{\max}}.$$

When $k \to \infty$, clearly $f(x_{l(k)}) \to -\infty$, which is a contradiction. In fact, f is a continuous function and so $f(x_k)$ converges to $f(\bar{x})$. □

THEOREM 2.4. *Algorithm SPG2 is well defined, and any accumulation point of the sequence $\{x_k\}$ that it generates is a constrained stationary point.*

Proof. If x_k is not a constrained stationary point, then by Lemma 2.1

$$\langle g(x_k), d_k \rangle = \langle g(x_k), g_{\alpha_k}(x_k) \rangle \leq -\frac{1}{\alpha_{\max}} \|g_{\alpha_k}(x_k)\|_2^2 < 0,$$

and the search direction is a descent direction. Hence, a stepsize satisfying (3) will be found after a finite number of trials, and Algorithm SPG2 is well defined.

Let $\bar{x} \in \Omega$ be an accumulation point of $\{x_k\}$, and relabel $\{x_k\}$ a subsequence converging to $\bar{x}$. We consider two cases.

Case 1. Assume that $\inf \lambda_k = 0$. Suppose, by contradiction, that $\bar{x}$ is not stationary. By continuity and compactness, there exists $\delta > 0$ such that

$$\left\langle g(\bar{x}), \frac{P(\bar{x} - \alpha g(\bar{x})) - \bar{x}}{\|P(\bar{x} - \alpha g(\bar{x})) - \bar{x}\|} \right\rangle < -\delta \quad \text{for all} \quad \alpha \in [\alpha_{\min}, \alpha_{\max}].$$

This implies that

$$\left\langle g(x_k), \frac{P(x_k - \alpha g(x_k)) - x_k}{\|P(x_k - \alpha g(x_k)) - x_k\|} \right\rangle < -\delta/2 \quad \text{for all} \quad \alpha \in [\alpha_{\min}, \alpha_{\max}] \tag{9}$$

and k large enough on the subsequence that converges to $\bar{x}$.

Since $\inf \lambda_k = 0$, there exists a subsequence $\{x_k\}_K$ such that

$$\lim_{k \in K} \lambda_k = 0.$$

In that case, from the way λ_k is chosen in (3), there exists an index $\bar{k}$ sufficiently large such that for all $k \geq \bar{k}$, $k \in K$, there exists ρ_k, $0 < \sigma_1 \leq \rho_k \leq \sigma_2$, for which $\lambda_k/\rho_k > 0$ fails to satisfy condition (3); i.e.,

$$f\left(x_k + \frac{\lambda_k}{\rho_k} d_k\right) > \max_{0 \leq j \leq M-1} f(x_{k-j}) + \gamma \frac{\lambda_k}{\rho_k} \langle g(x_k), d_k \rangle \geq f(x_k) + \gamma \frac{\lambda_k}{\rho_k} \langle g(x_k), d_k \rangle.$$

Hence,

$$\frac{f(x_k + \frac{\lambda_k}{\rho_k} d_k) - f(x_k)}{\lambda_k/\rho_k} > \gamma \langle g(x_k), d_k \rangle.$$

By the mean value theorem, this relation can be written as

$$\langle g(x_k + t_k d_k), d_k \rangle > \gamma \langle g(x_k), d_k \rangle \quad \text{for all } k \in K,\ k \geq \bar{k}, \tag{10}$$

where t_k is a scalar in the interval $[0, \lambda_k/\rho_k]$ that goes to zero as $k \in K$ goes to infinity.

Taking a convenient subsequence such that $d_k/\|d_k\|$ is convergent to d, and taking limits in (10), we deduce that $(1-\gamma)\langle g(\bar{x}), d \rangle \geq 0$. (In fact, observe that $\{\|d_k\|\}_K$ is bounded and so $t_k \|d_k\| \to 0$.) Since $(1-\gamma) > 0$ and $\langle g(x_k), d_k \rangle < 0$ for all k, then $\langle g(\bar{x}), d \rangle = 0$.

By continuity and the definition of d_k this implies that for k large enough on that subsequence we have that

$$\left\langle g(x_k), \frac{P(x_k - \alpha_k g(x_k)) - x_k}{\|P(x_k - \alpha_k g(x_k)) - x_k\|} \right\rangle > -\delta/2,$$

which contradicts (9).

Case 2. Assume that $\inf \lambda_k \geq \rho > 0$. Let us suppose by way of contradiction that $\bar{x}$ is not a constrained stationary point. Therefore $\|g_\lambda(\bar{x})\| > 0$ for all $\lambda \in (0, \alpha_{\max}]$. By continuity and compactness, there exists $\delta > 0$ such that $\|g_\lambda(\bar{x})\| \geq \delta > 0$ for all $\lambda \in [\rho, \alpha_{\max}]$.

As in the proof of the second case of Theorem 2.3,

$$f(x_{l(k)}) = \max_{0 \leq j \leq \min\,\{k, M-1\}} f(x_{k-j})$$

is a monotonically nonincreasing sequence. From (3) it follows that, for $k > M-1$,

$$f(x_{l(k)}) \leq f(x_{l(l(k)-1)}) + \gamma \lambda_{l(k)-1} \langle g(x_{l(k)-1}), g_{\alpha_{l(k)-1}}(x_{l(k)-1}) \rangle.$$

By continuity, for $k \geq \bar{k}$ sufficiently large, $\|g_{\alpha_k}(\bar{x_k})\| \geq \delta/2$. Hence, using Lemma 2.1, we obtain

$$f(x_{l(k)}) \leq f(x_{l(l(k)-1)}) - \frac{\gamma\,\rho}{\alpha_{\max}} \|g_{\alpha_{l(k)-1}}(x_{l(k)-1})\|_2^2 \leq f(x_{l(l(k)-1)}) - \frac{\gamma \delta^2 \rho}{4\alpha_{\max}}.$$

When $k \to \infty$, clearly $f(x_{l(k)}) \to -\infty$, which is a contradiction. In fact, f is a continuous function and so $f(x_k)$ converges to $f(\bar{x})$. □

3. Numerical results.

The algorithms SPG1 and SPG2 introduced in the previous section compute at least one projection on the feasible set Ω per iteration. Therefore, these algorithms are especially interesting in the case in which this projection is easy to compute. An important situation in which the projection is trivial is when Ω is an n-dimensional box, possibly with some infinite bounds. In fact, good algorithms for box constrained minimization are the essential tool for the development of efficient augmented Lagrangian methods for general nonlinear programming (see [8, 10, 13]). With this in mind, we implemented the spectral projected gradient algorithms for the case in which Ω is described by bounds on the variables. In order to assess the reliability of SPG algorithms, we tested them against the well-known

TABLE 1
Problem sets according to the CUTE classification.

Set #	Objective type	Problem interest
1	other	academic
2	other	modeling
3	other	real application
4	sum of squares	academic
5	sum of squares	modeling
6	quadratic	academic
7	quadratic	modeling
8	quadratic	real application

package LANCELOT [9] using *all* the bound constrained problems with more than 50 variables from the CUTE [10] collection. Only problem GRIDGENA was excluded from our tables because it gives an "exception error" when evaluated at some point by SPG algorithms. For all the problems with variable dimension, we used the largest dimension that is admissible without modification of the internal variables of the "double large" installation of CUTE.

Altogether, we solved 50 problems. The horizontal lines in Tables 2–5 divide the CUTE problems into 8 classes according to objective function type (quadratic, sum of squares, other) and problem interest (academic, modeling, real application). All problems are bound constrained only, twice continuously differentiable, and with more than 50 variables. The 8 sets, in the order in which they appear in the tables, are described in Table 1.

In the numerical experiments we used the default options for LANCELOT, i.e.,

- `exact-second-derivatives-used`,
- `bandsolver-preconditioned-cg-solver-used 5`,
- `exact-cauchy-point-required`,
- `infinity-norm-trust-region-used`,
- `gradient-tolerance 1.0D-05`.

We are deeply concerned with the reproducibility of the numerical results presented in this paper. For this reason, all the used codes are available by e-mail request to any of the authors, who are also available to discuss computational details.

All the experiments were run in a SPARCstation Sun Ultra 1, with an UltraSPARC 64-bit processor, 167 MHz clock and 128 MBytes of RAM memory. SPG codes are in Fortran77 and were compiled with the optimization compiler option -O4.

For the SPG methods we used $\gamma = 10^{-4}$, $\alpha_{min} = 10^{-30}$, $\alpha_{max} = 10^{30}$, $\sigma_1 = 0.1$, $\sigma_2 = 0.9$, and $\alpha_0 = 1/\|g_1(x_0)\|_\infty$. After running a few problems with $M \in \{5, 10, 15\}$, we decided to use $M = 10$, as the tests did not show meaningful differences. To decide when to stop the execution of the algorithms declaring convergence we used the criterion $\|g_1(x_k)\|_\infty \leq 10^{-5}$. We also stopped the execution of SPG when 50,000 iterations or 200,000 function evaluations were completed without achieving convergence.

To complete the numerical insight into the behavior of SPG methods, we also ran the projected gradient algorithm (PGA), which turns out to be identical to SPG1, with the initial choice of steplength $\alpha_k \equiv 1$. In this case we implemented both the monotone version of PGA, which corresponds to $M = 1$, and the nonmonotone one with $M = 10$. The convergence of the nonmonotone version is a particular case of our Theorem 2.3. The performance of the nonmonotone version of PGA, which is more efficient than the monotone version, is reported in Table 2.

Table 2
Performance of nonmonotone ($M = 10$) projected gradient.

Problem	n	IT	FE	GE	Time	$f(x)$	$\|g_1(x)\|_\infty$
BDEXP	5000	13065	13066	13066	459.99	3.464D−03	9.999D−06
EXPLIN	120	30608	200001	30609	15.08	−7.238D+05	7.768D−05
EXPLIN2	120	19581	126328	19582	9.87	−7.245D+05	8.192D−06
EXPQUAD	120	7899	200001	7900	22.06	−3.626D+06	3.875D−03
MCCORMCK	10000	16080	47939	16081	2755.50	−9.133D+03	2.485D−09
PROBPENL	500	888	10249	889	11.39	3.992D−07	7.265D−06
QRTQUAD	120	3464	38175	3465	3.76	−3.625D+06	5.303D−06
S368	100	2139	12532	2140	317.55	−7.085D+01	9.966D−06
HADAMALS	1024	1808	11468	1809	157.88	3.067D+04	9.611D−06
CHEBYQAD	50	5287	50893	5288	607.89	5.386D−03	9.918D−06
HS110	50	1	2	2	0.00	−9.990D+09	0.000D+00
LINVERSE	1999	19563	200001	19564	1465.91	6.820D+02	9.202D−02
NONSCOMP	10000	3737	25220	3738	559.04	7.632D−13	9.933D−06
QR3DLS	610	17272	200001	17273	735.62	3.051D−01	3.638D−01
SCON1LS	1002	40237	200001	40238	1512.18	6.572D+01	8.501D−02
DECONVB	61	6536	35665	6537	10.00	2.713D−03	1.814D−06
BIGGSB1	1000	50001	104775	50002	190.46	1.896D−02	1.362D−03
BQPGABIM	50	2222	22640	2223	1.68	−3.790D−05	9.972D−06
BQPGASIM	50	1247	12394	1248	0.94	−5.520D−05	9.334D−06
BQPGAUSS	2003	13482	200001	13483	986.07	−1.294D−01	1.037D+00
CHENHARK	1000	50001	173351	50002	323.09	−2.000D+00	5.299D−04
CVXBQP1	10000	1	2	2	0.10	2.250D+06	0.000D+00
HARKERP2	100	100	304	101	0.26	−5.000D−01	0.000D+00
JNLBRNG1	15625	13681	28689	13682	3332.51	−1.806D−01	5.686D−06
JNLBRNG2	15625	21444	107760	21445	8427.10	−4.150D+00	9.624D−06
JNLBRNGA	15625	12298	27172	12299	2666.47	−2.685D−01	5.388D−06
JNLBRNGB	15625	32771	200001	32772	12672.71	−5.569D+00	3.744D+00
NCVXBQP1	10000	1	2	2	0.10	−1.986D+10	0.000D+00
NCVXBQP2	10000	18012	200001	18013	4053.97	−1.334D+10	5.798D−01
NCVXBQP3	10000	15705	200001	15706	3955.02	−6.559D+09	2.609D+00
NOBNDTOR	14884	3649	7300	3650	718.13	−4.405D−01	8.604D−06
OBSTCLAE	15625	5049	11402	5050	1119.07	1.901D+00	1.000D−05
OBSTCLAL	15625	2734	6838	2735	634.97	1.901D+00	9.986D−06
OBSTCLBL	15625	3669	9084	3670	846.45	7.296D+00	9.995D−06
OBSTCLBM	15625	2941	7634	2942	694.42	7.296D+00	9.983D−06
OBSTCLBU	15625	3816	9403	3817	880.51	7.296D+00	9.981D−06
PENTDI	1000	50001	199995	50002	460.38	−7.500D− 01	2.688D−05
TORSION1	14884	4540	9082	4541	890.47	−4.257D−01	6.673D−06
TORSION2	14884	8704	17294	8705	1703.87	−4.257D−01	6.599D−06
TORSION3	14884	1941	4525	1942	406.85	−1.212D+00	9.957D−06
TORSION4	14884	4273	9062	4274	862.93	−1.212D+00	9.897D−06
TORSION5	14884	672	1651	673	144.80	−2.859D+00	9.813D−06
TORSION6	14884	1569	3322	1570	316.06	−2.859D+00	9.908D−06
TORSIONA	14884	4155	8312	4156	953.30	−4.184D−01	8.980D−06
TORSIONB	14884	8274	16417	8275	1899.52	−4.184D−01	8.829D−06
TORSIONC	14884	1933	4563	1934	476.48	−1.204D+00	9.976D−06
TORSIOND	14884	4325	9218	4326	1013.10	−1.204D+00	9.854D−06
TORSIONE	14884	688	1695	689	172.87	−2.851D+00	9.727D−06
TORSIONF	14884	1493	3143	1494	349.72	−2.851D+00	9.712D−06
ODNAMUR	11130	13222	200001	13223	5249.00	1.209D+04	5.192D+00

The complete performance of LANCELOT on this set of problems is reported in Table 3. In Tables 4 and 5 we show the behavior of SPG1 and SPG2, respectively.

For LANCELOT, we report the number of outer iterations (or function evaluations) (IT$_{out}$-FE), gradient evaluations (GE), conjugate gradient (or inner) iterations

Table 3
Performance of LANCELOT.

Problem	n	IT$_{out}$-FE	GE	IT$_{in}$-CG	Time	$f(x)$	$\|g_1(x)\|_\infty$
BDEXP	5000	10	11	26	3.19	1.964D−03	6.167D−06
EXPLIN	120	13	14	50	0.08	−7.238D+05	5.183D−09
EXPLIN2	120	11	12	24	0.07	−7.245D+05	1.012D−06
EXPQUAD	120	18	16	52	0.14	−3.626D+06	1.437D−06
MCCORMCK	10000	7	6	5	4.71	−9.133D+03	5.861D−06
PROBPENL	500	1	2	0	0.17	3.992D−07	3.424D−07
QRTQUAD	120	168	137	187	1.23	−3.625D+06	3.568D−06
S368	100	7	7	11	2.19	−1.337D+02	3.314D−06
HADAMALS	1024	33	34	5654	157.60	7.444D+02	7.201D−06
CHEBYQAD	50	65	48	829	5.41	5.386D−03	7.844D−06
HS110	50	1	2	0	0.02	−9.990D+09	0.000D+00
LINVERSE	1999	35	30	2303	77.52	6.810D+02	8.407D−06
NONSCOMP	10000	8	9	9	4.74	3.055D−14	9.749D−09
QR3DLS	610	255	226	25036	434.02	3.818D−08	4.051D−06
SCON1LS	1002	1604	1372	1357	56.51	7.070D−10	8.568D−06
DECONVB	61	17	16	233	0.40	1.236D−08	2.147D−06
BIGGSB1	1000	501	502	500	6.17	1.500D−02	4.441D−16
BQPGABIM	50	3	4	10	0.03	−3.790D−05	6.120D−06
BQPGASIM	50	3	4	9	0.03	−5.520D−05	5.733D−06
BQPGAUSS	2003	8	9	2345	42.60	−3.626D−01	4.651D−06
CHENHARK	1000	205	206	484	5.02	−2.000D+00	6.455D−06
CVXBQP1	10000	1	2	1	3.69	2.250D+06	0.000D+00
HARKERP2	100	1	2	2	0.11	−5.000D−01	7.514D−13
JNLBRNG1	15625	24	25	1810	217.19	−1.806D−01	4.050D−06
JNLBRNG2	15625	14	15	912	108.93	−4.150D+00	9.133D−07
JNLBRNGA	15625	21	22	1327	155.93	−2.685D−01	1.191D−06
JNLBRNGB	15625	10	11	329	42.58	−6.281D+00	2.602D−06
NCVXBQP1	10000	1	2	0	3.27	−1.986D+10	0.000D+00
NCVXBQP2	10000	3	4	407	6.62	−1.334D+10	5.821D−11
NCVXBQP3	10000	5	6	360	6.67	−6.558D+09	2.915D−06
NOBNDTOR	14884	36	37	790	117.34	−4.405D−01	2.758D−06
OBSTCLAE	15625	4	5	7409	1251.08	1.901D+00	1.415D−06
OBSTCLAL	15625	24	25	480	58.05	1.901D+00	5.323D−06
OBSTCLBL	15625	18	19	2761	397.58	7.296D+00	1.996D−06
OBSTCLBM	15625	5	6	1377	233.70	7.296D+00	2.243D−06
OBSTCLBU	15625	19	20	787	112.55	7.296D+00	1.529D−06
PENTDI	1000	1	2	0	0.20	−7.500D−01	0.000D+00
TORSION1	14884	37	38	793	96.88	−4.257D−01	1.237D−06
TORSION2	14884	9	10	4339	722.28	−4.257D−01	4.337D−06
TORSION3	14884	19	20	241	27.36	−1.212D+00	2.234D−06
TORSION4	14884	15	16	5639	894.13	−1.212D+00	6.469D−07
TORSION5	14884	9	10	72	10.48	−2.859D+00	3.186D−06
TORSION6	14884	10	11	4895	579.62	−2.859D+00	8.124D−07
TORSIONA	14884	37	38	795	103.70	−4.184D−01	9.590D−07
TORSIONB	14884	10	11	4025	722.79	−4.184D−01	1.329D−06
TORSIONC	14884	19	20	241	29.77	−1.205D+00	2.236D−06
TORSIOND	14884	9	10	9134	1369.14	−1.205D+00	5.184D−06
TORSIONE	14884	9	10	72	11.25	−2.851D+00	3.201D−06
TORSIONF	14884	10	11	5008	631.14	−2.851D+00	8.796D−07
ODNAMUR	11130	11	12	26222	1416.03	9.237D+03	7.966D−06

(IT$_{in}$-CG), CPU time in seconds (Time), functional value at the final iterate ($f(x)$), and ∞-norm of the "continuous projected gradient" at the final iterate ($\|g_1(x)\|_\infty$). For SPG methods, we report number of iterations (IT), function evaluations (FE), gradient evaluations (GE), CPU time in seconds (Time), best function value found ($f(x)$),

TABLE 4
Performance of SPG1.

