SIAM J. OPTIM.
Vol. 11, No. 1, pp. 1–30

PROVABLY GOOD GLOBAL ROUTING OF INTEGRATED CIRCUITS*

T. LENGAUER† AND M. LÜGERING‡

Abstract. This paper investigates the global routing problem for integrated circuits. We introduce a formulation on the basis of integer programming which minimizes the routing area among a limited set of Steiner trees for each net. Indeed, the involved cost function depends on the channel density of the routing which has a direct influence on the routing area.

Our methods for solving the global routing problem employ local search heuristics, sequential routing, genetic routing, and randomized procedures. Our methods for computing lower bounds are based on linear and Lagrange relaxation. An analysis on the tightness of the bounds indicates that the difference between the cost of the optimal integer solutions and the cost of the optimal fractional solutions is only a small number of tracks in practice. Moreover, the analysis leads to the concept of linear preprocessing by which we exclude a large number of high-cost solutions.

We introduce several versions of preprocessing, one of which preserves the opportunity of obtaining a globally optimal solution in general; all of them do so in practice. Linear preprocessing enables us to solve problem instances with several thousand nets provably optimal or at least provably close to optimal.

All methods have been implemented in the software package ERIDANUS. We present computational results.

The global routing problem assumes the placement of the chip components to be fixed. An extension of the problem, which we call global layout of integrated circuits, allows the placement to be variable and searches for a placement that minimizes the routing area among a limited set of alternatives. We show that the results concerning the global routing problem can be easily extended to global layout.

Key words. routing, integer programming, integrated circuits, combinatorial optimization

AMS subject classifications. 90C10, 90C27, 05C85

PII. S1052623497331786

1. Problem definition.

Global routing is an essential part of the physical design of integrated circuits. The global routing phase usually follows the placement or floorplanning phase. During the global routing phase, the approximate course of all wires is determined. A detailed introduction to the global routing problem and its role in circuit layout can be found in [26].

Combinatorially, an instance of the global routing problem has the following three elements.

A routing graph. This undirected graph is denoted by $G = (V, E), |V| = k, |E| = m$. The routing graph is the representation of the routing regions on the chip that result from the preceding placement phase. In many applications, G is planar. We do not require this to be the case, however. Intuitively, the vertices of the routing graph represent possible positions of wire terminals, and the edges represent channels along or regions through which wiring can be performed. In order to support this intuition, each edge e in the routing

*Received by the editors December 23, 1997; accepted for publication (in revised form) July 30, 1999; published electronically July 25, 2000. This research was supported in part by grant Le 491/4 of Deutsche Forschungsgeweinschaft

http://www.siam.org/journals/siopt/11-1/33178.html

†German National Research Center for Information Technology, Schloß Birlinghoven, Sankt Augustin, Germany (thomas.lengauer@gmd.de). Department of Computer Science, University of Bonn, Bonn, Germany.

‡Department of Computer Science, University of Bonn, Bonn, Germany.

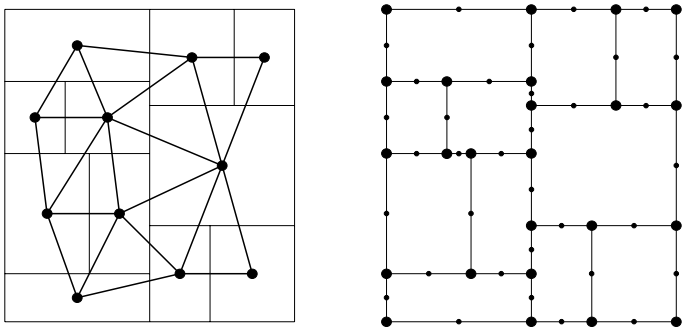


FIG. 1. *Two different wiring models.*

graph has two labels, its length $l(e) \geq 0$ and its capacity $c(e) \geq 0$. We assume capacities to be integer. Depending on the wiring model, the interpretation of G can take on different forms. Figure 1 depicts two different wiring models. In channelless routing the vertices are located at centers of the cells of the floorplan. The edges represent adjacencies between the cells. In this model, a reasonable choice of the labels for an edge $e = \{v, w\}$ is to choose $l(e)$ to be the actual length of e in the planar embedding shown in the figure—e.g., according to the Manhattan distance or the Euclidean distance—and $c(e)$ to be the length of the dual edge e' of e in the floorplan, in an appropriate unit. In this way, the capacity $c(e)$ measures the number of wires that can cross e' when moving from cell v to cell w. In the model of routing in channels, G represents the channel structure of the floorplan. In this case, the length of an edge should again be its actual length in the embedding. Its capacity should be an estimate of the width of the corresponding channel.

A set of nets. This multiset is denoted by $N \subseteq 2^V$ with $|N| = n$. Each net is specified by the set of its terminals, and each terminal is a vertex in V. Thus, in channelless routing, all terminals for a cell are clustered in the center of this cell. Using routing in channels, vertices can be added as desired to represent specific terminal positions. Of course, the same net can occur multiply, with each instance of the net being routed differently. Therefore, N is a multiset, in general. We denote a net by ν and different copies of ν by (ν, i), $i = 1, \ldots, k_\nu$. (k_ν is the multiplicity of net ν.) Furthermore, each net $(\nu, i) \in N$ has an integer weight $w(\nu, i) > 0$ that represents the cost of a unit-length wire for (ν, i). Often, all weights will be unity. Different weights can, for instance, model the different bit-width of buses that are represented by a single net each.

A set of admissible routes for each net. In principle, all Steiner trees between the vertices of (ν, i) may be admissible. In practice, however, the Steiner trees are restricted for technical reasons to having special features, e.g., a limited length or a limited number of bends. The admissible routes for net (ν, i) will be denoted by $T^1_{\nu,i}, \ldots, T^{I_{\nu,i}}_{\nu,i}$. ($I_{\nu,i}$ is the number of admissible routes for (ν, i).) The set of all admissible routes is denoted by

$$\mathcal{T} = (T^j_{\nu,i})_{1 \leq j \leq I_{\nu,i}, (\nu,i) \in N}$$

with $|\mathcal{T}| =: \xi$. Usually, the number $I_{\nu,i}$ of admissible routes for (ν, i) is large (exponential in the number of terminals of (ν, i)). But often, the set of admissible routes can be defined concisely in small space.

There are a multitude of variants of the global routing problem. In general, they fall into two classes, constrained global routing and unconstrained global routing. In the constrained global routing problem, the edge capacities in G are strictly adhered to; violations lead to illegal routings. In the unconstrained version, routings exceeding the edge capacities are allowed but punished in the cost function. The total wire length enters the cost measure with low priority in both versions of the global routing problem.

We now formally define the solutions of these two versions of the global routing problem. For this purpose, let $\mathcal{I} = (G, N, \mathcal{T})$ be the given problem instance.

Routing. A routing $\mathcal{R} = (T_{\nu,i})_{(\nu,i)\in N'}$ is a set of admissible routes $T_{\nu,i}$ for a subset N' of the nets in N. The nets in $N \setminus N'$ have no routes in $\mathcal{R}$. If $N \setminus N' \neq \emptyset$, the routing $\mathcal{R}$ is said to be incomplete.

Traffic. The traffic $U(\mathcal{R}, e)$ across an edge $e \in E$ in a routing $\mathcal{R}$ is the total weighted cost of all nets that are wired across e in $\mathcal{R}$, i.e.,

$$U(\mathcal{R}, e) := \sum_{\substack{(\nu,i)\in N' \\ e\in T_{\nu,i}}} w(\nu, i).$$

Load. The load $\Lambda(\mathcal{R}, e)$ of an edge $e \in E$ in a routing $\mathcal{R}$ is defined to be

$$\Lambda(\mathcal{R}, e) := U(\mathcal{R}, e) - c(e).$$

If $\Lambda(\mathcal{R}, e) < 0$, then the edge e is said to be unsaturated, if $\Lambda(\mathcal{R}, e) = 0$, then e is saturated, and if $\Lambda(\mathcal{R}, e) > 0$, then e is oversaturated. In the following we measure both traffic and load of edges in tracks. (In [26], the load is defined as $\Lambda(\mathcal{R}, e) := U(\mathcal{R}, e)/c(e)$. The difference between these two definitions is technical, but the definition given here is more attractive in many practical settings, because channels with many free tracks are avoided.)

Constrained global routing. A legal routing with respect to the constrained global routing problem is a routing $\mathcal{R}$ such that no edge $e \in E$ is oversaturated. The cost of a routing $\mathcal{R}$ is the pair

$$\left(\sum_{(\nu,i)\in N\setminus N'} w(\nu, i), W(\mathcal{R}) \right),$$

where

$$W(\mathcal{R}) := \sum_{e\in E} U(\mathcal{R}, e) \cdot l(e)$$

f is the total weighted wire length of the routing. Costs of different routings are compared using the lexicographic ordering. The number of routed nets is maximized with first priority. Among all routings that provide routes for a maximum number of nets, the one with the smallest total weighted wire length is chosen.

Unconstrained global routing. With respect to the unconstrained global routing problem, each complete routing is legal. The cost of a routing $\mathcal{R}$ is the pair

$$(\max_{e\in E} \Lambda(\mathcal{R}, e), W(\mathcal{R})).$$

Costs of different routings are again compared using the lexicographic ordering. Thus, an optimal routing is one with minimum maximal edge load and, among those, one with minimum total weighted wire length.

Both versions of the global routing problem are strongly NP-hard. In fact, this holds even for severely restricted cases of these problems.

- If $|N| = 1$, we obtain the minimum Steiner tree problem, which is strongly NP-hard [21].
- Kramer and van Leeuwen [25] have shown the restriction to be strongly NP-hard, in which all nets have exactly two terminals, all edge capacities are unity, all edge lengths are zero, all net weights are unity, and the routing graph is a square grid graph. (The restriction of the constrained global routing problem, in which all edge capacities are unity, all edge lengths are zero, and all net weights are unity, is also called Steiner tree packing.)
- Korte, Prömel, and Steger [24] have proved the Steiner tree packing problem to be strongly NP-hard even when $|N| = 2$ and G is planar.
- Karp et al. [22] have shown the restriction to be strongly NP-hard, in which only one-bend routes (of two-terminal nets) are admissible.

Three special cases of global routing problems come up particularly often in practice:

- $w(\nu, i) = 1$ for all $(\nu, i) \in N$. This case occurs if we do not route buses, but rather just single strand wires.
- $l(e) = 0$ for all $e \in E$. In this case we are not concerned with total weighted wire length.
- The routing graph is a partial or a complete grid graph.

By the results of Kramer and van Leeuwen [25], all these special cases are strongly NP-hard, even if only two-terminal nets are involved.

In the past, both versions of the global routing problem have received a great deal of attention within the CAD community. In the following we summarize concepts for solving the global routing problem.

2. Survey of solution methods.

2.1. Representation of routes. One critical point in the investigation of the global routing problem is the representation of routes. There are two approaches, in general. In the explicit approach to global routing, a special binary variable is chosen to represent each route. The corresponding integer program has a large number of variables and comparatively few constraints. The advantage of explicit global routing is that we can take technical restraints on the admissibility of routes into account. In the implicit approach to global routing, the legality of routes is implicitly secured by appropriate constraints involving variables that represent only nets and edges. More specifically, we generate a constraint for each cut in the routing graph that ensures that each net with terminals on either side of the cut is connected across the cut. The resulting integer program has few variables but a large number of constraints. Since routes are not represented by special variables, no complex technical restraints on the routes can be formulated in the implicit approach.

The explicit approach to global routing is rendered practical by preselecting a feasible number of routes to choose from. The selection of these routes turns out to be quite difficult. One idea is to compute minimum Steiner trees on the nets before global routing. But there is no clear indication that the consideration of minimum Steiner trees leads to low-density solutions; in fact, it can be shown that often the best global routing does not use minimum Steiner trees. A more advanced idea is to

use flow arguments in order to generate candidate routes. In this paper, we use the following scheme to produce routes. We construct a complete graph on the terminals of a net and compute a spanning tree on this graph. For every edge in this spanning tree we compute a path between the respective wire terminals which has at most one bend. There are at most two possibilities. A candidate route is the union of such paths. We call such routes one-bend routes. The number of one-bend routes for a net with r terminals is at most $r^{r-2} \cdot 2^{r-1}$. (Thus, the numbers of candidate routes for nets with between two and five terminals are limited by 2, 12, 128, and 2000, respectively.) Indeed, a complete enumeration of all one-bend routes is often possible in small space. Moreover, one-bend routes seem to be attractive in many practical settings. Note that by the result of Karp et al. [22] even this special case of the global routing problem is strongly NP-hard.

2.2. Sequential global routing. A whole host of algorithms for solving global routing are based on the idea of sequential global routing of the nets. In such an approach, the nets are first ordered according to a suitable notion of "difficulty to route." Then a Steiner tree is generated for each net in the prescribed order. The Steiner tree tries to avoid congested areas. For this purpose, edge load costs that direct the Steiner tree generation are maintained dynamically. Ting and Tien [36] presented an early version of this approach. More recent work is given by Chiang, Sarrafzadeh, and Wong [5], Chiang [6], and Chiang, Wong, and Sarrafzadeh [7]. Relying on sequential routing alone discriminates against the nets routed late. Therefore, *rip-up-and-reroute (RR)* strategies have been developed that iterate the sequential routing process; see [26].

2.3. Genetic global routing. Esbensen solves both the minimum Steiner tree problem [10] and the (unconstrained) global routing problem [9] using genetic algorithms. For global routing, the genotype of a specific solution consists of one entry for each net $(\nu, i) \in N$ which holds the index j of the route $T^j_{\nu,i}$ which is realized in that solution. In order to generate routes, Esbensen uses his own genetic algorithm for the minimum Steiner tree problem. Thus, his intention is to provide low-cost Steiner trees as candidate routes for the nets.

2.4. Hierarchical global routing. The method of hierarchical global routing solves the constrained global routing problem. This method was introduced by Burstein and Pelavin [2] in the context of gate array layout and was subsequently extended to general floorplans by Luk et al. [29]. A further improvement of this method is given by Heistermann and Lengauer [16]. Burstein and Hong [1] extend this approach to include placement in the context of gate array layout, and Lengauer and Müller [28] extended the approach for general floorplans. Hierarchical global routing methods use a cut tree for the floorplan. Such trees are computed by floorplanning methods based on circuit partitioning. A small global routing problem is associated with each node in the cut tree and solved exactly using integer programming methods. The partial solutions thus obtained are assembled to a solution of the whole problem instance. Whereas the solutions of the small global routing problems pertaining to the nodes in the cut tree are optimal, no statement about the quality of the resulting overall solutions can be made. However, experience indicates that the solutions are quite good in practice.

2.5. Routing by linear relaxation. The method of routing by linear relaxation considers the explicit approach to the unconstrained global routing problem. The first algorithm of this kind was introduced by Hu and Shing [19]. They solve

the linear relaxation of the integer programming formulation of the problem in order to obtain an initial (fractional) solution, and then they proceed to deduce an integer solution by selecting, for each net, the Steiner tree with the highest fractional value. In order to provide potential routes, Hu uses column generating techniques [18] which are based on flow arguments. Instead of solving the linear relaxation of global routing, Shragowitz and Keel [35] extend the algorithm of Hu and Shing [19] by formulating the linear relaxation of the routing problem as a multicommodity network flow problem. Vannelli [37] applies Karmarkar's linear programming algorithm [20], which is based on Khachian's ellipsoid method [23], to Hu and Shing's linear programming formulation. Karp et al. [22] suggest the idea of doing global routing by linear programming followed by randomized rounding, a technique discussed in detail by Ng, Raghavan, and Thompson [30], Raghavan [31], and Raghavan and Thompson [32, 33]. They use the fractional outcomes of the linear programming phase as biases for appropriate coin tosses that generate a solution in a randomized fashion. Repeating this random experiment an appropriate number of times generates a provably good solution with high probability. A worst case analysis of this method has been given in [32]. A deterministic version of the method has been presented in [31].

2.6. Multicommodity network flow. An extension of the idea of Karp, Raghavan, and others has been presented by Carden and Cheng [3] and Carden, Li, and Cheng [4]. In order to pursue the idea of representing the linear relaxation of global routing as a multicommodity network flow problem, they utilize Shahrokhi and Matula's algorithm [34] to derive a fractional flow solution. They extend this method to handle multiterminal Steiner trees instead of just shortest paths. Furthermore, they exhibit, at any stage, the error bound of the current result from an optimal solution of the linear programming formulation. Then they use a randomized rounding technique to derive a discrete net connection with an error bound on the derivation from the optimal fractional value.

2.7. Polyhedral algorithms. Grötschel, Martin, and Weismantel [11, 12, 13, 14, 15] studied the Steiner tree packing problem from a polyhedral point of view. They identify several classes of facet-defining inequalities and use them in order to perform branch-and-cut on the problem.

Lengauer and Lügering [27] discuss possibilities for defining both global routing and global layout as integer programs. Moreover, they discuss the related polyhedral theory. In contrast to the work of Grötschel et al. they observe that polyhedral algorithms for the unconstrained formulation are not really applicable because often the identification of a facet requires the knowledge of the optimal value of x_L. Moreover, sometimes an inequality of the form $x_L \geq \beta$ (where β is the optimal value of x_L) becomes facet defining.

3. Outline of the paper. Constrained global routing is the harder problem insofar as it is not sufficient to come up with just any routing; it has to be a legal one. Ensuring legality of the routings is quite a difficult issue, in general. So far, this problem has stood in the way of the use of randomized algorithms based on the linear relaxation of constrained global routing integer programs, for instance. In general, if we are looking for heuristic methods for finding good routings, it is very helpful not to have to deal with legality issues. Furthermore, most often the capacities that are attached to the edges of a routing graph are only rough estimates of routing cost. Even if the edge costs are precise, the solution of an unconstrained version of the global routing problem affords us with helpful information if no complete routing with

respect to the constrained version exists, e.g., the location of overcongested regions in a low-cost routing. The constrained global routing problem cannot produce such information. Thus, in our opinion, the unconstrained global routing problem is the more attractive version of the global routing problem.

In this paper we present a set of methods for solving large instances of the (unconstrained) global routing problem optimally or provably near optimally. Section 4 formulates global routing as an integer program and section 5 proposes methods for solving global routing. In section 6 we investigate lower bounds on the optimum cost of the underlying integer program. Section 6.4 gives an analysis on the tightness of the bounds. The results of this section suggest applying a linear preprocessing, which is discussed in section 7. In section 8 we combine the results from the previous sections in order to present overall strategies for solving the global routing problem. Section 9 extends the results to the global layout problem, and, finally, section 10 gives conclusions.

4. The integer program. We follow the explicit approach and provide for each candidate route $T^j_{\nu,i}$ for net $(\nu,i) \in N$ its own variable $x_{\nu,i,j}$ with the intention that

$$x_{\nu,i,j} := \begin{cases} 1 & \text{if net } (\nu,i) \text{ takes route } T^j_{\nu,i}, \\ 0 & \text{otherwise.} \end{cases}$$

In addition, we provide a lid variable x_L that measures the maximum load over all edges. Then, a solution of the integer program for global routing has the form

$$x := \left((x_{\nu,i,j})_{(\nu,i)\in N, T^j_{\nu,i}}, x_L \right).$$

For the remainder of the paper we disregard total weighted wire length. Thus, we set $l(e) = 0$ for all $e \in E$. Furthermore, we do not consider different net weights, i.e., we set $w(\nu,i) = 1$ for all $(\nu,i) \in N$. The traffic of an edge $e \in E$ is computed as follows:

$$U(x,e) := \sum_{(\nu,i)\in N} \sum_{j \mid e \in T^j_{\nu,i}} x_{\nu,i,j}.$$

The integer program for the global routing problem has completeness constraints that ensure that every net is routed by exactly one route and load constraints that ensure that the variable x_L represents the maximum edge load. In addition, the variables of the form $x_{\nu,i,j}$ have to take on binary values. The resulting integer program is the following.

DEFINITION 1 (the integer program for global routing).

$$(IP^e): \quad \min_x \quad x_L$$

such that (s.t.)

$$\sum_{j=1}^{I_{\nu,i}} x_{\nu,i,j} = 1 \text{ for all } (\nu,i) \in N \tag{1}$$

(completeness constraints),

$$U(x,e) - x_L \le c(e) \text{ for all } e \in E \tag{2}$$

(load constraints),

$$x_{\nu,i,j} \in \{0,1\} \text{ for all } (\nu,i) \in N, 1 \le j \le I_{\nu,i} \tag{3}$$

(integrality constraints).

The cost $\vartheta(\check{x})$ of an optimal solution $\check{x}$ of IP^e is denoted by $\vartheta(IP^e)$.

In the following we often do not distinguish between a problem instance and the related integer program.

5. Upper bounds. All proposed methods have been implemented in the software package ERIDANUS using the C programming language under SOLARIS 2.5.1 on a Sparc Ultra-Enterprise. In our experiments we used several of the benchmarks from the Microelectronics Center of North Carolina (MCNC) and benchmarks from other authors. All of these examples are small enough to be solved exactly by our methods. Therefore the quality of the resulting routing depends only on the choice of routing variants, which we regard as being outside the scope of this paper. This is also one of the reasons why we do not present comparisons of chip area on the MCNC benchmarks with other authors. A second reason is that we solve only part of the physical design problem, excluding placement and detailed routing.

For a thorough assessment of ERIDANUS, we have generated a large number of gate arrays of our own, covering a large range of problem sizes. In this way, we are able to rate how the suggested methods scale up with problem size, especially with an increasing number of nets. We observed that the quality of the results does not depend on the origin of the examples. Therefore, in this paper, we focus our attention on two of the examples we constructed. While both examples are constructed on the same grid size, namely 15 rows and 20 columns, they differ in the number of nets by a factor of 16. The first example is named MOZART-M05 and has 1600 nets while the second, denoted as MOZART-M09, has 25600 nets. For both examples, all edge capacities are set to zero. Furthermore, the number of terminals lies between two and five for each net. In order to generate routes, we applied a one-bend route computation on each net which comes up with an average of 21.5 routes per net for MOZART-M05 and 22.2 routes per net for MOZART-M09.

We define the problem size $Z(IP^e)$ of an instance of global routing as the number of nonzero entries in the coefficient matrix of IP^e. Thus, we have

$$Z(IP^e) = O(\xi \cdot q + m),$$

where q is the average number of edges taken by an arbitrary route.

5.1. Local search. In order to apply local search to the global routing problem we introduce the following neighborhood relationship on the set of routings.

DEFINITION 2 (SNE-neighborhood). *Let x' and x'' be two arbitrary solutions of the same instance of the global routing problem.*

$$x' \sim^1 x'' :\Longleftrightarrow x' \text{ and } x'' \text{ differ in the route of exactly one net}$$

$\sim^1$ is called single-net-exchange(SNE)-neighborhood.

In [27] we proved that $\sim^1$ is a subset of the convex neighborhood of the underlying polytope.

For the purposes of local search we refine the cost function, which in the integer program consists solely of the variable x_L. Here, we count the number of edges with maximum load with second priority. Thus, of two solutions x' and x'' with $x'_L = x''_L$,

the solution with fewer maximum-loaded edges has lower cost. This improves the performance of local search significantly.

The conventional local search (*LS*) method, which iterates to better solutions in the neighborhood, has runtime (in the worst case as well as in practice)

$$t(LS) = O(\alpha \cdot \xi \cdot q + m),$$

where α is the maximum number of times every net is rerouted. In the following we assume that α is bounded by a (small) constant. We observe that, while this condition is not true in general, our experience shows that it appropriately reflects the performance of the algorithm in practice. Thus, we assume that the method has a runtime which grows linear in the problem size.

In addition to the conventional method we introduce a more refined procedure which we call swinging search (*SS*), and which is similar to the well-known simulated annealing method. The swinging search heuristic randomly adds a number, the so-called vibration, to the capacity of an edge. Vibrations differ from edge to edge and lie between zero and the maximum vibration which decreases with increasing runtime, until the algorithm behaves like conventional local search. The starting value of the maximum vibration is set to n and decreases exponentially down to zero (integer arithmetic). Thus, assuming α to be constant for every involved local search, we obtain a runtime of

$$t(SS) = O(\log n \cdot \xi \cdot q + m).$$

In our computational experiments we apply the conventional local search 100 times and the swinging method 10 times. Moreover, we generate 100 random solutions (*TR*). The results are shown in Table 1. For every method, the runtime is given in the form *hh:mm:ss* and the number of tracks of the best solution found is presented.

5.2. Sequential routing. A straightforward implementation of the classical sequential routing (*SR*) procedure does not yield satisfactory results. Indeed, the cost of the solutions lies somewhere midway between the cost of locally optimal solutions (according to $\sim^1$) and random solutions. This is because sequential routing does not provide locally optimal solutions according to the SNE-neighborhood, in general. The runtime of this method (in the worst case as well as in practice) is linear in the problem size

$$t(SR) = O(\xi \cdot q + m).$$

In order to improve the method, we added an *RR* strategy which is tantamount to applying a conventional local search after each iteration of sequential routing that increases the value of x_L. The runtime of the *RR* heuristic (assuming α to be constant for every involved local search) is

$$t(RR) = O(\kappa \cdot \xi \cdot q + m),$$

where κ is the maximum load achieved by an edge. (κ does not grow as fast as n in practice.) In our computational experiments we apply the *RR* method once. The results are shown in Table 1.

TABLE 1
The upper bound results.

		MOZART-M05	MOZART-M09
TR	*t*	0:00:01	0:00:13
	UB	112	1871
LS	*t*	0:00:10	0:06:14
	UB	65	1065
SS	*t*	0:00:15	0:18:34
	UB	62	1013
RR	*t*	0:00:08	0:40:37
	UB	60	989
MD	*t*	0:01:07	0:20:15
	UB	62	1033
RD	*t*	0:03:09	5:23:57
	UB	59	971

5.3. Genetic routing. The application of genetic algorithms [17] to global routing is taken from Esbensen [9]. It provides genotypes which consist of one entry for each net which holds the index of the route which is realized for that net in the respective solution. We have found that conventional genetic routing (GR) produces results that are only slightly better than random solutions. The runtime of this method (in the worst case as well as in practice) is

$$t(GR) = O(g \cdot p \cdot n \cdot q + g \cdot p \cdot m + g \cdot p \cdot \log p),$$

where g is the number of generations and p the size of the population. (The term $g \cdot p \cdot \log p$ results from the required sorting of the individuals in each generation.)

In order to improve the method, we added a mutation-driven (MD) strategy which applies conventional local search to every newly generated solution. The runtime of the *MD* heuristic (assuming α to be constant for every involved local search) is

$$t(MD) = O(g \cdot p \cdot \xi \cdot q + g \cdot p \cdot m + g \cdot p \cdot \log p).$$

In our experiments we apply the *MD* method once with $g = 10$ and $p = 128$. The results are shown in Table 1.

5.4. Randomized rounding. The idea of randomized rounding (RD) is taken from Ng, Raghavan, and Thompson [30], Raghavan [31], and Raghavan and Thompson [32, 33]. It solves the linear relaxation LP^e of the underlying integer program IP^e and sets the fractional outcomes to integer values, in a randomized fashion. Here, the probability that a specific route is realized is exactly its fractional value. The runtime of this method is dominated by the solution of the linear program LP^e:

$$t(RD) = t(LP^e) + O(\xi + n \cdot q + m).$$

In our computational experiments we apply the algorithm with 100 trials of rounding. The results are shown in Table 1.

The quality of the solutions of the *RD* procedure is superior to the quality of the solutions of all other methods. The reason for this will be discussed later.

6. Lower bounds.

6.1. Linear relaxation. As mentioned already, we denote the linear relaxation of the integer program IP^e with LP^e. The cost $\vartheta(\bar{x})$ of an optimal solution $\bar{x}$ of LP^e is denoted with $\vartheta(LP^e)$.

Computational experience shows that the solution of the linear relaxation dominates the runtime for large problem instances. In our experiments we use the commercial package CPLEX in order to solve linear programs by the simplex algorithm as well as integer linear programs by branch-and-bound. Table 2 shows the computational results for the solution of the linear relaxation in terms of the circuits MOZART-M05 and MOZART-M09. In order to apply branch-and-bound to large routing problems we first have to reduce the size of their integer programs. In section 7 we will show how to do this.

TABLE 2
The lower bound results.

		MOZART-M05	MOZART-M09
LP	*t*	0:03:08	5:23:43
	LB	57.5174	969.326
SO	*t*	0:00:51	0:32:35
	LB	57.5101	969.303

6.2. Duality. The dual of the linear relaxation, or, for short, the linear dual DP^e of the global routing integer program has the following shape.

DEFINITION 3 (the linear dual for global routing).

$$(DP^e): \quad \max_{\pi} \sum_{(\nu,i)\in N} \pi_{\nu,i} - \sum_{e\in E} c(e)\cdot\pi_e$$

s.t.

$$\pi_{\nu,i} \le \sum_{e\in T^j_{\nu,i}} \pi_e \text{ for all } (\nu,i)\in N, 1\le j\le I_{\nu,i}, \tag{4}$$

$$\sum_{e\in E} \pi_e = 1, \tag{5}$$

$$\pi_e \ge 0 \text{ for all } e\in E. \tag{6}$$

The cost $\vartheta(\pi)$ of an optimal solution π of DP^e is denoted by $\vartheta(DP^e)$.

For some combinatorial optimization problems it is easier to solve the dual than the original (primal) program. This does not hold for global routing, however. The reason we introduce duality is that it affords structural insight into the global routing problem. Applied to global routing the complementary slackness conditions take on the following form.

LEMMA 4 (the complementary slackness conditions for global routing). *Let $\bar{x}$ be a solution of LP^e and π be a solution of DP^e. $\bar{x}$ and π are both optimal if and only if*

$$\pi_e\cdot(\bar{x}_L - U(\bar{x},e) + c(e)) = 0 \text{ for all } e\in E, \tag{7}$$

$$\left(\sum_{e\in T^j_{\nu,i}} \pi_e - \pi_{\nu,i}\right)\cdot \bar{x}_{\nu,i,j} = 0 \text{ for all } (\nu,i)\in N, 1\le j\le I_{\nu,i}. \tag{8}$$

6.3. Lagrange relaxation. In order to apply Lagrange relaxation to global routing, we relax the load constraints. The remaining coefficient matrix turns out to be totally unimodular. We obtain the following Lagrange dual LD^e.

DEFINITION 5 (the Lagrange dual for global routing).

$$(LD^e) : \max_{\lambda \geq 0} \quad \min_{x} \quad x_L + \sum_{e \in E} (U(x,e) - c(e) - x_L) \cdot \lambda_e$$

s.t.

$$\sum_{j=1}^{I_{\nu,i}} x_{\nu,i,j} = 1 \text{ for all } (\nu,i) \in N, \tag{9}$$

$$x_{\nu,i,j} \in \{0,1\} \text{ for all } (\nu,i) \in N, 1 \leq j \leq I_{\nu,i}. \tag{10}$$

The cost $\vartheta(\lambda)$ of an optimal solution λ of LD^e is denoted by $\vartheta(LD^e)$. The Lagrange program $LD^e(\lambda)$ is the Lagrange dual for a fixed λ.

Since the linear relaxation for global routing has a finite optimum and the polytope related to the Lagrange dual is integral we obtain the following.

THEOREM 6.

$$\vartheta(LP^e) = \vartheta(DP^e) = \vartheta(LD^e).$$

The following result relates the linear dual and the Lagrange dual.

THEOREM 7.

(a) *For every optimal solution π of DP^e there is an optimal solution λ of LD^e s.t.*

$$\lambda_e = \pi_e \text{ for all } e \in E.$$

(b) *For every optimal solution λ of LD^e there is an optimal solution π of DP^e s.t.*

$$\pi_e = \lambda_e \text{ for all } e \in E$$

and

$$\pi_{\nu,i} = \min_{1 \leq j \leq I_{\nu,i}} \sum_{e \in T^j_{\nu,i}} \lambda_e \text{ for all } (\nu,i) \in N.$$

Proof. Consider the cost function of the Lagrange dual LD^e

$$\max_{\lambda \geq 0} \quad \min_{x} \quad x_L + \sum_{e \in E} (U(x,e) - c(e) - x_L) \cdot \lambda_e.$$

After a trivial transformation we obtain

$$\max_{\lambda \geq 0} \quad \min_{x} \quad \left(1 - \sum_{e \in E} \lambda_e\right) \cdot x_L + \sum_{e \in E} (U(x,e) - c(e)) \cdot \lambda_e.$$

Since x_L is not restricted, in an optimal assignment to λ we must have

$$\sum_{e \in E} \lambda_e = 1. \tag{11}$$

This simplifies the cost function to

$$\max_{\lambda \geq 0} \quad \min_{x} \quad \sum_{e \in E} (U(x,e) - c(e)) \cdot \lambda_e.$$

After another trivial transformation we obtain

$$\max_{\lambda \geq 0} \quad \min_{x} \quad \sum_{e \in E} U(x,e) \cdot \lambda_e - \sum_{e \in E} c(e) \cdot \lambda_e.$$

After expanding the terms $U(x,e) \cdot \lambda_e$, we obtain the expression

$$\max_{\lambda \geq 0} \quad \min_{x} \sum_{(\nu,i) \in N} \sum_{1 \leq j \leq I_{\nu,i}} \left(\sum_{e \in T^j_{\nu,i}} \lambda_e \right) \cdot x_{\nu,i,j} - \sum_{e \in E} c(e) \cdot \lambda_e.$$

For every net $(\nu,i) \in N$ and every route $T^j_{\nu,i}$ of (ν,i) we define $\lambda_{\nu,i,j}$ to be

$$\lambda_{\nu,i,j} := \sum_{e \in T^j_{\nu,i}} \lambda_e$$

and call this quantity the Lagrange multiplier sum (LMS) of $T^j_{\nu,i}$. Thus, we get

$$\max_{\lambda \geq 0} \quad \min_{x} \quad \sum_{(\nu,i) \in N} \sum_{1 \leq j \leq I_{\nu,i}} \lambda_{\nu,i,j} \cdot x_{\nu,i,j} - \sum_{e \in E} c(e) \cdot \lambda_e.$$

Now, for every net $(\nu,i) \in N$ we choose the route that is associated with the smallest LMS. This, in effect, fulfills the completeness constraints of LD^e, which can therefore be left out in the following. Formally, for every net $(\nu,i) \in N$ we define $\lambda_{\nu,i}$ as

$$\lambda_{\nu,i} := \min_{1 \leq j \leq I_{\nu,i}} \lambda_{\nu,i,j} \tag{12}$$

and call this quantity the minimum Lagrange multiplier sum (minimum LMS) for net (ν,i). Thus, we end up with the following cost function:

$$\max_{(3)\lambda \geq 0} \quad \sum_{(\nu,i) \in N} \lambda_{\nu,i} - \sum_{e \in E} c(e) \cdot \lambda_e.$$

Furthermore, the integrality constraints of LD^e can be eliminated since the related coefficient matrix is totally unimodular. According to the labeled inequalities (4), (5), and (6) above we obtain the following form of the Lagrange dual.

$$(LD^e): \quad \max_{\lambda} \quad \sum_{(\nu,i) \in N} \lambda_{\nu,i} - \sum_{e \in E} c(e) \cdot \lambda_e$$

s.t.

$$\lambda_{\nu,i} \leq \lambda_{\nu,i,j} \text{ for all } (\nu,i) \in N, 1 \leq j \leq I_{\nu,i}, \tag{13}$$

$$\sum_{e \in E} \lambda_e = 1, \tag{14}$$

$$\lambda_e \geq 0 \text{ for all } e \in E. \tag{15}$$

Thus, LD^e has taken on the form of DP^e which proves the theorem. □

The equivalence of LD^e and DP^e plays an important role in the following sections. An immediate consequence is that we are able to solve the linear dual by Lagrange relaxation in principle. In practice, Lagrange duals are solved with subgradient optimization (SO). This procedure is much faster than the solution of the linear relaxation or its dual. On the other hand, using SO we do not obtain the exact values of the optimal Lagrange multipliers λ. But experimental experience indicates that SO performs satisfactorily, in practice. Table 2 shows the computational results for the solution of the Lagrange dual by SO in terms of the circuits MOZART-M05 and MOZART-M09.

In the following, we use the notation from Lagrange relaxation. Thus, we prefer to write λ instead of π. The approximated Lagrange multipliers, computed by SO, will be denoted by λ'.

6.4. Tightness of the bounds. We now investigate the difference $\Delta(x)$ between the cost $\vartheta(x)$ of an arbitrary solution x and the cost $\vartheta(\bar{x})$ of an optimal solution $\bar{x}$ of the linear relaxation

$$\Delta(x) := \vartheta(x) - \vartheta(\bar{x}).$$

If x is a global optimum of IP^e then we denote $\Delta(x)$ by $\Delta(IP^e)$. We first show that $\Delta(IP^e)$ can be very large in bad cases.

DEFINITION 8 (bottleneck networks). *Let $n, t \in \mathbb{N}, n, t \geq 2$. The instance $\mathcal{I}$ consists of n nets. Every net has two terminals with t edge-disjoint routes. The sets of terminals for different nets are disjoint. The routing graph $G = (V, E)$ contains t^n special edges which we call bottleneck edges. For an arbitrary solution x we define a string S_x which consists of the numbers of routes taken by x, i.e.,*

$$S_x := (j)_{(\nu,i)\in N | x_{\nu,i,j}=1}.$$

A bijective function assigns to every string a bottleneck edge which has the respective string as a label. The routes for the nets are chosen in such a way that for each (ν, i) every $T^j_{\nu,i}$ takes exactly the bottleneck edge whose label at the position for (ν, i) is j. The routing graph contains additional edges such that a meaningful problem instance of the described form is created. This can always be done, even in such a way that the resulting routing graph is planar. The edge capacities are zero for all edges. The resulting problem instance $\mathcal{I}$ is called bottleneck network for n nets and t routes per net and is abbreviated by $BN_{n,t}$. Figure 2 sketches a planar realization of the bottleneck network $BN_{4,2}$ on a grid.

For $n, t \in \mathbb{N}$, $n, t \geq 2$, let us consider a realization of the bottleneck network $BN_{n,t}$. This network is constructed in such a way that, for every solution x, we have an edge $e \in E$ with load n, i.e.,

$$\vartheta(IP^e(BN_{n,t})) = n.$$

An optimal solution $\bar{x}$ of the linear relaxation is

$$\bar{x}_{\nu,i,j} = \frac{1}{t} \text{ for all } (\nu, i) \in N \text{ and all } T^j_{\nu,i}.$$

Further we have

$$\vartheta(LP^e(BN_{n,t})) = \frac{n}{t}.$$

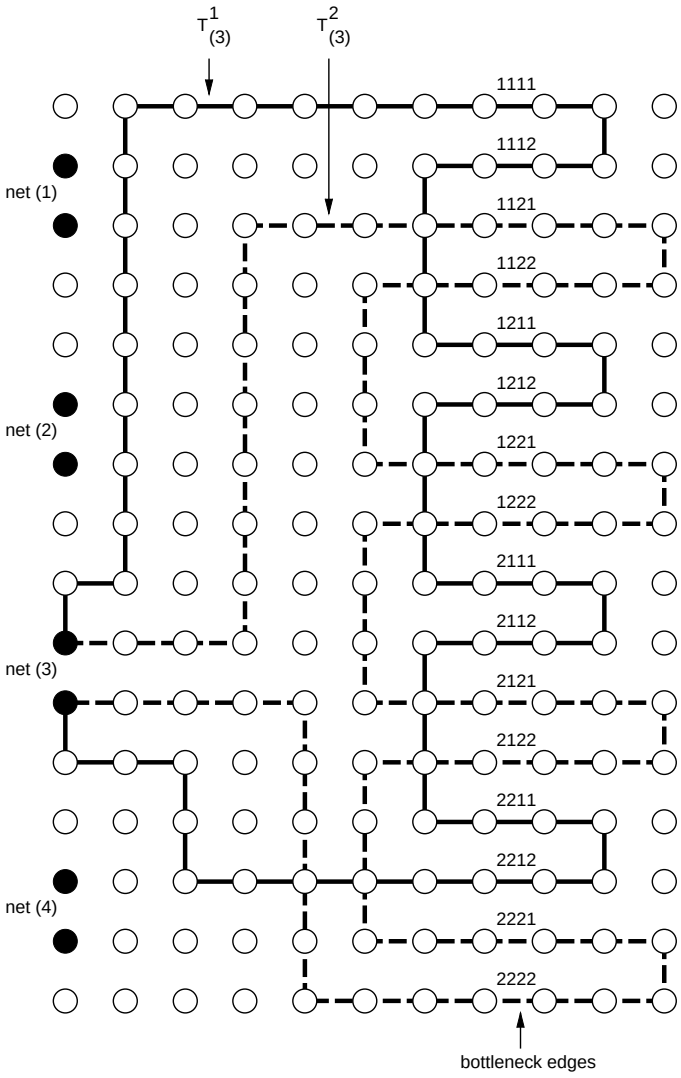


FIG. 2. *A planar realization of $BN_{4,2}$ on a grid.*

Finally, we obtain

$$\Delta(IP^e(BN_{n,t})) = n\left(\frac{t-1}{t}\right).$$

For $t = 2$ we have

$$\Delta(IP^e(BN_{n,t})) = \frac{n}{2}.$$

For $t = n$ we have

$$\Delta(IP^e(BN_{n,t})) = n - 1.$$

Bottleneck networks show that $\Delta(IP^e)$ cannot be bounded by a constant. On the other hand, such networks never seem to turn up in real life. Rather, $\Delta(IP^e)$ is quite

small in practice. Indeed, we have not come across a real circuit for which $\Delta(IP^e)$ is greater than 1.5 tracks. On the other hand, we observe that $\Delta(IP^e)$ slightly increases with an increasing number of edges of the routing graph.

We now introduce a helpful decomposition of $\Delta(x)$ into two terms. For this analysis we extend the notion of LMSs to complete solutions x:

$$\lambda_x := \sum_{(\nu,i)\in N} \sum_{1\le j\le I_{\nu,i}} \lambda_{\nu,i,j} \cdot x_{\nu,i,j}.$$

λ_x can also be expressed as

$$\lambda_x = \sum_{e\in E} U(x,e) \cdot \lambda_e.$$

Moreover, we denote the minimum LMS over all solutions by

$$\lambda_a := \sum_{(\nu,i)\in N} \lambda_{\nu,i}.$$

The cost of an arbitrary solution x is

$$\vartheta(x) = x_L.$$

The cost of an optimal solution $\bar{x}$ of LP^e is

$$\vartheta(\bar{x}) = \lambda_a - \sum_{e\in E} c(e) \cdot \lambda_e.$$

And therefore

$$\Delta(x) = x_L - \lambda_a + \sum_{e\in E} c(e) \cdot \lambda_e.$$

A simple transformation of this equation yields the following.

THEOREM 9. *Let x be an arbitrary solution of an instance of the global routing problem, and let λ be an optimal solution of the Lagrange dual of that instance.*

$$\Delta(x) = \Delta_1(x,\lambda) + \Delta_2(x,\lambda)$$

with

$$\Delta_1(x,\lambda) = \sum_{e\in E} (x_L - U(x,e) + c(e)) \cdot \lambda_e,$$

$$\Delta_2(x,\lambda) = \lambda_x - \lambda_a.$$

$\Delta_1(x,\lambda)$ is the difference between x_L and λ_x plus the sum of the edge capacities (weighted by the optimal Lagrange multipliers). If x is an optimal solution of $LD^e(\lambda)$, $\Delta(x)$ equals $\Delta_1(x,\lambda)$. Otherwise, $\Delta(x)$ is increased by the difference between λ_x and λ_a. This difference is represented by $\Delta_2(x,\lambda)$.

On the other hand, $\Delta_1(x,\lambda)$ is the sum over the free tracks in the routing channels, weighted by the optimal Lagrange multipliers. Computational experience indicates that the positive Lagrange multipliers distribute in a natural way across those edges of the routing graph which are bottlenecks. Note that the sum of the Lagrange multipliers equals one. Thus, $\Delta_1(x,\lambda)$ can be supposed to be small. Indeed, experiments indicate that $\Delta_2(x,\lambda)$ dominates $\Delta_1(x,\lambda)$. Thus, in section 7, we adopt the strategy of avoiding solutions x with a large value of $\Delta_2(x,\lambda)$.

6.5. Load tuning. Since the edge capacities are restricted to integer values, lower bounds can be rounded up to the next integer. This action, which we call load tuning, has several advantages. First, it renders lower bounds obtained by linear relaxation and SO identical in most cases. In the other cases we observed, the lower bound obtained by linear relaxation stays one above the lower bound obtained by SO. Second, load tuning increases the chance of proving a high-quality solution globally optimal. Recall that we did not come across a circuit where $\Delta(IP^e)$ is greater than 1.5 tracks. Thus, high-quality solutions can often be proved to be at most one track away from the optimum.

7. Linear preprocessing. Our general approach in this section is to provide a preprocessing step that solves a linear program. Based on this program we eliminate a (hopefully large) set of solutions, thus substantially reducing the solution space. We hope that the resulting subspace still contains the global optimum and we will analyze if it does. The small integer program pertaining to the subspace is then solved in the second step. Based on the kind of linear program that we use in the first step, we obtain several kinds of preprocessing which we now discuss in detail.

7.1. Dual preprocessing. In this variant, we first solve the Lagrange dual LD^e obtaining an optimal solution λ. Then we eliminate all solutions x for which $\Delta_2(x, \lambda) > 0$. We call this procedure dual preprocessing.

DEFINITION 10 (dual preprocessing (DPP)). *Let λ be an optimal solution of LD^e.*

(a) *A solution x of IP^e is called dual-reduced (according to λ) if*

$$\lambda_{\nu,i,j} > \lambda_{\nu,i} \Longrightarrow x_{\nu,i,j} = 0 \text{ for all } (\nu, i) \in N, T^j_{\nu,i}.$$

(b) *Let IP^e_λ be the integer program obtained from IP^e by the exclusion of all routes with*

$$\lambda_{\nu,i,j} > \lambda_{\nu,i}.$$

IP^e_λ is called dual-reduced integer program of IP^e (according to λ). The linear relaxation LP^e_λ of IP^e_λ is called dual-reduced linear relaxation of IP^e (according to λ).

DPP reduces the problem size $Z(IP^e)$ of IP^e. We quantify the amount of reduction via the quantity

$$R(\lambda) := 1 - \frac{\xi(IP^e_\lambda)}{\xi(IP^e)}.$$

Experiments show that $R(\lambda)$ correlates with the number of edges $e \in E$ with $\lambda_e > 0$, i.e., the number of positive Lagrange multipliers. A large value yields a large reduction, sometimes of more than 90%. A small value usually leads to small reductions, sometimes below 10%.

Furthermore, we have observed that DPP substantially reduces $\Delta(x)$. Table 3 summarizes the results of DPP on the circuits MOZART-M05 and MOZART-M09.

The following theorem shows that, by DPP, we do not lose the global optimum of IP^e and the related linear relaxation LP^e.

THEOREM 11. *Let λ be an optimal solution of LD^e.*

(a) $\vartheta(LP^e_\lambda) = \vartheta(LP^e)$.

(b) $\vartheta(IP^e_\lambda) = \vartheta(IP^e)$.

TABLE 3
The linear preprocessing results.

		MOZART-M05	MOZART-M09
DPP	t	0:03:09	5:24:02
	R	66.8 %	56.5 %
	TR	112 $\rightsquigarrow$ 71	1871 $\rightsquigarrow$ 1110
	LS	65 $\rightsquigarrow$ 59	1065 $\rightsquigarrow$ 973
	SS	62 $\rightsquigarrow$ 58	1013 $\rightsquigarrow$ 971
	RR	60 $\rightsquigarrow$ 59	989 $\rightsquigarrow$ 971
	MD	62 $\rightsquigarrow$ 59	1033 $\rightsquigarrow$ 971
	RD	59 $\rightsquigarrow$ 60	971 $\rightsquigarrow$ 971
	LP	57.5174	969.326
QDPP	t	0:00:52	0:33:02
	R	64.1 %	52.4%
	TR	112 $\rightsquigarrow$ 71	1871 $\rightsquigarrow$ 1189
	LS	65 $\rightsquigarrow$ 59	1065 $\rightsquigarrow$ 981
	SS	62 $\rightsquigarrow$ 59	1013 $\rightsquigarrow$ 972
	RR	60 $\rightsquigarrow$ 59	989 $\rightsquigarrow$ 972
	MD	62 $\rightsquigarrow$ 59	1033 $\rightsquigarrow$ 973
	RD	59 $\rightsquigarrow$ 59	971 $\rightsquigarrow$ 971
	LP	57.5101	969.303
PPP	t	0:03:08	5:23:47
	R	95.0 %	95.5 %
	TR	112 $\rightsquigarrow$ 60	1871 $\rightsquigarrow$ 972
	LS	65 $\rightsquigarrow$ 59	1065 $\rightsquigarrow$ 970
	SS	62 $\rightsquigarrow$ 58	1013 $\rightsquigarrow$ 970
	RR	60 $\rightsquigarrow$ 59	989 $\rightsquigarrow$ 971
	MD	62 $\rightsquigarrow$ 58	1033 $\rightsquigarrow$ 970
	RD	59 $\rightsquigarrow$ 59	971 $\rightsquigarrow$ 971
	LP	57.5174	969.326
QPPP	t	0:01:02	0:39:08
	R	95.0 %	95.5 %
	TR	112 $\rightsquigarrow$ 60	1871 $\rightsquigarrow$ 972
	LS	65 $\rightsquigarrow$ 59	1065 $\rightsquigarrow$ 970
	SS	62 $\rightsquigarrow$ 58	1013 $\rightsquigarrow$ 970
	RR	60 $\rightsquigarrow$ 59	989 $\rightsquigarrow$ 971
	MD	62 $\rightsquigarrow$ 58	1033 $\rightsquigarrow$ 970
	RD	59 $\rightsquigarrow$ 59	971 $\rightsquigarrow$ 971
	LP	57.5174	969.326

Proof. (a) The proof is obvious, since

$$\vartheta(LP^e_\lambda) = \vartheta(LP^e) = \sum_{(\nu,i)\in N} \lambda_{\nu,i} - \sum_{e\in E} c(e)\cdot\lambda_e.$$

(b) Let x' be an optimal solution of IP^e and x'' be an optimal solution of IP^e_λ. Obviously

$$\vartheta(x') \le \vartheta(x'').$$

It remains to show that

$$\vartheta(x') \ge \vartheta(x'').$$

In order to do this, we iteratively replace the routes of x'' by routes of x' and show that no such step improves the cost.

Possibly, x' contains routes with a minimum LMS. We replace these routes immediately. This does not improve the cost because x'' remains dual-reduced.

We now explain how to replace a route that has no minimum LMS. Let x_a be x'' before the replacement step and x_b after it. Assume that

$$\vartheta(x_a) > \vartheta(x_b).$$

Since x_a and x_b differ in the route of just one net, we have

$$\vartheta(x_a) = \vartheta(x_b) + 1.$$

As a consequence of (a) we have

$$\Delta_1(x_a) + \Delta_2(x_a) = \Delta_1(x_b) + \Delta_2(x_b) + 1.$$

Since x_a and x_b differ in the route of just one net and, moreover, $\sum_{e \in E} \lambda_e = 1$, we obtain

$$-1 \leq \Delta_1(x_a) - \Delta_1(x_b) \leq 1.$$

Since the replacement step turns a route with minimum LMS into a route which does not have this property, we observe

$$\Delta_2(x_a) < \Delta_2(x_b).$$

In summary we obtain

$$\Delta_1(x_a) + \Delta_2(x_a) = \Delta_1(x_b) + \Delta_2(x_b) + 1$$

$$> \Delta_1(x_b) + \Delta_2(x_a) + 1 \geq \Delta_1(x_a) + \Delta_2(x_a),$$

which is a contradiction. □

The following observation follows directly from Theorem 15 and Theorem 16 below.

THEOREM 12. *Every solution x obtained by RD is dual-reduced (according to some optimal solution λ of LD^e).*

DPP is quite powerful but, unfortunately, we need an optimal solution of the Lagrange dual which requires large runtime on large problem instances. In order to avoid the exact solution of LD^e we perform SO and use the approximate solution λ' of LD^e as a basis for a linear preprocessing. For the decision whether to exclude a specific route $T^j_{\nu,i}$ we introduce an additional value, the so-called accuracy $\varepsilon \geq 1$. We call this procedure quasidual preprocessing.

DEFINITION 13 (quasidual preprocessing (QDPP)). *Let λ' be an approximated solution of LD^e and $\varepsilon \geq 1$.*

(a) *A solution x of IP^e is called quasidual-reduced (according to λ' and ε) if*

$$\lambda'_{\nu,i,j} > \lambda'_{\nu,i} \cdot \varepsilon \Longrightarrow x_{\nu,i,j} = 0 \text{ for all } (\nu,i) \in N, T^j_{\nu,i}.$$

(b) *Let $IP^e_{\lambda',\varepsilon}$ be the integer program obtained from IP^e by the exclusion of all routes with*

$$\lambda'_{\nu,i,j} > \lambda'_{\nu,i} \cdot \varepsilon.$$

$IP^e_{\lambda',\varepsilon}$ is called a quasidual-reduced integer program of IP^e (according to λ and ε). The linear relaxation $LP^e_{\lambda',\varepsilon}$ of $IP^e_{\lambda',\varepsilon}$ is called quasidual-reduced linear relaxation of IP^e (according to λ and ε).

The accuracy ε should be chosen small enough such that

$$\vartheta(LP^e_{\lambda',\varepsilon}) = \vartheta(LP^e).$$

In this case we call the quasidual preprocessing proper. Unfortunately, we cannot test the above condition efficiently. Thus, we use SO and hope that the condition will be fulfilled. If we come up with matching bounds later on, this will have been the case. Our computational experience shows that $\varepsilon = 1.002$ is an appropriate choice. Table 3 summarizes the results in terms of the circuits MOZART-M05 and MOZART-M09.