Problem	n	IT	FE	GE	Time	$f(x)$	$\|g_1(x)\|_\infty$
BDEXP	5000	12	13	13	0.45	2.744D−03	7.896D−06
EXPLIN	120	66	75	67	0.01	−7.238D+05	3.100D−06
EXPLIN2	120	48	54	49	0.01	−7.245D+05	9.746D−07
EXPQUAD	120	92	107	93	0.03	−3.626D+06	4.521D−06
MCCORMCK	10000	16	17	17	1.78	−9.133D+03	4.812D−06
PROBPENL	500	2	7	3	0.01	3.992D−07	1.721D−07
QRTQUAD	120	1693	5242	1694	0.74	−3.625D+06	5.125D−06
S368	100	8	14	9	0.67	−1.200D+02	1.566D−07
HADAMALS	1024	33	42	34	1.49	3.107D+04	4.828D−08
CHEBYQAD	50	970	1545	971	35.52	5.386D−03	9.993D−06
HS110	50	1	2	2	0.00	−9.990D+09	0.000D+00
LINVERSE	1999	1707	2958	1708	45.42	6.810D+02	9.880D−06
NONSCOMP	10000	43	44	44	2.28	3.419D−10	7.191D−06
QR3DLS	610	50001	106513	50002	884.18	2.118D−04	9.835D−03
SCON1LS	1002	50001	75083	50002	882.43	1.329D+01	7.188D−03
DECONVB	61	1786	2585	1787	1.68	4.440D−08	9.237D−06
BIGGSB1	1000	6820	11186	6821	23.15	1.621D−02	9.909D−06
BQPGABIM	50	30	39	31	0.01	−3.790D−05	8.855D−06
BQPGASIM	50	32	39	33	0.01	−5.520D−05	9.100D−06
BQPGAUSS	2003	50001	86373	50002	930.52	−3.623D−01	1.930D−02
CHENHARK	1000	3563	6113	3564	14.89	−2.000D+00	9.993D−06
CVXBQP1	10000	1	2	2	0.10	2.250D+06	0.000D+00
HARKERP2	100	33	46	34	0.06	−5.000D−01	0.000D+00
JNLBRNG1	15625	1335	1897	1336	283.55	−1.806D−01	9.624D−06
JNLBRNG2	15625	1356	2121	1357	296.46	−4.150D+00	9.738D−06
JNLBRNGA	15625	629	933	630	116.77	−2.685D−01	9.809D−06
JNLBRNGB	15625	8531	13977	8532	1635.15	−6.281D+00	9.903D−06
NCVXBQP1	10000	1	2	2	0.10	−1.986D+10	0.000D+00
NCVXBQP2	10000	60	83	61	3.47	−1.334D+10	8.219D−06
NCVXBQP3	10000	112	118	113	5.31	−6.558D+09	6.019D−06
NOBNDTOR	14884	568	817	569	99.62	−4.405D−01	9.390D−06
OBSTCLAE	15625	749	1028	750	136.98	1.901D+00	7.714D−06
OBSTCLAL	15625	290	411	291	53.56	1.901D+00	7.261D−06
OBSTCLBL	15625	354	500	355	65.52	7.296D+00	9.024D−06
OBSTCLBM	15625	249	343	250	45.74	7.296D+00	9.139D−06
OBSTCLBU	15625	325	468	326	60.44	7.296D+00	7.329D−06
PENTDI	1000	12	14	13	0.07	−7.500D−01	8.523D−07
TORSION1	14884	574	832	575	101.00	−4.257D−01	9.525D−06
TORSION2	14884	586	862	587	102.79	−4.257D−01	9.712D−06
TORSION3	14884	231	350	232	41.47	−1.212D+00	9.593D−06
TORSION4	14884	190	259	191	32.66	−1.212D+00	8.681D−06
TORSION5	14884	83	101	84	13.84	−2.859D+00	9.169D−06
TORSION6	14884	82	97	83	13.58	−2.859D+00	7.987D−06
TORSIONA	14884	722	1057	723	147.94	−4.184D−01	8.590D−06
TORSIONB	14884	527	765	528	107.52	−4.184D−01	9.475D−06
TORSIONC	14884	190	270	191	38.50	−1.204D+00	9.543D−06
TORSIOND	14884	241	340	242	48.43	−1.204D+00	9.575D−06
TORSIONE	14884	57	76	58	11.42	−2.851D+00	8.700D−06
TORSIONF	14884	67	85	68	14.16	−2.851D+00	9.352D−06
ODNAMUR	11130	50001	82984	50002	4187.58	9.250D+03	9.690D−02

and ∞-norm of the continuous projected gradient at the final iterate ($\|g_1(x)\|_\infty$).

The numerical results of 10 problems deserve special comments:

(1) BDEXP ($n = 5,000$): LANCELOT obtained $f(x) = 1.964 \times 10^{-3}$ in 3.19 seconds, whereas SPG1 and SPG2 got $f(x) = 2.744 \times 10^{-3}$ in 0.45 seconds. Since the gradient norm is computed in LANCELOT only after each outer

Table 5
Performance of SPG2.

Problem	n	IT	FE	GE	Time	$f(x)$	$\|g_1(x)\|_\infty$
BDEXP	5000	12	13	13	0.45	2.744D−03	7.896D−06
EXPLIN	120	54	57	55	0.01	−7.238D+05	4.482D−06
EXPLIN2	120	56	59	57	0.01	−7.245D+05	5.633D−06
EXPQUAD	120	92	110	93	0.03	−3.626D+06	7.644D−06
MCCORMCK	10000	16	17	17	1.78	−9.133D+03	4.812D−06
PROBPENL	500	2	6	3	0.01	3.992D−07	1.022D−07
QRTQUAD	120	598	1025	599	0.19	−3.624D+06	8.049D−06
S368	100	16	19	17	1.15	−1.403D+02	1.963D−08
HADAMALS	1024	30	42	31	1.27	3.107D+04	2.249D−07
CHEBYQAD	50	1240	2015	1241	45.73	5.386D−03	8.643D−06
HS110	50	1	2	2	0.00	−9.990D+09	0.000D+00
LINVERSE	1999	1022	1853	1023	26.75	6.810D+02	8.206D−06
NONSCOMP	10000	43	44	44	2.22	3.419D−10	7.191D−06
QR3DLS	610	50001	107915	50002	869.25	2.312D−04	1.599D−02
SCON1LS	1002	50001	76011	50002	835.10	1.416D+01	1.410D−02
DECONVB	61	1670	2560	1671	1.38	4.826D−08	9.652D−06
BIGGSB1	1000	7571	12496	7572	24.41	1.626D−02	9.999D−06
BQPGABIM	50	24	37	25	0.01	−3.790D−05	8.640D−06
BQPGASIM	50	33	46	34	0.01	−5.520D−05	8.799D−06
BQPGAUSS	2003	50001	87102	50002	902.26	−3.624D−01	2.488D−03
CHENHARK	1000	2464	4162	2465	9.60	−2.000D+00	9.341D−06
CVXBQP1	10000	1	2	2	0.10	2.250D+06	2.776D−17
HARKERP2	100	33	46	34	0.06	−5.000D−01	1.110D−16
JNLBRNG1	15625	1664	2524	1665	349.19	−1.806D−01	6.265D−06
JNLBRNG2	15625	1443	2320	1444	309.22	−4.150D+00	9.665D−06
JNLBRNGA	15625	981	1530	982	180.92	−2.685D−01	6.687D−06
JNLBRNGB	15625	17014	28077	17015	3180.14	−6.281D+00	1.000D−05
NCVXBQP1	10000	1	2	2	0.10	−1.986D+10	2.776D−17
NCVXBQP2	10000	84	93	85	4.00	−1.334D+10	2.956D−06
NCVXBQP3	10000	111	117	112	5.13	−6.558D+09	2.941D−06
NOBNDTOR	14884	566	834	567	98.52	−4.405D− 01	8.913D−06
OBSTCLAE	15625	639	936	640	116.86	1.901D+00	9.343D−06
OBSTCLAL	15625	176	243	177	31.69	1.901D+00	6.203D−06
OBSTCLBL	15625	321	460	322	58.49	7.296D+00	3.731D−06
OBSTCLBM	15625	143	192	144	25.63	7.296D+00	8.294D−06
OBSTCLBU	15625	311	449	312	56.72	7.296D+00	9.703D−06
PENTDI	1000	1	3	2	0.01	−7.500D−01	0.000D+00
TORSION1	14884	685	1023	686	119.38	−4.257D−01	9.404D−06
TORSION2	14884	728	1117	729	127.62	−4.257D−01	9.616D−06
TORSION3	14884	183	264	184	31.72	−1.212D+00	6.684D−06
TORSION4	14884	226	325	227	38.99	−1.212D+00	9.398D−06
TORSION5	14884	73	105	74	12.68	−2.859D+00	8.751D−06
TORSION6	14884	63	75	64	10.39	−2.859D+00	9.321D−06
TORSIONA	14884	496	756	497	100.13	−4.184D−01	6.442D−06
TORSIONB	14884	584	866	585	116.70	−4.184D−01	7.917D−06
TORSIONC	14884	247	350	248	48.81	−1.204D+00	9.683D−06
TORSIOND	14884	226	317	227	44.62	−1.204D+00	9.467D−06
TORSIONE	14884	65	89	66	12.90	−2.851D+00	9.459D−06
TORSIONF	14884	68	84	69	13.07	−2.851D+00	9.302D−06
ODNAMUR	11130	50001	80356	50002	3927.97	9.262D+03	4.213D−01

iteration, which involves considerable computer effort, LANCELOT usually stops at points where this norm is considerably smaller than the tolerance 10^{-5}. On the other hand, SPG methods, which test the projected gradient more frequently, stop when $\|g_1(x)\|_\infty$ is slightly smaller than that tolerance. In a small number of cases this affects the quality of the solution, reflected in the objective function value.

(2) S368 ($n = 100$): LANCELOT, SPG1, and SPG2 arrived at different solutions, the best of which was the one obtained by SPG2. SPG1 was the winner in terms of computer time.

(3) HADAMALS ($n = 1,024$): LANCELOT obtained $f(x) = 74.44$ in 157.6 seconds. SPG1 and SPG2 obtained stationary points with $f(x) = 31,070$ in less than 2 seconds.

(4) NONSCOMP ($n = 10,000$): As in BDEXP, the SPG methods found a solution slightly worse than the one found by LANCELOT but used less computer time.

(5) QR3DLS ($n = 610$): LANCELOT found a better solution ($f(x) \approx 4 \times 10^{-8}$ against $f(x) \approx 2.3 \times 10^{-4}$) and used less computer time than the SPG methods.

(6) SCON1LS ($n = 1,002$): LANCELOT found the solution whereas the SPG methods did not converge after 50,000 iterations.

(7) DECONVB ($n = 61$): LANCELOT found the (slightly) best solution and used less computer time than the SPG methods.

(8) BIGGSB1 ($n = 1,000$): LANCELOT found $f(x) = 0.015$ in 6.17 seconds, whereas the SPG methods got $f(x) \approx 0.016$ in ≈ 24 seconds.

(9) BQPGAUSS ($n = 2,003$): LANCELOT beat SPG methods in this problem, in terms of both computer time and quality of solution.

(10) ODNAMUR ($n = 11,130$): LANCELOT obtained a better solution than the SPG methods for this problem and used less computer time.

Four of the problems considered above (QR3DLS, SCON1LS, BQPGAUSS, and ODNAMUR) can be considered failures of both SPG methods, since convergence to a stationary point was not attained after 50,000 iterations. In the four cases, the final point seems to be in the local attraction basin of a local minimizer, but local convergence is very slow. In fact, in the first three problems, the final projected gradient norm is $\approx 10^{-2}$, and in ODNAMUR the difference between $f(x)$ and its optimal value is ≈ 0.1 %. Slow convergence of SPG methods when the Hessian at the local minimizer is very ill conditioned is expected, and preconditioning schemes tend to alleviate this inconvenient. See [21].

In the remaining 40 problems, LANCELOT, SPG1, and SPG2 found the same solutions. In terms of computer time, SPG1 was faster than LANCELOT in 29 problems (72.5%) and SPG2 outperformed LANCELOT also in 29 problems. There are no meaningful differences between the performances of SPG1 and SPG2.

Excluding problems where the difference in CPU time was less than 10%, SPG1 beat LANCELOT 28-9 and SPG2 beat LANCELOT 28-11.

Excluding, from the 40 problems above, the ones in which the 3 algorithms converged in less than 1 second, we are left with 31 problems. Considering this set, SPG1 beat LANCELOT 20-11 (or 19-9 if we exclude, again, differences smaller than 10%) and SPG2 beat LANCELOT 20-11 (or 19-11).

As we mentioned above, we also implemented the projected gradient algorithm PGA, using the same framework as SPG in terms of interpolation schemes, both with monotone and nonmonotone strategies. The performance of both alternatives is very poor, in comparison to the algorithms SPG1 and SPG2 and other box-constraint minimizers. The performance of the nonmonotone version is given in Table 2. This confirms that the spectral choice of the steplength is the essential feature that puts efficiency in the projected gradient methodology.

4. Final remarks. It is customary to interpret the first trial step of a minimization algorithm as the minimizer of a quadratic model $q(x)$ on the feasible region or an approximation to it. It is always imposed that the first-order information at the current point should coincide with the first order information of the quadratic model. So, the quadratic approximation at x_{k+1} should be

$$q(x) = \frac{1}{2}\langle x - x_{k+1}, B_{k+1}(x - x_{k+1})\rangle + \langle g(x_{k+1}), x - x_{k+1}\rangle + f(x_{k+1})$$

and

$$\nabla q(x) = B_{k+1}(x - x_{k+1}) + g(x_{k+1}).$$

Secant methods are motivated by the interpolation condition $\nabla f(x_k) = \nabla q(x_k)$. Let us impose here the weaker condition

$$D_{s_k} q(x_k) = D_{s_k} f(x_k), \tag{11}$$

where $D_d\varphi(x)$ denotes the directional derivative of φ along the direction d (so $D_d\varphi(x) = \langle \nabla\varphi(x), d\rangle$). A short calculation shows that condition (11) is equivalent to

$$\langle s_k, B_{k+1}s_k\rangle = \langle s_k, y_k\rangle. \tag{12}$$

Clearly, the spectral choice

$$B_{k+1} = \frac{\langle s_k, y_k\rangle}{\langle s_k, s_k\rangle} I \tag{13}$$

(where I is the identity matrix) satisfies (12). Now, suppose that z is orthogonal to s_k and that x belongs to $\mathcal{L}_k$, the line determined by x_k and x_{k+1}. Computing the directional derivative of q along z at any point $x \in \mathcal{L}_k$, and using (13), we obtain

$$D_z q(x) = \langle B_{k+1}(x - x_{k+1}) + g(x_{k+1}), z\rangle = \langle g(x_{k+1}), z\rangle = D_z f(x_{k+1}).$$

Moreover, the properties (12) and

$$D_z q(x) = D_z f(x_{k+1}) \quad \text{for all} \quad x \in \mathcal{L}_k \quad \text{and} \quad z \perp s_k \tag{14}$$

imply that s_k is an eigenvector of B_{k+1} with eigenvalue $\langle s_k, y_k\rangle / \langle s_k, s_k\rangle$. Clearly, (13) is the most simple choice that satisfies this property. Another remarkable property of (13) is that the resulting algorithms turn out to be invariant under change of scale of both f and the independent variables.

In contrast to the property (14), satisfied by the spectral choice of B_{k+1}, models generated by the secant choice have the property that the directional derivatives of the model coincide with the directional derivatives of the objective function *at* x_k. Property (14) says that the model was chosen in such a way that the first order information with respect to orthogonal directions to s_k is the same as the first order information of the true objective function at x_{k+1} *for all* the points on the line $\mathcal{L}_k$. This means that first order information at the current point is privileged in the construction of the quadratic model, in relation to second order information that comes from the previous iteration. Perhaps this is one of the reasons underlying the unexpected efficiency of spectral gradient algorithms in relation to some rather arbitrary secant methods. Needless to say, the special form of B_{k+1} trivializes the problem of minimizing the model on the feasible set when this is simple enough, a fact that is fully exploited in SPG1 and SPG2.

Boxes are not the only type of sets on which it is trivial to project. The norm-constrained regularization problem [18, 23, 24, 32], defined by

$$\text{minimize } f(x) \quad \text{subject to} \quad x^T A x \leq r, \tag{15}$$

where A is symmetric positive definite, can be reduced to ball constrained minimization by a change of variables and, in this case, projections can be trivially computed. A particular case of (15) is the classical trust-region subproblem, where f is quadratic. Recently (see [20, 25]) procedures for escaping from nonglobal stationary points of this problem have been found, and so it becomes increasingly important to obtain fast algorithms for finding critical points, especially in the large-scale case. (See [28, 29, 31].)

Perhaps the most important characteristic of SPG algorithms is that they are extremely simple to code, to the point that anyone can write her or his own code using any scientific language in a couple of hours. (Fortran, C, and Matlab codes written by the authors are available by request.) Moreover, their extremely low memory requirements make them very attractive for large-scale problems. It is quite surprising that such a simple tool can be competitive with rather elaborate algorithms that use extensively tested subroutines and numerical procedures. The authors would like to encourage readers to write their own codes and to verify for themselves the nice properties of these algorithms in practical situations. Papers [6] and [4] illustrate the use of SPG methods in applications.

Acknowledgment. The authors are indebted to two anonymous referees whose comments helped considerably to improve this paper.

REFERENCES

[1] J. Barzilai and J. M. Borwein, *Two point step size gradient methods*, IMA J. Numer. Anal., 8 (1988), pp. 141–148.

[2] D. P. Bertsekas, *On the Goldstein-Levitin-Polyak gradient projection method*, IEEE Trans. Automat. Control, 21 (1976), pp. 174–184.

[3] D. P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[4] E. G. Birgin, R. Biloti, M. Tygel, and L. T. Santos, *Restricted optimization: A clue to a fast and accurate implementation of the common reflection surface stack method*, J. Appl. Geophys., 42 (1999), pp. 143–155.

[5] E. G. Birgin, I. Chambouleyron, and J. M. Martínez, *Estimation of the optical constants and the thickness of thin films using unconstrained optimization*, J. Comput. Phys., 151 (1999), pp. 862–880.

[6] E. G. Birgin and Y. G. Evtushenko, *Automatic differentiation and spectral projected gradient methods for optimal control problems*, Optim. Methods Softw., 10 (1998), pp. 125–146.

[7] P. H. Calamai and J. J. Moré, *Projected gradient methods for linearly constrained problems*, Math. Programming, 39 (1987), pp. 93–116.

[8] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 25 (1988), pp. 433–460; see also SIAM J. Numer. Anal., 26 (1989), pp. 764–767.

[9] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *LANCELOT: A Fortran Package for Large-Scale Nonlinear Optimization (Release A)*, Springer Ser. Comput. Math. 17, Springer-Verlag, New York, Berlin, Heidelberg, 1992.

[10] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds*, SIAM J. Numer. Anal., 28 (1991), pp. 545–572.

[11] J. C. Dunn, *Global and asymptotic convergence rate estimates for a class of projected gradient processes*, SIAM J. Control Optim., 19 (1981), pp. 368–400.

[12] J. C. Dunn, *Gradient-related constrained minimization algorithms in function spaces: Convergence properties and computational implications*, in Large Scale Optimization: State

of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer, Dordrecht, the Netherlands, 1994.

[13] A. Friedlander, J. M. Martínez, and S. A. Santos, *A new trust region algorithm for bound constrained minimization*, Appl. Math. Optim., 30 (1994), pp. 235–266.

[14] E. M. Gafni and D. P. Bertsekas, *Convergence of a Gradient Projection Method*, Report LIDS-P-1201, Lab. for Info. and Dec. Systems, MIT, Cambridge, MA, 1982.

[15] W. Glunt, T. L. Hayden, and M. Raydan, *Molecular conformations from distance matrices*, J. Comput. Chem., 14 (1993), pp. 114–120.

[16] A. A. Goldstein, *Convex programming in Hilbert space*, Bull. Amer. Math. Soc., 70 (1964), pp. 709–710.

[17] L. Grippo, F. Lampariello, and S. Lucidi, *A nonmonotone line search technique for Newton's method*, SIAM J. Numer. Anal., 23 (1986), pp. 707–716.

[18] M. Heinkenschloss, *Mesh independence for nonlinear least squares problems with norm constraints*, SIAM J. Optim., 3 (1993), pp. 81–117.

[19] E. S. Levitin and B. T. Polyak, *Constrained Minimization Problems*, USSR Comput. Math. Math. Phys., 6 (1966), pp. 1–50.

[20] S. Lucidi, L. Palagi, and M. Roma, *On some properties of quadratic programs with a convex quadratic constraint*, SIAM J. Optim., 8 (1998), pp. 105–122.

[21] F. Luengo, M. Raydan, W. Glunt, and T. L. Hayden, *Preconditioned Spectral Gradient Method for Unconstrained Optimization Problems*, Technical Report R.T. 96-08, Computer Science Department, Universidad Central de Venezuela, Caracos Venezuela.

[22] G. P. McCormick and R. A. Tapia, *The gradient projection method under mild differentiability conditions*, SIAM J. Control, 10 (1972), pp. 93–98.

[23] J. M. Martínez and S. A. Santos, *A trust region strategy for minimization on arbitrary domains*, Math. Programming, 68 (1995), pp. 267–302.

[24] J. M. Martínez and S. A. Santos, *Convergence results on an algorithm for norm constrained regularization and related problems*, RAIRO Rech. Opér., 31 (1997), pp. 269–294.

[25] P. D. Tao and L. T. H. An, *D.C. optimization algorithm for solving the trust-region subproblem*, SIAM J. Optim., 8 (1998), pp. 476–505.

[26] M. Raydan, *On the Barzilai and Borwein choice of steplength for the gradient method*, IMA J. Numer. Anal., 13 (1993), pp. 321–326.

[27] M. Raydan, *The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem*, SIAM J. Optim., 7 (1997), pp. 26–33.

[28] F. Rendl and H. Wolkowicz, *A semidefinite framework to trust region subproblems with application to large scale minimization*, Math. Programming, 77 (1997), pp. 273–299.

[29] M. Rojas, S. A. Santos and D. C. Sorensen, *A new matrix-free algorithm for the large-scale trust-region subproblem*, SIAM J. Optim, to appear.

[30] A. Schwartz and E. Polak, *Family of projected descent methods for optimization problems with simple bounds*, J. Optim. Theory Appl., 92 (1997), pp. 1–31.

[31] D. C. Sorensen, *Minimization of a large-scale quadratic function subject to spherical constraint*, SIAM J. Optim., 7 (1997), pp. 141–161.

[32] C. R. Vogel, *A constrained least squares regularization method for nonlinear ill-posed problems*, SIAM J. Control Optim., 28 (1990), pp. 34–49.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1212–1227

COMPUTATIONAL EXPERIENCE WITH AN INTERIOR POINT CUTTING PLANE ALGORITHM*

JOHN E. MITCHELL†

Abstract. There has been a great deal of success in the last 20 years with the use of cutting plane algorithms to solve specialized integer programming problems. Generally, these algorithms work by solving a sequence of linear programming relaxations of the integer programming problem, and they use the simplex algorithm to solve the relaxations. In this paper, we describe experiments using a predictor-corrector interior point method to solve the relaxations. For some problems, the interior point code requires considerably less time than a simplex based cutting plane algorithm.