7.2. Primal preprocessing. This variant of preprocessing begins with the linear relaxation LP^e.

DEFINITION 14 (primal preprocessing (PPP)). *Let $\bar{x}$ be an optimal solution of LP^e.*

(a) *A solution x of IP^e is called primal-reduced (according to $\bar{x}$) if*

$$\bar{x}_{\nu,i,j} = 0 \Longrightarrow x_{\nu,i,j} = 0 \text{ for all } (\nu,i) \in N, T^j_{\nu,i}.$$

(b) *Let $IP^e_{\bar{x}}$ be the integer program obtained from IP^e by the exclusion of all routes with*

$$\bar{x}_{\nu,i,j} = 0.$$

$IP^e_{\bar{x}}$ is called a primal-reduced integer program of IP^e (according to $\bar{x}$). The linear relaxation $LP^e_{\bar{x}}$ of $IP^e_{\bar{x}}$ is called a primal-reduced linear relaxation of IP^e (according to $\bar{x}$).

THEOREM 15. *Let x be an arbitrary solution of IP^e, $\bar{x}$ an optimal solution of LP^e, and λ an optimal solution of LD^e.*

If x is primal-reduced (according to $\bar{x}$) then x is dual-reduced (according to λ).

Proof. The proof follows directly from the complementary slackness conditions for global routing. □

The following result is obvious.

THEOREM 16. *Every solution x obtained by* RD *is primal-reduced (according to some optimal solution $\bar{x}$ of LP^e).*

Primal preprocessing strengthens the advantages of DPP substantially. Indeed, the reduction $R(\bar{x})$ (for every optimal solution $\bar{x}$ of LP^e) is above 90% for every instance of global routing we investigated. Furthermore, primal preprocessing improves $\Delta(x)$ to values which are close to zero. As a result, branch-and-bound can often be done efficiently, i.e., in a few minutes, on primal-reduced instances even if the original instance was very large. Table 3 shows the computational results on the circuits MOZART-M05 and MOZART-M09.

Primal preprocessing is enormously powerful but it has two disadvantages. First, primal preprocessing may lose the global optimum of the integer program IP^e. To illustrate this we introduce the following class of examples which are derived from the bottleneck networks by a slight modification.

DEFINITION 17 (shortcut networks). *Let $BN_{n,t}$ be a bottleneck network for n nets and t routes per net with $t = n$. We do not change the circuit of this problem instance, but modify the routes in the following way. We add another column of nodes to the routing grid on its left side and displace all terminals to the left by* 1. *(The old locations of the terminals are depicted with squares, the new ones with dark dots.) Then we extend all routes of the bottleneck network in the natural way to the new terminals locations across the so-called bridge edges (see Figure* 3*). Finally, we add a new route*

for each net going across the relevant two bridge edges and the corresponding shortcut edge (see figure). These new routes are called shortcut routes. Thus, the number of routes per net is $n+1$. The resulting problem instance $\mathcal{I}$ is called shortcut network for n nets and $n+1$ routes per net and is abbreviated with SN_n. Figure 3 sketches a planar realization of the shortcut network SN_2 on a grid.

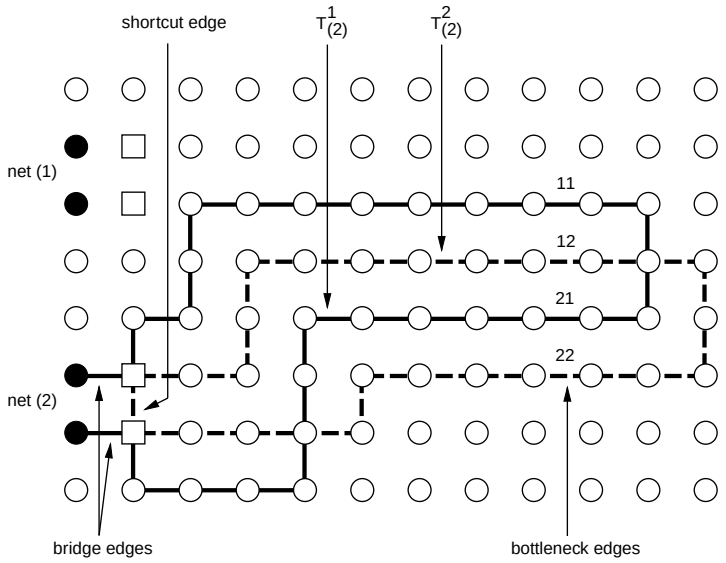


FIG. 3. *A planar realization of SN_2 on a grid.*

For $n \in \mathbb{N}, n \geq 2$, let us consider a realization of the shortcut network SN_n. The construction ensures that

$$\vartheta(LP^e(SN_n)) = 1.$$

This can be realized using all shortcut routes. We denote the shortcut routes by $T^{j^*}_{\nu,i}$ for every $(\nu,i) \in N$ and all the other routes with $T^{j'}_{\nu,i}$.

The solutions $\bar{x}^1$ and $\bar{x}^2$ with

$$\bar{x}^1_{\nu,i,j'} := \frac{1}{n} \text{ for all } (\nu,i) \in N \text{ and all } T^{j'}_{\nu,i},$$

$$\bar{x}^1_{\nu,i,j^*} := 0 \text{ for all } (\nu,i) \in N,$$

$$\bar{x}^2_{\nu,i,j'} := 0 \text{ for all } (\nu,i) \in N \text{ and all } T^{j'}_{\nu,i},$$

$$\bar{x}^2_{\nu,i,j^*} := 1 \text{ for all } (\nu,i) \in N$$

are both optimal for $LP^e(SN_n)$. Thus, primal preprocessing (based on $\bar{x}^1$) excludes all shortcut routes. We obtain

$$\vartheta(IP^e_{\bar{x}^1}(SN_n)) = n.$$

On the other hand we have

$$\vartheta(IP^e_{\bar{x}^2}(SN_n)) = \vartheta(IP^e(SN_n)) = 1$$

which shows that linear preprocessing may lose the global optimum on these networks. However, we never encountered such a situation in practice.

The second disadvantage of primal preprocessing appears more serious; we need an optimal solution of the linear relaxation which requires large runtime on large problem instances. In order to avoid the need of solving LP^e exactly we can use one of two strategies. The first is to represent the linear relaxation as a multicommodity network flow problem, as proposed by Carden and Cheng [3] and Carden, Li, and Cheng [4]. In this paper we follow the second strategy. DPP (based on a specific optimal solution λ of LD^e) reduces the problem size and improves $\Delta(x)$ in such a way that a subsequent solution of LP^e_λ is possible in reasonable time, even for large circuits. Moreover, Theorem 15 shows that this procedure yields the same result as conventional primal preprocessing. Unfortunately, LD^e is not easier to solve than LP^e. But we can use SO in order to approximate the desired result. In total, we perform a sequence of reductions, beginning with QDPP and executing primal preprocessing afterwards. We call this procedure quasiprimal preprocessing (QPPP).

QPPP is efficient even on very large circuits. Table 3 summarizes the results in terms of the circuits Mozart-M05 and Mozart-M09. As a whole, QPPP is the most effective of all preprocessing strategies we considered.

8. Putting it all together. In this section we present results on provably good global routing that combines the results from the previous sections. In section 6 we introduced several methods for obtaining upper bound solutions and observed that the RD procedure produces the best results. By now we know that the reason is that RD implicitly performs a primal preprocessing. Indeed, after primal preprocessing, all other upper bound methods mostly produce better results than RD.

In our opinion, the SS procedure is the most advantageous upper bound method. A main reason for this is the speed of this method. A combination with various versions of linear preprocessing yields the following variants for a provably good global routing of integrated circuits:

- Dual swinging search (DSS);
- Quasidual swinging search (QDSS);
- Primal swinging search (PSS);
- Quasiprimal swinging search (QPSS).

Table 4 summarizes the results on the circuits Mozart-M05 and Mozart-M09. The table also gives a comparison with RD. QPSS performs best and routes both circuits optimally.

9. Extension to global layout. The placement process precedes global routing. The goal of placement is to distribute the circuit components over the chip in such a way that wiring is possible in small space. The placement process is hampered severely by the fact that it is very difficult to come up with easy-to-compute and accurate estimates of wiring area. Thus, recently, research has been directed toward integrating placement with global routing, i.e., the router itself is used to provide wiring estimates. Heuristic versions of this approach have been presented in [1, 8, 28, 38, 39].

In this section we describe results on extensions of our provably good global routing methods to include placement. We call the respective problem global layout. We will first define the relevant integer program and then discuss heuristic algorithms.

TABLE 4
Results of provably good global routing methods.

		MOZART-M05	MOZART-M09
DSS	*t*	0:03:13	5:26:12
	LB	57.5174	969.326
	UB	58	971
QDSS	*t*	0:00:58	0:36:50
	LB	57.5101	969.303
	UB	59	972
PSS	*t*	0:03:12	5:25:00
	LB	57.5174	969.326
	UB	58	970
QPSS	*t*	0:01:05	0:40:21
	LB	57.5174	969.326
	UB	58	970
RD	*t*	0:03:09	5:23:57
	LB	57.5174	969.326
	UB	59	971

We again distinguish between a constrained and an unconstrained version of the problem and concentrate on the unconstrained case, for the same reasons as outlined before. Clearly, both versions of global layout are strongly NP-hard.

In order to tailor the global routing integer program to the global layout problem, we investigate two approaches. In the explicit-explicit approach to global layout, a special binary variable is chosen to represent each placement. Moreover, we provide explicitly generated routes for each placement alternative. The advantage of explicit-explicit global layout is that we can take technical restraints on the admissibility of placements into account. In the implicit-explicit approach to global layout, the legality of placements is implicitly secured by appropriate constraints which ensure that the circuit components, the so-called gates, are placed in a respectable fashion into so-called slots on the chip surface. Since placements are not represented by special variables, no complex restraints on the placements can be formulated in the implicit-explicit approach. Lengauer and Lügering [27] have investigated the implicit-explicit approach in more detail. In this paper we follow the explicit-explicit approach to global layout. The main problem with this approach is that we have to preselect a number of candidate placements. We will use placement heuristics [26] in order to do this. In this sense, our approach is heuristic and not exhaustive in terms of placement.

For each candidate placement $\rho \in P$ in the placement set P we provide a variable x_ρ with the intention that

$$x_\rho := \begin{cases} 1 & \text{if placement } \rho \text{ is realized,} \\ 0 & \text{otherwise.} \end{cases}$$

For each candidate route $T^j_{\rho,\nu,i}$ for placement $\rho \in P$ and net $(\nu, i) \in N$ we provide a variable $x_{\rho,\nu,i,j}$ with the intention that

$$x_{\rho,\nu,i,j} := \begin{cases} 1 & \text{if net } (\nu, i) \text{ takes route } T^j_{\rho,\nu,i}, \\ 0 & \text{otherwise.} \end{cases}$$

We denote the number of admissible routes for a placement $\rho \in P$ and a net $(\nu, i) \in N$ by $I_{\rho,\nu,i}$.

Again, we need a variable x_L to measure the maximum load over all edges. Thus, a solution of the integer program for global layout has the form

$$x := \left((x_\rho)_{\rho\in P}, (x_{\rho,\nu,i,j})_{\rho\in P,(\nu,i)\in N,T^j_{\rho,\nu,i}}, (x_L)\right).$$

The traffic of an edge $e \in E$ is computed as follows:

$$U(x,e) := \sum_{\rho\in P}\sum_{(\nu,i)\in N}\sum_{j|e\in T^j_{\rho,\nu,i}} x_{\rho,\nu,i,j}.$$

The integer program for global layout consists of a placement completion constraint which ensures that exactly one placement is realized, the routing completion constraints which ensure that every net is routed by exactly one route of the realized placement, and the load constraints which force x_L to be the maximum edge load. In addition, the variables x_ρ and $x_{\rho,\nu,i,j}$ have to take on binary values. The resulting integer program is the following.

DEFINITION 18 (the integer program for global layout).

$$(IP^{ee}): \quad \min_x \quad x_L$$

s.t.

$$\sum_{\rho\in P} x_\rho = 1 \tag{16}$$

(placement completion constraint),

$$\sum_{j=1}^{I_{\rho,\nu,i}} x_{\rho,\nu,i,j} - x_\rho = 0 \text{ for all } \rho \in P \text{ and all } (\nu,i) \in N \tag{17}$$

(routing completion constraints),

$$U(x,e) - x_L \le c(e) \text{ for all } e \in E \tag{18}$$

(load constraints),

$$x_\rho \in \{0,1\} \text{ for all } \rho \in P \tag{19}$$

(placement integrality constraints),

$$x_{\rho,\nu,i,j} \in \{0,1\} \text{ for all } \rho \in P, (\nu,i) \in N, 1 \le j \le I_{\rho,\nu,i} \tag{20}$$

(routing integrality constraints).

The cost $\vartheta(\check{x})$ of an optimal solution $\check{x}$ of IP^{ee} is denoted by $\vartheta(IP^{ee})$.

We denote the linear relaxation of the integer program IP^{ee} by LP^{ee} and the cost $\vartheta(\bar{x})$ of an optimal solution $\bar{x}$ of LP^{ee} by $\vartheta(LP^{ee})$.

The linear dual DP^{ee} of the global layout integer program has the following shape.

DEFINITION 19 (the linear dual for global layout).

$$(DP^{ee}): \quad \max_{\pi} \quad \pi_0 - \sum_{e \in E} c(e) \cdot \pi_e$$

s.t.

$$\pi_0 \leq \sum_{(\nu,i) \in N} \pi_{\rho,\nu,i} \text{ for all } \rho \in P, \tag{21}$$

$$\pi_{\rho,\nu,i} \leq \sum_{e \in T^j_{\rho,\nu,i}} \pi_e \text{ for all } \rho \in P, (\nu,i) \in N, 1 \leq j \leq I_{\rho,\nu,i}, \tag{22}$$

$$\sum_{e \in E} \pi_e = 1, \tag{23}$$

$$\pi_e \geq 0 \text{ for all } e \in E. \tag{24}$$

The cost $\vartheta(\pi)$ of an optimal solution π of DP^{ee} is denoted by $\vartheta(DP^{ee})$.

The complementary slackness conditions take on the following form.

LEMMA 20 (the complementary slackness conditions for global layout). *Let $\bar{x}$ be a solution of LP^{ee} and π be a solution of DP^{ee}. $\bar{x}$ and π are both optimal if and only if*

$$\pi_e \cdot (\bar{x}_L - U(\bar{x}, e) + c(e)) = 0 \text{ for all } e \in E, \tag{25}$$

$$(\sum_{e \in T^j_{\rho,\nu,i}} \pi_e - \pi_{\rho,\nu,i}) \cdot \bar{x}_{\rho,\nu,i,j} = 0 \text{ for all } \rho, (\nu,i), T^j_{\nu,i}, \tag{26}$$

$$(\sum_{(\nu,i) \in N} \pi_{\rho,\nu,i} - \pi_0) \cdot \bar{x}_\rho = 0 \text{ for all } \rho \in P. \tag{27}$$

In order to apply Lagrange relaxation to the global layout problem, we again relax the load constraints. The remaining coefficient matrix is totally unimodular, again. The construction of the Lagrange dual LD^{ee} is straightforward. The cost $\vartheta(\lambda)$ of an optimal solution λ of LD^{ee} is denoted by $\vartheta(LD^{ee})$.

Since the linear relaxation for the global layout problem has a finite optimum and the polytope pertaining to the Lagrange dual is integral we obtain the following result.

THEOREM 21.

$$\vartheta(LP^{ee}) = \vartheta(DP^{ee}) = \vartheta(LD^{ee}).$$

The following result extends Theorem 7 to the global layout problem.

THEOREM 22.

(a) *For every optimal solution π of DP^{ee} there is an optimal solution λ of LD^{ee} s.t.*

$$\lambda_e = \pi_e \text{ for all } e \in E.$$

(b) *For every optimal solution* λ *of* LD^{ee} *there is an optimal solution* π *of* DP^{ee} *s.t.*

$$\pi_e = \lambda_e \text{ for all } e \in E$$

and

$$\pi_{\rho,\nu,i} = \min_{1 \le j \le I_{\rho,\nu,i}} \sum_{e \in T^j_{\rho,\nu,i}} \lambda_e \text{ for all } \rho \in P, (\nu,i) \in N$$

and

$$\pi_0 = \min_{\rho \in P} \sum_{(\nu,i) \in N} \pi_{\rho,\nu,i}.$$

In the following, we again use the notation from Lagrange relaxation. Thus, we prefer to write λ instead of π and denote the approximated Lagrange multipliers by λ'. Moreover, for every placement $\rho \in P$, every net $(\nu,i) \in N$, and every route $T^j_{\rho,\nu,i}$, we define $\lambda_{\rho,\nu,i,j}$ to be

$$\lambda_{\rho,\nu,i,j} := \sum_{e \in T^j_{\rho,\nu,i}} \lambda_e$$

and call this quantity the LMS of $T_{\rho,\nu,i}$. For every placement $\rho \in P$ and every net $(\nu,i) \in N$ we define $\lambda_{\rho,\nu,i}$ as

$$\lambda_{\rho,\nu,i} := \min_{1 \le j \le I_{\rho,\nu,i}} \lambda_{\rho,\nu,i,j}$$

and call this quantity the minimum LMS for placement $\rho \in P$ and net $(\nu,i) \in N$. Finally, we define λ_0 as

$$\lambda_0 := \min_{\rho \in P} \sum_{(\nu,i) \in N} \lambda_{\rho,\nu,i}$$

and call this quantity the minimum overall Lagrange multiplier sum (minimum overall LMS).

Extending the discussion from section 6.4 we again denote the difference between the cost $\vartheta(x)$ of an arbitrary solution x and the cost of an optimal solution $\bar{x}$ of the linear relaxation by

$$\Delta(x) := \vartheta(x) - \vartheta(\bar{x}).$$

If x is a global optimum of IP^{ee} then we again denote $\Delta(x)$ by $\Delta(IP^e)$. We can easily modify the bottleneck networks in order to show that the difference $\Delta(IP^{ee})$ may be large, even for global layout. Nevertheless, as experiments indicate, $\Delta(IP^{ee})$ is quite small in practice. Indeed, we have found no circuit for which $\Delta(IP^{ee})$ exceeds 1.5 tracks, even in the case of global layout.

We now introduce a helpful decomposition of $\Delta(x)$ into three terms. In order to do so, we extend the notion of LMSs of complete solutions x to the global layout problem.

$$\lambda_x := \sum_{\rho \in P} \sum_{(\nu,i) \in N} \sum_{1 \le j \le I_{\rho,\nu,i}} \lambda_{\rho,\nu,i,j} \cdot x_{\rho,\nu,i,j}.$$

Recall that we can also express λ_x as

$$\lambda_x = \sum_{e \in E} U(x,e) \cdot \lambda_e.$$

Moreover, we denote the minimum LMS over all solutions x which realize a fixed placement ρ by

$$\lambda_\rho := \sum_{(\nu,i) \in N} \lambda_{\rho,\nu,i}.$$

Recall that λ_0 is the minimum overall LMS, i.e., the minimum LMS over all solutions x for all placements $\rho \in P$.

The cost of an arbitrary solution x is

$$\vartheta(x) = x_L.$$

The cost of an optimal solution $\bar{x}$ of the linear relaxation is

$$\vartheta(\bar{x}) = \lambda_0 - \sum_{e \in E} c(e) \cdot \lambda_e.$$

And therefore

$$\Delta(x) = x_L - \lambda_0 + \sum_{e \in E} c(e) \cdot \lambda_e.$$

Equivalently, we can arrange this figure as follows.

THEOREM 23. *Let x be an arbitrary solution of an instance of the global layout problem, and let λ be an optimal solution of the Lagrange dual of that instance.*

$$\Delta(x) = \Delta_1(x,\lambda) + \Delta_2(x,\lambda) + \Delta_3(x,\lambda)$$

with

$$\Delta_1(x,\lambda) = \sum_{e \in E} (x_L - U(x,e) + c(e)) \cdot \lambda_e,$$

$$\Delta_2(x,\lambda) = \lambda_x - \sum_{\rho \in P} \lambda_\rho \cdot x_\rho,$$

$$\Delta_3(x,\lambda) = \sum_{\rho \in P} \lambda_\rho \cdot x_\rho - \lambda_0.$$

The term $\Delta_2(x,\lambda)$ from the global routing problem is now split into two terms. Analogous to global routing, $\Delta_1(x,\lambda)$ can be supposed to be quite small. With an application of the preprocessing methods for global routing, $\Delta_2(x,\lambda)$ can be forced to be close to zero. In order to tightly bound $\Delta(x)$, $\Delta_3(x,\lambda)$ should be small, as well.

An extension of the idea of linear preprocessing to the global layout problem solves LP^{ee} or LD^{ee} and first excludes all placements $\rho \in P$ with $\bar{x}_\rho = 0$ or $\lambda_\rho > \lambda_0$, respectively. Subsequently, linear preprocessing excludes all routes $T^j_{\rho,\nu,i}$ for all remaining placements ρ with $\bar{x}_{\rho,\nu,i,j} = 0$ or $\lambda_{\rho,\nu,i,j} > \lambda_{\rho,\nu,i}$, respectively. By extending Theorem 15 to the global layout problem we can show that every primal-reduced solution created in this way is also dual-reduced in the above sense.

From a practical point of view the intention of linear preprocessing is not strong enough. Indeed, it is sufficient to concentrate on just one arbitrary placement ρ' with $x_{\rho'} > 0$ or $\lambda_{\rho'} = \lambda_0$, respectively. We will now discuss the reasons for this.

Let IP^e_ρ be the integer program obtained from IP^{ee} by adding the restriction

$$x_\rho = 1.$$

Obviously, IP^e_ρ is a global routing integer program. Furthermore, let LP^e_ρ be the linear relaxation of IP^e_ρ, DP^e_ρ the related linear dual, and LD^e_ρ the Lagrange dual of IP^e_ρ. We obtain the following theorem which can be easily derived from previous results.

THEOREM 24. *Let $\bar{x}$ be an optimal solution of LP^{ee} and λ be an optimal solution of LD^{ee}.*

(a) *For every placement $\rho' \in P$ with $\bar{x}_{\rho'} > 0$ we have*

$$\vartheta(LP^{ee}) = \vartheta(LP^e_{\rho'}).$$

(b) *For every placement $\rho' \in P$ with $\lambda_{\rho'} = \lambda_0$ we have*

$$\vartheta(LD^{ee}) = \vartheta(LD^e_{\rho'}).$$

Furthermore, λ is an optimal solution of $LD^e_{\rho'}$.

Proof. (b) Since all involved coefficient matrices are totally unimodular we have

$$\vartheta(LD^{ee}) = \min_{\rho \in P} \vartheta(LD^e_\rho).$$

Obviously, every placement $\rho' \in P$ with $\lambda_{\rho'} = \lambda_0$ achieves this minimum and λ is an optimal solution of $LD^e_{\rho'}$.

(a) As a result of Theorem 6 and Theorem 21 we have

$$\vartheta(LP^{ee}) = \min_{\rho \in P} \vartheta(LP^e_\rho).$$

According to the extension of Theorem 15 to the global layout problem, for every placement $\rho' \in P$ with $\bar{x}_{\rho'} > 0$ and every optimal solution λ of LD^{ee} we have

$$\lambda_{\rho'} = \lambda_0.$$

Thus, the rest follows from (b). □

With respect to Theorem 24 and the results concerning the global routing problem, we propose to restrict the solution space of global layout to those solutions which realize an arbitrary but fixed placement ρ' such that $x_{\rho'} > 0$ or $\lambda_{\rho'} = \lambda_0$. We call this procedure linear preselection and obtain a primal and a dual version. We observe that there is no advantage of primal preselection over dual preselection. Furthermore, there is no guarantee that linear preselection preserves optimal integer solutions. Indeed, it is easy to find a placement for a bottleneck network which increases the lower bound and decreases the upper bound of the original placement. Nevertheless, such a situation never occurred in our experiments.

In order to do linear preselection, we need an optimal solution of LP^{ee} or LD^{ee}. In order to avoid the large runtimes for linear preselection on large problem instances, we use SO to solve LD^{ee} approximately. We call this procedure quasidual preselection. The selected placement ρ' does not ensure that $\Delta_3(x, \lambda')$ is exactly zero for all x realizing ρ', in general. Nevertheless, experimental results show that quasidual preselection approximates dual preselection with sufficient accuracy. A subsequent application of load tuning to the global routing instance defined by ρ' in most cases eliminates errors made previously.

As a whole, an effective method of solving the integer program for global layout is to start with quasidual preselection and then perform QPSS on the resulting instance of the global routing problem. Experimental experience shows the effectiveness of this procedure. On the other hand, if we should ever come up with an unsatisfactory result we could restart the whole procedure and, what is more, disregard the placement selected to that point.

10. Conclusions. We have formulated the global routing problem as an integer program. In order to provide upper bounds we introduced several heuristics. Moreover, we suggested methods for finding lower bounds. An analysis on the tightness of the bounds shows that the cost of the optimal integer solutions and the cost of the optimal fractional solutions are only a small number of tracks apart in practice. We introduce several versions of preprocessing that substantially reduce the size of the integer program and some of which can be shown to preserve the global optimum of the integer program. In fact, all of them do so in practice. Preprocessing establishes the basis for an efficient and provably good global routing. We extended these results to the global layout problem.

We implemented all the discussed methods in the software package ERIDANUS. With this software, we are able to solve circuits with more than 10000 nets to optimality or at least close to optimality, in acceptable time.

REFERENCES

[1] M. BURSTEIN AND S. J. HONG, *Hierarchical VLSI layout: Simultaneous wiring and placement*, in Proceedings of VLSI '83, F. Anceau and E. J. Aas, eds., Elsevier Science Publishers B.V., Amsterdam, 1983, pp. 45–60.

[2] M. BURSTEIN AND R. PELAVIN, *Hierarchical wire routing*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 2 (1983), pp. 223–234.

[3] R. C. CARDEN AND C.-K. CHENG, *A global router using an efficient approximate multicommodity multiterminal flow algorithm*, in Proceedings of the 28th Design Automation Conference, San Francisco, ACM/IEEE, 1991, pp. 316–321.

[4] R. C. CARDEN, J. LI, AND C.-K. CHENG, *A global router with a theoretical bound on the optimal solution*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 15 (1996), pp. 208–216.

[5] C. CHIANG, M. SARRAFZADEH, AND C. K. WONG, *A powerful global router: Based on Steiner min-max trees*, in Proceedings of the International Conference on Computer-Aided Design, Sanata Clara, CA, IEEE, 1989, pp. 2–5.

[6] C. CHIANG, M. SARRAFZADEH, AND C. K. WONG, *Global routing based on Steiner min-max trees*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 9 (1990), pp. 1318–1325.

[7] C. CHIANG, C. K. WONG, AND M. SARRAFZADEH, *A weighted Steiner tree-based global router with simultaneous length and density minimization*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 13 (1994), pp. 1461–1469.

[8] W. W.-M. DAI AND E. S. KUH, *Simultaneous floorplanning and global routing for hierarchical building block layout*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 6 (1987), pp. 828–837.

[9] H. ESBENSEN, *A macro-cell global router based on two genetic algorithms*, in Proceedings of the European Design Automation Conference, Grenoble, 1994, pp. 428–433.

[10] H. ESBENSEN AND P. MAZUMDER, *A genetic algorithm for the Steiner problem in a graph*, in Proceedings of the European Design and Test Conference, Paris, 1994, pp. 402–406.