Key words. interior point methods, integer programming, cutting planes, linear ordering, Ising spin glasses, max cut

AMS subject classifications. 90C10, 90C05, 90C51, 90C57

PII. S1052623497324242

1. Introduction.

Any integer linear programming problem can be expressed as $\min\{c^T x : x \in S,\, x_i = 0, 1\, \forall i\}$, where S is a polyhedron. Often, a good solution can be found by heuristic methods such as local search, tabu search, simulated annealing, genetic algorithms, or algorithms specific to the particular problem; this heuristic solution may well be optimal. It is usually harder to prove optimality. Algorithms such as branch and bound, cutting plane approaches, and branch and cut can be used to obtain lower bounds on the optimal value, and if the algorithms are allowed to run for long enough, they will reduce the gap between the upper and lower bounds to zero and thus find the optimal solution. Cutting plane algorithms form a linear programming relaxation of the integer programming problem, solve the relaxation to obtain a lower bound on the optimal value of the integer program, and, if the upper and lower bounds do not agree, improve the relaxation and repeat the process. Cutting plane methods can be incorporated into a branch and bound method to give a branch and cut algorithm.

Cutting plane and branch and cut algorithms have been successfully used to solve many types of integer linear programming problems, including the traveling salesman problem [1, 20, 39], the linear ordering problem [21], clustering problems [24], and the maximum cut problem [2]. See Jünger, Reinelt, and Thienel [26] for a survey. The simplex algorithm was used to solve the linear programming relaxations in all of these references.

Interior point algorithms are now a very good alternative to the simplex method for linear programming problems, and they are superior for large problems where the structure of the nonzeroes in the constraint matrix is not too unfavorable. (See, for example, [30].) It is natural to investigate the use of interior point methods in a cutting plane algorithm. The successful use of an interior point method in this setting requires the ability to exploit a *warm start*: the solution to one relaxation should be close to the solution to the next relaxation in some sense, so it should require relatively few

*Received by the editors July 9, 1997; accepted for publication (in revised form) January 12, 2000; published electronically July 11, 2000.

http://www.siam.org/journals/siopt/10-4/32424.html

†Department of Mathematical Sciences, Rensselaer Polytechnic Institute, Troy, NY 12180 (mitchj@rpi.edu). This author's research was supported in part by ONR grant N00014-94-0391 and NSF grant CCR–9901822.

iterations to solve the next relaxation from this warm start as opposed to starting from a cold start that does not exploit this information. The simplex method appears to be fairly adept at exploiting the warm starts provided in a cutting plane algorithm, but equally efficient ways to restart when using an interior point method are not known. A general methodology is proposed by Gondzio [19] with encouraging computational results; as with the results we present in this paper, the principal emphasis in a restart method has to be to restart with an iterate that is centered. The principal technique we use is *early termination*: the relaxations are solved approximately, which results in an initial iterate for the next relaxation that is somewhat centered, leading to better performance.

Mitchell and Todd [38] presented a promising first attempt at using an interior point cutting plane algorithm, solving matching problems. An interior point cutting plane algorithm for the linear ordering problem was described in Mitchell and Borchers [35]. The computational times in that paper were comparable to those obtained by Grötschel, Jünger, and Reinelt [21] and Reinelt [43] with a cutting plane algorithm which used the simplex solver CPLEX3.0 [10] to solve the linear programs. Interior point approaches to integer programming problems are surveyed in [37]; this reference includes discussions of the theoretical performance of interior point cutting plane algorithms and of other applications of interior point column generation methods.

In the current paper, we present results on two classical integer programming problems, namely, the MAXCUT problem and the linear ordering problem. The MAXCUT instances arise from finding the ground state of Ising spin glasses, a problem in statistical physics. Our results appear to be considerably better under one distribution of the data than recent results in the literature [13] obtained using the simplex solver in CPLEX3.0. These are the hardest problems considered in this paper, requiring a more conservative choice of parameters than the other problems in order to obtain a robust implementation. We improve somewhat on the results in [35] for real-world linear ordering problems and also look at some larger randomly generated problems, obtaining better runtimes on some of these problems with our interior point method than with a cutting plane algorithm using the simplex solver in CPLEX4.0. Because of extensive experimentation, we are able to be more confident and therefore more specific about our choices of parameters than in [35]. Our algorithm is presented in section 2. The results for linear ordering problems and Ising spin glass problems are contained in sections 3.1 and 3.2, respectively.

Many different integer programming problems can be formulated using the framework (IP) that we present in section 2; the great majority of research on polyhedral theory and cutting plane algorithms is on problems that can be written in this form (see, for example, [7, 26, 34]). Of course, not all of these problems are equally amenable to the interior point cutting plane approach that we present in this paper. We return to the issue of determining appropriate problems for the interior point approach in our conclusions, section 4. One requirement for this investigation is that the linear programming relaxations should be large and yet the integer programming problems are solvable, so we examined problems where the time required to solve the linear programming relaxations is a substantial portion of the total solution time. For this paper we restrict our attention to problems that can be solved at the root node of a branch and cut tree, for several reasons, including the following two: interior point branch and bound is not well understood (see, for example, [28]), and the time to solve large problems that require branching is impracticable for this investigation.

2. An interior point cutting plane algorithm.

We assume we have an integer programming problem of the form

$$
\text{(IP)}\qquad
\begin{array}{lrcrcl}
\min & & & c^Tx & & \\
\text{subject to} & & & Ax & = & b, \\
 & 0 & \leq & x & \leq & u, \\
 & & & x_i & = & 0 \text{ or } 1 \text{ for } i \in I, \\
 & & & x & & \text{satisfies some additional conditions,}
\end{array}
$$

where x, c, and u are n-vectors, b is an m-vector, A is an $m \times n$ matrix of rank m, and I is the set of integer variables. We assume $u_i = 1$ for $i \in I$. We assume the additional conditions can be modeled as a (possibly exponential) set of linear constraints. Many problems can be cast in this framework; for example, the traveling salesman problem can be represented in this form, with the additional conditions being the subtour elimination constraints [20, 39] and the conditions $Ax = b$ representing the degree constraints that the tour must enter and leave each vertex exactly once. Some problems do not need additional conditions, and we regard such problems as also falling in our general framework. We let Q denote the convex hull of feasible solutions to (IP). We assume that the dimension of Q is $n - m$. The linear programming relaxation (or LP relaxation) of (IP) is

$$
\text{(LP)}\qquad
\begin{array}{lrcrcl}
\min & & & c^Tx, & & \\
\text{subject to} & & & Ax & = & b, \\
 & 0 & \leq & x & \leq & u
\end{array}
$$

with dual

$$
\text{(LD)}\qquad
\begin{array}{lrcrcrcl}
\max & b^Ty & - & u^Tw & & & & \\
\text{subject to} & A^Ty & - & w & + & z & = & c, \\
 & & & & & w, z & \geq & 0,
\end{array}
$$

where y is an m vector and w and z are n-vectors. The value of any feasible solution to (LD) provides a lower bound on the optimal value of (IP). We solve (LP) and (LD) using a predictor-corrector primal-dual interior point method similar to those described in Lustig, Marsten, and Shanno [30] and Mehrotra [31]. This algorithm keeps x, w, z, and the primal slacks $s := u - x$ strictly positive. We call such a point an *interior point*. The method is a barrier method, finding a sequence of approximate analytic centers in order to approach the optimal solution, where an analytic center is a solution to $\min\{c^Tx - \mu \sum_i \ln(x_i(u_i - x_i)) : Ax = b\}$ for some positive scalar μ. All iterates generated by the algorithm will satisfy $Ax = b$, as described later.

If the optimal solution to (LP) is feasible in (IP), then we can stop with optimality. If the optimal basic feasible solution x^{LP} to (LP) is not in Q, then we cut off x^{LP} by adding an extra constraint or *cutting plane* of the form ${a^0}^T x \leq b_0$. If the integer programming problem is NP-hard, then it is also NP-hard to find a violated cutting plane [23], so heuristics are usually used to generate cuts. This gives the relaxation

$$
\text{(LPnew)}\qquad
\begin{array}{lrcrcl}
\min & c^Tx & & & & \\
\text{subject to} & Ax & & & = & b, \\
 & {a^0}^Tx & + & x_0 & = & b_0, \\
 & 0 & \leq & x & \leq & u, \\
 & 0 & \leq & x_0 & \leq & u_0,
\end{array}
$$

where x_0 is a new fractional variable giving the slack in the added constraint. The cutting plane is a valid inequality for (IP), but it is violated by the optimal solution x^{LP}. We then solve (LPnew), and repeat the process. In this paper, the cutting planes we add are generally facets of Q, and we use specialized routines to find the cutting planes. The dual problem to (LPnew) is

$$\text{(LDnew)}\qquad \begin{array}{llcccccccc} \max & b^Ty & & & - & u^Tw & - & u_0w_0 & & \\ \text{subject to} & A^Ty & + & a_0y_0 & - & w & + & z & = & c, \\ & & & y_0 & - & w_0 & + & z_0 & = & 0, \\ & & & & & w, z & \geq & 0, & & \\ & & & & & w_0, z_0 & \geq & 0. & & \end{array}$$

Every iterate $\hat{x}, \hat{y}, \hat{w}, \hat{z}$ generated by an interior point method before reaching optimality will satisfy $0 < \hat{x} < u$ and $\hat{w} > 0$, $\hat{z} > 0$. These can be used to obtain a new feasible solution to (LDnew) by taking $y = \hat{y}$, $w = \hat{w}$, $z = \hat{z}$, $y_0 = 0$, and $w_0 = z_0$. If we pick $w_0 = z_0$ to be strictly positive, then all the nonnegativity constraints will be satisfied strictly. It is not so simple to obtain a feasible solution to (LPnew) because we have ${a^0}^T\hat{x} > b_0$ if the new constraint was a cutting plane.

It has been observed that if an interior point method is started from close to the boundary, it will move towards the center of the feasible region before starting to move towards the optimal solution. Thus, the optimal solution to (LP) is not a very good starting point for trying to solve (LPnew), so we search for cutting planes violated by $\hat{x}$ before reaching optimality. Such cutting planes may well be *deeper cuts* and cut off more of the part of the feasible region that is close to the optimal solution to (LP), because the iterate is further than the optimal solution from the boundary of the polyhedron.

The two principal disadvantages of looking for cuts before solving the current relaxation to optimality are, first, that we may be unable to find any cuts, so the search is a waste of time, and, second, that the search may return cuts which are violated by the current iterate, but which are not violated by the optimal solution, so we may end up solving additional relaxations. The second disadvantage can be minimized by moving towards the optimal solution from the center of the polyhedron, reducing the likelihood of violating cutting planes that are satisfied by the optimal solution to (LP). To reduce the impact of the first disadvantage, we use a *dynamically altered tolerance* τ for deciding when to search for violated cutting planes, searching only when the duality gap drops below this tolerance. This tolerance is increased if we find a large number of violated constraints, and decreased if we find only a few violated constraints.

As mentioned earlier, we can obtain a new feasible interior iterate for (LDnew) by setting $y_0 = 0$ and $w_0 = z_0 = \epsilon_D$ for some appropriate small positive value of ϵ_D. We chose $\epsilon_D = 10^{-3}$, which is considerably larger than the 10^{-6} used in [35]. To improve stability and performance, it is useful to also increase any small components of w and z up to ϵ_D.

We update the primal iterate using a point that is known to be feasible and interior in (LPnew). Any interior point which is a convex combination of feasible integral points will satisfy all cutting planes, so it will be feasible in (LPnew). In addition, it will be interior in (LPnew) provided it satisfies all the cutting planes strictly. Any point in the relative interior of Q will be feasible and interior in (LPnew). We used the vector of all halves as an initial point of this type for both problem classes considered in this paper. This point is updated as the algorithm progresses, by combining it with

1. **Initialize.** Set up the initial relaxation. Find initial interior primal and dual points. Find a feasible point in Q. Find a restart point x^{FEAS} in the relative interior of Q for use in Step 10.

2. **Inner iteration.** Perform one iteration of the primal dual algorithm.

3. **Check for early termination.** If the relative duality gap is larger than the tolerance τ, return to Step 2.

4. **Primal heuristics.** Use the primal heuristics to try to improve on the current best solution to (IP).

5. **Check for optimality.** The current dual solution provides a lower bound and the value of the best known feasible point provides an upper bound. If the difference between these two is sufficiently small, **Stop** with optimality.

6. **Look for cutting planes.** If possible, also update the known feasible point x^{FEAS}.

7. **Add cutting planes.** If any cutting planes were found in Step 6 then add an appropriate subset; otherwise, reduce τ and return to Step 2.

8. **Drop cutting planes.** If any cutting plane appears to no longer be important, drop it.

9. **Fix variables.** If possible, fix variables at zero or one.

10. **Modify current iterate.** Increase any small components of w and z to a small value ϵ_D. If necessary, increase appropriate components of w and/or z to regain dual feasibility. Update the primal solution to a convex combination of the current iterate and x^{FEAS}, giving a point which is interior in the new relaxation. Increase any small components of x and the vector of primal slacks to ϵ_P. Modify the tolerance τ. Return to Step 2.

Fig. 1. *An interior point cutting plane algorithm.*

any iterate which is in the convex hull. We can restart either at this feasible point or at an appropriate convex combination of this point and the previous iterate. To improve stability and performance, it is useful to also increase any small components of x and s to $\epsilon_P := 10^{-5}$.

In practice, many constraints are added at once. The same procedures for finding initial solutions to the new primal and dual relaxations can still be used.

Cutting plane algorithms are useful for proving optimality by generating lower bounds on the optimal value of (IP). Fractional primal points x can also be used to generate new feasible solutions to (IP) by using problem-specific rounding heuristics. If the interior point method is converging to a point in the interior of the optimal face of Q, then the primal heuristics may well provide one of the optimal solutions to (IP), so we can terminate the algorithm, because the value of the relaxation will agree with the value of the integer solution. Without good primal heuristics, the algorithm may search in vain for cutting planes, and be forced to branch, resulting in longer run times.

It is useful to drop constraints that no longer appear important. This has the advantage of shrinking the size of the relaxation, with the principal benefit of reducing the time required for each iteration, and the marginal benefit of very slightly reducing the number of iterations to solve a relaxation. Generally, we do not discard a constraint for several stages, and we drop the constraint if its slack variable is large—see section 3 for more details. Note that if the slack variable is large, then the corresponding dual variable y will be close to zero. More sophisticated tests are available, but the costs of these outweigh the benefits of the reduction in the size of the relaxations.

Simplex branch and cut methods can use reduced costs to fix variables at zero or one. The reduced costs are not available at the current interior solution to the relaxation (LP), but the dual variables are available, and these can be used to fix variables, as described in [32]. Fixing variables has the practical disadvantage of making the old restart point for (LPnew) no longer feasible, because this restart point is interior. Fixing some variables may impose logical constraints on other variables, so the restart point usually has to be modified and these additional logical constraints sometimes have to be added to the model. We did not find it necessary to fix variables for problems with integral objective function coefficients.

We summarize the complete algorithm in Figure 1. Note that more details can be found in section 3 for the two problem classes considered in this paper. We say that we have completed a *stage* every time we enter step 10. We complete the final stage when we enter step 5 for the last time. The set of appropriate constraints in step 7 is usually obtained using a bucket sort. The results in this paper represent an improvement over those obtained with a similar algorithm for linear ordering problems in [35], with a reduction in the number of iterations as well as the runtime. The principal differences are the use of larger restart parameters ϵ_P and ϵ_D in step 10, keeping constraints for more stages before allowing them to be dropped in step 8, slightly changing the method for updating τ in step 7, and using different parameters to choose the appropriate subset, including adding a larger number of constraints.

3. Computational results.

We have used this algorithm to solve several different problems in combinatorial optimization. In this section, we describe the modifications made to the basic algorithm and give computational results for each problem. The computer code was written in FORTRAN 77. We have a framework where the majority of the code remains the same for each problem, and we use problem specific subroutines for initializing the problem, finding primal integral solutions using heuristics, finding cutting planes, and modifying the relaxation by adding and dropping constraints. All the computational testing was performed on a Sun SPARC 20/71 UNIX workstation. All runtimes are reported in seconds.

We use the Yale Sparse Matrix Package [16] to calculate the projections, using the routine `mmd` due to Liu [29] to find an ordering of the columns of ADA^T for the Cholesky factorization of this matrix, where D is an appropriate diagonal matrix. Our interior point linear programming solver could be improved. It is probably about two to three times slower than commercial solvers such as CPLEX [10]. In particular, some of the linear algebra routines could be improved. We do not use a publicly available code such as HOPDM [18] or PCx [11], because none of these codes makes it easy to access the current solution after each iteration, stop the process when desired, suggest a new starting point, and not preprocess each relaxation, which are all required features of our algorithm.

3.1. The linear ordering problem.

3.1.1. Definition of the problem.

The linear ordering problem is a combinatorial optimization problem with a wide variety of applications, such as triangulation of input-output matrices, archeological seriation, minimizing total weighted completion time in one-machine scheduling, and aggregation of individual preferences. It is NP-hard (Karp [27]), and a complete description of the facets of its convex hull is not known. The polyhedral structure of the linear ordering problem has been investigated by Grötschel, Jünger, and Reinelt [21], Jünger [25], and Reinelt [42].

The problem requires placing p sectors (or objects) in order, where there is a cost g_{ij} for placing sector i before sector j. It was shown by Grötschel, Jünger, and Reinelt [21] that the linear ordering problem with p sectors is equivalent to the following integer programming problem:

$$\begin{array}{llrl} & & \min & \sum_{1\le i<j\le p} c_{ij}x_{ij} \\ (3.1) & (\mathrm{LO}) & \text{subject to} & x_{ij}+x_{jk}-x_{ik} \le 1 \text{ for } 1\le i<j<k\le p \\ (3.2) & & & -x_{ij}-x_{jk}+x_{ik} \le 0 \text{ for } 1\le i<j<k\le p, \\ & & & x_{ij} = 0 \text{ or } 1 \text{ for } 1\le i<j\le p, \end{array}$$

where $c_{ij} = g_{ij} - g_{ji}$ for $1 \le i < j \le p$. Here, we obtain $x_{ij} = 1$ if i is before j in the ordering, and $x_{ij} = 0$ otherwise. Equations (3.1) and (3.2) are called *triangle inequalities*; they prevent solutions x which correspond to, for example, sector i before sector j, sector j before sector k, and sector k before sector i.

3.1.2. Details of the algorithm. The initial linear programming relaxation of (LO) is $\min\{c^Tx : 0 \le x \le e\}$, where c and x are $p(p-1)/2$ vectors and e is the $p(p-1)/2$ vector of ones. (Throughout, we use e to denote the vector of ones of an appropriate dimension.)

The only cutting planes we add are triangle inequalities of the form given in (3.1) and (3.2)—these were sufficient to solve most of the problems in our test set. We first called the separation routines when the relative duality gap (the duality gap divided by the larger of the absolute value of the dual value and 1) was below $\tau = 0.3$. When cutting planes were found, this tolerance τ was multiplied by 1.4^k, where $k = \lfloor 10(MAXVIOL + 0.1)\rfloor - 9$ and $MAXVIOL$ is the maximum cutting plane violation.

The separation routine comprised complete enumeration of all the triangle inequalities. These were bucket sorted by violation. We add only constraints that have violation at least $0.5MAXVIOL$. The algorithm proceeds through the inequalities in order of decreasing violation until an edge-disjoint set of at most 500 constraints has been found, which is then added to the relaxation. (We say several constraints are *edge-disjoint* if they use distinct sets of variables.) Adding an edge-disjoint subset has the beneficial effect of reducing the amount of fill-in in the matrix product AA^T, and thus reducing the linear algebra required to calculate projections when finding the next interior point iterate. Note that if we chose to translate the cutting planes so that they are satisfied at equality, then it is easy to find a restart direction if the cuts are orthogonal [41, 17], as they are if they are edge-disjoint.

Our primal heuristics are similar to those suggested in Grötschel, Jünger, and Reinelt [21]. We round the current iterate. An ordering is constructed from this rounded solution using a greedy heuristic: at step k it picks the kth element in the ordering, breaking ties arbitrarily. A local optimization routine is then applied to this greedy ordering, where each sector is examined in a different position in the ordering.

We dropped any constraint which had been in the relaxation for at least five stages and which still had a slack of at least 0.4.

We initialized the restart point to be $x_{ij}^{FEAS} = 0.5$. This was updated at each iteration to x if x did not violate any of the cutting planes. If x violated any triangle constraint, then we updated x^{FEAS} by taking a step of length α from x^{FEAS} in the direction towards x, where α is 90% of the distance to the closest triangle inequality.

Table 1
Results on real-world input-output matrices.

Sectors	44	50	56	60	79
Number	29	3	11	2	1
Iterations	53	64	62	67	104
Time (seconds)	9.1	21.1	32.1	52.9	487.4
Stages	14	17	17	18	24
Cuts added	322	539	732	891	1985
Cuts dropped	77	140	225	269	646

Christof and Reinelt [9] have developed a simplex based branch and cut algorithm for hard instances of the linear ordering problem where the cutting planes come from small-dimensional versions of the problem, as in Christof and Reinelt [8]. The instances we examine in this paper are larger, but they do not generally require branching or extensive separation routines to find violated cutting planes. We are interested in large instances because they have large linear programming relaxations, so the amount of time spent solving the relaxations will be a significant proportion of the total solution time. We expect that the methods described in this paper, in conjunction with the methods described in [9], will make it possible to solve large, hard instances.

3.1.3. Real-world problems. Table 1 contains the results of our algorithm on 46 real-world linear ordering problems. All the problems come from input-output tables in economics; except for the 79-sector problem *usa*79, they are all available from LOLIB at the URL http://www.iwr.uni-heidelberg.de/iwr/comopt/software/LOLIB. For a discussion of the origins of these problems, see Grötschel, Jünger, and Reinelt [21] or Mitchell and Borchers [35]; for a discussion of the economic interpretation of the results see Grötschel, Jünger, and Reinelt [22]. All the problems in Table 1 except for those with 50 sectors were attacked using the algorithm discussed in [35]. The costs in all of these input-output tables are integral, so we terminated when the gap between our upper and lower bounds was smaller than 1.

The rows of the tables convey the following information. The first row gives the number of sectors and the second row the number of instances of that particular size that were solved. The rows labeled *Iterations*, *Time (seconds)*, and *Stages* give the means of, respectively, the total number of primal-dual predictor-corrector iterations required to solve the integer programming problem, the total time in seconds required to solve the problems, and how often the LP relaxation was modified so the total number of LP relaxations formed for a particular problem is one more than the number of stages. The rows labeled *Cuts added* and *Cuts dropped* give, respectively, the total number of cutting planes added to the relaxations and the number of these cuts that were subsequently dropped. The numbers are rounded to the number of digits shown.

As can be seen, all these problems can be solved easily with our code. The algorithm requires only around 4 iterations per stage; as would be expected, the number of iterations required on a stage increases as the algorithm proceeds, so the last stage may well require about 10 iterations. Of course, this last stage is the only one that has to be solved exactly. The proportion of time spent actually solving the linear programming relaxations increases as the problem size increases, accounting for over 90% of the time on the largest problem *usa*79. The number of stages is larger than in some simplex based implementations because we add a set of edge-disjoint constraints at each stage, which keeps the Cholesky factor from becoming too dense.

The iteration counts and the number of stages are better than those contained in [35]. The Sun SPARC 20/71 used in the experiments in this paper is about twice as fast as the Sun SPARC 10/30 used for the experiments in [35]. After adjusting for this, the runtimes for the 44 and 56 sector problems in Table 1 are similar to those in the earlier paper, but the runtimes for the larger problems are two to three times better than those in [35]. It was argued in [35] that the runtimes in that paper were comparable to those obtained by the simplex based cutting plane algorithm due to Grötschel, Jünger, and Reinelt [21] and Reinelt [43]—they were somewhat worse, but the difference was shrinking as the problem size increased. Thus, the new results give a runtime that is very similar to that in [43] for the largest problem *usa*79.

Our results can also be compared with a simplex based cutting plane algorithm for these problems [6], which is written in C and uses the simplex solver in CPLEX4.0 to solve the relaxations. It adds all the violated constraints to the relaxation and resolves. We obtained a copy of this code and used it to solve the problems in our test set. Most of them required only two or three stages, and the runtimes are better than those obtained with the interior point code—the ratio decreases as the problem size increases, but the runtimes are still perhaps three times better for the problem *usa*79. This is still a good result for the interior point code, since it was all written "in-house" whereas CPLEX4.0 is an excellent commercial code. We are also comparing runtimes of codes written in different languages, so it is hard to draw definitive conclusions. For these problems, CPLEX4.0 uses Devex pricing in the dual and it introduces perturbations in the data; these choices aid the solution procedure considerably.

3.1.4. Random problems. We also solved some larger randomly generated problems, and in addition some of these problems were solved using the code described in [6]. We generated these problems by first setting pz% of the entries g_{ij} to 0 and generating a random permutation τ; the remaining entries were then uniformly distributed integers between 0 and 99 if $\tau(i) < \tau(j)$ or between 0 and 39 if $\tau(i) > \tau(j)$. The problems become harder as pz increases. Many of the real-world problems contain a number of zeroes in the g_{ij} entries. The generated problems all had linearity between 70.9 and 74.2—the linearity measures the proportion of the total weight accounted for by the ordering. The extreme cases are, first, that every entry in the matrix takes the same value, when the linearity would be 50, and second, when there are no nonzero entries below the diagonal, in which case the linearity is 100. The randomly generated problems had similar linearity to the real-world problems.

The results are contained in Table 2. We let pz take the values 0, 10, 20, and 30, and the number of sectors was set to 50, 75, 100, 150, and 200; five problems were generated with each combination. The table contains the mean results for each set of problems. Because of memory limitations, we were unable to solve problems with more than 100 sectors using the simplex code, and we were also unable to solve problems with 150 sectors and $pz \geq 20$ or with 200 sectors and $pz \geq 10$ using the interior point code. In addition, again because of memory limitations, we could not solve problems with 100 sectors and $pz = 30$ with either code, and we were able to solve only one problem with 100 sectors and $pz = 20$ using the simplex code—on the remaining problems, the code ran for roughly 1000 seconds before running out of memory. The triangle inequalities were not sufficient to solve four of the problems, one each with 75 sectors and pz equal to 10, 20, and 30 and one with 100 sectors and $pz = 20$; we have omitted these problems from the tables. It appears that the simplex code spends well over 90% of its time within CPLEX, at least for the harder problems. The columns in Table 2 contain the same information as the rows in Table 1, with

Table 2

Results for random linear ordering problems.

pz	Sectors	Interior point				Simplex
		Time	Iters	Stages	Added	time
0	50	6.0	26.0	7.0	236.0	3.3
0	75	20.2	30.2	8.2	543.2	13.8
0	100	51.1	33.6	9.2	1003.2	98.4
0	150	206.4	44.8	12.2	2919.0	—
0	200	754.8	46.2	12.4	6406.4	—
10	50	10.1	35.8	9.6	362.2	6.1
10	75	50.8	47.5	13.0	871.5	73.9
10	100	155.6	53.0	13.8	1510.0	280.9
10	150	2071.9	72.4	12.4	6406.4	—
20	50	19.7	50.8	12.6	500.2	10.4
20	75	240.7	90.5	17.3	1247.5	119.6
20	100	1405.4	89.5	18.5	2313.8	—
30	50	70.1	73.6	15.2	732.6	29.5
30	75	771.3	102.3	17.8	1588.0	251.9

the addition that the last column contains the runtimes with the cutting plane code that uses the simplex solver in CPLEX 4.0. Runtimes are quoted in seconds.

As can be seen, the interior point code outperforms the simplex based code for problems with at least 100 sectors where pz is no bigger than 10. Furthermore, it can be seen that the rate of increase in the runtimes as the problem size increases is far smaller for the interior point code than for the simplex code. When pz is as big as 30, the Cholesky factors become dense and the simplex code outperforms the interior point code. For $pz = 20$, the simplex code outperforms the interior point code for 50 and 75 sector problems, but it appears that the codes would take similar times for 100 sector problems, were it not for memory limitations.

As the proportion of zeroes pz increases, the linear ordering problems should become more dual degenerate, with multiple optimal solutions. For linear programming problems, degeneracy is normally favorable for an interior point method. However, for these problems, the degeneracy results in the addition of many cutting planes that use the same variables so the constraint matrix A eventually contains several dense columns and there is considerable fill in the Cholesky factor of the matrix AA^T. This increases the time for one iteration of the interior point method, and thus the simplex code outperforms the interior point code when $pz = 30$. One possible remedy for this problem is to use a preconditioned conjugate gradient algorithm to calculate the directions in the interior point method; this is a subject for future research.

We have recently investigated combining an interior point cutting plane method with a simplex cutting plane method [36], with results that appear to be superior to using either method on its own. The random problems used in both [36] and this paper are available at the URL http://www.math.rpi.edu/~mitchj/generators.

We also examined a formulation of a clustering problem proposed by Grötschel and Wakabayashi [24]. This problem can be written in a manner similar to the linear ordering problem, with triangle inequalities, although the triangle inequalities have a different structure. The computational results were similar to those for the linear ordering problem, in that they were comparable to the results obtained with a simplex method, and the relative performance of the interior point code improved as the problems increased in size. The algorithm appears to perform worse than one described by Palubeckis [40], at least for smaller problems. As the problem sizes increased, the gap between the algorithms decreased. The random instances of both this clustering problem and also the Ising spin glass problem used in this paper are also available from the URL given above.

3.2. The ground states of Ising spin glasses.

3.2.1. Definition of the problem. Finding the ground states of Ising spin glasses is an important problem in physics. We examine two-dimensional Ising spin glasses. This problem was originally discussed in the operations research literature by Barahona, Jünger, and Reinelt [3], who modeled the problem as a MAXCUT problem and developed a simplex based cutting plane algorithm to solve the problem. Recently, some of these authors and other colleagues have returned to this problem and have improved their computational results considerably [44, 12, 13]. We have previously sketched our experience on a smaller set of these problems in [33]. Facets of the cut polytope are described in [4, 14, 15].

We are given a collection of points, and we know the interaction between the points; we want to determine which points have a positive charge and which points have a negative charge. Our model places vertices at points of an $L \times L$ grid on a torus. Each vertex has four neighbors: to the left, to the right, above, and below. There are weights on the edges joining a vertex to its neighbors which correspond to the bonds or interactions between the vertices. We generate edge weights using two different distributions, and we report results for problems with grids of size up to 100×100. We assume there is no external field—it was shown by Barahon, Jünger, and Reinelt [3] that an external field can be modeled by including an extra vertex; the resulting problem appears to be easier to solve than a problem with no external field, at least when the edge weights have a Gaussian distribution.

The problem can be modeled on an undirected graph $G = (V, E)$ as

$$\begin{array}{ll} \min & \sum_{i=1}^{p} \sum_{j>i,(i,j)\in G} c_{ij} x_{ij} \\ \text{subject to} & x \text{ is the incidence vector of a cut,} \end{array}$$

where p is the number of vertices, there is a variable x_{ij} for each edge, and the cost c_{ij} of each edge is derived from the interaction between the vertices. Each vertex has four neighbors, so a $k \times k$ grid will have k^2 vertices and $2k^2$ edges.

Cutting planes can be derived by using the observation that every cycle and every cut intersect in an even number of edges. Every subset F of odd cardinality of every chordless cycle C gives the facet-defining inequality

$$x(F) - x(C \setminus F) \leq |F| - 1, \tag{3.3}$$

where $x(S)$ denotes $\sum_{(i,j)\in S} x_{ij}$ for any subset $S \subseteq E$. The cycles of length four (the *squares*) in the graph are chordless cycles, and there are many other chordless cycles. There are other families of facet-defining inequalities; we searched only for facets of the form (3.3).

3.2.2. Details of the algorithm. The initial relaxation is $\min\{c^T x : 0 \leq x \leq e\}$. All the cutting planes are of the form (3.3).

The separation routine consists of three parts. We first search for cutting planes corresponding to the squares in the graph using complete enumeration. The violated constraints are bucket sorted by violation and the most violated constraints are added. We are prepared to add constraints that correspond to squares that share edges. We add at most 500 square constraints; further, if $k < 500$ constraints are violated by at least 0.1, then we add at most $\max\{L-k,0\}$ constraints with violation less than 0.1. If this does not return at least L constraints which are violated by cutting planes or if the largest violation of a square constraint is no more than 0.2, we then use a heuristic procedure similar to that described in Barahona, Jünger, and Reinelt [3] to find longer chordless cycles with violated constraints. The heuristic is restricted to add at most 100 violated constraints; further, we restrict it so that it adds at most L^2 nonzeroes to the constraint matrix A (excluding the columns corresponding to the slack variables).

If the heuristic was called and it did not find at least 20 cutting planes, we use an implementation of the exact algorithm due to Barahona and Mahjoub [4], which has complexity $O(p^3)$ (p is the number of nodes), and is guaranteed to find a violated cycle inequality, if one exists. We place an upper limit of L on the number of these constraints that we will add, and we add a constraint only if it has a violation that is at least half of the violation of the most violated constraint found by this exact procedure on this stage. The routine looks for cycles starting from each vertex in the graph; to limit the time spent on this, we start from a maximum of 50 further vertices after finding a constraint with violation at least 0.05. We insist that the set of added constraints arising from longer cycles be edge-disjoint at each stage. The nonsquare constraints usually contain many more than 4 edges. We found it advantageous to scale an added constraint with $|C|$ edges, normalizing so that the L_1-norm of the constraint was $4/\sqrt{|C|}$.

We solved every tenth LP relaxation to a relative duality gap of 10^{-8}. Several of the problems took a large number of stages, and solving the relaxations accurately is a way to limit the number of stages, at a cost of an increased number of iterations. This approach reduces the variability of the runtimes.

Adding the longer cycles makes it hard to update the restart point: the restart point found in one stage may well be infeasible at a future stage because we do not check every possible constraint. Thus, we restarted in step 10 of the algorithm by moving towards the point $0.5e$. We move so that the restart point is 5% of the way from the boundary of the feasible region of the new relaxation towards $0.5e$. The dual iterate was updated to an earlier dual iterate, namely the last point where the relative duality gap was at least 10%.

Our primal heuristic used the primal point x to generate the incidence vectors of several cuts. Edges with x_{ij} smaller than 0.01 or greater than 0.99 forced vertices onto the same side or opposite sides of the cut and then unassigned vertices were assigned in a greedy manner. In order to get several cuts, the order in which initially unassigned vertices were examined was randomized. The number of cuts generated at stage k is $(1+(k/6))$. Once an incidence vector has been generated, it is modified using a local improvement process. The local improvement process looks for paths of vertices—all vertices on a path are moved to the other side of the cut if this results in improvement. We start off looking for paths consisting of just a single vertex, and eventually we look for paths containing up to 10 vertices. We use each vertex in turn as the starting vertex. We use a breadth first search to explore all paths starting from the vertex; if a path results in an increase in the size of the cut of at least 2.5, then we stop searching along this branch and backtrack. If we are unable to find an improving path starting from any vertex, we look for paths that do not hurt the solution. If we are then still unable to find improving paths, we terminate the local improvement process. This idea of looking for paths was proposed by Berry and Goldberg [5].

Each edge appears only in eight of the possible cycle constraints of length 4, so

Table 3
Results for Ising spin glass problems.

L	N_L	Time	Iters	Stages	Added	Energy
10	1946	0.47	9	2.0	69.0	–1.3895
20	1946	4.79	21	4.0	327.7	–1.3985
30	1546	24.38	39	6.7	809.4	–1.4003
40	1200	93.08	64	9.8	1550.6	–1.4001
50	720	290.75	99	13.8	2502.9	–1.4005
60	440	772.37	154	19.5	3670.0	–1.4019
70	384	2245.92	216	25.8	5294.2	–1.4012
80	310	5787.28	310	34.5	7219.3	–1.4012
90	280	11320.24	400	42.8	9501.6	–1.4017
100	229	11873.59	391	44.1	10975.0	–1.4023

the columns of the constraint matrix did not become dense. Therefore, we dropped a constraint only if none of the corresponding edge variables remained unfixed.

3.2.3. Computational results. We generated random problems using two different probability distributions. First, we generated random edge weights with a Gaussian distribution with mean 0 and standard deviation 1. Second, we generated edge weights of ±1, with 1 or -1 equally likely. Our results were far better for the second class. The principal properties of real spin glasses (for example, amorphous alloys) are represented well by the ±1 spin glass model on a rectangular lattice. The results in [12] are far better than our results with the first distribution, so we do not report these results in detail.

The results from problems where the edge weights were ±1 are contained in Table 3. We give the number N_L of problems of each size solved, the number of primal-dual iterations required, the total CPU time to solve the problems, the number of stages, the number of cuts added, and the average ground state energy. Typically, over 40% of the total CPU time for the larger instances was spent on the primal heuristics. In addition, we were unable to solve 2, 18, 10, and 7 instances with $L = 70, 80, 90, 100$, respectively, using just cutting planes of type (3.3), so these instances are omitted from the table.

When solving these problems, we exploited the fact that every cut will have even value, so we can terminate the algorithm with the optimal solution when the gap between the upper and lower bounds falls below two. With this termination criterion, we found that we were rarely able to fix any edges in step 9 of the algorithm. (This contrasts markedly with our experience with problems with a Gaussian distribution of edge weights, where fixing variables made it possible to solve problems which were otherwise beyond the reach of our implementation due to memory requirements.)

These results compare very favorably with those in De Simone et al. [13], who used the simplex solver in CPLEX3.0 in a branch and cut algorithm for problems with the same distribution, on a Sun SPARC 10, which is approximately half as fast as our machine. They report results for problems of size up to 70×70. Problems of size 50×50 took them roughly an hour, problems of size 60×60 took roughly two to three hours, and problems of size 70×70 required on the order of fifteen hours.

One reason for the better results for the problems with ±1 edge weights than with Gaussian edge weights is that the problems do not have to be solved as accurately: we can terminate if the gap becomes less than two. An interior point method is good at getting close to an optimal solution, but it may take a while in the cutting plane setting to get a relative gap of, say, 10^{-6}. Our primal heuristic worked well for the ±1 problems, almost always finding the optimal solution at least one or two stages before it was possible to prove optimality; this was not the case for the Gaussian problems, with the optimal solution often not discovered until the final stage.

The number of stages and iterations for problems with either distribution are sensitive to slight changes in the parameters of the algorithm. We found a slight change may well halve the number of iterations required to solve one problem but double the number of iterations required to solve another. The table contains the results with a set of parameters that appeared to produce reasonable results, producing some of the better runs for most problems and respectable results for most of the remaining problems.

4. Conclusions. We have presented cutting plane algorithms for several integer programming problems. These algorithms use a predictor-corrector interior point method to solve the LP relaxations. For some MAXCUT problems and linear ordering problems, we have obtained runtimes that are comparable with or better than those obtained using a cutting plane method that employs the simplex solver in CPLEX to solve the relaxations.

It appears from the results detailed in the current paper and from other experiments, that the most suitable problems are of the following types.

- The linear programming relaxations are large, with the number of variables and/or constraints numbering in the thousands. This is because of the well-documented observation that the performance of interior point methods relative to simplex methods for linear programs improves as the problem size increases.
- The objective function coefficients are integer. It then suffices to reduce the duality gap to be less than 1 in order to prove optimality. This is useful for an interior point cutting plane method because such an approach can typically get close to optimality quickly but then may take a long time to reduce the duality gap to, say, 10^{-6}. Problems with integral coefficients are more likely to suffer from primal or dual degeneracy, which is more harmful to the performance of a simplex cutting plane algorithm than an interior point cutting plane algorithm. When the objective function coefficients are fractional, an appropriate method may be to use an interior point cutting plane algorithm initially and switch over to a simplex cutting plane algorithm as optimality is approached.
- It should be possible to find a strictly feasible point in the convex hull of feasible integral points efficiently, because such a point can then be used to restart the algorithm after cutting planes have been added. If it is not possible to restart in this manner, the method proposed by Gondzio [19] can be used.

We have recently experimented with combining interior point cutting plane algorithms with dual simplex cutting plane algorithms, using the interior point solver for the early stages and the simplex solver for the later stages. The performance of this algorithm has been outstanding for linear ordering problems [36]. It may well be that such a hybrid cutting plane method is an appropriate choice for a wide variety of integer programming problems.

Acknowledgments. I am grateful to Brian Borchers for providing the cutting plane code for linear ordering problems that uses CPLEX4.0. I am also grateful to Ulrich Dorndorf and Gerd Reinelt for sending me the clustering problems and some of the linear ordering problems, respectively. Further, I thank the two referees and the associate editor for their thorough reading of the paper and useful comments.

REFERENCES

[1] D. Applegate, R. Bixby, V. Chvátal, and W. Cook, *On the solution of traveling salesman problems*, Doc. Math., Extra Vol. III (1998), pp. 645–656.

[2] F. Barahona, M. Grötschel, M. Jünger, and G. Reinelt, *An application of combinatorial optimization to statistical physics and circuit layout design*, Oper. Res., 36 (1988), pp. 493–513.

[3] F. Barahona, M. Jünger, and G. Reinelt, *Experiments in quadratic* 0-1 *programming*, Math. Programming, 44 (1989), pp. 127–137.

[4] F. Barahona and A. R. Mahjoub, *On the cut polytope*, Math. Programming, 36 (1986), pp. 157–173.

[5] J. Berry and M. Goldberg, *Path optimization for graph partitioning problems*, Discrete Appl. Math., 90 (1999), pp. 27–50.

[6] B. Borchers, *Private communication: Project of R. M. Cooke*, New Mexico Tech., Socorro, NM, 1996.

[7] A. Caprara and M. Fischetti, *Branch and cut algorithms*, in Annotated Bibliographies in Combinatorial Optimization, M. Dell'Amico, F. Maffioli, and S. Martello, eds., John Wiley, New York, 1997, Chapter 4.

[8] T. Christof and G. Reinelt, *Low-dimensional Linear Ordering Polytopes*, Tech. report, IWR Heidelberg, Germany, 1997.

[9] T. Christof and G. Reinelt, *Algorithmic aspects of using small instance relaxations in parallel branch-and-cut*, Algorithmica, to appear.