[11] M. GRÖTSCHEL, A. MARTIN, AND R. WEISMANTEL, *Packing Steiner trees: A cutting plane algorithm and computational results*, Math. Programming, 72 (1996), pp. 125–145.

[12] M. GRÖTSCHEL, A. MARTIN, AND R. WEISMANTEL, *Packing Steiner trees: Further facets*, European J. Combin., 17 (1996), pp. 39–52.

[13] M. GRÖTSCHEL, A. MARTIN, AND R. WEISMANTEL, *Packing Steiner trees: Polyhedral investigations*, Math. Programming, 72 (1996), pp. 101–123.

[14] M. Grötschel, A. Martin, and R. Weismantel, *Packing Steiner trees: Separation algorithms*, SIAM J. Discrete Math., 9 (1996), pp. 233–257.
[15] M. Grötschel, A. Martin, and R. Weismantel, *The Steiner tree packing problem in VLSI-design*, Math. Programming, 78 (1997), pp. 265–281.
[16] J. Heistermann and T. Lengauer, *The efficient solution of integer programs for hierarchical global routing*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 10 (1991), pp. 748–753.
[17] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, 1975.
[18] T. C. Hu, *Integer Programming and Network Flows*, Addison-Wesley, Reading, MA, 1969.
[19] T. C. Hu and M. T. Shing, *A decomposition algorithm for circuit routing*, in VLSI Circuit Layout: Theory and Design, T. C. Hu and E. S. Kuh, eds., IEEE, New York, 1985, pp. 144–152.
[20] N. Karmarkar, *A new polynomial algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.
[21] 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.
[22] R. M. Karp, F. T. Leighton, R. L. Rivest, C. D. Thompson, U. V. Vazirani, and V. V. Vazirani, *Global wire routing in two-dimensional arrays*, Algorithmica, 2 (1987), pp. 113–129.
[23] L. G. Khachian, *A polynomial algorithm for linear programming*, Doklady Akad. Nauk. USSR, 244 (1979), pp. 1093–1096 (in Russian); Soviet Math. Doklady, 20 (1979), pp. 191–194 (in English).
[24] B. Korte, H. J. Prömel, and A. Steger, *Steiner trees in VLSI-layout*, in Paths, Flows, and VLSI-Layout, B. Korte, L. Lovász, H. J. Prömel, and A. Schrijver, eds., Springer-Verlag, Berlin, Heidelberg, New York, 1990, pp. 185–214.
[25] M. R. Kramer and J. van Leeuwen, *The complexity of wire routing and finding minimum area layouts for arbitrary VLSI circuits*, in Advances in Computing Research, Vol. 2: VLSI Theory, F. P. Preparata, ed., JAI Press, Reading, MA, 1984, pp. 129–146.
[26] T. Lengauer, *Combinatorial Algorithms for Integrated Circuit Layout*, Teubner–Wiley Series of Applicable Theory in Computer Science, John Wiley & Sons, New York, 1990.
[27] T. Lengauer and M. Lügering, *Integer program formulations of global routing and placement problems*, in Algorithmic Aspects of VLSI Layout, M. Sarrafzadeh and D. T. Lee, eds., Lecture Notes Series on Computing 2, World Scientific, Singapore, 1993, pp. 167–197.
[28] T. Lengauer and R. Müller, *Robust and accurate hierarchical floorplanning with integrated global wiring*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 12 (1993), pp. 802–809.
[29] W. K. Luk, P. Sipala, M. Tamminen, D. Tang, L. S. Woo, and C. K. Wong, *A hierarchical global wiring algorithm for custom chip design*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 6 (1987), pp. 518–533.
[30] A. P.-C. Ng, P. Raghavan, and C. D. Thompson, *Experimental results for a linear program global router*, Comput. Artificial Intelligence, 6 (1987), pp. 229–242.
[31] P. Raghavan, *Probabilistic construction of deterministic algorithms: Approximating packing integer programs*, J. Comput. System Sci., 37 (1988), pp. 130–143.
[32] P. Raghavan and C. D. Thompson, *Randomized rounding: A technique for provably good algorithms and algorithmic proofs*, Combinatorica, 7 (1987), pp. 365–374.
[33] P. Raghavan and C. D. Thompson, *Multiterminal global routing: A deterministic approximation scheme*, Algorithmica, 6 (1991), pp. 73–82.
[34] F. Shahrokhi and D. W. Matula, *The maximum concurrent flow problem*, J. Association for Computing Machinery, 37 (1990), pp. 318–334.
[35] E. Shragowitz and S. Keel, *A global router based on a multicommodity flow model*, Integration, 5 (1987), pp. 3–16.
[36] B. S. Ting and B. N. Tien, *Routing techniques for gate array*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 2 (1983), pp. 301–312.
[37] A. Vannelli, *An adaptation of the interior point method for solving the global routing problem*, IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 10 (1991), pp. 193–203.
[38] G. Zimmermann, *Top-down design of digital systems*, in Advances in CAD for VLSI, Volume 2: Logic Design and Simulation, E. Hörbst, ed., North-Holland, New York, 1986, pp. 185–206.
[39] G. Zimmermann, *A new area and shape function estimation technique for VLSI layouts*, in Proceedings of the 25th Design Automation Conference, Anaheim, CA, ACM/IEEE, 1988, pp. 60–65.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 31–52

CONSTRAINT QUALIFICATIONS FOR SEMI-INFINITE SYSTEMS OF CONVEX INEQUALITIES*

WU LI†, CHANDAL NAHAK‡, AND IVAN SINGER§

Abstract. We introduce and study the Abadie constraint qualification, the weak Pshenichnyi–Levin–Valadier property, and related constraint qualifications for semi-infinite systems of convex inequalities and linear inequalities. Our main results are new characterizations of various constraint qualifications in terms of upper semicontinuity of certain multifunctions. Also, we give some applications of constraint qualifications to linear representations of convex inequality systems, to convex Farkas–Minkowski systems, and to formulas for the distance to the solution set. Some of our concepts and results are new even in the particular case of finite inequality systems.

Key words. constraint qualifications, semi-infinite inequality systems, convex Farkas–Minkowski systems, distance formulas

AMS subject classifications. 90C34

PII. S1052623499355247

1. Introduction.

Let $g_i : \mathbb{R}^n \to \mathbb{R} = (-\infty, +\infty)$ $(i \in I)$ be a family of convex functions, where I is an arbitrary (but nonempty) index set, and let us consider the system of "convex inequalities"

$$g_i(x) \le 0 \qquad (i \in I). \tag{1}$$

Throughout this paper we shall consider only the above framework, which is sufficient for many applications. However, let us mention that some of our results and proofs can be extended to arbitrary (finite or infinite dimensional) normed linear spaces X and to inequality systems (1) with convex functions $g_i : X \to \overline{\mathbb{R}} = [-\infty, +\infty]$ $(i \in I)$.

In what follows we shall assume, without any special mention, that the solution set S of the system (1) is nonempty, i.e.,

$$S := \{x \in \mathbb{R}^n \,|\, g_i(x) \le 0 \ (i \in I)\} \ne \emptyset. \tag{2}$$

We shall often consider the important particular case when each g_i is affine, say,

$$g_i(x) = \langle a_i, x\rangle - b_i \qquad (i \in I), \tag{3}$$

where $a_i \in \mathbb{R}^n$, $b_i \in \mathbb{R}$, and $\langle a_i, x\rangle$ denotes the dot product of vectors in $\mathbb{R}^n$. In this case, (1) becomes a system of linear inequalities,

$$\langle a_i, x\rangle \le b_i \qquad (i \in I), \tag{4}$$

*Received by the editors April 16, 1999; accepted for publication February 22, 2000; published electronically July 25, 2000.

http://www.siam.org/journals/siopt/11-1/35524.html

†Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529 (wli@odu.edu). The research of this author was partially supported by National Science Foundation grant NSF-DMS-9973218.

‡National Institute of Science & Technology, Palur Hills, Berhampur University, Berhampur-761008, Orissa, India (cnahak@hotmail.com). Current address: Department of Mathematics and Statistics, Old Dominion University, Norfolk, VA 23529. The work of this author was supported by DST, Government of India, under a BOYSCAST fellowship.

§Institute of Mathematics, P.O. Box 1-764, RO-70700, Bucharest, Romania (Ivan.Singer@imar.ro). The work of this author was partially supported by Ministry of Research and Technology grant 4073/1998.

and (2) becomes

$$S := \{x \in \mathbb{R}^n | \, \langle a_i, x\rangle \leq b_i \ (i \in I)\} \neq \emptyset. \tag{5}$$

Note that one can formally convert (1) to one convex inequality,

$$G(x) \leq 0, \tag{6}$$

where $G(\cdot)$ is the sup-function [7] of (1), defined as

$$G(x) := \sup_{i\in I} g_i(x) \qquad (x \in \mathbb{R}^n). \tag{7}$$

The system (1) is said to be a system *with finite-valued sup-function*, if

$$G(x) < +\infty \qquad (x \in \mathbb{R}^n). \tag{8}$$

In this paper, we always assume that (8) holds.

For the inequality system (1) and for any x in $\mathbb{R}^n$, we shall denote by $I(x)$ the set of "active indices" at x, i.e.,

$$I(x) := \{i \in I | \, g_i(x) = G(x)\}. \tag{9}$$

Note that if $G(x) = 0$ (in particular, if $x \in \operatorname{bd} S$), then $I(x) = \{i \in I | \, g_i(x) = 0\}$ is the classical definition of active indices.

One of the reasons for the difficulty of extending the results from finite inequality systems to semi-infinite inequality systems is that in the semi-infinite case for $x \in \operatorname{bd} S$ the set $I(x)$ may be empty or may be infinite. As we shall see in what follows, some other reasons, which explain why many results cannot be extended at all, or can be extended only under some additional assumptions (and sometimes only with different proofs), include the following: while in the finite case the index set I is compact, in our main results on the general semi-infinite case we shall assume no topology on I; also, while in the finite case for each $x \in \mathbb{R}^n$ the set $A_x := \{g_i(x) | \, i \in I\}$ is closed in $\mathbb{R}$, and the sup-function $G(x) := \sup_{i\in I} g_i(x)$ is always finite-valued on $\mathbb{R}^n$, in the general semi-infinite case these are no longer true. Furthermore, it is well known that for a linear inequality system (4) with a finite index set I we have

$$N_S(x) = \operatorname{cone}\{a_i\}_{i\in I(x)} \qquad (x \in \operatorname{bd} S), \tag{10}$$

where $N_S(x)$ and $\operatorname{bd} S$ denote the normal cone of S at x and the boundary of S, respectively. In general, (10) does not hold for a linear inequality system (4) if I is infinite. Given a convex system (1), another important well-known property for a finite I is (with the convention $\cup_{i\in\emptyset} A_i = \emptyset$)

$$\partial G(x) = \operatorname{co}\left\{\bigcup_{i\in I(x)} \partial g_i(x)\right\} \qquad (x \in \mathbb{R}^n), \tag{11}$$

where $\operatorname{co}(A)$ denotes the convex hull of a set A and $\partial g(x)$ denotes the subdifferential of a convex function g at x:

$$\partial g(x) := \{y \in \mathbb{R}^n | \ \langle y, z - x\rangle \leq g(z) - g(x) \ (z \in \mathbb{R}^n)\}. \tag{12}$$

In general, (11) does not hold if I is infinite.

In the present paper we shall give a detailed discussion of constraint qualifications for semi-infinite systems of convex inequalities and linear inequalities and the relations among them. We shall introduce and study the Abadie constraint qualification, the weak Pshenichnyi–Levin–Valadier (PLV) property, and related constraint qualifications. Our main results are new characterizations of various constraint qualifications in terms of upper semicontinuity of certain multifunctions. Also, we shall give some applications of constraint qualifications to linear representations of convex inequality systems, convex Farkas–Minkowski (FM) systems, and formulas for the distance to the solution set. Moreover, some of our concepts and results on semi-infinite convex inequality systems will yield new contributions *even when applied to the particular case of finite inequality systems* (such as Corollary 3.4).

Let us describe now, briefly, the sections of our paper.

It is well known (see, e.g., [7, pp. 307–309]) that in the theory of convex minimization over the solution set of a finite system of convex inequalities the so-called basic constraint qualification (BCQ), which requires that the normal cone at each point of the boundary of the solution set should coincide with "the cone of the active constraints" at that point, plays an important role; for example, it is satisfied if and only if the Karush–Kuhn–Tucker (KKT) sufficient optimality conditions are also necessary for optimality (see, e.g., [7, Proposition 2.2.1, p. 308]). Recently, the BCQ has been extended to semi-infinite linear inequality systems by Puente and Vera de Serio [14], who have used the term "locally Farkas–Minkowski systems," or briefly, LFM systems, and further extended to semi-infinite systems of convex inequalities by Goberna and López [5, p. 162], who have used the term "convex locally FM systems" (CLFM systems). In section 2 we shall introduce a weaker constraint qualification than the BCQ, which is different from the BCQ even in the particular case of finite convex inequality systems and which we shall call the *Abadie CQ*, requiring only that the normal cone at each point of the boundary of the solution set should coincide with the *closure* of the cone of the active constraints at that point. We shall give new characterizations of the Abadie CQ and the BCQ in terms of upper semicontinuity of certain associated convex cone-valued multifunctions.

In section 3 we shall introduce and study *the PLV property* and the *weak PLV property* of a semi-infinite convex inequality system at a point x, requiring that the subdifferential of the sup-function $G(\cdot)$ at x should coincide with (respectively, with the closure of) the convex hull of the subdifferentials of constraints corresponding to the active indices at that point; when this property holds for all points in the boundary of the solution set, we shall simply use the terms *PLV property* or *weak PLV property*, respectively. In the particular case of finite linear inequality systems the BCQ (and hence the Abadie CQ) is always satisfied, and for finite convex inequality systems so is the PLV property (whence also the weak PLV property) at all points of $\mathbb{R}^n$, but for semi-infinite inequality systems the situation is different. We shall give new characterizations of the PLV and weak PLV properties in terms of upper semicontinuity of certain associated multifunctions. We shall also show some connections among the PLV, weak PLV properties, the BCQ, and Abadie CQ.

In section 4 we shall be concerned with Slater conditions. In contrast with the case of finite systems of convex inequalities, in the semi-infinite case two different Slater conditions appear in a natural way: the usual one, requiring the existence of a point in the solution set, at which all inequalities of the system are satisfied as strict inequalities, and the so-called *strong Slater condition* (following the terminology of [5, p. 128]), in which the inequalities of the system are required to be satisfied uniformly

strictly, that is, in which the sup-function of the system is required to satisfy the usual Slater condition. We shall study the connections between the Slater conditions and the constraint qualifications discussed in sections 2 and 3; it will turn out that the situation concerning these connections is different from that occurring in the case of finite inequality systems. Also, we shall see that in the general semi-infinite case the usual Slater condition is too weak.

The final section is devoted to some applications of constraint qualifications.

Given a system of convex inequalities, we recall that any equivalent system of linear inequalities (i.e., a system of linear inequalities with the same solution set) is called a *linear representation* of the given convex system. It is well known that linear representations, and especially a certain simple one, which we shall call the "standard" linear representation, are useful tools in the study of convex inequality systems (see, e.g., [5] and the references therein). In subsection 5.1 we shall give a new linear representation of (finite or semi-infinite) convex inequality systems satisfying the Abadie CQ, which uses a much smaller subset of inequalities of the standard linear representation. Also, we shall show the connections between some properties of the initial convex inequality system and its representation.

In subsection 5.2 we shall extend from semi-infinite linear inequality systems to semi-infinite convex inequality systems the concepts of consequence relations and FM systems and, using the standard linear representation, we shall extend a known relation between linear FM systems and the BCQ, given in [14] and [5], to the case of convex FM systems. We shall also give a direct proof of this result, which, in contrast with the known proof for the linear case, does not use any subset of $\mathbb{R}^{n+1}$.

The exact formulas for the distance of a point to the solution set of a convex inequality system are important, among other reasons, for their connection with "asymptotic constraint qualifications" and for obtaining results on error bounds for such a system (see [11]). The well-known general formulas for the distance of a point to a closed convex set are not sufficiently useful for this purpose, since they do not exploit the special structure of the constraints of the inequality system. Up to the present, only a formula for the distance to the solution set of a semi-infinite *linear* inequality system has been known (for a dual version, see [4] and [19], and for the finite case, see [2]). In subsection 5.3, assuming the Abadie CQ or the BCQ, we shall give the first formulas for the distance of a point to the solution set of a semi-infinite system of *convex* inequalities, which are new even in the finite case. Also, using this result, we shall show that the distance of a point to the solution set of a convex inequality system (1) satisfying the BCQ is equal to the distance of that point to some finite subsystem of (1).

We conclude this section by introducing some notation which we shall use in this paper.

We shall consider $\mathbb{R}^n$ endowed with the usual scalar product $\langle \cdot, \cdot \rangle$, the Euclidean norm $\|\cdot\|$, and the topology induced by this norm. For an index set J, $|J|$ denotes the cardinality of J. Let A be a subset of $\mathbb{R}^n$. Then $\bar{A}$, $\operatorname{int}(A)$, and $\operatorname{bd}(A)$ denote the closure, the interior, and the boundary of A, respectively; $\operatorname{co}(A)$ and $\overline{\operatorname{co}}\,(A)$ are the convex hull and the closed convex hull of A, respectively; $\operatorname{cone}(A)$ and $\overline{\operatorname{cone}}\,(A)$ are the convex cone and the closed convex cone, generated by vectors in A, respectively; A^0 is the polar of A, i.e.,

$$A^0 := \{y \in \mathbb{R}^n | \, \langle x, y \rangle \leq 1 \ (x \in A)\}, \tag{13}$$

and $A^{00} = (A^0)^0$ is the bipolar of A; in the particular case when A is a cone, A^0

coincides with the "negative polar of A," i.e., we have

$$A^0 = \{y \in \mathbb{R}^n | \, \langle x, y\rangle \leq 0 \ (x \in A)\}. \tag{14}$$

The results of the present paper and of [11] were presented at the Workshop on Error Bounds and Applications in Mathematical Programming in Hong Kong, December 8–14, 1998. Auslender and Rubinov provided the references [13] and [8] in connection with Theorem 3.1. We received López [3] after the present paper had been completed. In order to compare our results with those of [3], we have inserted Remarks 2, 3(a), and 5(a).

2. Abadie CQ and BCQ. For characterizations of constraint qualifications for (1) we consider the convex cone generated by the subdifferentials of the active members of $G(\cdot)$ at x:

$$N'(x) := \operatorname{cone}\left(\bigcup_{i\in I(x)} \partial g_i(x)\right) \qquad (x \in \mathbb{R}^n). \tag{15}$$

We use $\overline{N'}(x)$ to denote the closure of $N'(x)$.

If every g_i is an affine function as defined in (3), then

$$N'(x) = \operatorname{cone}\{a_i\}_{i\in I(x)} \tag{16}$$

is the cone generated by the "active constraints" at x. Thus, in [5], $N'(x)$ is called "the cone of active constraints at x."

Let $T_S(x)$ be the tangent cone of S at x, i.e., $T_S(x) = \overline{\operatorname{cone}}(S - x)$. The normal cone of S at $x \in S$ is defined as

$$N_S(x) := \{y \in \mathbb{R}^n | \ \langle y, z - x\rangle \leq 0 \text{ for all } z \in S\}. \tag{17}$$

It is well known that $N_S(x) = T_S(x)^\circ$ (see, e.g., [7, Proposition 5.2.4, p. 137]).

Definition 2.1. *We shall say that the convex inequality system* (1) *satisfies*

(a) *the Abadie CQ at a point* $x \in \operatorname{bd} S$, *if*

$$T_S(x) = N'(x)^\circ \quad \textit{or equivalently} \quad N_S(x) = \overline{N'}(x); \tag{18}$$

(b) *the BCQ, at a point* $x \in \operatorname{bd} S$, *if*

$$N_S(x) = N'(x); \tag{19}$$

(c) *the Abadie CQ* (*respectively, the BCQ*), *if it satisfies the Abadie CQ* (*respectively, the BCQ*) *at all points* $x \in \operatorname{bd} S$.

Remark 1. (a) The equivalence of the two formulas in (18) follows from the bipolar theorem. In fact, if $N_S(x) = \overline{N'}(x)$, then, since $T_S(x)$ is a closed convex cone, by the bipolar theorem and by $N_S(x) = T_S(x)^0$ (see, e.g., [7, Proposition 5.2.4, p. 137]) we have

$$T_S(x) = T_S(x)^{00} = N_S(x)^0 = \overline{N'(x)}^0 = N'(x)^0.$$

On the other hand, if $T_S(x) = N'(x)^0$, then, by $N_S(x) = T_S(x)^0$ and the bipolar theorem, and since $N'(x)$ is a cone, we obtain

$$N_S(x) = T_S(x)^0 = N'(x)^{00} = \overline{N'(x)}.$$

(b) Since we always have (see, e.g., [7, Lemma 4.4.1, p. 267] and [7, Lemma 2.1.3, p. 305])

$$\overline{N'}(x) \subseteq \overline{\text{cone}}(\partial G(x)) \subseteq N_S(x) \quad \text{for } x \in \text{bd}\, S, \tag{20}$$

$$N'(x) \subseteq \text{cone}(\partial G(x)) \subseteq N_S(x) \quad \text{for } x \in \text{bd}\, S, \tag{21}$$

the system (1) satisfies the Abadie CQ at $x \in \text{bd}\, S$ if and only if

$$N_S(x) \subseteq \overline{N'}(x) \tag{22}$$

and it satisfies the BCQ at $x \in \text{bd}\, S$ if and only if

$$N_S(x) \subseteq N'(x). \tag{23}$$

Moreover, for $x \in \text{bd}\, S$, by (20), (1) satisfies the Abadie CQ at x if and only if $\overline{N'}(x) = \overline{\text{cone}}(\partial G(x))$ and (6) satisfies the Abadie CQ at x. Similarly, by (21), (1) satisfies the BCQ at x if and only if $N'(x) = \text{cone}(\partial G(x))$ and (6) satisfies the BCQ at x. Clearly, (1) satisfies the BCQ at x if and only if it satisfies the Abadie CQ at x and $N'(x)$ is closed. Thus, (1) satisfies the BCQ at x if and only if $T_S(x) = N'(x)^\circ$ and $N'(x)$ is closed. The points $x \in \text{bd}\, S$ with the latter property have been called "Lagrangian regular points" in [12, Definition 3.3].

(c) When I is finite, Definition 2.1 is the classical definition of the Abadie CQ introduced by Abadie (see [1], also [10]) and, respectively, of the BCQ (see [7, p. 207]). If I is finite and each g_i is a differentiable convex function, then $\partial g_i(x) = \{\nabla g_i(x)\}$ (where $\nabla g_i(x)$ is the gradient of g_i at x) and the cone $N'(x)$ of (15) is closed (see, e.g., [7, Lemma 4.3.3, p. 130]). In this case, the Abadie CQ and the BCQ coincide.

(d) The BCQ, in an equivalent form, has been introduced for semi-infinite linear inequality systems (4) in [14] (called LFM systems) and extended to convex inequality systems (1) in [5, pp. 162–163] (called convex LFM constraint systems).

When I is finite and each g_i is an affine function, the BCQ, and hence also the Abadie CQ, are satisfied (see, e.g., [7, Example 5.2.6(b), p. 138]). When each g_i is affine, for semi-infinite systems of linear inequalities, the Abadie CQ may not hold and the Abadie CQ is not the same as the BCQ, as shown by the following example.

Example 1. Let $n = 2$.

(a) Semi-infinite linear systems without the Abadie CQ.
Let $I = \{0, 1, 2, \dots\}$ and

$$g_i(x) := \begin{cases} x_2 - 1 & \text{if } i = 0, \\ -x_1 & \text{if } i = 1, \\ x_1 - 1 & \text{if } i = 2, \\ -x_2 - \frac{1}{i} & \text{if } i = 3, 4, \dots \end{cases} \qquad (x = (x_1, x_2) \in \mathbb{R}^2). \tag{24}$$

Then $S = \{(x_1, x_2) |\ \ 0 \le x_1 \le 1,\ 0 \le x_2 \le 1\}, 0 \in \text{bd}\, S$, and $N_S(0) = \{(t_1, t_2) |\ \ t_1 \le 0,\ t_2 \le 0\}$. Also, $I(0) = \{i \in I |\, g_i(0) = 0\} = \{1\}$, whence $N'(0) = \text{cone}\{(-1, 0)\} = \{(-t, 0) |\, t \ge 0\}$. Thus the family (24) does not satisfy the Abadie CQ.

(b) Semi-infinite linear systems with the Abadie CQ but without BCQ.
Let $I = \{1, 2, \dots\}$ and

$$g_i(x) = x_1 + \frac{1}{i} x_2 \qquad (x = (x_1, x_2) \in \mathbb{R}). \tag{25}$$

Then $S = \{x \in \mathbb{R}^2 |\, x_1 \le 0, x_1 + x_2 \le 0\}, 0 \in \operatorname{bd} S$, and $N_S(0) = \{(t_1, t_2) |\, 0 \le t_2 \le t_1\}$. Also, $I(0) = \{i \in I |\, g_i(0) = 0\} = I$, whence $N'(0) = \operatorname{cone}\left\{\left(1, \frac{1}{i}\right) | i \in I\right\} = \{(t_1, t_2) |\, 0 < t_2 \le t_1\} \cup \{(0,0)\}$. Thus the family (25) satisfies the Abadie CQ, but not the BCQ.

Next we give new characterizations of the Abadie CQ and the BCQ for (1), in terms of the upper semicontinuity of the multifunctions $\overline{N'}(\cdot)$, $\overline{\operatorname{cone}}(\partial G(\cdot))$, $N'(\cdot)$, and $\operatorname{cone}(\partial G(\cdot))$. We recall (see, e.g., [16, p. 55]) that a multifunction (i.e., a set-valued function) $Q : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ (the collection of subsets of $\mathbb{R}^n$) is said to be *upper semicontinuous in the sense of Kuratowski,* or briefly, *upper semicontinuous, at* $x \in \mathbb{R}^n$, if the relations $\lim_{k\to+\infty} x_k = x$, $\lim_{k\to+\infty} y_k = y \in \mathbb{R}^n, y_k \in Q(x_k)$ $(k = 1, 2, \dots)$ imply $y \in Q(x)$. Clearly, the graph of Q (i.e., the set $\{(x, y) |\, x \in \mathbb{R}^n, y \in Q(x)\}$) is closed if and only if Q is upper semicontinuous at all $x \in \mathbb{R}^n$.