[10] CPLEX Optimization Inc., *CPLEX Linear Optimizer and Mixed Integer Optimizer*, Suite 279, 930 Tahoe Blvd., Bldg 802, Incline Village, NV.

[11] J. Czyzyk, S. Mehrotra, M. Wagner, and S. J. Wright, *PCx User Guide (Version* 1.1*)*, Tech. report, Optimization Technology Center, Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL, 1997.

[12] C. De Simone, M. Diehl, M. Jünger, P. Mutzel, G. Reinelt, and G. Rinaldi, *Exact ground states of Ising spin glasses: New experimental results with a branch and cut algorithm*, J. Statist. Phys., 80 (1995), pp. 487–496.

[13] C. De Simone, M. Diehl, M. Jünger, P. Mutzel, G. Reinelt, and G. Rinaldi, *Exact ground states of two-dimensional* $\pm J$ *Ising spin glasses*, J. Statist. Phys., 84 (1996), pp. 1363–1371.

[14] M. Deza and M. Laurent, *Facets for the cut con* I, Math. Programming, 56 (1992), pp. 121–160.

[15] M. Deza and M. Laurent, *Facets for the cut con* II: *Clique-web inequalities*, Math. Programming, 56 (1992), pp. 161–188.

[16] S. C. Eisenstat, M. C. Gurshy, M. H. Schultz, and A. H. Sherman, *The Yale Sparse Matrix Package,* I. *The symmetric codes*, Internat. J. Numer. Methods Engrg., 18 (1982), pp. 1145–1151.

[17] J.-L. Goffin and J.-P. Vial, *Multiple cuts in the analytic center cutting plane method*, SIAM J. Optim., to appear.

[18] J. Gondzio, *HOPDM (ver.* 2.12*)—A fast LP solver based on a primal-dual interior point method*, European J. Oper. Res., 85 (1995), pp. 221–225.

[19] J. Gondzio, *Warm start of the primal-dual method applied in the cutting plane scheme*, Math. Programming, 83 (1998), pp. 125–143.

[20] M. Grötschel and O. Holland, *Solution of large-scale travelling salesman problems*, Math. Programming, 51 (1991), pp. 141–202.

[21] M. Grötschel, M. Jünger, and G. Reinelt, *A cutting plane algorithm for the linear ordering problem*, Oper. Res., 32 (1984), pp. 1195–1220.

[22] M. Grötschel, M. Jünger, and G. Reinelt, *Optimal triangulation of large real-world input-output matrices*, Statistiche Hefte, 25 (1984), pp. 261–295.

[23] M. Grötschel, L. Lovasz, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer-Verlag, Berlin, 1988.

[24] M. Grötschel and Y. Wakabayashi, *A cutting plane algorithm for a clustering problem*, Math. Programming, 45 (1989), pp. 59–96.

[25] M. Jünger, *Polyhedral Combinatorics and the Acyclic Subdigraph Problem*, Heldermann, Berlin, 1985.

[26] M. Jünger, G. Reinelt, and S. Thienel, *Practical problem solving with cutting plane algorithms in combinatorial optimization*, in Combinatorial Optimization, DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 20, AMS, Providence, RI, 1995, pp. 111–152.

[27] R. M. Karp, *Reducibility among combinatorial problems*, in Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum Press, New York, 1972, pp. 85–103.

[28] E. K. Lee and J. E. Mitchell, *Computational experience of an interior point SQP algorithm in a parallel branch-and-bound framework*, in High Performance Optimization, H. L. Frenk, K. Roos, T. Terlaky, and S. Zhang, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 329–347.

[29] J. Liu, *The evolution of the minimum degree ordering algorithm*, SIAM Rev., 31 (1989), pp. 1–19.

[30] I. J. Lustig, R. E. Marsten, and D. F. Shanno, *Interior point methods for linear programming: Computational state of the art*, ORSA J. Comput., 6 (1994), pp. 1–14. See also the following commentaries and rejoinder.

[31] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

[32] J. E. Mitchell, *Fixing variables and generating classical cutting planes when using an interior point branch and cut method to solve integer programming problems*, European J. Oper. Res., 97 (1997), pp. 139–148.

[33] J. E. Mitchell, *An interior point cutting plane algorithm for Ising spin glass problems*, in Operations Research Proceedings, SOR 1997, Jena, Germany, P. Kischka and H.-W. Lorenz, eds., Springer-Verlag, Berlin, 1998, pp. 114–119.

[34] J. E. Mitchell, *Branch-and-cut algorithms for combinatorial optimization problems*, in Handbook of Applied Optimization, P. Pardalos and M. G. C. Resende, eds., Oxford University Press, London, to appear.

[35] J. E. Mitchell and B. Borchers, *Solving real-world linear ordering problems using a primal-dual interior point cutting plane method*, Ann. Oper. Res., 62 (1996), pp. 253–276.

[36] J. E. Mitchell and B. Borchers, *Solving linear ordering problems with a combined interior point/simplex cutting plane algorithm*, in High Performance Optimization, H. L. Frenk, K. Roos, T. Terlaky, and S. Zhang, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 349–366.

[37] J. E. Mitchell, P. P. Pardalos, and M. G. C. Resende, *Interior point methods for combinatorial optimization*, in Handbook of Combinatorial Optimization, Vol. 1, D.-Z. Du and P. Pardalos, eds., Kluwer Academic Publishers, Boston, 1998, pp. 189–297.

[38] J. E. Mitchell and M. J. Todd, *Solving combinatorial optimization problems using Karmarkar's algorithm*, Math. Programming, 56 (1992), pp. 245–284.

[39] M. Padberg and G. Rinaldi, *A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems*, SIAM Rev., 33 (1991), pp. 60–100.

[40] G. Palubeckis, *A branch-and-bound approach using polyhedral results for a clustering problem*, INFORMS J. Comput., 9 (1997), pp. 30–42.

[41] S. Ramaswamy and J. E. Mitchell, *On Updating the Analytic Center After the Addition of Multiple Cuts*, Tech. Report 37–94–423, DSES, Rensselaer Polytechnic Institute, Troy, NY, 1994. Substantially revised August 1998.

[42] G. Reinelt, *The Linear Ordering Problem: Algorithms and Applications*, Heldermann, Berlin, 1985.

[43] G. Reinelt, *Private communication*, 1995.

[44] H. Rieger, L. Santen, U. Blasum, M. Diehl, M. Jünger, and G. Rinaldi, *The critical exponents of the two-dimensional Ising spin glass revisited: Exact ground state calculations and Monte Carlo simulations*, J. Phys. A, 29 (1996), pp. 3939–3950.

SIAM J. OPTIM.
Vol. 10, No. 4, pp. 1228–1248

ERROR BOUNDS FOR LINEAR MATRIX INEQUALITIES[*]

JOS F. STURM[†]

Abstract. For iterative sequences that converge to the solution set of a linear matrix inequality, we show that the distance of the iterates to the solution set is at most $O(\epsilon^{2^{-d}})$. The nonnegative integer d is the so-called degree of singularity of the linear matrix inequality, and ϵ denotes the amount of constraint violation in the iterate. For infeasible linear matrix inequalities, we show that the minimal norm of ϵ-approximate primal solutions is at least $1/O(\epsilon^{1/(2^d-1)})$, and the minimal norm of ϵ-approximate Farkas-type dual solutions is at most $O(1/\epsilon^{2^d-1})$. As an application of these error bounds, we show that for any bounded sequence of ϵ-approximate solutions to a semidefinite programming problem, the distance to the optimal solution set is at most $O(\epsilon^{2^{-k}})$, where k is the degree of singularity of the optimal solution set.

Key words. semidefinite programming, error bounds, linear matrix inequality, regularized duality

AMS subject classifications. 90C31, 90C22, 65G99, 15A42

PII. S1052623498338606

1. Introduction.

Linear matrix inequalities play an important role in system and control theory; see the book by Boyd et al. [6]. Recently, considerable progress has been made in optimization over linear matrix inequalities, i.e., semidefinite programming (see [1, 9, 13, 14, 22, 24, 25, 32, 34] and the references cited therein).

We study the linear matrix inequality (LMI)

$$\begin{cases} X \in B + \mathcal{A}, \\ X \succeq 0, \end{cases} \tag{1.1}$$

where $X \succeq 0$ means positive semidefiniteness, B is a given (real) symmetric matrix, and $\mathcal{A}$ is a linear subspace of symmetric matrices.

The LMI (1.1) is in conic form; see, e.g., [23, 32]. Since we leave complete freedom as to the formulation of $\mathcal{A}$, it is in general not difficult to fit a given LMI into conic form. Consider for instance a linear matrix inequality

$$F_0 + \sum_{j=1}^{m} y_j F_j \succeq 0,$$

where $F_0, F_1, \ldots, F_m$ are given symmetric matrices. This is a conic form LMI (1.1) where $B = F_0$ and $\mathcal{A}$ is the span of $\{F_1, F_2, \ldots, F_m\}$.

Recently developed interior point codes for semidefinite programming make it possible to solve LMIs numerically. Such algorithms generate sequences of increasingly good approximate solutions, provided that the LMI is solvable. For a discussion of interior point methods for semidefinite programming, see, e.g., [13, 32]. A typical

[*] Received by the editors May 11, 1998; accepted for publication (in revised form) December 28, 1999; published electronically July 11, 2000. Research for this work was performed at Communications Research Laboratory, McMaster University, Hamilton, Canada, and was supported by Netherlands Organization for Scientific Research (NWO) grant S62-459.

http://www.siam.org/journals/siopt/10-4/33860.html

[†] Department of Quantitative Economics, Maastricht University, P.O. Box 616, NL-6200MD Maastricht, The Netherlands (j.sturm@ke.unimaas.nl).

way to measure the quality of an approximate solution is by evaluating its *constraint violation.*

For instance, if we denote the smallest eigenvalue of an approximate solution $\tilde{X}$ by $\lambda_{\min}(\tilde{X})$, then we may say that $\tilde{X}$ violates the constraint "$X \succeq 0$" by an amount of $[-\lambda_{\min}(\tilde{X})]_+$, where the operator $[\cdot]_+$ yields the positive part. In fact, $[-\lambda_{\min}(\tilde{X})]_+$ is the distance, measured in the matrix 2-norm, of the approximate solution $\tilde{X}$ to the cone of positive semidefinite matrices. The matrix 2-norm is a convenient measure for the amount by which the positive semidefiniteness constraint is violated, but other matrix norms can in principle be used as well.

Similarly, we say that $\tilde{X}$ violates the constraint "$X \in B + \mathcal{A}$" by an amount of $\operatorname{dist}(\tilde{X}, B + \mathcal{A})$, where $\operatorname{dist}(\cdot,\cdot)$ denotes the distance function (for a given norm). The total amount of constraint violation in $\tilde{X}$, i.e.,

$$\operatorname{dist}(\tilde{X}, B + \mathcal{A}) + [-\lambda_{\min}(\tilde{X})]_+, \tag{1.2}$$

is called the *backward error* of $\tilde{X}$ with respect to the LMI (1.1). The backward error indicates how much we should perturb the data of the problem, such that $\tilde{X}$ is an exact solution to the perturbed problem. (The perturbation parameters are symmetric matrices U and V such that $\tilde{X} \in (B + U) + \mathcal{A}$ and $\tilde{X} \succeq V$. One may restrict to perturbations in B by taking the positive semidefinite part of $\tilde{X}$, which has essentially the same backward error.)

However, the backward error does not (immediately) tell us the distance from $\tilde{X}$ to the solution set of the original LMI; this distance is called the *forward error* of $\tilde{X}$.

Without knowing any exact solution, there is no straightforward way to estimate the forward error. For linear inequality and equation systems, however, the forward error and backward error are of the same order of magnitude; see Hoffman [11]. The equivalence between forward and backward errors also holds true for systems that are described by convex quadratic inequalities if a Slater condition holds; see Luo and Luo [17]. In these cases, we have a relation of the form

$$\text{forward error} = O(\text{backward error}),$$

which is called a Lipschitzian error bound. For systems of convex quadratic inequalities without Slater's condition, an error bound of the form

$$\text{forward error} = O((\text{backward error})^{1/2^d}) \tag{1.3}$$

was obtained by Wang and Pang [35]. They also showed that $d \leq n+1$, where n is the dimension of the problem. The error bounds for linear and convex quadratic inequality systems hold without any boundedness assumption. This is remarkable, since in other cases where an error bound is known, the error bounds usually depends on the norm of the approximate solution [26]. Error bounds for systems with a nonconvex quadratic inequality are given in Luo and Sturm [19] and references cited therein.

An error bound of the form (1.3) is called a Hölderian error bound with exponent $\gamma = 1/2^d$. A Hölderian error bound has been demonstrated for analytic inequality and equation systems if the size of the approximate solutions is bounded by a fixed constant; see Luo and Pang [18]. However, there are no known positive lower bounds on the exponent γ, except in the linear and quadratic cases that are mentioned above, or when a Slater condition holds [7]. For a comprehensive survey of error bounds, we refer to Pang [26].

Some issues on error bounds for LMIs and semidefinite programming were addressed in [4, 7, 8, 22, 31, 33]. Deng and Hu [7] derived upper bounds on the Lipschitz constant (or condition number) for LMIs if Slater's condition holds. Luo, Sturm, and Zhang [22] and Sturm and Zhang [33] proved some Lipschitzian type error bounds for central solutions for semidefinite programs under strict complementarity.

There is a rich perturbation theory for nonlinear optimization problems which also applies to semidefinite programming; see the recent survey by Bonnans and Shapiro [4]. Under regularity and nondegeneracy conditions, Shapiro [31] showed that the then unique optimal solution is differentiable in the perturbation parameter; this is an application of the inverse function theorem. Bonnans and Shapiro [4] also gave an example with unique primal and dual optimal solutions, while violating the strict complementarity condition in which the optimal solution is not Lipschitz stable. Without imposing nondegeneracy conditions, little is known about the sensitivity of the optimal solution set with respect to perturbations; much more is known about the optimal value function. In particular, it is already known from Rockafellar [30] that the directional derivatives of the optimal value function exist in any perturbation direction. Goldfarb and Scheinberg [8] investigated how these directional derivatives can be computed. Helmberg [10] demonstrated how to use dual solutions to estimate the optimal objective value when new constraints are added.

In this paper, we show for LMIs in $n \times n$ matrices that (1.3) holds for a certain $d \in \{0, 1, 2, \ldots, n-1\}$, the so-called *degree of singularity*, provided that the size of the approximate solutions is bounded. The boundedness assumption is not very restrictive. Namely, the interior point method generates a bounded sequence of approximate solutions whenever the LMI has a feasible solution. We will show that if the LMI is infeasible, then the norm of ϵ-approximate solutions must be at least $1/O(\epsilon^{1/(2^d-1)})$.

The error bound results in this paper hold without any assumptions on the LMI. In particular, the solution set of the LMI can be unbounded and nonsolid.

We interpret the degree of singularity in the context of regularized duality as defined by Borwein and Wolkowicz [5] and Ramana [28]. (See [29] for the relation between the dual in [28] and the regularization scheme in [5].) The degree of singularity is basically the number of elementary regularizations that are needed to obtain a fully regularized dual. Under Slater's constraint qualification, the irregularity level d is zero. (Notice that this is also true for convex quadratic systems; see Wang and Pang [35].) Unfortunately, it is not easy to determine the irregularity level in general. But even if d is unknown, a nontrivial error bound is obtained by replacing d with its upper bound, $d \leq n-1$.

It is a natural idea to apply error bound results for LMIs to obtain sensitivity results for semidefinite optimization and vice versa. However, we cannot use the same argumentation that links results for systems of linear inequalities with results for parametric linear programs. Namely, the linear case has two crucial properties: the optimal solution set is characterized by the primal-dual optimality conditions, and a strictly complementary solution exists. In the nonlinear case, however, there may not exist any Lagrangean dual solution even if the primal problem has an optimal solution. The existence of a dual optimal solution can be guaranteed by imposing (possibly restrictive) constraint qualifications, but even then a strictly complementary solution may not exist. Perturbation theory has been developed for the situation where no dual optimal solution exists (and the primal Slater condition fails). However, these results require some second order conditions [4], which imply a Hölderian error bound of degree 1/2.

In section 4, we will apply our error bound for LMIs to obtain an error bound for the optimal solution set of a semidefinite program. In general, however, there is not a sensible way to define a backward error for the optimal solution set of a semidefinite program, using only approximate solutions to the standard primal-dual pair. Therefore, we will assume that a complementary solution exists. The optimal solution set is then described by the LMI

$$\left\{\begin{array}{c} X \in B + \mathcal{A}, \\ \bar{C} \bullet X = 0, \\ X \succeq 0, \end{array}\right. \tag{1.4}$$

where $\bar{C}$ is a dual optimal solution. For (1.4), i.e., the optimal solution set of a semidefinite programming problem, the degree of singularity is at most one, if strict complementarity holds. The concept of singularity degrees thus embeds the Slater and strict complementarity conditions in a hierarchy of singularity for LMIs. We will argue that if a feasible interior point method is used, then one may simply take the duality gap as the backward error measure in the error bound. This applies in particular to interior point methods that use the self-dual embedding technique [20, 36]. It is left as a subject of further research to develop an error bound that is based on approximate solutions to Ramana's perfect dual [28, 13], instead of the standard Lagrangian dual.

This paper is organized as follows. In section 2, we introduce the concept of regularized backward errors, which is closely related to the concept of minimal faces [5]. In this section, we also show that there is a close connection between the regularized backward error and the forward error. We will then estimate in section 3 how the regularized backward error depends on the usual backward error. In section 4, we apply the error bound for LMIs to semidefinite programming problems. The paper is concluded in section 5.

Notation. Let $\mathcal{S}^{n\times n}$ denote the space of $n \times n$ real symmetric matrices. The cone of all positive semidefinite matrices in $\mathcal{S}^{n\times n}$ is denoted by $\mathcal{S}^{n\times n}_{+}$, and we write $X \succeq 0$ if and only if $X \in \mathcal{S}^{n\times n}_{+}$. The interior of $\mathcal{S}^{n\times n}_{+}$ is the set of positive definite matrices $\mathcal{S}^{n\times n}_{++}$, and we write $X \succ 0$ if and only if $X \in \mathcal{S}^{n\times n}_{++}$. We let $N := n(n+1)/2$ denote the dimension of the real linear space $\mathcal{S}^{n\times n}$. The standard inner product for two symmetric matrices X and Y is $X \bullet Y = \operatorname{tr} XY$. The matrix norm $\|X\|_2$ is the operator 2-norm that is associated with the Euclidean norm for vectors, namely

$$\|X\|_2 = \max\{\|Xy\|_2 \mid \|y\|_2 = 1\}.$$

For symmetric matrices, $\|X\|_2$ is the largest eigenvalue of X or the largest eigenvalue of $-X$, whichever is larger.

2. The regularized backward error.

Let $\bar{\mathcal{A}}$ denote the smallest linear subspace containing $B + \mathcal{A}$, i.e.,

$$\bar{\mathcal{A}} = \{X \in \mathcal{S}^{n\times n} \mid X + tB \in \mathcal{A} \text{ for some } t \in \Re\}. \tag{2.1}$$

We are naturally interested in the intersection of this linear subspace with the cone of positive semidefinite matrices. It holds that

$$\bar{\mathcal{A}} \cap \mathcal{S}^{n\times n}_{+} = \{0\} \iff \bar{\mathcal{A}}^{\perp} \cap \mathcal{S}^{n\times n}_{++} \neq \emptyset; \tag{2.2}$$

the above characterization is a simple illustration of a duality theorem for convex cones which will play an important role in our analysis.

This duality theorem states that given a linear subspace $\mathcal{L}$ and a convex cone $\mathcal{K} \subseteq \Re^N$ with relative interior $\operatorname{relint} \mathcal{K}$ and dual cone $\mathcal{K}^* := \{z \in \Re^N \mid z^{\mathrm{T}}\mathcal{K} \subseteq \Re_+\}$, it holds that

$$\mathcal{L} \cap \text{ relint } \mathcal{K} \neq \emptyset \iff \mathcal{L}^{\perp} \cap \mathcal{K}^{*} \subseteq -\mathcal{K}^{*}; \tag{2.3}$$

see Corollary 2 in Luo, Sturm, and Zhang [21] and Corollary 2.2 in [32]. This result generalizes a classical duality theorem of Gordan and Stiemke for linear inequalities.

To see why (2.2) is a special case of (2.3), we must interpret $\mathcal{S}_{+}^{n\times n}$ as a convex cone in $\Re^N$. This can be established by choosing an orthonormal basis of $\mathcal{S}^{n\times n}$, say, an orthonormal set of symmetric matrices $\{S[1], S[2], \ldots, S[N]\}$, where $N := n(n+1)/2$ is the dimension of $\mathcal{S}^{n\times n}$. We can then associate with any matrix $X \in \mathcal{S}^{n\times n}$ a coordinate vector $x \in \Re^N$ into this basis and vice versa. Namely, we let $x_i = S[i] \bullet X$ for $i = 1, \ldots, N$, and $X = \sum_{i=1}^{N} x_i S[i]$. Due to the orthonormality of the basis, we have $X \bullet Y = x^{\mathrm{T}} y$ for all matrices $X, Y \in \mathcal{S}^{n\times n}$ with coordinate vectors $x, y \in \Re^N$.