We shall first prove a lemma, in which we shall use the convex hull of the subdifferentials of the active members of $G(x)$, that is, the set

$$D'(x) := \operatorname{co}\left(\bigcup_{i\in I(x)} \partial g_i(x)\right) \qquad (x \in \mathbb{R}^n). \tag{26}$$

LEMMA 2.2. *Let $x \in \mathbb{R}^n$ and let $Q : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ be a multifunction such that, for all z in a neighborhood of x, $Q(z)$ is a convex set, $I(z) \neq \emptyset$, and $D'(z) \subset Q(z)$. If Q is upper semicontinuous at x, then $\partial G(x) \subset Q(x)$.*

Proof. Assume on the contrary that there exists $\overline{y} \in \partial G(x) \setminus Q(x)$. Since Q is upper semicontinuous at x and $Q(x)$ is a convex set, $Q(x)$ is a closed convex set. Then, by the strict separation theorem [7, Theorem 4.1.1, p. 121], there exists $u \in \mathbb{R}^n \setminus \{0\}$ such that

$$\langle \overline{y}, u\rangle > \sup_{y\in Q(x)} \langle y, u\rangle.$$

Let $G'(x; u)$ be the directional derivative of G at x in the direction u. By $\overline{y} \in \partial G(x)$ and a well-known formula for $G'(x; u)$ (see, e.g., [7, p. 240]), we get

$$G'(x; u) = \sup_{y\in\partial G(x)} \langle y, u\rangle \ge \langle \overline{y}, u\rangle > \sup_{y\in Q(x)} \langle y, u\rangle. \tag{27}$$

Let $t_k = \frac{1}{k}$ $(k = 1, 2, \dots)$. Since $I(z) \neq \emptyset$ for z in a neighborhood of x, without loss of generality we can assume $I(x + t_k u) \neq \emptyset$. Let $i_k \in I(x + t_k u)$ and $y_k \in \partial g_{i_k}(x + t_k u) \subseteq \partial G(x + t_k u)$ $(k = 1, 2, \dots)$. Then

$$-t_k \langle y_k, u\rangle = \langle y_k, x - (x + t_k u)\rangle \le G(x) - G(x + t_k u),$$

whence, by $t_k > 0$ $(k = 1, 2, \dots)$,

$$\langle y_k, u\rangle \ge \frac{G(x + t_k u) - G(x)}{t_k} \qquad (k = 1, 2, \dots). \tag{28}$$

Since $y_k \in \partial G(x + t_k u)$ $(k = 1, 2, \dots)$ and since $\{x + t_k u |\, k \ge 1\}$ is bounded, the set $\{y_k : k \ge 1\}$ is bounded as well (see, e.g., [7, Proposition 6.2.2, p. 282]). Hence, we may assume, without loss of generality, that $y_k \to \widehat{y}$. Then, letting $k \to +\infty$ in (28) and using (27), we obtain

$$\langle \widehat{y}, u\rangle \ge G'(x; u) > \sup_{y\in Q(x)} \langle y, u\rangle. \tag{29}$$

But, since

$$\lim_{k\to+\infty}(x+t_ku)=x,\ \ \lim_{k\to+\infty}y_k=\widehat{y},$$
$$y_k\in\partial g_{i_k}(x+t_ku)\subseteq D'(x+t_ku)\subseteq Q(x+t_ku)\qquad(k=1,2,\dots),$$

and since Q is upper semicontinuous at x, we have $\widehat{y}\in Q(x)$, which contradicts (29). □

THEOREM 2.3. *Let $x\in\operatorname{bd}S$ and $I(z)\neq\emptyset$ for z in a neighborhood of x. Then the following two statements are equivalent.*

(a) (1) *satisfies the Abadie CQ at x.*

(b) *Both $\overline{\operatorname{cone}}(\partial G(\cdot))$ and $\overline{N'}(\cdot)$ are upper semicontinuous at x.*

Proof. (a)⇒(b): Let $x_k\to x$, $y_k\in\overline{N'}(x_k)$ (or $y_k\in\overline{\operatorname{cone}}[\partial G(x_k)]$) and $y_k\to y$. We claim that $y\in N_S(x)$. To prove the claim, let

$$S_k:=\{z\in\mathbb{R}^n\mid G(z)\le G(x_k)\}. \tag{30}$$

Since $y_k\in\overline{N'}(x_k)\subseteq\overline{\operatorname{cone}}(\partial G(x_k))\subseteq N_{S_k}(x_k)$ (see (20)), we have

$$\langle y_k,z-x_k\rangle\le 0\quad\text{for } z\in S_k. \tag{31}$$

We consider two cases.

Case 1. (6) does not satisfy the Slater condition, that is, $G(z)\ge 0$ for all $z\in\mathbb{R}^n$. Then $G(x_k)\ge 0$ and $S\subseteq S_k$. Thus, (31) implies

$$\langle y_k,z-x_k\rangle\le 0\quad\text{for } z\in S.$$

Letting $k\to+\infty$ in the above inequality we obtain

$$\langle y,z-x\rangle\le 0\quad\text{for } z\in S.$$

That is, $y\in N_S(x)$.

Case 2. (6) satisfies the Slater condition, that is, $G(\hat{x})<0$ for some $\hat{x}\in\mathbb{R}^n$. Then $\operatorname{int}S=\{z\in\mathbb{R}^n:G(z)<0\}$ and $\overline{\operatorname{int}S}=S$. Let $z\in\operatorname{int}S$. Then there is $k_0>0$ such that $z\in S_k$ for $k\ge k_0$ (since $G(x_k)\to G(x)=0$). By (31), we have

$$\langle y,z-x\rangle=\lim_{k\to+\infty}\langle y_k,z-x_k\rangle\le 0. \tag{32}$$

Since (32) holds for any $z\in\operatorname{int}S$, it also holds for $z\in\overline{\operatorname{int}S}=S$. Thus, $y\in N_S(x)$.

If (1) satisfies the Abadie CQ at x, we get $y\in N_S(x)=\overline{\operatorname{cone}}(\partial G(x))=\overline{N'}(x)$ (see (20)). Therefore, both $\overline{N'}(\cdot)$ and $\overline{\operatorname{cone}}(\partial G(\cdot))$ are upper semicontinuous at x.

(b)⇒(a): Let $y\in N_S(x)\setminus\{0\}$. Let $x_0:=x+y$. Then $x_0\notin S$ (since $x_0=x+y\in S$ and $y\in N_S(x)\setminus\{0\}$ would imply $\langle y,x\rangle\ge\langle y,x_0\rangle=\langle y,x\rangle+\langle y,y\rangle>\langle y,x\rangle$, which is impossible) and hence $G(x_0)>0$. Let

$$S_k:=\left\{z\in\mathbb{R}^n\mid G(z)\le\frac{1}{k}\right\} \tag{33}$$

and assume k large enough so that $\frac{1}{k}<G(x_0)$, that is, $x_0\notin S_k$. Let $x_k^*\in S_k$ be such that $\operatorname{dist}(x_0,S_k)=\|x_0-x_k^*\|$. Then, since $x\in S_k$, we have

$$\|x_0-x_k^*\|\le\|x_0-x\|. \tag{34}$$

Thus, $\{x_k^*\}$ is a bounded sequence. Without loss of generality we may assume that $x_k^* \to \hat{x}$. Since $G(\hat{x}) = \lim_{k\to+\infty} G(x_k^*) \le \lim_{k\to+\infty} \frac{1}{k} = 0$, we have $\hat{x} \in S$. Letting $k \to +\infty$ in (34) we obtain

$$\|x_0 - \hat{x}\| \le \|x_0 - x\|. \tag{35}$$

On the other hand, since $x_0 - x = y \in N_S(x)$ and $x \in \operatorname{bd} S$, x is the projection of x_0 onto S. Hence, (35) and $\hat{x} \in S$ imply that $\hat{x} = x$. Consequently, $x_k^* \to x$ and $x_0 - x_k^* \to x_0 - x = y$.

We claim that $x_0 - x_k^* \in \operatorname{cone}(\partial G(x_k^*))$. Indeed, by $\operatorname{dist}(x_0, S_k) = \|x_0 - x_k^*\|$ we have $x_k^* \in \operatorname{bd} S_k$ and $x_0 - x_k^* \in N_{S_k}(x_k^*)$. Since $G(z) \le \frac{1}{k}$ satisfies the Slater condition and $x_k^* \in \operatorname{bd} S_k$, we have $N_{S_k}(x_k^*) = \operatorname{cone}(\partial G(x_k^*))$ (see, e.g., [7, Theorem 1.3.5, p. 245]). Thus, $x_0 - x_k^* \in \operatorname{cone}(\partial G(x_k^*))$, which proves the claim.

Since $x_k^* \to x, x_0 - x_k^* \in \operatorname{cone}(\partial G(x_k^*))$, and $x_0 - x_k^* \to y$, by the upper semicontinuity of $\overline{\operatorname{cone}}(\partial G(\cdot))$ at x we get $y \in \overline{\operatorname{cone}}(\partial G(x))$. Hence, since y was an arbitrary nonzero element in $N_S(x)$, $N_S(x) \subseteq \overline{\operatorname{cone}}(\partial G(x))$. On the other hand, by the upper semicontinuity of $\overline{N'}(\cdot)$ at x and Lemma 2.2 (applied to $Q = \overline{N'}$), we have $\partial G(x) \subseteq \overline{N'}(x)$, which implies $\overline{\operatorname{cone}}(\partial G(x)) \subseteq \overline{N'}(x)$. Therefore, we have (22), and thus (1) satisfies the Abadie CQ at x. □

THEOREM 2.4. *Let $x \in \operatorname{bd} S$ and $I(z) \ne \emptyset$ for z in a neighborhood of x. Then the following two statements are equivalent.*

(a) (1) *satisfies the BCQ at x.*

(b) *Both* $\operatorname{cone}(\partial G(\cdot))$ *and* $N'(\cdot)$ *are upper semicontinuous at x.*

Proof. (a)⇒(b): By Theorem 2.3, both $\overline{\operatorname{cone}}(\partial G(\cdot))$ and $\overline{N'}(\cdot)$ are upper semicontinuous at x. Since $N'(x) = \operatorname{cone}(\partial G(x)) = N_S(x)$ and $N_S(x)$ is closed, we know that $N'(x)$ and $\operatorname{cone}(\partial G(x))$ must be closed. Thus, both $\operatorname{cone}(\partial G(\cdot))$ and $N'(\cdot)$ are upper semicontinuous at x.

(b)⇒(a): By Theorem 2.3, (1) satisfies the Abadie CQ at x. But (b) also implies that $N'(x)$ is a closed set. So $N'(x) = \overline{N'}(x) = N_S(x)$ and (1) satisfies the BCQ. □

From Theorems 2.3 and 2.4 we obtain the following characterizations of the Abadie CQ and the BCQ.

COROLLARY 2.5. *Suppose that $I(z) \ne \emptyset$ for z in a neighborhood of* $\operatorname{bd} S$. *Then the following two statements are true.*

(a) (1) *satisfies the Abadie CQ if and only if both* $\overline{\operatorname{cone}}(\partial G(\cdot))$ *and* $\overline{N'}(\cdot)$ *are upper semicontinuous at every x in* $\operatorname{bd} S$.

(b) (1) *satisfies the BCQ if and only if both* $\operatorname{cone}(\partial G(\cdot))$ *and* $N'(\cdot)$ *are upper semicontinuous at every x in* $\operatorname{bd} S$.

Remark 2. Fajardo and López [3, Theorem 3.1(i)] proved that if (1) satisfies the BCQ, then the multifunction $A(x) := \{y \in N'(x) | \ \|y\| \le 1\}$ $(x \in \mathbb{R}^n)$ is (B)-upper semicontinuous on S (which is equivalent to the upper semicontinuity of $N'(\cdot)$ on S) and that if (6) satisfies the Slater condition, then the converse is also true [3, Theorem 3.1(iib)].

Here a mapping $Q : \mathbb{R}^n \to 2^{\mathbb{R}^n}$ is said to be (B)-*upper semicontinuous* at $x \in \mathbb{R}^n$, if for every open set W in $\mathbb{R}^n$ containing $Q(x)$ there exists a neighborhood $V(x)$ of x such that $Q(z) \subset W$ for each $z \in V(x)$; furthermore, Q is said to be (B)-*upper semicontinuous* on a set $M \subset \mathbb{R}^n$ if it is (B)-upper semicontinuous at each $x \in M$. It is well known (see, e.g., [5, p. 128]) that if Q is (B)-upper semicontinuous (at x), then it is upper semicontinuous (at x), but the converse is not true. One can show that if the set $\{z | \ z \in Q(y)$ for y in some neighborhood of $x\}$ is bounded, then the (B)-upper semicontinuity of $Q(\cdot)$ at x is equivalent to the upper semicontinuity of $Q(\cdot)$ at x.

3. Subdifferentials of the sup-function and its active members. In this section, we study the relations between the subdifferential of the sup-function G and the subdifferentials of its active member functions $\{g_i|\ i \in I(x)\}$.

We shall use the set $D'(x)$ defined by (26). Note that

$$N'(x) = \text{cone}(D'(x)) \qquad (x \in \mathbb{R}^n). \tag{36}$$

We shall denote by $\overline{D'}(x)$ the closure of the set $D'(x)$.

One important property of (1) when I is finite is the equality (11) (see, e.g., [13, Theorem 1.4]), which can be rewritten as

$$\partial G(x) = D'(x) \qquad (x \in \mathbb{R}^n). \tag{37}$$

The above equality means that the subdifferential of the sup-function $G(\cdot)$ is the convex hull of subdifferentials of its active members.

In general, (37) does not hold if I is infinite. The following sufficient (but not necessary) condition that guarantees $\partial G(x) = D'(x)$ for all $x \in \mathbb{R}^n$, and is satisfied when I is finite, has been given by Levin [8, Theorem 2] (and, at the same time, Valadier [18, Theorem 2] has obtained, in an arbitrary topological linear space instead of $\mathbb{R}^n$, the weaker conclusion in which D' is replaced by $\overline{D'}$).

THEOREM 3.1 (PLV theorem [13, 8, 18]). *If I is a compact set (in some metric space), and $g_i : \mathbb{R}^n \to \mathbb{R}$ $(i \in I)$ is a family of convex functions such that for each fixed $x \in \mathbb{R}^n$ the function $i \to g_i(x)$ is upper semicontinuous on I, then* (37) *holds at all $x \in \mathbb{R}^n$.*

Here a real-valued function $f(t)$ is said to be upper semicontinuous at $t = t_0$ if

$$\overline{\lim_{t \to t_0}}\ f(t) \le f(t_0), \tag{38}$$

i.e.,

$$\lim_{\epsilon \to 0+}\ \sup_{|t - t_0| \le \epsilon} f(t) \le f(t_0). \tag{39}$$

Note that (39) is equivalent to the upper semicontinuity of the multifunction $F(t) = \{y \in \mathbb{R}|\ y \le f(t)\}$ at t_0.

For convenience, we shall introduce the following definition related to (37).

DEFINITION 3.2. *Let* (1) *be a convex inequality system with a finite-valued sup-function. We shall say that the family $\{g_i|\, i \in I\}$, or the system* (1)*, has*

(a) *the weak PLV property at a point $x \in \mathbb{R}^n$, if*

$$\partial G(x) = \overline{D'}(x); \tag{40}$$

(b) *the PLV property at a point $x \in \mathbb{R}^n$, if*

$$\partial G(x) = D'(x); \tag{41}$$

(c) *the weak PLV property (respectively, the PLV property), if it has the weak PLV property (respectively, the PLV property) at all $x \in \text{bd}\, S$.*

One major problem with an infinite I is the possibility of $I(x) = \emptyset$ at $x \in \text{bd}\, S$ (which cannot happen when I is finite). For example, given a family of convex functions $g_i : \mathbb{R}^n \to \mathbb{R}$, let $g_{i,k}(x) := g_i(x) - \frac{1}{k}$ for $i \in I$ and $k = 1, 2, \ldots$. Then (1) holds if and only if

$$g_{i,k}(x) \le 0 \quad \text{for } k = 1, 2, \ldots , \quad i \in I. \tag{42}$$

However, the active index set for (42) is empty at every $x \in \operatorname{bd} S$ and so there is no (weak) PLV property for (42).

One can avoid the inconvenience of having $I(x) = \emptyset$ by requiring the closedness of $\{g_i(x) | \, i \in I\}$.

PROPOSITION 3.3. *Let $x \in \operatorname{bd} S$. If the set $A_x := \{g_i(x) | \, i \in I\}$ is closed in $\mathbb{R}$, then $I(x) \neq \emptyset$.*

Proof. Since $x \in \operatorname{bd} S$, we have $G(x) = 0$ (by the continuity of G). Taking any sequence $\{i_k\} \subseteq I$ such that $g_{i_k}(x) \to \sup_{i \in I} g_i(x) = G(x) = 0$ as $k \to +\infty$, by the closedness of A_x we obtain $0 \in A_x$, so $I(x) \neq \emptyset$. □

However, even if $I(x) \neq \emptyset$, (4) (and hence, in general, (1)) might not have the weak PLV property. Also, in general, the weak PLV property is not the same as the PLV property.

Example 2. Let $n = 1$.

(a) A semi-infinite linear system without weak PLV property.
Let $I = \{0; (1,k); (2,k) | \, k = 1, 2, \dots\}$ and

(43)
$$g_0(x) = 3x, \ g_{1,k}(x) = 2x - \frac{1}{k}, \ g_{2,k}(x) = 4x - \frac{1}{k} \quad (x \in \mathbb{R}, k = 1, 2, \dots).$$

Then
$$G(x) = \sup_{k \geq 1} \{g_0(x), g_{1,k}(x), g_{2,k}(x)\} = \begin{cases} 2x & \text{if } x < 0, \\ 4x & \text{if } x \geq 0. \end{cases}$$

Thus, $S = (-\infty, 0]$, $\operatorname{bd} S = \{0\}$, $I(0) = \{i \in I | \, g_i(0) = G(0) = 0\} = \{0\}$, and $\partial G(0) = [2, 4]$, $D'(0) = \{3\}$. Therefore, the system does not have the weak PLV property at $x = 0$.

(b) A semi-infinite linear system with weak PLV property, but without PLV property.
Let $I = \{(1,k); (2,k) | \, k = 1, 2, \dots\}$ and

(44)
$$g_{1,k}(x) = \left(1 + \frac{1}{k}\right) x, \ g_{2,k}(x) = \left(5 - \frac{1}{k}\right) x \quad (x \in \mathbb{R}, k = 1, 2, \dots).$$

Then
$$G(x) = \sup_{k \geq 1} \{g_{1,k}(x), g_{2,k}(x)\} = \begin{cases} x & \text{if } x < 0, \\ 5x & \text{if } x \geq 0. \end{cases}$$

Thus, $S = (-\infty, 0]$, $\operatorname{bd} S = \{0\}$, $I(0) = \{i \in I | \, g_i(0) = G(0) = 0\} = I$, and $\partial G(0) = [1, 5]$, $D'(0) = (1, 5)$. Therefore, the system has the weak PLV property at $x = 0$ but not the PLV property.

There is no relation of implication between the CQs and the PLV properties. Indeed, in Example 2(a), the system actually satisfies the BCQ at $x = 0$, but the PLV property does not hold at $x = 0$; while for any I with one index the PLV property is trivially true, but (1) may not satisfy the Abadie CQ. However, when (1) satisfies the PLV property (in particular, when I is finite), $N'(x) \equiv \operatorname{cone}(\partial G(x))$ and the characterizations for the Abadie CQ and the BCQ can be simplified.

COROLLARY 3.4. *Suppose that* (1) *satisfies the PLV property and $I(x) \neq \emptyset$ for x in a neighborhood of* $\operatorname{bd} S$ (*e.g., this happens when I is finite*). *Let $x \in \operatorname{bd} S$. Then the following two statements are true.*

(a) (1) *satisfies the Abadie CQ at x if and only if $\overline{N'}(\cdot)$ is upper semicontinuous at x.*

(b) (1) *satisfies the BCQ at x if and only if $N'(\cdot)$ is upper semicontinuous at x.*

Proof. By the assumption, $D'(x) = \partial G(x)$ for all x, which implies $N'(x) = \text{cone}\,(D'(x)) = \text{cone}(\partial G(x)) \neq \emptyset$ for all x. Thus Corollary 3.4 follows from Corollary 2.5. □

The following proposition shows that if (1) satisfies the (weak) PLV property, then (1) and (6) are "equivalent" in terms of CQs.

PROPOSITION 3.5. *Let $x \in \text{bd}\, S$.*

(a) *Suppose that* (1) *satisfies the weak PLV property at x. Then* (1) *satisfies the Abadie CQ at x if and only if* (6) *satisfies the Abadie CQ at x.*

(b) *Suppose that* (1) *satisfies the PLV property at x. Then* (1) *satisfies the BCQ at x if and only if* (6) *satisfies the BCQ at x.*

Proof. (a) If (1) satisfies the weak PLV property at x, then, by (40) and (36), $\overline{\text{cone}}(\partial G(x)) = \overline{\text{cone}}(\overline{D'}(x)) = \overline{N'}(x)$. Thus, $N_S(x) = \overline{N'}(x)$ if and only if $N_S(x) = \overline{\text{cone}}(\partial G(x))$.

(b) If (1) satisfies the PLV property at x, then, by equations (41) and (36), $\text{cone}(\partial G(x)) = \text{cone}(D'(x)) = N'(x)$. Thus, $N_S(x) = N'(x)$ if and only if $N_S(x) = \text{cone}(\partial G(x))$. □

It turns out the the weak PLV property or the PLV property at a point $x \in \mathbb{R}^n$ can be characterized by the upper semicontinuity at x of the multifunction $\overline{D'}(\cdot)$ or $D'(\cdot)$, respectively.

THEOREM 3.6. *Let $x \in \mathbb{R}^n$ and $I(z) \neq \emptyset$ for z in a neighborhood of x. Then $\partial G(x) = \overline{D'}(x)$ if and only if $\overline{D'}(\cdot)$ is upper semicontinuous at x.*

Proof. If $\overline{D'}(\cdot)$ is upper semicontinuous at x, then, by Lemma 2.2, $\partial G(x) \subseteq \overline{D'}(x)$. Since $\partial G(x) \supseteq \overline{D'}(x)$ always holds, we have $\partial G(x) = \overline{D'}(x)$.

Next we assume that $\partial G(x) = \overline{D'}(x)$ and prove the upper semicontinuity of $\overline{D'}(\cdot)$ at x.

Let $\lim_{k\to+\infty} x_k = x$, $\lim_{k\to+\infty} y_k = y \in \mathbb{R}^n, y_k \in \overline{D'}(x_k)$ $(k = 1, 2, \dots)$. By $\overline{D'}(x_k) \subseteq \partial G(x_k)$, we have $y_k \in \partial G(x_k)$. Since G is a finite convex function, $\partial G(z) \neq \emptyset$ for all $z \in \mathbb{R}^n$ and ∂G is upper semicontinuous (see, e.g., [7, Proposition 6.2.1, p. 282]). Thus, $y \in \partial G(x)$. Since $\partial G(x) = \overline{D'}(x)$, we have $y \in \overline{D'}(x)$ and so $\overline{D'}(\cdot)$ is upper semicontinuous at x. □

THEOREM 3.7. *Let $x \in \mathbb{R}^n$ and $I(z) \neq \emptyset$ for z in a neighborhood of x. Then $\partial G(x) = D'(x)$ if and only if $D'(\cdot)$ is upper semicontinuous at x.*

Proof. Note that $D'(\cdot)$ is upper semicontinuous at x if and only if $\overline{D'}(\cdot)$ is upper semicontinuous at x and $D'(x)$ is a closed set.

If $D'(\cdot)$ is upper semicontinuous at x, then, by Theorem 3.6, $\partial G(x) = \overline{D'}(x) = D'(x)$. On the other hand, if $\partial G(x) = D'(x)$, then by the upper semicontinuity of $\partial G(\cdot)$, $D'(x)$ is a closed set and $\partial G(x) = \overline{D'}(x)$. Hence, by Theorem 3.6, $\overline{D'}(\cdot)$ is upper semicontinuous at x. Therefore, $D'(\cdot)$ is upper semicontinuous at x. □

Using Theorems 3.6 and 3.7 we obtain the following characterizations of the weak PLV and PLV properties at all $x \in \mathbb{R}^n$.

THEOREM 3.8. *The following statements are true.*

(a) *$\partial G(x) = \overline{D'}(x)$ for all $x \in \mathbb{R}^n$ if and only if $\overline{D'}(x) \neq \emptyset$ for all $x \in \mathbb{R}^n$ and $\overline{D'}(\cdot)$ is upper semicontinuous on $\mathbb{R}^n$.*

(b) *$\partial G(x) = D'(x)$ for all $x \in \mathbb{R}^n$ if and only if $D'(x) \neq \emptyset$ for all $x \in \mathbb{R}^n$ and $D'(\cdot)$ is upper semicontinuous on $\mathbb{R}^n$.*

Proof. Since $G(\cdot)$ is a finite convex function, $\partial G(x) \neq \emptyset$ for any x. Therefore,

every condition in the above theorem implies $I(x) \neq \emptyset$ for all $x \in \mathbb{R}^n$. Consequently, Theorem 3.8 follows from Theorems 3.6 and 3.7. □

Remark 3. (a) Fajardo and López [3, Theorem 4.1(i)] proved that if $\overline{D'}(x) \neq \emptyset$ for all $x \in \mathbb{R}^n$ and $\overline{D'}(\cdot)$ is (B)-upper semicontinuous on $\mathbb{R}^n$ (see Remark 2 above), then $\partial G(x) = \overline{D'}(x)$ for all $x \in \mathbb{R}^n$.

Since the (B)-upper semicontinuity of $\overline{D'}(\cdot)$ is equivalent to the upper semicontinuity of $\overline{D'}(\cdot)$ (see Remark 2), the "if" part of Theorem 3.8(a) is equivalent to [3, Theorem 4.1(i)].

(b) Using Theorem 3.8, we can give a new proof of Theorem 3.1, which seems simpler and more natural than the proofs known in the literature (see, e.g., the proof of Theorem 4.4.2 [7, p. 267]. To this end, let us first prove the following fact.