As a convention, we use upper-case symbols, like X and B, for symmetric matrices, and we implicitly define the corresponding lower-case symbols, like x and b, to be the associated coordinate vectors, as described above. Furthermore, we use calligraphic letters, such as $\mathcal{S}_{+}^{n\times n}$, to denote sets. With the established one-to-one correspondence between $\mathcal{S}^{n\times n}$ and $\Re^N$ in mind, we do not only use $\mathcal{S}_{+}^{n\times n}$ for the set of positive semidefinite matrices in $\mathcal{S}^{n\times n}$, but also for the set of coordinate vectors of positive semidefinite matrices, which is a convex cone in the Euclidean space $\Re^N$. We will also use such a convention for other sets of symmetric matrices. In particular, we reformulate (2.1) as

$$\bar{\mathcal{A}} = \mathcal{A} + \text{ Img } b,$$

where Img $b \subset \Re^N$ is the line of all multiples of b. The orthogonal complement of $\bar{\mathcal{A}}$ is

$$\bar{\mathcal{A}}^{\perp} = \mathcal{A}^{\perp} \cap \text{ Ker } b^{\mathrm{T}} = \{X \in \mathcal{A}^{\perp} \mid B \bullet X = 0\}.$$

The all-zero matrix is obviously the only matrix that is both positive and negative semidefinite, i.e., $\mathcal{S}_{+}^{n\times n} \cap -\mathcal{S}_{+}^{n\times n} = \{0\}$. Also, the cone of positive semidefinite matrices is self-dual, i.e., $(\mathcal{S}_{+}^{n\times n})^{*} = \mathcal{S}_{+}^{n\times n}$. Thus, taking $\mathcal{K} = \mathcal{S}_{+}^{n\times n}$ and $\mathcal{L} = \bar{\mathcal{A}}^{\perp}$ in (2.3) yields (2.2).

Relation (2.2) states that if $\bar{\mathcal{A}}$ and $\mathcal{S}_{+}^{n\times n}$ intersect only at the origin, then there exists a positive definite matrix $Z \in \bar{\mathcal{A}}^{\perp}$. Now consider a sequence of increasingly accurate solutions $\{X(\epsilon) \mid \epsilon > 0\}$ satisfying

$$\operatorname{dist}(X(\epsilon), B + \mathcal{A}) \leq \epsilon \text{ and } \lambda_{\min}(X(\epsilon)) \geq -\epsilon \text{ for all } \epsilon > 0; \tag{2.4}$$

notice that the parameter ϵ measures the backward error in $X(\epsilon)$. It follows that since $Z \perp (B + \mathcal{A})$, we must have $|Z \bullet X(\epsilon)| = O(\epsilon)$. Using the fact that $X(\epsilon) + \epsilon I \succeq 0$ and Z is positive definite, this implies that $\|X(\epsilon)\| = O(\epsilon)$. The above reasoning establishes the relation

$$\bar{\mathcal{A}} \cap \mathcal{S}_{+}^{n\times n} = \{0\} \Longrightarrow \|X(\epsilon)\| = O(\epsilon), \tag{2.5}$$

which is an error bound for the case that $\bar{\mathcal{A}}$ intersects the semidefinite cone only at the origin.

Now assume that $\bar{\mathcal{A}} \cap \mathcal{S}_{+}^{n\times n} \neq \{0\}$, and let $X^{*} \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_{+}^{n\times n})$. By applying a basis transformation if necessary, we may assume without loss of generality that we can partition X^{*} as

$$X^{*} = \begin{bmatrix} X_B^{*} & 0 \\ 0 & 0 \end{bmatrix}, \quad X_B^{*} \succ 0. \tag{2.6}$$

Using this notation, we can partition an arbitrary matrix $X \in \mathcal{S}^{n\times n}$ as

$$X = \begin{bmatrix} X_B & X_U \\ X_U^{\mathrm{T}} & X_N \end{bmatrix}.$$

A face of $\mathcal{S}_+^{n\times n}$ is by definition a cone $\mathcal{F} \subseteq \mathcal{S}_+^{n\times n}$ such that for all $X, Y \in \mathcal{S}_+^{n\times n}$,

$$X + Y \in \mathcal{F} \Longrightarrow X, Y \in \mathcal{F}.$$

Since the faces of $\mathcal{S}_+^{n\times n}$ are exposed, we can parametrize the faces of $\mathcal{S}_+^{n\times n}$ as

$$\text{face}(\mathcal{S}_+^{n\times n}, Z) = \{X \in \mathcal{S}_+^{n\times n} \mid Z \bullet X = 0\}, \tag{2.7}$$

where the parameter Z is a given positive semidefinite matrix. In particular, if

$$Z = \begin{bmatrix} 0 & 0 \\ 0 & Z_N \end{bmatrix}, \qquad Z_N \succ 0,$$

then

$$\text{face}(\mathcal{S}_+^{n\times n}, Z) = \left\{ X = \begin{bmatrix} X_B & 0 \\ 0 & 0 \end{bmatrix} \middle| X_B \in \mathcal{S}_+^{n\times n} \right\}, \tag{2.8}$$

and X is in the relative interior of face$(\mathcal{S}_+^{n\times n}, Z)$ if $X_B \succ 0$ (and $X_U = 0$, $X_N = 0$). The facial structure of $\mathcal{S}_+^{n\times n}$ has been studied in detail by Bohnenblust [3], Barker and Carlson [2], Lewis [16], and Pataki [27]. The following two lemmas can be derived using the results of Barker and Carlson [2]. We give elementary proofs for completeness.

LEMMA 2.1. *Let $X^* \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n})$, and suppose without loss of generality that X^* is of the form* (2.6). *Then it holds for all $X \in \bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}$ that $X_U = 0$ and $X_N = 0$.*

Proof. Suppose to the contrary that X_N is not the all-zero matrix, and let $y^{\mathrm{T}} = \begin{bmatrix} 0 & y_N^{\mathrm{T}} \end{bmatrix}$ be such that $X_N y_N \neq 0$. Then for any $\alpha \in \Re$,

$$y^{\mathrm{T}}(X^* + \alpha X)y = \alpha\, y_N^{\mathrm{T}} X_N y_N, \qquad y_N^{\mathrm{T}} X_N y_N > 0,$$

where we used the fact that X is positive semidefinite. Consequently, we have for all $\alpha > 0$ that

$$X^* + \alpha X \in \bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}, \qquad X^* - \alpha X \notin \mathcal{S}_+^{n\times n},$$

which contradicts the fact that by definition X^* is in the relative interior of $\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}$. We have now shown by contradiction that $X_N = 0$. Since X is positive semidefinite, it also follows that $X_U = 0$. □

LEMMA 2.2. *Let $X^* \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n})$ and suppose without loss of generality that X^* is of the form* (2.6). *Then*

$$\text{relint}\ ((B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}) = (B + \mathcal{A}) \cap\ \text{relint}\ \ \text{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} \right).$$

Proof. The lemma holds trivially true if $(B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n} = \emptyset$. Suppose now that there exists $X \in (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}$. Since $X^* \in \bar{\mathcal{A}}$, there exists $t \in \Re$ such that $X^* - tB \in \mathcal{A}$. However, for all $\alpha > 0$ satisfying $\alpha t > -1$, we have

$$\frac{1}{1 + \alpha t}(X + \alpha X^*) \in (B + \mathcal{A}) \cap\ \text{relint}\ \ \text{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix} \right),$$

where we used Lemma 2.1. This shows that

$$(B+\mathcal{A}) \cap \text{ relint } \text{ face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right) \neq \emptyset.$$

Using Lemma 2.1 once again, the lemma follows from the above relation. □

Due to the above result, the face

$$\text{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)$$

is sometimes called the minimal face [5] or the regularized semidefinite cone [21] for the affine space $B + \mathcal{A}$.

The backward error of $X(\epsilon)$ with respect to the regularized system

$$(B+\mathcal{A}) \cap \text{ face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)$$

is naturally defined as

$$\text{dist}(X(\epsilon), B+\mathcal{A}) + [-\lambda_{\min}(X(\epsilon))]_+ + \|X_U(\epsilon)\| + \|X_N(\epsilon)\|. \tag{2.9}$$

An implication of Lemma 2.3 below is that if $\{X(\epsilon) \mid \epsilon > 0\}$ is bounded, then the regularized backward error is of the same order as the forward error $\text{dist}(X(\epsilon), (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n})$. Notice that there are no conditions made on the LMI itself.

Lemma 2.3. *Let $X^* \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n})$ and suppose without loss of generality that X^* is of the form (2.6). If $\{X(\epsilon) \mid \epsilon > 0\}$ is such that*

$$\text{dist}(X(\epsilon), B+\mathcal{A}) \le \epsilon, \quad \lambda_{\min}(X(\epsilon)) \ge -\epsilon, \quad \|X_U(\epsilon)\| + \|X_N(\epsilon)\| \le \epsilon \tag{2.10}$$

for all $\epsilon > 0$, then $(B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n} \neq \emptyset$. Moreover, there exists $\{\delta(\epsilon) \in \Re \mid \epsilon > 0\}$, such that

$$\text{dist}((1+\delta(\epsilon))X(\epsilon), (B+\mathcal{A}) \cap \mathcal{S}_+^{n\times n}) = O(\epsilon), \quad |\delta(\epsilon)| = O(\epsilon)$$

for $0 < \epsilon \le 1$.

Proof. As is well known, the backward and forward error for a system of linear equations are of the same order [11]. Therefore, the relations

$$\text{dist}(X(\epsilon), B+\mathcal{A}) \le \epsilon, \quad \|X_U(\epsilon)\| + \|X_N(\epsilon)\| \le \epsilon$$

imply that

$$\text{dist}(X(\epsilon), \{X \in B+\mathcal{A} \mid X_U = 0,\, X_N = 0\}) = O(\epsilon).$$

This bound implies the existence of $\{Y(\epsilon) \mid \epsilon > 0\}$ such that

$$X(\epsilon) + Y(\epsilon) \in \{X \in B+\mathcal{A} \mid X_U = 0,\, X_N = 0\}, \quad \|Y(\epsilon)\| = O(\epsilon). \tag{2.11}$$

Using also the fact that X_B^* is positive definite, it follows that

$$X(\epsilon) + Y(\epsilon) + \alpha(\epsilon)X^* \succeq 0 \quad \text{for all} \quad \epsilon > 0,$$

with

$$\alpha(\epsilon) := \frac{[-\lambda_{\min}(X(\epsilon))]_+ + \|Y(\epsilon)\|_2}{\lambda_{\min}(X_B^*)}.$$

Notice that $\alpha(\epsilon) = O(\epsilon)$. Since $X^* \in \bar{\mathcal{A}}$, there must exist $t \in \Re$ such that $X^* - tB \in \mathcal{A}$. Let $\bar{\epsilon} > 0$ be such that $t\alpha(\epsilon) > -1$ for all $\epsilon \in (0, \bar{\epsilon}]$. Then

$$\frac{1}{1 + t\alpha(\epsilon)}(X(\epsilon) + Y(\epsilon) + \alpha(\epsilon)X^*) \in (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n} \text{ for } 0 < \epsilon \leq \bar{\epsilon},$$

and hence

$$\operatorname{dist}\left(\frac{1}{1 + t\alpha(\epsilon)}X(\epsilon), (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}\right) = O(\epsilon)$$

for $0 < \epsilon \leq \bar{\epsilon}$. □

Under Slater's condition, i.e., if $(B + \mathcal{A}) \cap \mathcal{S}_{++}^{n\times n} \neq \emptyset$, Lemma 2.3 generalizes Hoffman's error bound [11] for systems of linear inequalities and equations to LMIs. Remark that error bounds for convex inequality systems under Slater's condition are well known, but require additional regularity conditions [12]. Here we do not impose additional conditions. In particular, no boundedness assumption is made, i.e., the error bound holds globally over $\mathcal{S}^{n\times n}$. However, the lemma requires a scaling factor $1 + \delta(\epsilon)$, which is not needed in the case of linear inequalities and equations. The following example shows that this scaling factor is essential in the case of LMIs.

Example 1. Consider the LMI in $\mathcal{S}^{2\times 2}$ with

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathcal{A} = \left\{ W = \begin{bmatrix} w_{11} & w_{12} \\ w_{12} & 0 \end{bmatrix} \middle| w_{11}, w_{12} \in \Re \right\},$$

i.e., we want to find w_{11} and w_{12} such that $w_{11} \geq |w_{12}|^2$. This LMI obviously has positive definite solutions (the identity matrix for instance). Therefore, the regularized backward error is identical to the usual backward error. The approximate solution

$$X(\epsilon) := \begin{bmatrix} 1/(\epsilon^2 + \epsilon^3) & 1/\epsilon \\ 1/\epsilon & 1 + \epsilon \end{bmatrix}$$

has backward error $\epsilon > 0$. However, $X(\epsilon) + Y(\epsilon) \in (B + \mathcal{A}) \cap \mathcal{S}_+^{2\times 2}$ if and only if

$$y_{22}(\epsilon) = -\epsilon, \quad y_{11}(\epsilon) \geq \frac{1}{\epsilon(1+\epsilon)} + \frac{2\, y_{12}(\epsilon)}{\epsilon} + |y_{12}(\epsilon)|^2,$$

which shows that the distance of $X(\epsilon)$ to $(B + \mathcal{A}) \cap \mathcal{S}_+^{2\times 2}$ is bounded from below by a positive constant as $\epsilon \downarrow 0$. However, we have $X(\epsilon)/(1 + \epsilon) \in (B + \mathcal{A}) \cap \mathcal{S}_+^{2\times 2}$, which agrees with the statement of Lemma 2.3.

Below are more remarks on the regularized error bound of Lemma 2.3.

Remark 1. Lemma 2.3 states that the mere existence of $\{X(\epsilon) \mid \epsilon > 0\}$ satisfying (2.10) for all $\epsilon > 0$ implies that $(B+\mathcal{A}) \cap \mathcal{S}_+^{n\times n} \neq \emptyset$, even though $X(\epsilon)$ is not necessarily bounded for $\epsilon \downarrow 0$. In the case of weak infeasibility, i.e., if

$$\operatorname{dist}(B + \mathcal{A}, \mathcal{S}_+^{n\times n}) = 0, \quad (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n} = \emptyset,$$

we can therefore conclude that if $X(\epsilon)$ satisfies (2.4), then

$$\liminf_{\epsilon \downarrow 0} \|X_N(\epsilon)\| + \|X_U(\epsilon)\| > 0.$$

Remark 2. If $X^{(1)}, X^{(2)}, \ldots$ is a bounded sequence with

$$\operatorname{dist}(X^{(k)}, B + \mathcal{A}) \to 0 \text{ and } [-\lambda_{\min}(X^{(k)})]_+ \to 0 \text{ for } k \to \infty,$$

then also $\|X_U^{(k)}\| + \|X_N^{(k)}\| \to 0$, as follows from Lemma 2.1. Letting

$$\epsilon_k := \operatorname{dist}(X^{(k)}, B+\mathcal{A}) + [-\lambda_{\min}(X^{(k)})]_+ + \|X_U^{(k)}\| + \|X_N^{(k)}\|,$$

it follows from Lemma 2.3 and the boundedness of the sequence $\{X^{(k)} \mid k = 1, 2, \ldots\}$ that

$$\operatorname{dist}(X^{(k)}, (B+\mathcal{A}) \cap \mathcal{S}_+^{n\times n}) = O(\epsilon_k).$$

3. Regularization steps.

In order to bound the regularized backward error (2.9) in terms of the original backward error (1.2), we use a sequence of regularization steps.

In section 2, we have partitioned $n \times n$ matrices according to the structure of X^* given by (2.6). In this section, we will also partition $n \times n$ matrices into blocks, but with respect to a possibly different eigenvector basis; the sizes of the blocks can be different as well. We will denote the blocks by the subscripts $_{11}$, $_{12}$ and $_{22}$, i.e.,

$$X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^{\mathrm{T}} & X_{22} \end{bmatrix}.$$

We will also encounter the dual cone of a face of $\mathcal{S}_+^{n\times n}$, namely,

$$\begin{aligned} \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)^* &= \{Z \mid Z_{11} \bullet X_{11} \geq 0 \text{ for all } X_{11} \succ 0\} \\ &= \left\{ \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^{\mathrm{T}} & Z_{22} \end{bmatrix} \middle| Z_{11} \succeq 0 \right\}. \end{aligned}$$

Obviously, we have

$$\operatorname{relint}\ \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)^* = \left\{ \begin{bmatrix} Z_{11} & Z_{12} \\ Z_{12}^{\mathrm{T}} & Z_{22} \end{bmatrix} \middle| Z_{11} \succ 0 \right\}.$$

In the following, we will allow the possibility that $X = X_{11}$, i.e., X_{12} and X_{22} are nonexistent. For this case, we use the convention that $\|X_{12}\| = \|X_{22}\| = 0$.

LEMMA 3.1. *Let $\bar{\mathcal{A}}$ be a linear subspace of $\mathcal{S}^{n\times n}$, and suppose that $\{X(\epsilon) \mid 0 < \epsilon \leq 1\}$ is such that*

$$\operatorname{dist}(X(\epsilon), \bar{\mathcal{A}}) \leq \epsilon, \quad \|X_{12}(\epsilon)\| + \|X_{22}(\epsilon)\| \leq \epsilon, \quad \lambda_{\min}(X(\epsilon)) \geq -\epsilon$$

for all $0 < \epsilon \leq 1$. Let

$$Z \in \operatorname{relint}\ \left(\bar{\mathcal{A}}^\perp \cap \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)^*\right).$$

It holds that

- $Z_{11} \succ 0$ *if and only if*

$$\bar{\mathcal{A}} \cap \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right) = \{0\}.$$

- $Z_{11} = 0$ *if and only if*

$$\bar{\mathcal{A}} \cap \operatorname{relint}\ \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right) \neq \emptyset.$$

- *For the remaining case that* $0 \neq Z_{11} \not\succ 0$*, let* $Q = \begin{bmatrix} Q_1, & Q_2 \end{bmatrix}$ *be an orthogonal matrix such that* $Z_{11}Q_1 = 0$*,* $Q_2^T Z_{11} Q_2 \succ 0$*, and hence* $Q^T Z_{11} Q = 0 \oplus (Q_2^T Z_{11} Q_2)$*. Then*

$$\|Q_2^T X_{11}(\epsilon) Q_2\| = O(\epsilon), \quad \|X_{11}(\epsilon)Q_2\| = O(\sqrt{\epsilon \|X(\epsilon)\|}).$$

Proof. The first two cases, i.e., $Z_{11} = 0$ or $Z_{11} \succ 0$, are immediate applications of (2.3). It remains to consider the case that Z_{11} is a nonzero but singular, positive semidefinite matrix. Since $\operatorname{dist}(X(\epsilon), \bar{\mathcal{A}}) \leq \epsilon$, there must exist $Y(\epsilon)$, such that

$$X(\epsilon) + Y(\epsilon) \in \bar{\mathcal{A}}, \quad \|Y(\epsilon)\| \leq \epsilon \tag{3.1}$$

for all $\epsilon > 0$. This implies that $Z \perp (X(\epsilon) + Y(\epsilon))$ because $Z \in \bar{\mathcal{A}}^{\perp}$, and therefore

$$\begin{aligned} Z_{11} \bullet X_{11}(\epsilon) &= Z \bullet \left(\begin{bmatrix} X_{11}(\epsilon) & 0 \\ 0 & 0 \end{bmatrix} - X(\epsilon) - Y(\epsilon) \right) \\ &\leq \|Z\|_F \left\| \begin{bmatrix} Y_{11}(\epsilon) & X_{12}(\epsilon) + Y_{12}(\epsilon) \\ (X_{12}(\epsilon) + Y_{12}(\epsilon))^{\mathrm{T}} & X_{22}(\epsilon) + Y_{22}(\epsilon) \end{bmatrix} \right\|_F , \end{aligned}$$

where we used the Cauchy–Schwarz inequality. Now recall that

$$\|X_{12}(\epsilon)\| = O(\epsilon), \quad \|X_{22}(\epsilon)\| = O(\epsilon), \quad \|Y(\epsilon)\| = O(\epsilon),$$

so that we further obtain

$$Z_{11} \bullet X_{11}(\epsilon) = O(\epsilon). \tag{3.2}$$

We will now apply a basis transformation so that the structure in $X_{11}(\epsilon)$ becomes apparent. In particular, we define $\Xi(\epsilon) := Q^T X_{11}(\epsilon) Q$. Partition $\Xi(\epsilon)$ according to the partition in Q, i.e.,

$$\Xi_{11}(\epsilon) := Q_1^T X_{11}(\epsilon) Q_1, \; \Xi_{12}(\epsilon) := Q_1^T X_{11}(\epsilon) Q_2, \; \Xi_{22}(\epsilon) := Q_2^T X_{11}(\epsilon) Q_2.$$

By definition, Q is such that

$$Q^{\mathrm{T}} Z_{11} Q = \begin{bmatrix} 0 & 0 \\ 0 & Q_2^{\mathrm{T}} Z_{11} Q_2 \end{bmatrix}, \quad Q_2^{\mathrm{T}} Z_{11} Q_2 \succ 0.$$

Using also that $\lambda_{\min}(\Xi(\epsilon)) = \lambda_{\min}(X_{11}(\epsilon)) \geq \lambda_{\min}(X(\epsilon)) \geq -\epsilon$, we get

$$\begin{aligned} 0 &\leq \operatorname{tr} (Q_2^{\mathrm{T}} Z_{11} Q_2)(\Xi_{22}(\epsilon) + \epsilon I) \\ &= \operatorname{tr} (Q^{\mathrm{T}} Z_{11} Q)(\Xi(\epsilon) + \epsilon I) \\ &= \operatorname{tr} Z_{11}(X_{11}(\epsilon) + \epsilon I) = O(\epsilon), \end{aligned}$$

where we applied estimation (3.2) in the last identity. Recalling that $Q_2^{\mathrm{T}} Z_{22} Q_2 \succ 0$, it follows easily from the above relation that

$$\|\Xi_{22}(\epsilon)\| = O(\epsilon). \tag{3.3}$$

Finally, since $\lambda_{\min}(X(\epsilon)) \geq -\epsilon$, we know that $\Xi(\epsilon) + 2\epsilon I$ is positive definite. The Schur complement

$$(\Xi_{11}(\epsilon) + 2\epsilon I) - \Xi_{12}(\epsilon) \left(\Xi_{22}(\epsilon) + 2\epsilon I\right)^{-1} \Xi_{12}(\epsilon)^{\mathrm{T}}$$

must therefore be positive definite as well. However, the eigenvalues of $\Xi_{22}(\epsilon)$ are all $O(\epsilon)$ in magnitude; see (3.3). The eigenvalues of $(\Xi_{22}(\epsilon)+2\epsilon I)^{-1}$ are therefore bounded below by $1/O(\epsilon)$. It thus follows that

$$\|\Xi_{12}(\epsilon)\|^2 = O(\epsilon\|\Xi_{11}(\epsilon)+2\epsilon I\|). \tag{3.4}$$

Furthermore, an immediate consequence of the definition of $\Xi(\epsilon)$ is that

$$\|\Xi_{11}(\epsilon)\| \le \|X_{11}(\epsilon)\|, \quad \|\Xi_{12}(\epsilon)\| \le \|X_{11}(\epsilon)\|. \tag{3.5}$$

Considering (3.4)–(3.5), we conclude that

$$\|\Xi_{12}(\epsilon)\|^2 = O(\epsilon\|X_{11}(\epsilon)\|). \tag{3.6}$$

The bound in (3.4) can be replaced by (3.6) without further conditions, because if $\|X_{11}(\epsilon)\| < \epsilon$, then also $\|X_{11}(\epsilon)\|^2 \le \epsilon\|X_{11}(\epsilon)\|$. Similarly, (3.3) and (3.5) imply that

$$\|\Xi_{22}(\epsilon)\|^2 = O(\epsilon\|X_{11}(\epsilon)\|). \tag{3.7}$$

This completes the proof: the claimed bounds follow directly from (3.3), (3.6), and (3.7). □

For a given linear subspace $\bar{\mathcal{A}} \subseteq \mathcal{S}^{n\times n}$, we define the *level of singularity* $d(\bar{\mathcal{A}})$ by recursively applying the construction of Lemma 3.1. This procedure, which is equivalent to the regularization scheme of Borwein and Wolkowicz [5], is outlined below.

PROCEDURE 1. *Definition of the level of singularity of a linear subspace* $\bar{\mathcal{A}} \subseteq \mathcal{S}^{n\times n}$.

Step (1) *Let* $Z^{(0)} \in \operatorname{relint}(\bar{\mathcal{A}}^\perp \cap \mathcal{S}^{n\times n}_+)$. *If* $Z^{(0)}=0$ *or* $Z^{(0)} \succ 0$, *then* $d(\bar{\mathcal{A}})=0$. *Otherwise, proceed with Step* 2.

Step (2) *Let* $Q_1^{(0)}, Q_2^{(0)}$ *be such that* $Z^{(0)}Q_1^{(0)}=0$ *and* $(Q_2^{(0)})^T Z^{(0)} Q_2^{(0)} \succ 0$. *Set* $d=1$ *and*

$$\bar{\mathcal{A}}_1 = \left\{ X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix} \middle| \begin{bmatrix} Q_1^{(0)}, & Q_2^{(0)} \end{bmatrix} X \begin{bmatrix} (Q_1^{(0)})^T \\ (Q_2^{(0)})^T \end{bmatrix} \in \bar{\mathcal{A}} \right\}.$$

Step (3) *Let*

$$Z^{(d)} \in \operatorname{relint}\left(\bar{\mathcal{A}}_d^\perp \cap \operatorname{face}\left(\mathcal{S}_+^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)^* \right).$$

If $Z_{11}^{(d)}=0$, *then set* $d(\bar{\mathcal{A}})=d$. *Otherwise, proceed with Step* 4.

Step (4) *Let* $Q_1^{(d)}, Q_2^{(d)}$ *be such that* $Z_{11}^{(d)}Q_1^{(d)}=0$ *and* $(Q_2^{(d)})^T Z_{11}^{(d)} Q_2^{(d)} \succ 0$, *and define*

$$\bar{Q}_1^{(d)} = \begin{bmatrix} Q_1^{(d)} \\ 0 \end{bmatrix}, \quad \bar{Q}_2^{(d)} = \begin{bmatrix} Q_2^{(d)} & 0 \\ 0 & I \end{bmatrix}, \quad \bar{Q}^{(d)} = \begin{bmatrix} \bar{Q}_1^{(d)}, & \bar{Q}_2^{(d)} \end{bmatrix}.$$

Let

$$\bar{\mathcal{A}}_{d+1} = \left\{ X = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^T & X_{22} \end{bmatrix} \middle| \begin{bmatrix} \bar{Q}_1^{(d)}, & \bar{Q}_2^{(d)} \end{bmatrix} X \begin{bmatrix} (\bar{Q}_1^{(d)})^T \\ (\bar{Q}_2^{(d)})^T \end{bmatrix} \in \bar{\mathcal{A}}_d \right\},$$

and therefore

$$\bar{\mathcal{A}}_{d+1} = \left\{ X \;\middle|\; \left(\prod_{k=0}^{d} \bar{Q}^{(k)} \right) X \left(\prod_{k=0}^{d} \bar{Q}^{(k)} \right)^T \in \bar{\mathcal{A}} \right\}.$$

Set $d = d + 1$ *and return to Step* 3.

Note that the order of the $_{11}$ block is reduced in each iteration of the above procedure. We start with the full dimensional cone $\mathcal{S}_+^{n\times n}$, and in the first iteration we determine a face of this cone. Next we arrive at a face of this face and so on. We claim that this procedure finally arrives at the minimal face. To see this, notice that at any given step $d = 0, 1, \dots, d(\bar{\mathcal{A}})$ above, we perform a regularization step as described in Lemma 3.1. Recall from (2.2) that $d(\bar{\mathcal{A}}) = 0$ and $Z^{(0)} \succ 0$ if and only if $\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n} = \{0\}$, and this case has already been treated in section 2. In any other case, we have $Z_{11}^{(d(\bar{\mathcal{A}}))} = 0$. It is easily seen from Lemma 3.1 that if $X \in \bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}$, then $X\bar{Q}_2^{(d(\bar{\mathcal{A}})-1)} = 0$. This means that all nonzeros of X must be contained in the (final) $_{11}$ block for $\bar{\mathcal{A}}_{d(\bar{\mathcal{A}})}$. On the other hand, since $Z_{11}^{(d(\bar{\mathcal{A}}))} = 0$ in the above procedure, it follows from (2.3) that there exists $\tilde{X} \in \bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}$ such that $\tilde{X}_{11} \succ 0$ and $\tilde{X}_{12} = 0$, $\tilde{X}_{22} = 0$. Since we just showed that $X_{12} = 0$ and $X_{22} = 0$ for all $X \in \bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n}$, we must have $\tilde{X} \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n})$. Hence, the face in the final iteration is the minimal face. For $\bar{\mathcal{A}} = \mathcal{A} +$ Img b, we may therefore take $X^* = \tilde{X}$ and $X_B^* = \tilde{X}_{11}$; see (2.6).

The columns of $\prod_{k=0}^{d(\bar{\mathcal{A}})-1} \bar{Q}^{(k)}$ in Procedure 1 define a new basis for $\Re^n$, and the matrices $Z^{(d)}$, $d = 0, 1, \dots, d(\bar{\mathcal{A}})$, are block diagonal with respect to this basis. Thus, by applying a basis transformation if necessary, we may assume without loss of generality that there is a $(d(\bar{\mathcal{A}}) + 1) \times (d(\bar{\mathcal{A}}) + 1)$ block partition, such that for $k = 1, 2, \dots, d(\bar{\mathcal{A}})$,

$$(3.8)\quad \left\{ \begin{array}{l} Z := Z^{(d(\bar{\mathcal{A}})-k)}, \\ Z = \left[\begin{array}{ccc} 0 & 0 & Z(1:k, k+2:d(\bar{\mathcal{A}})+1) \\ & Z(k+1,k+1) & Z(k+1, k+2:d(\bar{\mathcal{A}})+1) \\ & & Z(k+2:d(\bar{\mathcal{A}})+1, k+2:d(\bar{\mathcal{A}})+1) \end{array} \right], \\ Z(k+1,k+1) \succ 0. \end{array} \right.$$

Above, we used a Matlab-type[1] notation, thus $1 : k$ means $1, 2, \dots, k$, and $Z(i, j)$ denotes the block on the ith row and jth column in the $(d(\bar{\mathcal{A}})+1) \times (d(\bar{\mathcal{A}})+1)$ block partition. Since Z is symmetric, we specified only the upper block triangular part of Z. The relation between the $(d(\bar{\mathcal{A}}) + 1) \times (d(\bar{\mathcal{A}}) + 1)$ partition in (3.8) and the 2×2 partition in iteration $d = d(\bar{\mathcal{A}}) - k$ of Procedure 1 is that

$$Z_{11} = Z(1 : k + 1, 1 : k + 1).$$

The minimal face is the set of matrices X for which

$$X(1, 1) \succeq 0, \quad X(i, j) = 0 \text{ for all } (i, j) \neq (1, 1).$$

In iteration $d = d(\bar{\mathcal{A}}) - k$ of Procedure 1, we arrive at the face where

$$X(1 : k + 1, 1 : k + 1) \succeq 0, \quad X(i, j) = 0 \text{ if } \max(i, j) > k + 1,$$

[1] MATLAB is a registered trademark of The MathWorks, Inc.

which indeed includes the minimal face.

We remark that the third row and column in the 3×3 block form of (3.8) are nonexistent for $k = d(\bar{\mathcal{A}})$.

Using Lemma 3.1, we can now estimate the regularized backward error.

LEMMA 3.2. *Let $\bar{\mathcal{A}} = \mathcal{A} +$ Img b, and $X^* \in$ relint $(\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n})$. Suppose without loss of generality that X^* is of the form* (2.6)*. If $d(\bar{\mathcal{A}}) > 1$ and $\{X(\epsilon) \mid 0 < \epsilon \leq 1\}$ is such that for all $0 < \epsilon \leq 1$,*

$$\mathrm{dist}(X(\epsilon), B + \mathcal{A}) \leq \epsilon, \quad \lambda_{\min}(X(\epsilon)) \geq -\epsilon,$$

then

$$\|X_U(\epsilon)\| = O(\epsilon^\gamma \|X(\epsilon)\|^{1-\gamma}), \quad \|X_N(\epsilon)\| = O(\epsilon^{2\gamma}\|X(\epsilon)\|^{1-2\gamma}),$$

with $\gamma = 2^{-d(\bar{\mathcal{A}})}$, where $d(\bar{\mathcal{A}})$ is the degree of singularity of $\bar{\mathcal{A}}$.

Proof. Assume that a suitable basis transformation has taken place, so that the partition in (3.8) is valid for all $k = 1, 2, \ldots, d(\bar{\mathcal{A}})$. Applying Lemma 3.1 in iteration $d = 0$ of Procedure 1, we have that

$$\|X_\epsilon(1 : d(\bar{\mathcal{A}}), d(\bar{\mathcal{A}}) + 1)\| = O(\sqrt{\epsilon\|X_\epsilon\|}), \quad \|X_\epsilon(d(\bar{\mathcal{A}}) + 1, d(\bar{\mathcal{A}}) + 1)\| = O(\epsilon),$$

where we used X_ϵ as a synonym for $X(\epsilon)$. Suppose now that in iteration $d \in \{0, \ldots, d(\bar{\mathcal{A}}) - 2\}$, we have

$$\text{(3.9)} \qquad \begin{cases} \|X_\epsilon(1 : k, k+1 : d(\bar{\mathcal{A}}) + 1)\| = O(\epsilon^\phi \|X_\epsilon\|^{1-\phi}), \\ \|X_\epsilon(k+1 : d(\bar{\mathcal{A}}) + 1, k+1 : d(\bar{\mathcal{A}}) + 1)\| = O(\epsilon^{2\phi}\|X_\epsilon\|^{1-2\phi}), \end{cases}$$

where

$$k = d(\bar{\mathcal{A}}) - d, \quad \phi = 2^{-(d+1)}.$$

We can now apply Lemma 3.1 with "ϵ" replaced by "$O(\epsilon^\phi\|X_\epsilon\|^{1-\phi})$." This yields the conclusion that (3.9) also holds for $k' = k - 1$ and $\phi' = \phi/2$. By induction, we obtain that (3.9) holds for $d = d(\bar{\mathcal{A}}) - 1$, $k = 1$, and $\phi = 2^{-(d+1)} = \gamma$. Since $X_\epsilon(1, 1) = X_B(\epsilon)$, the lemma follows. □

We arrive now at the main result of this paper, namely an error bound for LMIs.

THEOREM 3.3. *Let $\bar{\mathcal{A}} = \mathcal{A} +$ Img b. If $\{X(\epsilon) \mid 0 < \epsilon \leq 1\}$ is such that $\|X(\epsilon)\|$ is bounded and*

$$\mathrm{dist}(X(\epsilon), B + \mathcal{A}) \leq \epsilon \textit{ and } \lambda_{\min}(X(\epsilon)) \geq -\epsilon \textit{ for all } \epsilon > 0,$$

then

$$\mathrm{dist}(X(\epsilon), (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}) = O(\epsilon^{2^{-d(\bar{\mathcal{A}})}}).$$

Proof. For the case that $d(\bar{\mathcal{A}}) > 0$, the theorem follows by combining Lemma 2.3 with Lemma 3.2. If $d(\bar{\mathcal{A}}) = 0$, there are two cases, either $\bar{\mathcal{A}} \cap \mathcal{S}_+^{n\times n} = \{0\}$ or $\bar{\mathcal{A}} \cap \mathcal{S}_{++}^{n\times n} \neq \emptyset$. In the former case, we have $\|X(\epsilon)\| = O(\epsilon)$, and hence the error bound holds; see section 2. In the latter case, we have that $X^* = X_B^* \succ 0$, and the error bound follows from Lemma 2.3. □

An LMI is said to be weakly infeasible if

(1) there is no solution to the LMI, i.e., $(B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n} = \emptyset$, but

(2) $\mathrm{dist}(B+\mathcal{A},\mathcal{S}_+^{n\times n})=0$.
For weakly infeasible LMIs, there exist approximate solutions with arbitrarily small constraint violations. However, the following theorem provides a lower bound on the size of such approximate solutions to weakly infeasible LMIs.

THEOREM 3.4. *Let $\bar{\mathcal{A}}=\mathcal{A}+$ Img b and suppose that*

$$(B+\mathcal{A})\cap\mathcal{S}_+^{n\times n}=\emptyset.$$

If $\{X(\epsilon)\mid\epsilon>0\}$ is such that

$$\mathrm{dist}(X(\epsilon),B+\mathcal{A})\le\epsilon \text{ and } \lambda_{\min}(X(\epsilon))\ge-\epsilon \text{ for all } \epsilon>0,$$

then, for ϵ small enough, we have $X(\epsilon)\neq 0$ and

$$\frac{1}{\|X(\epsilon)\|}=O(\epsilon^{1/(2^{d(\bar{\mathcal{A}})}-1)}).$$

Proof. Suppose to the contrary that there exists a sequence $\epsilon_1,\epsilon_2,\dots$ with $\epsilon_k\to 0$ and $\|X(\epsilon_k)\|=o(\epsilon_k^{-1/(2^{d(\bar{\mathcal{A}})}-1)})$. Applying Lemma 3.2, it follows that

$$\|X_U(\epsilon_k)\|+\|X_N(\epsilon_k)\|=O(\epsilon_k^{2^{-d(\bar{\mathcal{A}})}}\|X(\epsilon_k)\|^{1-2^{-d(\bar{\mathcal{A}})}})=o(1).$$

Together with Lemma 2.3, we obtain that $(B+\mathcal{A})\cap\mathcal{S}_+^{n\times n}\neq\emptyset$, a contradiction. □

There is an extension of Farkas' lemma from linear inequalities to convex cones, which states that

$$\text{(3.10)} \qquad \mathrm{dist}(B+\mathcal{A},\mathcal{K})>0 \iff \exists Z\in\mathcal{A}^\perp\cap\mathcal{K}^*:\ B\bullet Z<0,$$

where $\mathcal{K}\subset\mathcal{S}^{n\times n}$ is a convex cone, and $\mathcal{K}^*$ is the associated dual cone. See, e.g., Lemma 2.5 in [32]. If $\mathrm{dist}(B+\mathcal{A},\mathcal{S}_+^{n\times n})>0$, then we say that the LMI is strongly infeasible. Relation (3.10) states that strong infeasibility can be demonstrated by a matrix $Z\in\mathcal{A}^\perp\cap\mathcal{S}_+^{n\times n}$ with $B\bullet Z<0$, and such Z is called a *dual improving direction.*

For weakly infeasible LMIs, infeasibility cannot be demonstrated by a dual improving direction. However, an LMI is infeasible if and only if there exist approximate dual improving directions with arbitrarily small constraint violations. See, e.g., Lemma 2.6 in [32]. The next theorem gives an upper bound for the minimal norm of such approximate dual improving directions in the case of infeasibility.

THEOREM 3.5. *Let $\bar{\mathcal{A}}=\mathcal{A}+$ Img b. If $(B+\mathcal{A})\cap\mathcal{S}_+^{n\times n}=\emptyset$, then there exist $\{Y(\epsilon)\mid\epsilon>0\}$ such that for all $0<\epsilon\le 1$, it holds that*

$$\mathrm{dist}(Y(\epsilon),\mathcal{A}^\perp)=O(\epsilon),\qquad B\bullet Y(\epsilon)<-1+O(\epsilon),\qquad \lambda_{\min}(Y(\epsilon))\ge-\epsilon,$$

and

$$\|Y(\epsilon)\|=O(\epsilon^{1-2^{d(\bar{\mathcal{A}})}}).$$

Proof. Let $X^*\in$ relint $(\bar{\mathcal{A}}\cap\mathcal{S}_+^{n\times n})$, and suppose without loss of generality that X^* is of the form (2.6). Using the same 2×2 partition as in (2.6), it follows from Lemma 2.3 that

$$\mathrm{dist}\left(B+\mathcal{A},\ \mathrm{face}\left(\mathcal{S}_+^{n\times n},\begin{bmatrix}0&0\\0&I\end{bmatrix}\right)\right)>0.$$

Applying (3.10), it thus follows that there exists a matrix $Y^{(0)}$ such that

$$B \bullet Y^{(0)} < -1, \quad Y^{(0)} \in \mathcal{A}^{\perp} \cap \text{ face}\left(\mathcal{S}_{+}^{n\times n}, \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}\right)^{*}. \tag{3.11}$$

Partitioning $Y^{(0)}$, we have

$$Y^{(0)} = \begin{bmatrix} Y_B^{(0)} & Y_U^{(0)} \\ (Y_U^{(0)})^{\mathrm{T}} & Y_N^{(0)} \end{bmatrix}, \quad Y_B^{(0)} \succeq 0. \tag{3.12}$$

Assume that a suitable basis transformation has taken place, so that the partition in (3.8) is valid for all $k = 1, 2, \ldots, d(\bar{\mathcal{A}})$. We shall now construct $\{Y^{(k)} \mid k = 0, 1, \ldots, d(\bar{\mathcal{A}})\}$ such that

$$\begin{cases} \|Y^{(k)}\| = O(\epsilon^{1-2^k}), \\ Y^{(k)}(1:k+1, 1:k+1) \succeq 0, \\ B \bullet Y^{(k)} < -1 + O(\epsilon), \\ \operatorname{dist}(Y^{(k)}, \mathcal{A}^{\perp}) = O(\epsilon) \end{cases} \tag{3.13}$$

for $0 < \epsilon \leq 1$. Remark from (3.11)–(3.12) that (3.13) holds for $k = 0$. We will construct $Y^{(k)}$ for $k \in \{1, 2, \ldots, d(\bar{\mathcal{A}}) - 1\}$ in such a way that it satisfies (3.13), provided that $Y^{(k-1)}$ satisfies (3.13). We can then use induction.