LEMMA 3.9. *If I is a compact metric space and if $g_i : \mathbb{R}^n \to \mathbb{R}$ $(i \in I)$ is a family of convex functions such that for each $x \in \mathbb{R}^n$ the function $i \to g_i(x)$ is upper semicontinuous on I, then the set-valued mapping $x \to I(x)$ is upper semicontinuous on $\mathbb{R}^n$ and the set-valued mapping $(x, i) \to \partial g_i(x)$ is upper semicontinuous on W, where $W := \{(x, i) | \, i \in I(x)\}$. That is, if $\lim_{k\to+\infty} x_k = \hat{x}$, $\lim_{k\to+\infty} i_k = \hat{i}$, and $\lim_{k\to+\infty} y_k = \hat{y}$ with $i_k \in I(x_k)$ and $y_k \in \partial g_{i_k}(x_k)$ $(k = 1, 2, \dots)$, then $\hat{i} \in I(\hat{x})$ and $\hat{y} \in \partial g_{\hat{i}}(\hat{x})$.*

Proof. First we prove $\hat{i} \in I(\hat{x})$ by contradiction. In fact, if $g_{\hat{i}}(\hat{x}) < G(\hat{x})$, by the upper semicontinuity of $g_i(\hat{x})$ with respect to i, there exist a positive constant δ and a neighborhood $O(\hat{i})$ of $\hat{i}$ in I such that

$$g_i(\hat{x}) \leq G(\hat{x}) - \delta \quad \text{for } i \in O(\hat{i}). \tag{45}$$

Let $\hat{G}(x) := \sup_{i\in O(\hat{i})} g_i(x)$. By the assumptions, $\hat{G}(x) \leq G(x) < +\infty$ so $G(x)$ and $\hat{G}(x)$ are continuous convex functions. Thus,

$$G(\hat{x}) = \lim_{k\to+\infty} G(x_k) = \lim_{k\to+\infty} g_{i_k}(x_k) = \lim_{k\to+\infty, i_k\in O(\hat{i})} g_{i_k}(x_k) \leq \lim_{k\to+\infty} \hat{G}(x_k) = \hat{G}(\hat{x}). \tag{46}$$

But (45) implies that $\hat{G}(\hat{x}) \leq G(\hat{x}) - \delta$, a contradiction to (46). This proves that $\hat{i} \in I(\hat{x})$. Now, for any $z \in \mathbb{R}^n$, we have

$$\langle y_k, z - x_k\rangle \leq g_{i_k}(z) - g_{i_k}(x_k) \qquad (k = 1, 2, \dots). \tag{47}$$

Since $i_k \in I(x_k)$ $(k = 1, 2, \dots)$ and $\hat{i} \in I(\hat{x})$, we have

$$g_{i_k}(x_k) = G(x_k) \to G(\hat{x}) = g_{\hat{i}}(\hat{x}). \tag{48}$$

Thus, letting $k \to +\infty$ in (47) and using $\overline{\lim}_{k\to+\infty} g_{i_k}(z) \leq g_{\hat{i}}(z)$ (by the assumption of upper semicontinuity of the mapping $i \to g_i(z)$), we obtain

$$\langle y, z - \hat{x}\rangle \leq g_{\hat{i}}(z) - g_{\hat{i}}(\hat{x}). \tag{49}$$

Since (49) holds for any z, we have $\hat{y} \in \partial g_{\hat{i}}(\hat{x})$. □

Finally, in order to prove Theorem 3.1 it will be sufficient, by Theorem 3.7, to prove that under the assumptions of Theorem 3.1 the mapping $D'(\cdot)$ has closed graph and $I(x) \neq \emptyset$ for all $x \in \mathbb{R}^n$. Let $\lim_{k\to+\infty} x_k = x$ and $\lim_{k\to+\infty} y_k = y$ with $y_k \in D'(x_k)$ $(k = 1, 2, \dots)$. By the definition of $D'(x_k)$, there exist $i_{1,k}, i_{2,k}, \dots, i_{m_k,k}$ in $I(x_k)$, $\lambda_{1,k}, \dots, \lambda_{m_k,k}$, and $y_{j,k} \in \partial g_{i_{j,k}}(x_k)$ such that $\lambda_{j,k} \geq 0$, $\sum_{j=1}^{m_k} \lambda_{j,k} = 1$, and

$y_k = \sum_{j=1}^{m_k} \lambda_{j,k} y_{j,k}$. Since $y_k \in \mathbb{R}^n$, by Carathéodory's theorem (see, e.g., [7, Theorem 1.3.6, p. 98]) we may assume, without loss of generality, that $m_k \le n+1$. Since $G(\cdot)$ is finite-valued and $x_k \to \hat{x}$, the set $\{y \in \partial G(x_k) |\ k = 1, 2, \dots\}$ is bounded (see, e.g., [7, Proposition 6.2.2, p. 282]). By $y_{j,k} \in \partial g_{i_j}(x_k) \subset \partial G(x_k)$, we know that $\{y_{j,k} |\ 1 \le j \le m_k, k = 1, 2, \dots\}$ is a bounded set. Since I is compact and $\{m_k\}$, $\{y_{j,k}\}, \{\lambda_{j,k}\}$ are all bounded with respect to k, by repeatedly selecting subsequences we may assume, without loss of generality, that $m_k = m$ for all k , $i_{j,k} \to i_j$, $y_{j,k} \to y_j$, and $\lambda_{j,k} \to \lambda_j$ as $k \to +\infty$. By the fact proved above, we know that $i_j \in I(x)$ and $y_j \in \partial g_{i_j}(x)$. Thus, $y = \sum_{j=1}^{m} \lambda_j y_j \in D'(x)$, which proves that $D'(\cdot)$ has a closed graph. Finally, since I is compact and $i \to g_i(x)$ is upper semicontinuous, we have $I(x) \ne \emptyset$ for all $x \in \mathbb{R}^n$. This provides an alternative proof of Theorem 3.1.

4. Slater conditions. If there exists $\bar{x} \in \mathbb{R}^n$ such that

$$g_i(\bar{x}) < 0 \qquad (i \in I), \tag{50}$$

then (1) is said to satisfy *the Slater condition*. Let us recall the following well-known result, which gives a sufficient condition for the BCQ of (1) or (6).

PROPOSITION 4.1 (see [7, Theorem 1.3.5, p. 245] and [7, Remark 1.3.6, p. 246]). *If I is finite and* (1) *satisfies the Slater condition, then* (1) *satisfies the BCQ. In particular, if* (6) *satisfies the Slater condition, then* (6) *satisfies the BCQ.*

Remark 4. (a) If (6) satisfies the Slater condition, then (1) also satisfies the Slater condition. But the converse is not true. The Slater condition for (6) is sometimes called the strong Slater condition for the convex system (1) (see, e.g., [5, p.128]). However, the term "strong Slater condition" is also used in the literature in other senses (see, e.g., Lewis and Pang [9], where "strong Slater condition" means that 0 does not belong to the closure of the set $\partial G(G^{-1}(0))$, and, for a different sense, see [7, Definition 2.3.1, p. 311]).

(b) When I is finite, (1) satisfies the Slater condition if and only if (6) satisfies the Slater condition.

PROPOSITION 4.2. *Suppose that* (6) *satisfies the Slater condition.*

(a) *If* (1) *has the weak PLV property, then* (1) *satisfies the Abadie CQ.*

(b) *If* (1) *has the PLV property, then* (1) *satisfies the BCQ.*

Proof. (a) By Proposition 4.1, the Slater condition for (6) implies the BCQ for (6). Hence, by Proposition 3.5(a), we have the Abadie CQ at all $x \in \operatorname{bd} S$.

(b) The proof is similar, using Proposition 3.5(b). □

Remark 5. (a) Fajardo and López [3, Theorem 4.1(ii)] proved that if (6) satisfies the Slater condition, $\overline{D'}(\cdot)$ is (B)-upper semicontinuous (see Remark 2), $\overline{D'}(x) \ne \emptyset$ for $x \in \mathbb{R}^n$, and $N'(x)$ is closed for each x in S, then (1) satisfies the BCQ. But the (B)-upper continuity of $\overline{D'}(\cdot)$ is equivalent to the upper continuity of $\overline{D'}(\cdot)$ (see Remark 2), so this result also follows from Theorem 3.8(a) and Proposition 4.2(a).

(b) The assumptions in (a) and (b) of Proposition 4.2 cannot be omitted, as shown by Example 1(a), in which the Slater condition for (1) or (6) is satisfied (in fact, for $\bar{x} = (\frac{1}{2}, \frac{1}{2})$, $g_i(\bar{x}) \le -\frac{1}{2}$ for all i), but the Abadie CQ is not satisfied. When I is a finite set, the PLV property always holds. In this case, the Slater condition, the BCQ, and the Abadie CQ are all different.

We recall that for a convex system (1) a solution $\bar{x} \in S$ is called a *Slater point* if we have (50).

PROPOSITION 4.3. *If for each $x \in S$ the active index set $I(x) \ne \emptyset$, then every Slater point of* (1) *is a Slater point of* (6) *(and hence, in this case, the Slater condition for* (1) *and the Slater condition for* (6) *are equivalent).*

Proof. Let $\bar{x}$ be a Slater point of (1), i.e., let $\bar{x}$ be a point such that (50) holds. If $\bar{x}$ were not a Slater point of (6), i.e., if we had $\sup_{i\in I} g_i(\bar{x}) = 0$, then, since $I(\bar{x}) \neq \emptyset$, there would exist $i_0 \in I$ such that $g_{i_0}(\bar{x}) = \sup_{i\in I} g_i(\bar{x}) = 0$, a contradiction to the assumption (50). □

Using Theorem 3.1, one can give a stronger condition which ensures the BCQ. Indeed, combining Theorem 3.1 and Proposition 4.2, we obtain the following result, which has been proved with a more complicated method by López and Vercher [12, Theorem 3.8].

COROLLARY 4.4. *If I is a compact set (in some metric space), $g_i : \mathbb{R}^n \to \mathbb{R}$ $(i \in I)$ is a family of convex functions such that for each fixed $x \in \mathbb{R}^n$ the function $i \to g_i(x)$ is upper semicontinuous on I, and* (1) *satisfies the Slater condition, then the BCQ holds for* (1).

Proof. Since I is compact and $i \to g_i(x)$ is upper semicontinuous, $I(x) \neq \emptyset$ for any x. By Proposition 4.3, (6) satisfies the Slater condition. By Theorem 3.1, the PLV property holds. Thus, the corollary follows from Proposition 4.2. □

Even though we stated Corollary 4.4 in terms of the Slater condition of (1), obviously the Slater condition of (6) is also satisfied. In general, the Slater condition for (1) is not very meaningful if (6) does not satisfy the Slater condition. One might wonder whether we should use the following stronger version of (50):

$$g_i(\bar{x}) \leq -\delta < 0 \quad (i \in I), \tag{51}$$

where δ is a positive constant. In the case that $G(x) < +\infty$ for $x \in \mathbb{R}^n$, (51) is nothing more than the Slater condition for (6). If one allows $G(x)$ to be $+\infty$, then (51) does not provide any useful information about the system as shown in the following example.

Example 3. (a) For $\{g_i | \, i \in I\}$ and $\bar{x} \in \mathbb{R}^n$ satisfying (50), let

$$\bar{g}_i(x) := -\frac{1}{g_i(\bar{x})} g_i(x) \qquad (i \in I). \tag{52}$$

Then (51) holds with g_i and δ being replaced by $\bar{g}_i$ and 1, respectively. Note that $x \in S$ if and only if $\bar{g}_i(x) \leq 0$ $(i \in I)$. Also the sets $I(x)$ and $N'(x)$ remain unchanged for $x \in \operatorname{bd} S$. Thus, (1) satisfies the Abadie CQ (respectively, the BCQ) if and only if so does the system $\bar{g}_i(x) \leq 0$ $(i \in I)$. This example shows that replacing (50) by (51) without requiring a finite-valued G does not give any new information about the underlying system.

(b) Or we could make the situation worse. For example, let $\hat{g}_{i,k}(x) := k g_i(x) - 1$ for $i \in I, k \geq 1$. Then we always have

$$\hat{g}_{i,k}(x) \leq -1 < 0 \quad \text{for } x \in S. \tag{53}$$

It is easy to see that $x \in S$ if and only if $\hat{g}_{i,k}(x) \leq 0$ for $i \in I, k \geq 1$. In this case, the active index set is always empty for any $x \in \operatorname{bd} S$. Thus, it is not possible to study S by using $N'(x)$. This example shows that (51) without requiring a finite-valued $G(x)$ could be a meaningless condition, while (51) with a finite-valued $G(x)$ means that (6) satisfies the Slater condition, which is useful for constraint qualification properties of (1) (see Proposition 4.2).

5. Applications.

5.1. Linear representation of convex systems. Given a semi-infinite convex inequality system (1), we recall that a semi-infinite linear inequality system,

$$\langle \hat{a}_k, x\rangle \le \hat{b}_k \qquad (k \in K), \tag{54}$$

where $\hat{a}_k \in \mathbb{R}^n$ and $\hat{b}_k \in \mathbb{R}$, is said to be a *linear representation of the system* (1), provided that x^* is a solution of (1) if and only if x^* is a solution of (54) (i.e., provided that the systems of inequalities (1) and (54) are equivalent). Each linear representation (54) of (1) is also called a *linear system associated to the convex system* (1).

It is well known (see, e.g., the proof of Theorem 5.2 in [6]) that the system

$$\langle a, x\rangle \le \langle a, z\rangle - g_i(z) \qquad (z \in \mathbb{R}^n, i \in I, a \in \partial g_i(z)), \tag{55}$$

is a linear representation of the convex system (1), which we shall call *the standard linear representation of* (1). For the sake of completeness, we include the proof here.

Proof. If $x^* \in S$, then

$$g_i(z) + \langle a, x^* - z\rangle \le g_i(x^*) \le 0 \qquad (z \in \mathbb{R}^n, i \in I, a \in \partial g_i(z)),$$

so x^* satisfies (55). Conversely, if x^* satisfies (55), then

$$g_i(z) + \langle a, x^* - z\rangle \le 0 \qquad (z \in \mathbb{R}^n, i \in I, a \in \partial g_i(z)).$$

Taking here $z = x^*$, we obtain $g_i(x^*) \le 0$ $(i \in I)$, that is, $x^* \in S$, which completes the proof. □

A natural question is whether we can use a smaller subsystem of (55) to get a linear representation of (1). In particular, we study when the semi-infinite linear system

$$\langle y_i, x\rangle \le \langle y_i, z\rangle \qquad (z \in \operatorname{bd} S, i \in I(z), y_i \in \partial g_i(z)) \tag{56}$$

is a linear representation of (1). Note that (56) is indeed a subsystem of (55), since for $z \in \operatorname{bd} S$, $i \in I(z)$, and $y \in \partial g_i(z)$, we have $g_i(z) = 0$.

THEOREM 5.1.

(a) *If the convex system* (1) *satisfies the Abadie CQ, then the system* (56) *is a linear representation of* (1).

(b) *If the convex system* (1) *satisfies the Slater condition* (50) *and* $I(x) \ne \emptyset$ *for all* $x \in S$, *then the system* (56) *is a linear representation of* (1).

Proof. Obviously, if $x \in S$, then, since $\partial g_i(z) \subset N_S(z)$ for $i \in I(z)$ and $z \in \operatorname{bd} S$, (56) follows from $y_i \in N_S(z)$. Thus, every solution of (1) is a solution of (56). In order to prove that (56) is a linear representation of (1), it is sufficient to show that if $x \notin S$, then (56) does not hold.

(a) First we prove the following more general result:
Suppose that for any $x \in \operatorname{bd} S$ and $y \in N_S(x) \setminus \{0\}$, there is a vector $\hat{y} \in N'(x)$ such that $\langle y, \hat{y}\rangle > 0$. Then (56) is a linear representation of (1).

Let $x \notin S$ and let z be the projection of x onto S. Then $z \in \operatorname{bd} S$, $x - z \ne 0$, and $x - z \in N_S(z)$ [7, Theorem 3.1.1, p. 117]. If $\langle y_i, x\rangle \le \langle y_i, z\rangle$ for all $i \in I(z)$ and $y_i \in \partial g_i(z)$, then $\langle y, x - z\rangle \le 0$ for all $y \in N'(z)$, a contradiction to the assumption

there is $\hat{y} \in N'(x)$ such that $\langle \hat{y}, x - z\rangle > 0$. Hence, x does not satisfy (56). This proves that $x \in S$ if and only if x satisfies (56), i.e., (56) is a linear representation of (1).

Now, if the convex system (1) satisfies the Abadie CQ, i.e., $N_S(x) = \overline{N'}(x)$ for all $x \in \operatorname{bd} S$, then it is trivially true that for any $x \in \operatorname{bd} S$ and $y \in N_S(x) \setminus \{0\}$, there is a vector $\hat{y} \in N'(x)$ such that $\langle y, \hat{y}\rangle > 0$ (indeed, it is enough to take $\hat{y} \in N'(x)$ sufficiently close to y). So (56) is a linear representation of (1).

(b) By Proposition 4.3, (6) satisfies the Slater condition. Let $x \notin S$, and let $\bar{x} \in S$ be such that $G(\bar{x}) < 0$. Then $\operatorname{co}\{x, \bar{x}\} \cap \operatorname{bd} S$ contains exactly one point, say z. Thus, there is a positive constant θ such that

$$x - z = \theta(z - \bar{x}). \tag{57}$$

Since $I(z) \neq \emptyset$, choose $i \in I(z)$ and $y_i \in \partial g_i(z) \subset \partial G(z)$. Then, by the definition of $\partial G(z)$, we have

$$G(\bar{x}) - G(z) \geq \langle y_i, \bar{x} - z\rangle \quad \text{or} \quad \langle y_i, z - \bar{x}\rangle \geq G(z) - G(\bar{x}) = -G(\bar{x}) > 0. \tag{58}$$

Since $\theta > 0$, it follows from (57) and (58) that $\frac{1}{\theta}\langle y_i, x - z\rangle = \langle y_i, z - \bar{x}\rangle \geq -G(\bar{x}) > 0$, whence $\langle y_i, x\rangle > \langle y_i, z\rangle$. Thus, for any $x \notin S$, (56) does not hold. So (56) is a linear representation of (1). □

Remark 6. (a) In [12, the proof of Theorem 4.5], it has been observed that if (1) satisfies the assumptions of Corollary 4.4, then the linear inequality system

$$\langle u, x\rangle \leq \langle u, y\rangle \quad \text{for } y \in \operatorname{bd} S, u \in \partial G(y) \tag{59}$$

is a linear representation of (1). Let us observe that this follows also from Corollary 4.4 and Theorem 5.1(a) above, since $\partial G(x) = \operatorname{co}\{\partial g_i(x) : i \in I(x)\}$ for $x \in \operatorname{bd} S$ under the assumption of Corollary 4.4, hence (59) is equivalent to (56).

(b) In the particular case when all g_i $(i \in I)$ are convex and differentiable, Theorem 5.1(b) (even with a smaller subsystem of (56), obtained by choosing for each $x \in \operatorname{bd} S$, with the aid of the axiom of choice, an index $i(x) \in I(x)$ and an $y_{i(x)} \in \partial g_i(x)$) has been shown, essentially, in the proof of Theorem 5.4 in [6].

Some connections between the inequality systems (1) and (56) are given in the following proposition.

PROPOSITION 5.2. *Let* (56) *be a linear representation of the convex inequality system* (1). *Then*

(a) (1) *satisfies the Abadie CQ (respectively, the BCQ) if and only if so does* (56).

(b) *Denoting by* G *and* G_0 *the sup-functions of* (1) *and* (56) *respectively, we have*

$$G_0(x) \leq G(x) \ \ for \ \ x \in \mathbb{R}^n. \tag{60}$$

Proof. (a) Let (56) be a linear representation of (1) and let $x \in \operatorname{bd} S$. Since (1) and (56) have the same solution set S, and hence the same normal cone $N_S(x)$ at x, it will be enough to show that

$$N'(x) \text{ for (56)} = N'(x) \text{ for (1)}. \tag{61}$$

Since the linear system (56) is a subset of the linear system (55), we have

$$N'(x) \text{ for (56)} \subseteq N'(x) \text{ for (55)}. \tag{62}$$

Furthermore, by [5, proof of Theorem 10.7], there holds

$$N'(x) \text{ for } (55) = N'(x) \text{ for } (1). \tag{63}$$

Finally, from the definitions it is obvious that

$$N'(x) \text{ for } (1) \subseteq N'(x) \text{ for } (56), \tag{64}$$

which, together with (62) and (63), yields (61).

(b) By the definitions of the sup-function and of $\partial g_i(z)$ and $I(z)$, we have, for any $x \in \mathbb{R}^n$,

$$G_0(x) = \sup_{z \in bdS,\, i \in I(z), y_i \in \partial g_i(z)} \Big(\langle y_i, x \rangle - \langle y_i, z \rangle \Big) \leq \sup_{i \in I} g_i(x) = G(x). \qquad \square$$

Remark 7. (a) The inequality in Proposition 5.2(b) may be strict.

(b) From Proposition 5.2(b) above it follows that if (1) satisfies the Slater condition, then so does (56); also, if (6) satisfies the Slater condition, then $G_0(\overline{x}) < 0$ for some $\overline{x} \in \mathbb{R}^n$. However, the converse statements are not true.

5.2. Convex FM systems. Related to the BCQ are the convex FM systems, defined as follows.

Definition 5.3.

(a) *A linear inequality*

$$\langle a_0, x \rangle \leq b_0, \tag{65}$$

where $a_0 \in \mathbb{R}^n \setminus \{0\}$ *and* $b_0 \in \mathbb{R}$, *will be called a consequence relation of the convex inequality system* (1)*, if every* $x \in S$ *satisfies* (65).

(b) *The system* (1) *will be called a* convex FM *system if every linear consequence relation of system* (1) *is also a consequence relation of some finite subsystem of* (1).

Remark 8. In the particular case of a linear inequality system (4), the above definition reduces to the usual definition of consequence relations and FM systems [5].

One can extend some results on linear FM systems to convex FM systems. For example, the fact that a linear inequality system (4) satisfying the BCQ and with bounded solution set S is an FM system (see [5, Exercise 5.7]) admits the following extension.

Proposition 5.4. *A convex inequality system* (1) *satisfying the BCQ and with bounded solution set* S *is a convex FM system.*

Proof. By the above proof of Proposition 5.2(a), (1) satisfies the BCQ (if and) only if so does its standard linear representation (55). Furthermore, since (55) is a linear inequality system having the same bounded solution set S, it is an FM system (see, e.g., [5, Exercise 5.6]). Finally, let us show that if (55) is an FM system, then so is (1). Indeed, let

$$S \subseteq \{x \in \mathbb{R}^n | \, \langle a_0, x \rangle \leq b_0\}, \tag{66}$$

where $a_0 \in \mathbb{R}^n \setminus \{0\}$ and $b_0 \in \mathbb{R}$. Then, since (55) is an FM system and has the same solution set S, there exists a finite subsystem of (55), say

$$\langle a_j, x \rangle \leq \langle a_j, z_j \rangle - g_j(z_j) \qquad (z_j \in \mathbb{R}^n, j \in J, a_j \in \partial g_j(z_j)), \tag{67}$$

where $|J| < +\infty$, such that (65) is a consequence relation of (67). Let S_J be the solution set of the finite subsystem

$$g_j(x) \leq 0 \qquad (j \in J) \tag{68}$$

of (1) and let $x \in S_J$. Then for any $z_j \in \mathbb{R}^n, j \in J$, and $a_j \in \partial g_j(z_j)$, we have

$$\langle a_j, x\rangle - \langle a_j, z_j\rangle \leq g_j(x) - g_j(z_j) \leq -g_j(z_j), \tag{69}$$

so x is a solution of (67). Hence, since (65) is a consequence relation of (67), we obtain $\langle a_0, x\rangle \leq b_0$, which, since $x \in S_J$ was arbitrary, proves that (1) is an FM system. □

Remark 9. The proofs of some results on linear systems, given in [5], use certain cones of $\mathbb{R}^{n+1}$ associated to (4). However, let us observe that one can give, directly for the extensions of those results to convex systems, proofs which are new even for the case of linear inequality systems and which do not use any subsets of $\mathbb{R}^{n+1}$. Indeed, let us give such a proof of Proposition 5.4. Assume that S is bounded and (1) satisfies the BCQ, and let (65) be a consequence relation of (1). Let c be the smallest number such that $\langle a_0, x\rangle \leq c$ is still a consequence relation of (1) (such a number exists, since otherwise $S \subseteq \{x \in \mathbb{R}^n \,|\, \langle a_0, x\rangle = -\infty\} = \emptyset$, a contradiction to the general assumption made in this paper).

We claim that there exists $z \in S$ such that $\langle a_0, z\rangle = c$. Indeed, by the definition of c, we have $S \subseteq \{x \in \mathbb{R}^n | \langle a_0, x\rangle \leq c\}$ and for each $k = 1, 2, \ldots$ there exists $x_k \in S$ such that $\langle a_0, x_k\rangle > c - \frac{1}{k}$. Then, since S is bounded and closed, hence compact, $\{x_k\}$ has a subsequence converging to some $z \in S$. Clearly, $\langle a_0, z\rangle = c$, which proves our claim.

By the above, we have $S \subseteq \{x \in \mathbb{R}^n \,|\, \langle a_0, x\rangle \leq \langle a_0, z\rangle\}$, that is, $a_0 \in N_S(z)$. Hence, by the BCQ, there exists a finite subset J of $I(z)$, such that

$$a_0 \in \operatorname{co}\left(\cup_{i\in J}\partial g_i(z)\right). \tag{70}$$

Let

$$S_J := \{x \in \mathbb{R}^n | g_i(x) \leq 0 \ (i \in J)\}. \tag{71}$$

Then, by (70) and $J \subseteq I(z)$, we have $a_0 \in \operatorname{co}(\cup_{i\in J}\partial g_i(z)) \subseteq N_{S_J}(z)$, that is, $\langle a_0, x\rangle \leq \langle a_0, z\rangle = c$ for all $x \in S_J$. Thus, the inequality $\langle a_0, x\rangle \leq c$, whence also (65), is a consequence relation of the finite subsystem $g_i(x) \leq 0$ $(i \in J)$ of (1), which completes the proof.

Combining Proposition 5.4 and Corollary 4.4, there results the following corollary, which has been proved with more complicated methods in [12, Theorem 4.5].

COROLLARY 5.5. *If I is a compact set, $g_i : \mathbb{R}^n \to \mathbb{R}$ $(i \in I)$ is a family of convex functions such that for each fixed $x \in \mathbb{R}^n$ the function $i \to g_i(x)$ is upper semicontinuous on I,* (1) *satisfies the Slater condition, and the solution set S is bounded, then* (1) *is an* FM *system.*

Remark 10. The definition of a convex FM system given in [12] refers to its standard linear representation being a FM system. Under the assumptions of Corollary 5.5, both definitions are equivalent.

5.3. The distance to the solution set of a convex inequality system.

THEOREM 5.6. *Let $x \in \mathbb{R}^n \backslash S$ and $\hat{x}$ be the projection of x onto S.*

(a) *If* (1) *satisfies the Abadie CQ, then*

$$\text{dist}(x,S)=\sup_{\substack{I_0\subset I(\hat{x})\\ |I_0|<+\infty}}\ \sup_{y_j\in\partial g_j(\hat{x})\ (j\in I_0)}\ \max_{\substack{\lambda_j\geq 0\ (j\in I_0)\\ \left\|\sum_{j\in I_0}\lambda_j y_j\right\|=1}}\ \sum_{i\in I_0}\lambda_i\langle y_i,x-\hat{x}\rangle. \tag{72}$$

(b) *If* (1) *satisfies the BCQ, then*

$$\text{dist}(x,S)=\max_{\substack{I_0\subset I(\hat{x})\\ |I_0|<+\infty}}\ \sup_{y_j\in\partial g_j(\hat{x})\ (j\in I_0)}\ \max_{\substack{\lambda_j\geq 0\ (j\in I_0)\\ \left\|\sum_{j\in I_0}\lambda_j y_j\right\|=1}}\ \sum_{i\in I_0}\lambda_i\langle y_i,x-\hat{x}\rangle. \tag{73}$$

Proof. By [17, Remark 8(b)], we have

$$\text{dist}(x,S)=\|x-\hat{x}\|=\max_{y\in N_S(\hat{x}),\|y\|=1}\langle y,x-\hat{x}\rangle. \tag{74}$$

(a) By the Abadie CQ at $\hat{x}$, we have $y\in N_S(\hat{x})$ with $\|y\|=1$ if and only if there exist $I_0^k\subset I(\hat{x})$, $|I_0^k|<+\infty$, $y_j^k\in\partial g_j(\hat{x})$ $(j\in I_0^k)$, $\lambda_j^k\geq 0$ $(j\in I_0^k)$, $\left\|\sum_{j\in I_0^k}\lambda_j^k y_j^k\right\|=1$ such that $\sum_{j\in I_0^k}\lambda_j^k y_j^k\to y$. Thus, (72) is equivalent to (74).