Let

$$Y_t := Y^{(k-1)} + \epsilon I + t Z^{(d(\bar{\mathcal{A}})-k)}.$$

Since $Z^{(d(\bar{\mathcal{A}})-k)} \in \bar{\mathcal{A}}^{\perp} = \mathcal{A}^{\perp} \cap \text{ Ker } b^{\mathrm{T}}$, we immediately obtain from (3.13) that

$$B \bullet Y_t < -1 + O(\epsilon), \quad \operatorname{dist}(Y_t, \mathcal{A}^{\perp}) = O(\epsilon), \tag{3.14}$$

irrespective of t. Furthermore, since $Y^{(k-1)}(1:k, 1:k) \succeq 0$, it follows that $Y_t(1:k+1, 1:k+1)$ is positive semidefinite if and only if the Schur-complement

$$Y_t(k+1, k+1) - Y_t(1:k, k+1)^{\mathrm{T}} Y_t(1:k, 1:k)^{-1} Y_t(1:k, k+1)$$

is positive semidefinite. From (3.8) and the definition of Y_t, we have

$$Y_t(1:k, k+1) = Y^{(k-1)}(1:k, k+1),$$

and hence

$$\begin{aligned} & Y_t(k+1, k+1) - Y_t(1:k, k+1)^{\mathrm{T}} Y_t(1:k, 1:k)^{-1} Y_t(1:k, k+1) \\ & \succeq t\, Z^{(d(\bar{\mathcal{A}})-k)}(k+1, k+1) + Y^{(k-1)}(k+1, k+1) - \frac{1}{\epsilon}\|Y^{(k-1)}\|_2^2\, I. \end{aligned}$$

Thus, $Y_t(1:k+1, 1:k+1)$ is positive semidefinite if we choose t as

$$t = \frac{\|Y^{(k-1)}\|_2 + (\|Y^{(k-1)}\|_2^2/\epsilon)}{\lambda_{\min}(Z^{(d(\bar{\mathcal{A}})-k)}(k+1, k+1))} = O(\epsilon^{1-2^k}),$$

where we used that $\|Y^{(k-1)}\| = O(\epsilon^{1-2^{k-1}})$. Setting $Y^{(k)} = Y_t$, we obtain (3.13). The theorem follows by letting

$$Y(\epsilon) = Y^{(d(\bar{\mathcal{A}}))}. \qquad \square$$

We remark from the proof of Theorem 3.5 that the matrices $Y^{(0)}$ and $Z^{(k)}$, $k = 0, 1, \ldots, d(\bar{\mathcal{A}}) - 1$, provide a finite certificate of the infeasibility of the LMI. Together, these matrices form essentially a solution to the regularized Farkas-type dual of Ramana [28]; see also [15, 21]. Thus, the degree of singularity is the minimal number of layers that are needed in the perfect dual of Ramana. Equivalently, it is the number of regularization steps that is needed in the regularization scheme of Borwein and Wolkowicz [5, 29].

As discussed in the introduction, it is easy to calculate the backward error of an approximate solution. However, the error bound for the forward error of an LMI, as given in Theorem 3.3, does not only involve the backward error but also the degree of singularity. We will now provide some easily computable upper bounds on the degree of singularity.

LEMMA 3.6. *For the degree of singularity $d(\bar{\mathcal{A}})$ of a linear subspace $\bar{\mathcal{A}} \subseteq \mathcal{S}^{n\times n}$, it holds that*

$$d(\bar{\mathcal{A}}) \leq \min\{n-1,\ \dim \bar{\mathcal{A}},\ \dim \bar{\mathcal{A}}^{\perp}\}.$$

Proof. If $d(\bar{\mathcal{A}}) > 0$, then $\bar{\mathcal{A}} \cap \mathcal{S}_{+}^{n\times n} \neq \{0\}$, by definition of $d(\bar{\mathcal{A}})$. For this case, we have defined the $(d(\bar{\mathcal{A}})+1) \times (d(\bar{\mathcal{A}})+1)$ block partition (3.8), where each of the $d(\bar{\mathcal{A}})+1$ diagonal blocks is at least of size 1×1. Thus,

$$d(\bar{\mathcal{A}}) \leq n-1.$$

Furthermore, Lemma 3.1 defines a matrix $Z^{(k)} \in \bar{\mathcal{A}}^{\perp}$, for each regularization step $k \in \{0, 1, 2, \ldots, d(\bar{\mathcal{A}}) - 1\}$, and it is easily verified that these matrices are mutually independent. Therefore,

$$d(\bar{\mathcal{A}}) \leq \ \dim \bar{\mathcal{A}}^{\perp}.$$

Finally, using the $(d(\bar{\mathcal{A}})+1) \times (d(\bar{\mathcal{A}})+1)$ block partition (3.8), we claim that there exists $X^{(k)} \in \bar{\mathcal{A}}$ with

$$\left\{ \begin{array}{c} X^{(k)} = \left[\begin{array}{ccc} X^{(k)}(1:k, 1:k) & X^{(k)}(1:k, k+1) & 0 \\ X^{(k)}(1:k, k+1)^{\mathrm{T}} & 0 & 0 \\ 0 & 0 & 0 \end{array} \right], \\ X^{(k)}(1:k, 1:k) \succ 0. \end{array} \right.$$

Namely, if such $X^{(k)}$ does not exist, then by (2.3), there must exist $\Delta Z \in \bar{\mathcal{A}}^{\perp}$ such that

$$\left\{ \begin{array}{c} \Delta Z(1:k+1, 1:k+1) = \left[\begin{array}{cc} \Delta Z(1:k, 1:k) & 0 \\ 0 & \Delta Z(k,k) \end{array} \right], \\ 0 \neq \Delta Z(1:k, 1:k) \succeq 0, \end{array} \right.$$

and this contradicts the fact that $Z^{(d(\bar{\mathcal{A}})-k)}(1:k+1, 1:k+1)$ is of maximal rank; see its definition in Lemma 3.1. Again, it is easy to see that the matrices $X^{(k)} \in \bar{\mathcal{A}}$, $k = 1, 2, \ldots, d(\bar{\mathcal{A}})$, are mutually independent, and hence $d(\bar{\mathcal{A}}) \leq \ \dim \bar{\mathcal{A}}$. □

The bounds of Theorems 3.3 and 3.4 quickly become unattractive as the singularity degree increases. However, the next two examples show that these bounds can be tight. This means that problems with a large degree of singularity can be very hard to solve numerically.

Example 2. Consider the LMI

$$\left\{\begin{array}{c} x_{22}=0, \\ x_{k+1,k+1}=x_{1,k} \text{ for } k=2,3,\ldots,n-1, \\ X\in\mathcal{S}_+^{n\times n}. \end{array}\right.$$

Due to the restriction "$x_{22}=0$" and the positive semidefiniteness, we have $0=x_{12}=x_{33}$, which further implies $0=x_{13}=x_{44}$, and so on. With an inductive argument, we have $x_{1,k}=0$ for all $k=2,3,\ldots,n$. However, we can construct a sequence $\{X(\epsilon)\mid\epsilon>0\}$ with a constraint violation ϵ, but $x_{1,n}=\epsilon^{1/2^{n-1}}$, namely,

$$X(\epsilon)=\begin{bmatrix} n & \epsilon^{1/2^1} & \epsilon^{1/2^2} & \cdots & \epsilon^{1/2^{n-1}} \\ \epsilon^{1/2^1} & \epsilon & 0 & \cdots & 0 \\ \epsilon^{1/2^2} & 0 & \epsilon^{1/2^1} & & \\ \vdots & \vdots & & \ddots & \\ \epsilon^{1/2^{n-1}} & 0 & & & \epsilon^{1/2^{n-2}} \end{bmatrix}.$$

Notice that "$x_{22}=0$" is the only constraint that is violated by $X(\epsilon)$.

To see how unfortunate this example is, consider a backward error $\epsilon=10^{-8}$. Then, already for $n=25$, we have $x_{1,25}(10^{-8})>.99999$, whereas $\hat{x}_{1,25}=0$ for any solution $\hat{X}$ of the LMI.

Example 3. Extending Example 2 with the restriction "$x_{1,n}=1$," we obtain a (weakly) infeasible LMI:

$$\left\{\begin{array}{c} x_{22}=0, \quad x_{1,n}=1, \\ x_{k+1,k+1}=x_{1,k} \text{ for } k=2,3,\ldots,n-1, \\ X\in\mathcal{S}_+^{n\times n}. \end{array}\right.$$

However, we may construct a sequence $\{X(\epsilon)\mid\epsilon>0\}$ with constraint violation ϵ and $\|X(\epsilon)\|=O(1/\epsilon^{1/(2^{n-1}-1)})$. Namely, we let

$$\left\{\begin{array}{c} x_{11}(\epsilon)=n/\epsilon^{\alpha} \text{ with } \alpha:=1/(2^{n-1}-1), \\ x_{22}(\epsilon)=\epsilon, \\ x_{1,n}(\epsilon)=1, \\ x_{k+1,k+1}(\epsilon)=x_{1,k}(\epsilon)=\epsilon^{\beta} \text{ with } \beta=\left(2^{n-1-k}-1\right)/\left(2^{n-1}-1\right), \end{array}\right.$$

where $k\in\{2,3,\ldots,n-1\}$.

This example shows that (in)feasibility can be hard to detect. Namely, for $n=10$ and a backward error $\epsilon=10^{-8}$, we have $\|X(10^{-8})\|_2<11$, which is not unusually large; yet, the problem is infeasible.

4. Application to semidefinite programming.

4. Application to semidefinite programming. Error bounds for LMIs can be applied to semidefinite optimization models as well. A standard form semidefinite program is

$$\text{(P)} \qquad \min\{C\bullet X\mid X\in(B+\mathcal{A})\cap\mathcal{S}_+^{n\times n}\},$$

where B and C are given symmetric matrices. Associated with this optimization problem is a dual problem, namely,

$$\text{(D)} \qquad \min\{B\bullet Z\mid Z\in(C+\mathcal{A}^{\perp})\cap\mathcal{S}_+^{n\times n}\}.$$

An obvious property of the primal-dual pair (P) and (D) is the weak duality relation. Namely, if $X \in (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}$ and $Z \in (C + \mathcal{A}^\perp) \cap \mathcal{S}_+^{n\times n}$, then

$$0 \le X \bullet Z = C \bullet X + B \bullet Z - B \bullet C. \tag{4.1}$$

Clearly, if $X \bullet Z = 0$, then X and Z must be optimal solutions to (P) and (D), respectively; we say then that (X, Z) is a pair of complementary solutions. In general, such a pair may not exist, even if both (P) and (D) are feasible. (We say that (P) is feasible if $(B+\mathcal{A})\cap\mathcal{S}_+^{n\times n} \ne \emptyset$ and (D) is feasible if $(C+\mathcal{A}^\perp)\cap\mathcal{S}_+^{n\times n} \ne \emptyset$.) A sufficient condition for the existence of a complementary solution pair is that (P) and (D) are feasible and satisfy the primal-dual Slater condition, in which case $d(\mathcal{A} + \text{ Img } b) = d(\mathcal{A}^\perp + \text{ Img } c) = 0$.

Based on (4.1), we can formulate the set of complementary solutions as the LMI

$$\begin{cases} C \bullet X + B \bullet Z = B \bullet C, \\ X \in B + \mathcal{A}, \quad Z \in C + \mathcal{A}^\perp, \\ X \succeq 0, \quad Z \succeq 0. \end{cases}$$

In principle, we can apply our error bound results for LMIs directly to the above system. However, tighter bounds can be obtained by exploring its special structure.

Consider a bounded trajectory of approximate primal-dual solutions

$$\{(X(\epsilon), Z(\epsilon)) \mid \epsilon > 0\},$$

satisfying

$$\begin{cases} \operatorname{dist}(X(\epsilon), B + \mathcal{A}) \le \epsilon, \quad \operatorname{dist}(Z(\epsilon), C + \mathcal{A}^\perp) \le \epsilon, \\ \lambda_{\min}(X(\epsilon)) \ge -\epsilon, \quad \lambda_{\min}(Z(\epsilon)) \ge -\epsilon, \\ X(\epsilon) \bullet Z(\epsilon) \le \epsilon. \end{cases} \tag{4.2}$$

Let $(\bar{B}, \bar{C})$ be a complementary solution pair, i.e.,

$$\bar{B} \bullet \bar{C} = 0, \quad \bar{B} \in (B + \mathcal{A}) \cap \mathcal{S}_+^{n\times n}, \quad \bar{C} \in (C + \mathcal{A}^\perp) \cap \mathcal{S}_+^{n\times n}.$$

Such a pair must exist, since in particular any cluster point of $\{(X(\epsilon), Z(\epsilon) \mid \epsilon > 0\}$ for $\epsilon \downarrow 0$ is a complementary solution pair. Notice that $B + \mathcal{A} = \bar{B} + \mathcal{A}$ and similarly $C + \mathcal{A}^\perp = \bar{C} + \mathcal{A}^\perp$, from which we easily derive that

$$X \bullet Z = \bar{C} \bullet X + \bar{B} \bullet Z,$$

for feasible solutions X and Z, and

$$[\bar{C} \bullet X(\epsilon)]_+ = O(\epsilon), \quad [\bar{B} \bullet Z(\epsilon)]_+ = O(\epsilon)$$

for $(X(\epsilon), Z(\epsilon))$ satisfying (4.2). This means that $X(\epsilon)$ has an $O(\epsilon)$ constraint violation with respect to the LMI

$$\begin{cases} X \in \bar{B} + \mathcal{A}, \\ \bar{C} \bullet X \le 0, \\ X \succeq 0. \end{cases} \tag{4.3}$$

Notice that (4.3) describes the set of optimal solutions to (P). If we let

$$\bar{\mathcal{A}} := \text{ Img } \bar{b} + (\mathcal{A} \cap \text{ Ker } \bar{c}^{\mathrm{T}}), \tag{4.4}$$

then Theorems 3.3 and 3.4 are applicable to the LMI (4.3) and hence to the semidefinite program (P). Specifically, given a bounded trajectory $\{X(\epsilon), Z(\epsilon) \mid \epsilon > 0\}$ satisfying (4.2), we know that the distance from $X(\epsilon)$ to the set of optimal solutions to (P) is $O(\epsilon^{2^{-d(\bar{\mathcal{A}})}})$, where $d(\bar{\mathcal{A}})$ is the degree of singularity of the linear subspace defined in (4.4).

Since $\bar{B} \bullet \bar{C} = 0$, we can move the parentheses in definition (4.4) to get

$$\bar{\mathcal{A}} = (\text{ Img } \bar{b} + \mathcal{A}) \cap \text{ Ker } \bar{c}^{\mathrm{T}},$$

from which we get

$$\bar{\mathcal{A}}^{\perp} = \text{ Img } \bar{c} + (\mathcal{A}^{\perp} \cap \text{ Ker } \bar{b}^{\mathrm{T}}).$$

Noticing the primal-dual symmetry, we conclude that the distance from $Z(\epsilon)$ to the set of optimal solutions to (D) is $O(\epsilon^{2^{-d(\bar{\mathcal{A}}^{\perp})}})$, where $d(\bar{\mathcal{A}}^{\perp})$ is the degree of singularity of $\bar{\mathcal{A}}^{\perp}$.

5. Concluding remarks. Theorem 3.3 provides a Hölderian error bound for LMIs. For weakly infeasible LMIs, we have derived relations between backward errors and the size of approximate solutions; see Theorems 3.4 and 3.5. In section 4, we applied the error bound of Theorem 3.3 to semidefinite programming problems (SDPs). If the SDP has a strictly complementary solution, then its degree of singularity can be at most 1, and the bound becomes

$$\text{forward error} = O(\sqrt{\text{backward error}}).$$

For this case, Luo, Sturm, and Zhang [22] obtained a Lipschitzian error bound if the approximate solutions $(X(\epsilon), Z(\epsilon))$ are restricted to the central path. The sensitivity of central solutions with respect to perturbations in the right-hand side was studied by Sturm and Zhang [33].

Acknowledgments. Tom Luo's comments on an earlier version of this paper have resulted in substantial improvements in the presentation. Two anonymous referees and the associate editor, M. J. Todd, have assisted me in eliminating several sources of confusion. They have also stimulated me to extend the discussion in section 1.

REFERENCES

[1] F. Alizadeh, J.A. Haeberly, and M.L. Overton, *Primal-dual interior-point methods for semidefinite programming: Convergence rates, stability and numerical results*, SIAM J. Optim., 8 (1998), pp. 746–768.

[2] G. Barker and D. Carlson, *Cones of diagonally dominant matrices*, Pacific J. Math., 57 (1975), pp. 14–32.

[3] F. Bohnenblust, *Joint Positiveness of Matrices*, Tech. report, University of California, Los Angeles, CA, 1948.

[4] J.F. Bonnans and A. Shapiro, *Optimization problems with perturbations: A guided tour*, SIAM Rev., 40 (1998), pp. 228–264.

[5] J. Borwein and H. Wolkowicz, *Regularizing the abstract convex program*, J. Math. Anal. Appl., 83 (1981), pp. 495–530.

[6] S. Boyd, L.E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*, SIAM Stud. Appl. Math. 15, SIAM, Philadelphia, 1994.

[7] S. Deng and H. Hu, *Computable Error Bounds for Semidefinite Programming*, Tech. report, Department of Mathematical Sciences, Northern Illinois University, DeKalb, IL, 1996.

[8] D. Goldfarb and K. Scheinberg, *On parametric semidefinite programming*, Appl. Numer. Math., 29 (1999), pp. 361–377.

[9] C. Helmberg, F. Rendl, R.J. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[10] C. Helmberg, *Fixing Variables in Semidefinite Relaxations*, Tech. Report SC-gb-43, Konrad–Zuse–Centrum, Berlin, Germany, 1996.

[11] A. Hoffman, *On approximate solutions of systems of linear inequalities*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 263–265.

[12] D. Klatte and W. Li, *Asymptotic Constraint Qualifications and Global Error Bounds for Convex Inequalities*, Tech. report, Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA, 1996.

[13] E. de Klerk, *Interior Point Methods for Semidefinite Programming*, Ph.D. thesis, Faculty of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands, 1997.

[14] M. Kojima, S. Shindoh, and S. Hara, *Interior-point methods for the monotone semidefinite linear complementarity problem in symmetric matrices*, SIAM J. Optim., 7 (1997), pp. 86–125.

[15] K. Kortanek and Q. Zhang, *Perfect Duality in Semi–Infinite and Semidefinite Programming*, Tech. report, Department of Management Sciences, University of Iowa, Iowa City, IA, 1997.

[16] A. Lewis, *Eigenvalue–constrained faces*, Linear Algebra Appl., 269 (1998), pp. 159–181.

[17] X.-D. Luo and Z.-Q. Luo, *Extension of Hoffman's error bound to polynomial systems*, SIAM J. Optim., 4 (1994), pp. 383–392.

[18] Z.-Q. Luo and J.-S. Pang, *Error bounds for analytic systems and their applications*, Math. Programming, 67 (1994), pp. 1–28.

[19] Z.-Q. Luo and J. Sturm, *Error bounds for quadratic systems*, in High Performance Optimization, J. Frenk, C. Roos, T. Terlaky, and S. Zhang, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 383–404.

[20] Z.-Q. Luo, J. Sturm, and S. Zhang, *Duality and Self-Duality for Conic Convex Programming*, Tech. Report 9620/A, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 1996.

[21] Z.-Q. Luo, J. Sturm, and S. Zhang, *Duality results for conic convex programming*, Tech. Report 9719/A, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 1997.

[22] Z.-Q. Luo, J.F. Sturm, and S. Zhang, *Superlinear convergence of a symmetric primal-dual path following algorithm for semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 59–81.

[23] Y. Nesterov and A. Nemirovskii, *Interior point polynomial algorithms in convex programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.

[24] Y. Nesterov and M. Todd, *Self–scaled barriers and interior–point methods for convex programming*, Math. Oper. Res., 22 (1997), pp. 1–42.

[25] Y.E. Nesterov and M.J. Todd, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

[26] J. Pang, *Error bounds in mathematical programming*, Math. Programming, 97 (1998), pp. 299–332.

[27] G. Pataki, *On the rank of extreme matrices in semidefinite programs and the multiplicity of optimal eigenvalues*, Math. Oper. Res., 23 (1998), pp. 339–358.

[28] M. Ramana, *An exact duality theory for semidefinite programming and its complexity implications*, Math. Programming, 77 (1997), pp. 129–162.

[29] M.V. Ramana, L. Tunçel, and H. Wolkowicz, *Strong duality for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 641–662.

[30] R.T. Rockafellar, *Conjugate Duality and Optimization*, CBMS-NSF Regional Conf. Ser. Appl. Math., SIAM, Philadelphia, 1974.

[31] A. Shapiro, *First and second order analysis of nonlinear semidefinite programs*, Math. Programming, 77 (1997), pp. 301–320.

[32] J. Sturm, *Primal–Dual Interior Point Approach to Semidefinite Programming*, Tinbergen Institute Research Series 156, Thesis Publishers, Amsterdam, The Netherlands, 1997.

[33] J. Sturm and S. Zhang, *On Sensitivity of Central Solutions in Semidefinite Programming*, Tech. Report 9813/A, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 1998.

[34] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[35] T. Wang and J. Pang, *Global error bounds for convex quadratic inequality systems*, Optimization, 31 (1994), pp. 1–12.

[36] Y. Ye, M. Todd, and S. Mizuno, *An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm*, Math. Oper. Res., 19 (1994), pp. 53–67.