(b) By the BCQ at $\hat{x}$, we have $y\in N_S(\hat{x})$ with $\|y\|=1$ if and only if there exist $I_0\subset I(\hat{x})$, $|I_0|<+\infty$, $y_j\in\partial g_j(\hat{x})$ $(j\in I_0)$, $\lambda_j\geq 0$ $(j\in I_0)$, $\left\|\sum_{j\in I_0}\lambda_j y_j\right\|=1$ such that $\sum_{j\in I_0}\lambda_j y_j=y$. Thus, (73) is equivalent to (74). ∎

Remark 11. The assumption of Abadie CQ may be too strong, but at least the assumption $I(\widehat{x})\neq\emptyset$ (which is implied by the Abadie CQ) is necessary in order to have (72). Indeed, if $I(\widehat{x})=\emptyset$, then the right-hand side of (72) is meaningless.

When applied to the semi-infinite linear system (4), $\partial g_j(\hat{x})=\{a_j\}$ and $j\in I(\hat{x})$ means that $\langle a_j,\hat{x}\rangle=b_j$, and thus Theorem 5.6 reduces to the following form.

COROLLARY 5.7. *Let $x\in\mathbb{R}^n\backslash S$ and $\hat{x}$ be the projection of x onto S.*

(a) *If* (4) *satisfies the Abadie CQ, then*

$$\text{dist}(x,S)=\sup_{\substack{I_0\subset I(\hat{x})\\ |I_0|<+\infty}}\ \max_{\substack{\lambda_j\geq 0\ (j\in I_0)\\ \left\|\sum_{j\in I_0}\lambda_j a_j\right\|=1}}\ \sum_{i\in I_0}\lambda_i[\langle a_i,x\rangle-b_i]. \tag{75}$$

(b) *If* (4) *satisfies the BCQ, then*

$$\text{dist}(x,S)=\max_{\substack{I_0\subset I(\hat{x})\\ |I_0|<+\infty}}\ \max_{\substack{\lambda_j\geq 0\ (j\in I_0)\\ \left\|\sum_{j\in I_0}\lambda_j a_j\right\|=1}}\ \sum_{i\in I_0}\lambda_i[\langle a_i,x\rangle-b_i]. \tag{76}$$

Remark 12. (a) By a well-known theorem of Carathéodory (see, e.g., [15, Corollary 7.1(i), p. 94]), in each positive combination $\sum_{i\in I_0}\lambda_i y_i$ we may assume that $\{y_j|j\in I_0\}$ is linearly independent. That is,

$$\begin{aligned}N'(x)=\Bigg\{&\sum_{i\in I_0}\lambda_i y_i\ \Big|\ I_0\subset I(x), y_i\in\partial g_i(x)\ (i\in I_0),\\ &\{y_j\mid j\in I_0\}\text{ is linearly independent},\lambda_i\geq 0\ (i\in I_0)\Bigg\}.\end{aligned} \tag{77}$$

Since $y_i \in \mathbb{R}^n$ $(i \in I_0)$, we have

$$|I_0| \le n \quad \text{whenever } \{y_j | j \in I_0\} \text{ is linearly independent,} \tag{78}$$

and hence we also have the following representation of $N'(x)$:

$$N'(x) = \left\{ \sum_{i \in I_0} \lambda_i y_i \;\middle|\; I_0 \subset I(x), |I_0| \le n, y_i \in \partial g_i(x)\ (i \in I_0), \lambda_i \ge 0\ (i \in I_0) \right\}. \tag{79}$$

One could rewrite the distance formulas (72), (73), (75), and (76), based on either (77) or (79).

(b) In the particular case when I is finite, (4) satisfies the BCQ (see the observation before Example 1), and hence Corollary 5.7(b) reduces to [17, Remark 8(a)].

The following theorem shows that if the BCQ holds, then the distance of a point to the solution set S of an arbitrary convex inequality system (1) is equal to the distance of that point to the solution set of some finite subsystem of (1).

THEOREM 5.8. *If* (1) *satisfies the BCQ and* $\widehat{x}$ *is the projection of* x *onto* S*, then there exists* $\bar{J} \subseteq I(\widehat{x})$ *with* $|\bar{J}| < +\infty$*, such that*

$$\operatorname{dist}(x, S) = \operatorname{dist}(x, S_{\bar{J}}), \tag{80}$$

where

$$S_{\bar{J}} = \{z \in \mathbb{R}^n | \, g_i(z) \le 0\ (i \in \bar{J})\}. \tag{81}$$

Proof. Choose any $\bar{J} \subseteq I(\widehat{x})$ for which the first max in (73) is attained. Then, applying Theorem 5.6 to the inequality system

$$g_j(z) \le 0 \qquad (j \in \bar{J}) \tag{82}$$

and to its solution set $S_{\bar{J}}$ (of (81)), we obtain

$$\begin{aligned} \operatorname{dist}(x, S) &= \sup_{y_j \in \partial g_j(\hat{x})\ (j \in \bar{J})} \max_{\substack{\lambda_j \ge 0\ (j \in \bar{J}) \\ \left\| \sum_{j \in \bar{J}} \lambda_j y_j \right\| = 1}} \sum_{i \in \bar{J}} \lambda_i \langle y_i, x - \hat{x} \rangle \\ &\le \max_{J \subseteq \{j \in \bar{J} | g_j(\hat{x}) = 0\}} \max_{\substack{\lambda_j \ge 0\ (j \in J) \\ \left\| \sum_{j \in J} \lambda_j y_j \right\| = 1}} \sum_{i \in J} \lambda_i \langle y_i, x - \hat{x} \rangle = \operatorname{dist}(x, S_{\bar{J}}), \end{aligned} \tag{83}$$

which, since $S \subseteq S_{\bar{J}}$ (by (2) and (81)), yields (80). ∎

Remark 13. In the particular case when I is finite and each g_i is an affine function, (1) satisfies the BCQ, and hence Theorem 5.8 yields [2, Corollary 1.1].

Acknowledgments. We wish to thank A. Auslender and A. M. Rubinov for the references [13] and [8]. We also thank M. A. López for sending us the manuscript [3]. Finally, we wish to express our gratitude to López and M. D. Fajardo for their careful reading of the manuscript of the present paper and for their valuable remarks which contributed to its improvement.

REFERENCES

[1] J. ABADIE, *On the Kuhn-Tucker theorem*, in Nonlinear Programming, J. Abadie, ed., North–Holland, Amsterdam, 1967, pp. 19–36.

[2] C. Bergthaller and I. Singer, *The distance to a polyhedron*, Linear Algebra. Appl., 169 (1992), pp. 111–129.
[3] M. D. Fajardo and M. A. López, *Locally Farkas-Minkowski systems in convex semi-infinite programming*, J. Optim. Theory Appl., pp. 313–335.
[4] K. Fan, *On systems of linear inequalities*, in Linear Inequalities and Related Topics, H. W. Kuhn and A. W. Tucker, eds., Princeton University Press, Princeton, NJ, 1956, pp. 99–156.
[5] M. A. Goberna and M. A. López, *Linear Semi-infinite Optimization*, Wiley, Chichester, 1998.
[6] M. A. Goberna, M. A. López, and J. Pastor, *Farkas-Minkowski systems in semi-infinite programming*, Appl. Math. Optim., 7 (1981), pp. 295–308.
[7] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* I, Springer-Verlag, Berlin, Heidelberg, 1993.
[8] V. L. Levin, *Application of a theorem of E. Helly in convex programming, problems of best approximation and related topics*, Mat. Sb., 79 (1969), pp. 250–263, (in Russian).
[9] A. S. Lewis and J.-S. Pang, *Error bounds for convex inequality systems*, in Generalized Convexity, Generalized Monotonicity: Recent Results, J.-P. Crouzeix, J.-E. Martínez-Legaz, and M. Volle, eds., Kluwer Academic Publishers, Dordrecht, Boston, 1998, pp. 75–110.
[10] W. Li, *Abadie's constraint qualification, metric regularity, and error bounds for differentiable convex inequalities*, SIAM J. Optim., 7 (1997), pp. 966–978.
[11] W. Li and I. Singer, *Asymptotic Constraint Qualifications and Error Bounds for Semi-Infinite Systems of Convex Inequalities*, preprint, Dept. of Math. and Statistics, Old Dominion University, Norfolk, VA.
[12] M. A. López and E. Vercher, *Optimality conditions for nondifferentiable convex semi-infinite programming*, Math. Programming, 27 (1983), pp. 307-319.
[13] B. N. Pshenichnyi, *Convex programming in a normed space*, Kibernetika, 5 (1965), pp. 46–54, (in Russian).
[14] R. Puente and V. N. Vera de Serio, *Locally Farkas-Minkowski linear inequality systems*, Top, 7 (1999), pp. 103–121.
[15] A. Schrijver, *Theory of Linear and Integer Programming*, Wiley, Chichester, UK, 1986.
[16] I. Singer, *The Theory of Best Approximation and Functional Analysis*, CBMS-NSF Regional Conf. Ser. in Appl. Math., 13, SIAM, Philadelphia, PA, 1974.
[17] I. Singer, *Duality for optimization and best approximation over finite intersections*, Numer. Funct. Anal. Optim., 19 (1998), pp. 903–915.
[18] M. Valadier, *Sous-différentiels d'une borne supérieure et d'une somme continue de fonctions convexes*, CRAS Paris, 266 (1968), pp. 14–16.
[19] Y.-J. Zhu, *Generalizations of some fundamental theorems in linear inequalities*, Acta Math. Sinica, 16 (1966), pp. 25–39.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 53–69

OPTIMIZATION OF DISCONTINUOUS FUNCTIONS: A GENERALIZED THEORY OF DIFFERENTIATION*

LUC MOREAU† AND DIRK AEYELS†

Abstract. Problems of optimization and optimal control with discontinuous costs are considered. For that purpose, we introduce some preliminary ideas of a new generalized theory of differentiation, the main ideas of which are inspired by the work of Clarke [*Classics Appl. Math.* 5, SIAM, Philadelphia, 1990]. We present two calculus rules and apply the introduced theory to the study of some optimization and optimal control problems with discontinuous costs.

Key words. nonsmooth, calculus, optimization, optimal control

AMS subject classifications. 49J52, 49K10

PII. S1052623499354679

1. Introduction.

Many problems in pure and applied mathematics deal with nondifferentiable data. For instance, nondifferentiable objective functions arise naturally and frequently in optimization problems [4, section 1.1]. When theory and techniques are to be developed to optimize (e.g., minimize) such functions, a good generalization of the classical gradient concept seems indispensable.

Since the early 1960s several generalized theories of differentiation have been proposed by different authors. A first major step in this direction came with the dissertation of Rockafellar [7], who introduced subgradients for convex functions. Another breakthrough occurred when Clarke [2] found a way of extending Rockafellar's ideas to the broader class of lower semicontinuous, proper functions. This line of ideas has given rise to an extensive amount of research, continuing to the present. An overview of the most important results in a unifying framework may be found in the excellent recent exposition [9]. Related approaches to nondifferentiable problems may be found, e.g., in [11, 1, 10].

Consider a function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$. The classical gradient of f at x is defined only when f is differentiable at x. For nondifferentiable functions f, various generalizations of the gradient are summarized in [9], which are termed (regular, general, proximal, Clarke) subgradients. A common and very important feature of these subgradients is that they can be characterized geometrically in terms of normals to the epigraph of f at the point $(x, f(x))$. (This is a particular instance of the far-reaching duality between variational analysis and variational geometry.) We thus see that these different types of subgradient only capture information about the epigraph of f near $(x, f(x))$. Notice that if f is discontinuous at x, then the epigraph of f near the point $(x, f(x))$ does not contain the same information as f near the point

*Received by the editors July 1, 1999; accepted for publication (in revised form) November 2, 1999; published electronically July 25, 2000. This paper presents research results of the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology, and Culture. The scientific responsibility rests with its authors. A preliminary version of this work appeared as a conference paper [*Optimization of discontinuous functions and application to optimal control*, in Proceedings of the 13th International Symposium on Mathematical Theory of Networks and Systems, Il Poligrafo, Padova, Italy, 1998].

http://www.siam.org/journals/siopt/11-1/35467.html

†SYSTeMS group, Ghent University, Technologiepark 9, 9052 Zwijnaarde, Belgium (lmoreau@ensmain.rug.ac.be, dirk.aeyels@rug.ac.be). The first author was supported by BOF grant 011D0696 of Ghent University.

x. We illustrate this with a simple one-dimensional example. Consider the function

$$f : \mathbb{R} \to \mathbb{R} : x \mapsto f(x) = \begin{cases} \alpha x, & x < 0, \\ -x - 1, & x \geq 0, \end{cases}$$

with α a real parameter. Clearly, the epigraph of f in a neighborhood of $(0, -1)$ does not depend on the particular value of α, and thus we see that the (regular, general, proximal, Clarke) subgradients of f at 0 are also independent of α. For various applications, this is a favorable property. Suppose, however, that f represents a profit that has to be maximized. In this case, if $\alpha > 0$, then f has a supremum at $x = 0$ and we would certainly be interested in this point $x = 0$. We are therefore interested in a generalized theory of differentiation that takes into account the behavior of f for $x < 0$ and that is able to detect the supremum.

In the present paper we present some preliminary ideas of a generalized theory of differentiation that

(1) incorporates information of f in a complete neighborhood of x,

(2) leads to necessary conditions for a function f to have a minimum (maximum, infimum, supremum) at a point x,

(3) is applicable to arbitrary functions $f : \mathbb{R}^n \to \mathbb{R}$.

The main ideas that have led to this paper actually were inspired by the seminal work of Clarke [4]. The present theory coincides with Clarke's theory when f is locally Lipschitz but differs from it—and the other theories mentioned in [9]—when f is more general, although there still is a close relationship.

Furthermore, in the present paper

(4) we develop two calculus rules—one for scalar multiples and a sum rule;

(5) we apply the present theory to the study of the following three problems:

(i) Unconstrained optimization. Let $f : \mathbb{R}^n \to \mathbb{R}$ be an arbitrary function. Find $x \in \mathbb{R}^n$, where f has a local minimum (maximum, infimum, supremum).

(ii) Descent directions. Let $f : \mathbb{R}^n \to \mathbb{R}$ be an arbitrary function and let x be a point in $\mathbb{R}^n$. Find directions $v \in \mathbb{R}^n$ such that $f(x + \lambda v) < f(x)$ for $\lambda \in (0, \infty)$ sufficiently small.

(iii) Optimal control. Given a sufficiently smooth control system $\dot{x} = X(t, x, u)$ ($t \in \mathbb{R}$, $x \in \mathbb{R}^n$, $u \in \mathcal{U} \subseteq \mathbb{R}^s$) and an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ (the endpoint cost). Minimize $f(\chi(t_1))$ over all trajectories χ of the control system defined on a fixed interval $[t_0, t_1]$ and starting in a fixed point x_0.

(6) we compare obtained results with results from the literature.

This paper is organized as follows. Section 2 clarifies the relation with the literature. Sections 3 and 4 introduce a generalized theory of differentiation for arbitrary functions from $\mathbb{R}^n$ to $\mathbb{R}$. Section 5 presents two calculus rules. Section 6 applies the theory of the present paper to the problem of optimizing an arbitrary function from $\mathbb{R}^n$ to $\mathbb{R}$. Section 7 considers the problem of finding points $\overline{x}$, where $f(\overline{x}) < f(x)$. Section 8 is devoted to a particular class of nonsmooth optimal control problems. Section 9 concludes the paper.

2. Relation with the literature. Consider a function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$. As mentioned in the introduction, the generalized theories of differentiation mentioned in [9] only incorporate information of the epigraph of f in the neighborhood of $(x, f(x))$. Here we are interested in a theory that incorporates information of f in a complete neighborhood of the point x. Since the main ideas of the present theory were inspired by Clarke's seminal work [2], we start with recalling Clarke's theory.

Clarke's theory [4] associates with f and x two subsets of the dual space $(\mathbb{R}^n)^*$: the *generalized gradient*[1] of f at x, denoted $\boldsymbol{\partial} f(x)$, and the *singular generalized gradient*[2] of f at x, denoted $\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x)$. See [3, 4, 5] for some applications in the calculus of variations and optimal control theory. In order to recall the definition of $\boldsymbol{\partial} f(x)$ and $\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x)$, we first introduce some geometrical concepts. Let $S \subseteq \mathbb{R}^n$, $x \in S$. *Clarke's tangent cone to S at x*, denoted $T_S(x)$, is defined by

(1)
$$\begin{aligned} T_S(x) = \{v \in \mathbb{R}^n : &\forall \text{ sequence } \{x_i\}_{i=1}^{\infty} \text{ in } S \text{ converging to } x \\ &\forall \text{ sequence } \{t_i\}_{i=1}^{\infty} \text{ in } (0, \infty) \text{ converging to } 0 \\ &\exists \text{ sequence } \{v_i\}_{i=1}^{\infty} \text{ in } \mathbb{R}^n \text{ converging to } v : x_i + t_i v_i \in S \quad \forall i\}. \end{aligned}$$

Clarke's normal cone to S at x, denoted $N_S(x)$, is defined by

$$N_S(x) = \{\zeta \in (\mathbb{R}^n)^* : \zeta(v) \leq 0 \quad \forall v \in T_S(x)\}. \tag{2}$$

Given an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. The *epigraph of f*, denoted $\operatorname{epi} f$, is the following subset of $\mathbb{R}^{n+1}$:

$$\operatorname{epi} f = \{(x, r) \in \mathbb{R}^n \times \mathbb{R} : r \geq f(x)\}. \tag{3}$$

Clarke's generalized gradient $\boldsymbol{\partial} f(x)$ and Clarke's singular generalized gradient $\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x)$ are defined as follows:

$$\boldsymbol{\partial} f(x) = \{\zeta \in (\mathbb{R}^n)^* : (\zeta, -1) \in N_{\operatorname{epi} f}(x, f(x))\}, \tag{4}$$

$$\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x) = \{\zeta \in (\mathbb{R}^n)^* : (\zeta, 0) \in N_{\operatorname{epi} f}(x, f(x))\}. \tag{5}$$

If f is continuously differentiable at x, then $\boldsymbol{\partial} f(x)$ is just the singleton whose element is the classical gradient and $\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x) = \{0\}$.

We are now about to describe the approach taken in the present paper. However, before we do this, we first recall an alternative, equivalent definition of Clarke's generalized gradient valid for locally Lipschitz functions [4]. When $f : \mathbb{R}^n \to \mathbb{R}$ is a locally Lipschitz function and $x \in \mathbb{R}$, an alternative definition of $\boldsymbol{\partial} f(x)$ is based on *Clarke's generalized directional derivative*. Clarke's generalized directional derivative of f at x in the direction $v \in \mathbb{R}$, denoted $f^0(x; v)$, is defined as follows:

$$f^0(x; v) = \inf_{\varepsilon > 0} \sup_{\|\overline{x} - x\| < \varepsilon,\ 0 < t < \varepsilon} \frac{f(\overline{x} + tv) - f(\overline{x})}{t}. \tag{6}$$

Notice that, since f is locally Lipschitz, $f^0(x; v)$ is a real number. $\boldsymbol{\partial} f(x)$ may then be defined as the set

$$\{\zeta \in (\mathbb{R}^n)^* : \zeta(v) \leq f^0(x; v)\, \forall v \in \mathbb{R}^n\}. \tag{7}$$

The starting point of the present work was the observation that

(1) The objects defined by (6) and (7) capture information of f in a complete neighborhood of x, since all states $\overline{x}$ with $\|\overline{x} - x\| < \varepsilon$ are considered in (6).

[1] See [4, Definition 2.4.10]. Following the terminology of [9], Clarke's generalized gradient is the set of Clarke subgradients.

[2] See [4, p. 102]. Following the terminology of [9], Clarke's singular generalized gradient is the set of Clarke horizon subgradients.

(2) Formulas (6) and (7) still make sense when f is an arbitrary function from $\mathbb{R}^n$ to $\mathbb{R}$, not necessarily locally Lipschitz, provided we allow $f^0(x;v)$ to be an extended real number. This leads to a new generalized theory of differentiation, which will be studied in the present paper.

Since the subset of $(\mathbb{R}^n)^*$ obtained in this way will, in general, not coincide with Clarke's generalized gradient $\boldsymbol{\partial} f(x)$, we introduce a new term, the *semigradient* of f at x, and denote it by $\mathbf{SG}f(x)$. It is clear from the above discussion that $\boldsymbol{\partial} f(x)$ and $\mathbf{SG}f(x)$ coincide when f is locally Lipschitz at x. (In this case, $\boldsymbol{\partial} f(x)$ and $\mathbf{SG}f(x)$ may be used interchangeably. By convention, we will always use the notation $\boldsymbol{\partial} f(x)$ and speak about Clarke's generalized gradient when f is explicitly assumed to be locally Lipschitz at x.) In addition, we will introduce a *singular semigradient*, denoted $\mathbf{SG}^{\boldsymbol{\infty}} f(x)$, which will play a role similar to $\boldsymbol{\partial}^{\boldsymbol{\infty}} f(x)$.

3. Generalized directional derivative. For functions $f : \mathbb{R}^n \to \mathbb{R}$ which are *locally Lipschitz*, Clarke introduced the *generalized directional derivative* (e.g., [4]). In this section we investigate how this concept extends toward *arbitrary* functions $f : \mathbb{R}^n \to \mathbb{R}$. The generalized directional derivative for arbitrary f will serve as an intermediate step in defining what we call the (singular) semigradient for arbitrary f. This will be done in section 4.

DEFINITION 3.1. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. Let $x, v \in \mathbb{R}^n$. The generalized directional derivative of f at x in the direction v, denoted $f^0(x;v)$, is the extended real number defined by*

$$f^0(x;v) = \inf_{\varepsilon>0} \sup_{\|\overline{x}-x\|<\varepsilon,\ 0<t<\varepsilon} \frac{f(\overline{x}+tv) - f(\overline{x})}{t} \tag{8}$$

with $\|\cdot\|$ the Euclidean norm on $\mathbb{R}^n$.

It is clear that $f^0(x;0) = 0$.

Remark 1. Formula (8) is the one Clarke used in [4] to define the generalized directional derivative in the case of f being locally Lipschitz. In that case, $f^0(x;v)$ is a real number. Formula (8) makes sense for arbitrary f provided we allow $f^0(x;v)$ to be an extended real number.

The following proposition provides an alternative characterization of $f^0(x;v)$. (The proof is elementary and therefore omitted.)

PROPOSITION 3.2. *$f^0(x;v)$ is the maximum of $\limsup_{i\to\infty} \frac{f(x_i+t_iv)-f(x_i)}{t_i}$ over all the sequences $\{x_i\}_{i=1}^{\infty}$ in $\mathbb{R}^n$ converging to x and $\{t_i\}_{i=1}^{\infty}$ in $(0,\infty)$ converging to 0.*

In the remainder of this section we will study some properties of the generalized directional derivative for arbitrary functions $f : \mathbb{R}^n \to \mathbb{R}$.

PROPOSITION 3.3. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. Let $x \in \mathbb{R}^n$. The function $f^0(x;\cdot) : \mathbb{R}^n \to \mathbb{R} \cup \{-\infty, \infty\}$ is*

(i) *subadditive, that is, $f^0(x; v+w) \le f^0(x;v) + f^0(x;w)\ \forall v, w \in \mathbb{R}^n$ for which the sum in the right-hand side is defined;*[3]

(ii) *positively homogeneous, that is, $f^0(x;\lambda v) = \lambda f^0(x;v)\ \forall v \in \mathbb{R}^n, \lambda \in (0,\infty)$;*

(iii) *convex.*

Proof. The proof of (i) and (ii) is similar to the proof in the case of f being locally Lipschitz [4, Proposition 2.1.1.(a)] except for some straightforward modifications due to $f^0(x;\cdot)$ being extended real valued. We prove (iii). Take $v, w \in \mathbb{R}^n$ and $\lambda \in (0,1)$.

[3] In this paper, we follow the convention that the sums $-\infty+\infty$ and $\infty+(-\infty)$ are not defined.

By (i) and (ii)

$$f^0(x;(1-\lambda)v+\lambda w) \le (1-\lambda)f^0(x;v)+\lambda f^0(x;w) \tag{9}$$

if the sum in the right-hand side is defined. Hence, f is convex by [8, Theorem 4.2]. ∎

The previous proposition reveals that $f^0(x;\cdot)$ is a positively homogeneous convex function. (Positively homogeneous) convex functions are studied, e.g., in [8]. We recall some notions from that reference. The *effective domain* of $f^0(x;\cdot)$, denoted $\operatorname{dom} f^0(x;\cdot)$, is the set $\{v \in \mathbb{R}^n : f^0(x;v) < \infty\}$. $f^0(x;\cdot)$ is called *proper* iff it only takes values in $\mathbb{R} \cup \{\infty\}$. Otherwise, $f^0(x;\cdot)$ is called *improper*.[4] Let C be a subset of $\mathbb{R}^n$. C is a *cone* iff $\lambda v \in C\ \forall \lambda \in (0,\infty)$ and $v \in C$. Let C be a convex subset of $\mathbb{R}^n$. The *relative interior of* C, denoted $\operatorname{ri} C$, is the interior which results when C is regarded as a subset of its affine hull. The *relative boundary of* C, denoted $\operatorname{rbdy} C$, is defined as $\operatorname{cl} C \setminus \operatorname{ri} C$, with $\operatorname{cl} C$ the closure of the set C.

PROPOSITION 3.4. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. Let $x \in \mathbb{R}^n$.*

(i) $\operatorname{dom} f^0(x;\cdot)$, $\operatorname{ri}\operatorname{dom} f^0(x;\cdot)$, *and* $\operatorname{cl}\operatorname{dom} f^0(x;\cdot)$ *are nonempty convex cones.*

(ii) *If $f^0(x;\cdot)$ is proper, it is real valued continuous on* $\operatorname{ri}\operatorname{dom} f^0(x;\cdot)$. *If $f^0(x;\cdot)$ is improper, it is identically $-\infty$ on* $\operatorname{ri}\operatorname{dom} f^0(x;\cdot)$.

Proof. $\operatorname{dom} f^0(x;\cdot)$ is nonempty since $f^0(x;0) = 0 < \infty$, it is convex since $f^0(x;\cdot)$ is a convex function, and it is a cone since $f^0(x;\cdot)$ is positively homogeneous. Furthermore the relative interior and the closure of a nonempty convex cone are again nonempty convex cones [8, pp. 45, 50]. This establishes (i). (ii) follows from [8, Theorems 10.1 and 7.2]. ∎

4. Semigradient and singular semigradient. This section introduces the semigradient and the singular semigradient of an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ at a point $x \in \mathbb{R}^n$. The semigradient coincides with Clarke's generalized gradient when f is locally Lipschitz.

DEFINITION 4.1. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. Let $x \in \mathbb{R}^n$. The semigradient of f at x, denoted $\mathbf{SG}f(x)$, and the singular semigradient of f at x, denoted $\mathbf{SG}^{\infty}f(x)$, are subsets of $(\mathbb{R}^n)^*$, the dual of the vector space $\mathbb{R}^n$, defined by*

$$\mathbf{SG}f(x) = \{\zeta \in (\mathbb{R}^n)^* : \zeta(v) \le f^0(x;v)\ \forall v \in \mathbb{R}^n\}, \tag{10}$$

$$\mathbf{SG}^{\infty}f(x) = (\operatorname{dom} f^0(x;\cdot))^{\perp}, \tag{11}$$

where $(\operatorname{dom} f^0(x;\cdot))^{\perp}$ *denotes the polar*[5] *of* $\operatorname{dom} f^0(x;\cdot)$.

Remark 2. Formula (10) is the one Clarke used in [4] to define the generalized gradient for locally Lipschitz f. This formula still makes sense when working with arbitrary f. Formula (11) is introduced in the present paper. The resulting object $\mathbf{SG}^{\infty}f(x)$ will turn out to play a role similar to Clarke's singular generalized gradient $\boldsymbol{\partial}^{\infty}f(x)$ for arbitrary f.

$\mathbf{SG}f(x)$ is a closed convex set, possibly empty. This follows from [8, Corollary 13.2.1]. $\mathbf{SG}^{\infty}f(x)$ is a closed convex cone containing the origin.

The relation between the (singular) semigradient and Clarke's (singular) generalized gradient is the subject of the following theorem.

[4] This coincides with the notion of (im)properness in [8] since $f^0(x;0) = 0 < \infty$.

[5] The polar of a nonempty convex set $C \in \mathbb{R}^n$, denoted $C^{\perp}$, is the nonempty closed convex cone $\{\zeta \in (\mathbb{R}^n)^* : \zeta(v) \le 0\ \forall v \in C\}$. The polar of a nonempty convex set $D \in (\mathbb{R}^n)^*$, denoted $D^{\perp}$, is the nonempty closed convex cone $\{v \in \mathbb{R}^n : \zeta(v) \le 0\ \forall \zeta \in D\}$.

THEOREM 4.2 (relation with Clarke's theory). *Let $x \in \mathbb{R}^n$. For arbitrary $f : \mathbb{R}^n \to \mathbb{R}$ the following inclusions hold:*

$$\boldsymbol{\partial} f(x) \subseteq \mathbf{SG} f(x), \tag{12}$$

$$\{0\} \subseteq \boldsymbol{\partial}^{\infty} f(x) \subseteq \mathbf{SG}^{\infty} f(x). \tag{13}$$

In the particular case that $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz, these inclusions reduce to

$$\boldsymbol{\partial} f(x) = \mathbf{SG} f(x), \tag{14}$$

$$\{0\} = \boldsymbol{\partial}^{\infty} f(x) = \mathbf{SG}^{\infty} f(x). \tag{15}$$

Proof. The proof of (12) is similar to the proof of [4, Theorem 2.4.9.(i), "if" part], except for obvious modifications since now $f^0(x;\cdot)$ is extended real valued. The proof of (13) is similar to the proof of (12). (14) is proved in [4]. Finally, when f is locally Lipschitz, $f^0(x;\cdot)$ is real valued and hence $\mathbf{SG}^{\infty} f(x) = \{0\}$. Together with (13), this establishes (15). □

In the remainder of this section, we study some properties of $\mathbf{SG} f(x)$. In particular, we clarify the relationship between $\mathbf{SG} f(x)$ and $f^0(x;\cdot)$. As said above, $\mathbf{SG} f(x)$ is a closed convex subset of $(\mathbb{R}^n)^*$, possibly empty. We recall some known facts about closed convex subsets of $(\mathbb{R}^n)^*$ (e.g., [8]). Let C be a closed convex subset of $(\mathbb{R}^n)^*$. The support function σ_C of C is defined by

$$\sigma_C : \mathbb{R}^n \to \mathbb{R} \cup \{-\infty, \infty\} : v \mapsto \sup_{\zeta \in C} \zeta(v). \tag{16}$$

C and σ_C contain exactly the same information. That is, if C is known, then σ_C is known by (16), and, conversely, if σ_C is known, then C may be recovered by the following formula:

$$C = \{\zeta \in (\mathbb{R}^n)^* : \zeta(v) \leq \sigma_C(v)\, \forall v \in \mathbb{R}^n\}. \tag{17}$$

When $f : \mathbb{R}^n \to \mathbb{R}$ is locally Lipschitz, then $f^0(x;\cdot)$ is precisely the support function of $\boldsymbol{\partial} f(x)$ [4]. In other words, it is exactly equivalent to know $f^0(x;\cdot)$ or $\boldsymbol{\partial} f(x)$ when f is locally Lipschitz. However, this duality does not extend to the case of arbitrary functions f as studied in the present paper. That is, for arbitrary $f : \mathbb{R}^n \to \mathbb{R}$, $f^0(x;\cdot)$ might not be the support function of $\mathbf{SG} f(x)$, as may be seen from simple examples.[6] Although $f^0(x;\cdot)$ and the support function of $\mathbf{SG} f(x)$ are not necessarily the same, there still is a close relationship between these two functions. In order to reveal this relationship, we recall the following notion from [8]. The closure of the convex function $f^0(x;\cdot) : \mathbb{R}^n \to \mathbb{R} \cup \{-\infty, \infty\}$, denoted $\operatorname{cl} f^0(x;\cdot)$, is another convex function from $\mathbb{R}^n$ to $\mathbb{R} \cup \{-\infty, \infty\}$ defined as follows. If $f^0(x;\cdot)$ is proper, then $\operatorname{cl} f^0(x;\cdot)$ is the greatest lower semicontinuous function majorized by $f^0(x;\cdot)$. If $f^0(x;\cdot)$ is improper, then $\operatorname{cl} f^0(x;\cdot)$ is identically equal to $-\infty$.[7] The following proposition essentially says that $f^0(x;\cdot)$ and the support function of $\mathbf{SG} f(x)$ coincide on $\operatorname{ri} \operatorname{dom} f^0(x;\cdot)$.

PROPOSITION 4.3. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$. Let $x \in \mathbb{R}^n$.*

(i) *The support function of the closed convex set $\mathbf{SG} f(x)$ is $\operatorname{cl} f^0(x;\cdot)$.*

(ii) *If $f^0(x;\cdot)$ is proper, then $\operatorname{cl} f^0(x;\cdot)(v) = f^0(x;v) \forall v \in \mathbb{R}^n \setminus \operatorname{rbdy} \operatorname{dom} f^0(x;\cdot)$. If $f^0(x;\cdot)$ is improper, then $\operatorname{cl} f^0(x;\cdot)(v) = f^0(x;v)\ \forall v \in \operatorname{ri} \operatorname{dom} f^0(x;\cdot)$.*

Proof. (i) follows from [8, Corollary 13.2.1] since $f^0(x;0) = 0$. If $f^0(x;\cdot)$ is proper, then (ii) follows from [8, Theorem 7.4]. If $f^0(x;\cdot)$ is improper, then (ii) follows from Proposition 3.4(ii). □

[6] Take, for example, function f_2 from Example 2 at the origin.

[7] This coincides with the definition of closure of a function from [8] since $f^0(x;0) = 0$.

5. Calculus. In this section, two calculus rules will be presented: a calculus rule for scalar multiples and a sum rule.

5.1. Scalar multiples.

THEOREM 5.1. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a nonzero $\lambda \in \mathbb{R}$. Let $x \in \mathbb{R}^n$. Then,*

$$\mathbf{SG}(\lambda f)(x) = \lambda \mathbf{SG} f(x), \tag{18}$$

$$\mathbf{SG}^{\infty}(\lambda f)(x) = \lambda \mathbf{SG}^{\infty} f(x). \tag{19}$$

Remark 3. Equations (18) and (19) may not hold if $\lambda = 0$. Clearly, if $\lambda = 0$, then $\mathbf{SG}(\lambda f)(x) = \{0\}$ and $\mathbf{SG}^{\infty}(\lambda f)(x) = \{0\}$. Notice that $\mathbf{SG}(0f)(x) = \{0\}$ may be different from $0\mathbf{SG} f(x)$, since $\mathbf{SG} f(x)$ may be empty.

Notice that for the generalized theories of differentiation that are summarized in [9], the corresponding calculus rule for scalar multiples holds only for $\lambda > 0$ [9, p. 438]. This is illustrated in the following example.

Example 1. Consider $f_1 : \mathbb{R} \to \mathbb{R} : x \mapsto \sqrt{|x|}$. We have the equality

$$\mathbb{R} = \mathbf{SG}(-f_1)(0) = -\mathbf{SG} f_1(0) = \mathbb{R}.$$

Notice that any real number is a (regular, general, proximal, Clarke) subgradient for f_1 at 0, whereas $-f_1$ does not have any (regular, general, proximal, Clarke) subgradient at 0.

Proof of Theorem 5.1. Modulo modifications for $f^0(x; \cdot)$ being extended real valued, the proof is the same as in the locally Lipschitz case [4]. □

5.2. Sum rule.

THEOREM 5.2. *Assume two arbitrary functions f and g from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. Then, at least one of the following two cases holds.*

(1) *Good case.*

$$\mathbf{SG}(f + g)(x) \subseteq \mathrm{cl}\,(\mathbf{SG} f(x) + \mathbf{SG} g(x)), \tag{20}$$

$$\mathbf{SG}^{\infty}(f + g)(x) \subseteq \mathrm{cl}\,(\mathbf{SG}^{\infty} f(x) + \mathbf{SG}^{\infty} g(x)). \tag{21}$$

(2) *Bad case. There is a nonzero $\zeta \in \mathbf{SG}^{\infty} f(x)$ and a nonzero $\xi \in \mathbf{SG}^{\infty} g(x)$ such that* (a) $\zeta + \xi = 0$ *and* (b) $-\zeta \notin \mathbf{SG}^{\infty} f(x)$ *or* $-\xi \notin \mathbf{SG}^{\infty} g(x)$.

We illustrate the role of the bad case in Theorem 5.2 with a simple example, as follows.

Example 2. Consider

$$f_2 : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} 1 & \text{if } x_1 < x_2^2, \\ 0 & \text{if } x_1 \geq x_2^2, \end{cases}$$

and $f_3 : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto f_2(-x_1, x_2)$. The bad case is satisfied, and indeed we see that the expected sum rule (the good case) does not hold:

$$\mathbb{R}^2 = \mathbf{SG}(f_2 + f_3)(0,0) \not\subseteq \mathrm{cl}\ (\mathbf{SG} f_2(0,0) + \mathbf{SG} f_3(0,0)) = \mathbb{R} \times \{0\},$$
$$\mathbb{R}^2 = \mathbf{SG}^{\infty}(f_2 + f_3)(0,0) \not\subseteq \mathrm{cl}\ (\mathbf{SG}^{\infty} f_2(0,0) + \mathbf{SG}^{\infty} f_3(0,0)) = \mathbb{R} \times \{0\}.$$

There is a remarkable similarity between Theorem 5.2 and the sum rule for Clarke's generalized gradients [5, Proposition 5A.4.]:[8] let f and g be lower semicontinuous functions from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. Then, at least one of the following two cases holds.

(1) Good case.

$$\boldsymbol{\partial}(f+g)(x) \subseteq \boldsymbol{\partial} f(x) + \boldsymbol{\partial} g(x), \tag{22}$$

$$\boldsymbol{\partial}^{\boldsymbol{\infty}}(f+g)(x) \subseteq \boldsymbol{\partial}^{\boldsymbol{\infty}} f(x) + \boldsymbol{\partial}^{\boldsymbol{\infty}} g(x). \tag{23}$$

(2) Bad case. There is a nonzero $\zeta \in \boldsymbol{\partial}^{\boldsymbol{\infty}} f(x)$ and a nonzero $\xi \in \boldsymbol{\partial}^{\boldsymbol{\infty}} g(x)$ such that $\zeta + \xi = 0$.

Notice that, unlike the sum rule for Clarke's generalized gradients, Theorem 5.2 does not require f and g to be lower semicontinuous. Furthermore, the respective bad cases are somewhat different. We illustrate this with two simple examples, as follows.

Example 3. Consider

$$f_4 : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} -1 & \text{if } x_1 = 0, \\ 0 & \text{if } x_1 \neq 0, \end{cases}$$

and $f_5 = -f_4$. We leave it as an exercise to the reader to check that our bad case is not fulfilled, and hence the good case holds. Indeed,

$$\{(0,0)\} = \mathbf{SG}(f_4+f_5)(0,0) \subseteq \mathrm{cl}\,(\mathbf{SG} f_4(0,0) + \mathbf{SG} f_5(0,0)) = \mathbb{R} \times \{0\},$$
$$\{(0,0)\} = \mathbf{SG}^{\boldsymbol{\infty}}(f_4+f_5)(0,0) \subseteq \mathrm{cl}\,(\mathbf{SG}^{\boldsymbol{\infty}} f_4(0,0) + \mathbf{SG}^{\boldsymbol{\infty}} f_5(0,0)) = \mathbb{R} \times \{0\}.$$

The sum rule for Clarke's generalized gradients, however, is not applicable, since f_5 is not lower semicontinuous. Indeed,

$$\{(0,0)\} = \boldsymbol{\partial}(f_4+f_5)(0,0) \not\subseteq \boldsymbol{\partial} f_4(0,0) + \boldsymbol{\partial} f_5(0,0) = \emptyset,$$
$$\{(0,0)\} = \boldsymbol{\partial}^{\boldsymbol{\infty}}(f_4+f_5)(0,0) \subseteq \boldsymbol{\partial}^{\boldsymbol{\infty}} f_4(0,0) + \boldsymbol{\partial}^{\boldsymbol{\infty}} f_5(0,0) = \mathbb{R} \times \{0\}.$$

Example 4. Consider $f_6 : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \sqrt{|x_1|}$ and $f_7 = -f_6$. Our bad case is not fulfilled due to the second condition (b), and hence the good case holds:

$$\{(0,0)\} = \mathbf{SG}(f_6+f_7)(0,0) \subseteq \mathrm{cl}\,(\mathbf{SG} f_6(0,0) + \mathbf{SG} f_7(0,0)) = \mathbb{R} \times \{0\},$$
$$\{(0,0)\} = \mathbf{SG}^{\boldsymbol{\infty}}(f_6+f_7)(0,0) \subseteq \mathrm{cl}\,(\mathbf{SG}^{\boldsymbol{\infty}} f_6(0,0) + \mathbf{SG}^{\boldsymbol{\infty}} f_7(0,0)) = \mathbb{R} \times \{0\}.$$

Both functions are lower semicontinuous, but the bad case of Clarke's sum rule is fulfilled, and indeed,

$$\{(0,0)\} = \boldsymbol{\partial}(f_6+f_7)(0,0) \not\subseteq \boldsymbol{\partial} f_6(0,0) + \boldsymbol{\partial} f_7(0,0) = \emptyset,$$
$$\{(0,0)\} = \boldsymbol{\partial}^{\boldsymbol{\infty}}(f_6+f_7)(0,0) \subseteq \boldsymbol{\partial}^{\boldsymbol{\infty}} f_6(0,0) + \boldsymbol{\partial}^{\boldsymbol{\infty}} f_7(0,0) = \mathbb{R} \times \{0\}.$$

Proof of Theorem 5.2. This theorem follows immediately from Lemmas A.1 to A.4 stated and proved in the appendix. □

[8] The sum rule in that reference is somewhat more general than the version which we state here: in that reference it applies to the sum of a finite number of functions from $\mathbb{R}^n$ to $\mathbb{R} \cup \{\infty\}$.

6. Local optimization. In this section we consider the problem of optimizing a given function $f : \mathbb{R}^n \to \mathbb{R}$. Consider, for example, the function

$$f : \mathbb{R} \to \mathbb{R} : x \mapsto \begin{cases} -x & \text{if } x \leq 0, \\ x - 1 & \text{if } x > 0. \end{cases}$$

This function has a local infimum (which is actually global) at $x = 0$. If this function represents a certain cost, and we want to reduce the cost, then we would certainly be interested in the point $x = 0$. Hence it might be interesting to have a generalized theory of differentiation that detects this infimum. As argued in the introduction, the (regular, general, proximal, Clarke) subgradients studied, e.g., in [9] are not a suitable tool with which to detect this infimum. This is because these objects only incorporate information of the epigraph of f near $(0, 0)$ and thus discard the behavior of f for $x > 0$, which is crucial for f to have an infimum or not. Therefore, although this function does have an infimum at $x = 0$, the number 0 is not a (regular, general, proximal, Clarke) subgradient of f at 0. On the other hand, the semigradient does detect this infimum; that is,

$$0 \in (-\infty, 1] = \mathbf{SG}f(0).$$

This is because the semigradient is defined using the generalized directional derivative, and this latter uses information of f in complete neighborhoods of x and hence takes into account the behavior of f for $x > 0$ equally well as for $x < 0$. In general it turns out that the semigradient is a suitable tool with which to detect minima, maxima, finite infima, and finite suprema.

DEFINITION 6.1. *Let x be a point in $\mathbb{R}^n$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to have a local infimum at x if there exists a neighborhood U of x such that $f(\hat{x}) \geq \sup_{\varepsilon>0} \inf_{\|\overline{x}-x\|<\varepsilon} f(\overline{x})$ $\forall \hat{x} \in U$. In this case $\sup_{\varepsilon>0} \inf_{\|\overline{x}-x\|<\varepsilon} f(\overline{x})$ is called a local infimum. Similar definitions are made for a local supremum.*

THEOREM 6.2 (local optimization). *Let $f : \mathbb{R}^n \to \mathbb{R}$ be an arbitrary function and let x be a point in $\mathbb{R}^n$.*

(i) *If f has a local minimum or maximum at x, then $0 \in \mathbf{SG}f(x)$.*

(ii) *If* (a) *f has a local infimum or supremum at x and* (b) *this local infimum or supremum is finite, then $0 \in \mathbf{SG}f(x)$.*

We illustrate this theorem with two examples. The first example concerns the detection of maxima.

Example 5. Consider $f_8 : \mathbb{R} \to \mathbb{R} : x \mapsto -\sqrt{|x|}$. This function has a local maximum (which is actually global) at $x = 0$, and one verifies easily that $0 \in \mathbb{R} = \mathbf{SG}f_8(0)$ in agreement with Theorem 6.2. To compare, notice that f_8 does not have a (regular, general, proximal, Clarke) subgradient at 0 and thus in particular 0 is not a (regular, general, proximal, Clarke) subgradient of f_8 at 0.

The second example shows that *infinite* infima or suprema might not be detected by the semigradient.

Example 6. Consider

$$f_{10} : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} \ln|x_1| - x_2 & \text{if } x_1 \neq 0, \\ -x_2 & \text{if } x_1 = 0. \end{cases}$$

Clearly f_{10} has an infimum at $(0, 0)$. This infimum equals $-\infty$. However, $0 \notin \mathbf{SG}f_{10}(0, 0)$ since $f_{10}^0((0, 0); (0, 1)) = -1$.

Proof of Theorem 6.2 (i) follows immediately from (ii) so we proceed with the proof of (ii). Suppose f has a finite local *infimum* at x. Denote this finite local infimum by $\underline{m}$; that is, $\underline{m} = \sup_{\varepsilon>0} \inf_{\|\overline{x}-x\|<\varepsilon} f(\overline{x}) \in \mathbb{R}$. Let, for $\delta \in (0, \infty)$, $B^n_\delta(x)$ denote the open ball in $\mathbb{R}^n$ centered at x with radius δ. By definition, there is a $\delta \in (0, \infty)$ such that $f(\hat{x}) \geq \underline{m}\ \forall \hat{x} \in B^n_\delta(x)$. Furthermore, we may construct a sequence $\{x_i\}_{i=1}^\infty$ in $B^n_{\frac{\delta}{2}}(x)$ converging to x such that $\lim_{i\to\infty} f(x_i) = \underline{m}$. Fix a nonzero $v \in \mathbb{R}^n$. We will show that $f^0(x; v) \geq 0$. Let $t_i = \frac{\delta}{2^i\|v\|}$ $(i = 1, 2, \dots)$. We construct a subsequence of $\{x_i\}_{i=1}^\infty$ as follows. For every $i = 1, 2, \dots$, we pick an element $x_{i'}$ from this sequence for which $f(\hat{x}) \geq f(x_{i'}) - \frac{\delta}{i2^i\|v\|}\ \forall\ \hat{x} \in B^n_\delta(x)$. Hence, after relabeling this subsequence as $\{x_i\}_{i=1}^\infty$,

$$\frac{f(x_i + t_i v) - f(x_i)}{t_i} \geq \frac{-\frac{\delta}{i2^i\|v\|}}{\frac{\delta}{2^i\|v\|}} = \frac{-1}{i}.$$

From this, $\limsup_{i\to\infty} \frac{f(x_i+t_iv)-f(x_i)}{t_i} \geq 0$. Using Proposition 3.2 we have thus proved that, for all nonzero $v \in \mathbb{R}^n$, $f^0(x; v) \geq 0$. Therefore we conclude $0 \in \mathbf{SG}f(x)$. The proof of the supremum case is immediate from the observation that $\mathbf{SG}(-f)(x) = -\mathbf{SG}f(x)$. ∎

7. Descent directions. Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$. This section shows how $\mathbf{SG}f(x)$ and $\mathbf{SG}^\infty f(x)$ may lead to information about those points $\overline{x}$ where $f(\overline{x}) < f(x)$.

THEOREM 7.1 (descent directions). *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$.*

(i) *If $0 \notin \mathbf{SG}f(x) \neq \emptyset$, then at least one of the following two cases holds.*

(1) *Good case. For every nonempty closed convex cone $K \subseteq \mathrm{ri}\,(\mathbf{SG}f(x)^\perp) \cup \{0\}$, there is a neighborhood U of 0 in $\mathbb{R}^n$ with the property*

$$f(\overline{x}) < f(x) \qquad \forall \overline{x} \in \{x\} + (U \cap K \setminus \{0\}). \tag{24}$$

(2) *Bad case. There is a nonzero $\zeta \in$ closed convex cone generated by*[9] $\mathbf{SG}f(x)$ *and a nonzero $\xi \in \mathbf{SG}^\infty f(x)$ such that* (a) $\zeta + \xi = 0$ *and* (b) $-\zeta \notin$ *closed convex cone generated by $\mathbf{SG}f(x)$ or $-\xi \notin \mathbf{SG}^\infty f(x)$.*

(ii) *If $\mathbf{SG}f(x) = \emptyset$, then, for every nonempty closed convex cone K contained in $\mathrm{ri}\,(\mathbf{SG}^\infty f(x)^\perp) \cup \{0\}$, there is a neighborhood U of 0 in $\mathbb{R}^n$ with the property*

$$f(\overline{x}) < f(x) \qquad \forall \overline{x} \in \{x\} + (U \cap K \setminus \{0\}). \tag{25}$$

Proof. This theorem follows immediately from Lemmas A.5 and A.6 in the appendix. ∎

We illustrate this theorem with three examples.

Example 7. Let

$$f_{11} : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} x_1 & \text{if } |x_2| \leq -x_1, \\ x_1 + 1 & \text{if } |x_2| > -x_1. \end{cases}$$

$\mathbf{SG}f_{11}(0,0) = \cup_{\lambda\in[0,\,\infty)}\{\lambda+1\} \times [-\lambda,\, \lambda]$ and $\mathbf{SG}^\infty f_{11}(0,0) = \cup_{\lambda\in[0,\,\infty)}\{\lambda\} \times [-\lambda,\, \lambda]$. Part (i) of Theorem 7.1 applies and the bad case is not fulfilled. Hence the good case holds with $\mathrm{ri}\,(\mathbf{SG}f_{11}(0,0)^\perp) \cup \{(0,0)\} = \big(\cup_{\lambda\in(0,\,\infty)}\{-\lambda\} \times (-\lambda,\, \lambda)\big) \cup \{(0,0)\}$.

[9] The closed convex cone generated by $\mathbf{SG}f(x)$ is the set $\mathrm{cl}\,(\cup_{\lambda\in(0,\,\infty)}\lambda\mathbf{SG}f(x))$.

Example 8. Let

$$f_{12} : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} x_1 & \text{if } |x_2| \geq x_1, \\ x_1 + 1 & \text{if } |x_2| < x_1. \end{cases}$$

$\mathbf{SG} f_{12}(0,0) = \cup_{\lambda \in [0,\,\infty)} \{\lambda + 1\} \times [-\lambda,\, \lambda]$ and $\mathbf{SG}^{\infty} f_{12}(0,0) = \cup_{\lambda \in [0,\,\infty)} \{\lambda\} \times [-\lambda,\, \lambda]$. Part (i) of Theorem 7.1 applies and the bad case is not fulfilled. Hence the good case holds with $\operatorname{ri}(\mathbf{SG} f_{12}(0,0)^{\perp}) \cup \{(0,0)\} = \big(\cup_{\lambda \in (0,\,\infty)} \{-\lambda\} \times (-\lambda,\, \lambda)\big) \cup \{(0,0)\}$.

Example 9. Let

$$f_{13} : \mathbb{R}^2 \to \mathbb{R} : (x_1, x_2) \mapsto \begin{cases} (x_1)^{\frac{1}{3}} & \text{if } |x_2| \leq -x_1, \\ (x_1)^{\frac{1}{3}} + 1 & \text{if } |x_2| > -x_1. \end{cases}$$

$\mathbf{SG} f_{13}(0,0) = \emptyset$ and $\mathbf{SG}^{\infty} f_{13}(0,0) = \cup_{\lambda \in [0,\,\infty)} \{\lambda\} \times [-\lambda,\, \lambda]$. Part (ii) of Theorem 7.1 applies with $\operatorname{ri}(\mathbf{SG}^{\infty} f_{13}(0,0)^{\perp}) \cup \{(0,0)\} = \big(\cup_{\lambda \in (0,\,\infty)} \{-\lambda\} \times (-\lambda,\, \lambda)\big) \cup \{(0,0)\}$.

If we compare the estimates given by Theorem 7.1 with the actual descent directions in these examples, then we see the following. In Examples 7 and 9, Theorem 7.1 gives very good estimates; in Example 8 it gives conservative estimates.

8. Optimal control. Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x^* \in \mathbb{R}^n$. We will apply the results of the previous section. We will show how $\mathbf{SG} f(x^*)$ and $\mathbf{SG}^{\infty} f(x^*)$ may lead to a linear approximation of the set $S = \{x \in \mathbb{R}^n : f(x) < f(x^*)\} \cup \{x^*\}$ at the point x^*. The linear approximation considered here is a *generalized approximating cone* [10]. This will provide a possible application of (singular) semigradients to a class of nonsmooth optimal control problems.

The definition of a generalized approximating cone may be found in [10] and will not be repeated here. Instead we state a sufficient condition for a cone D in $\mathbb{R}^n$ to be a generalized approximating cone for a set $S \subseteq \mathbb{R}^n$ at a point $x^* \in S$.

LEMMA 8.1. *Let $S \subseteq \mathbb{R}^n$, $x^* \in S$. A closed convex cone D in $\mathbb{R}^n$, which is the closure of the union of an increasing sequence $D_1 \subseteq D_2 \subseteq D_3 \subseteq \cdots$ of closed convex cones such that for each $i = 1, 2, \ldots$ there exists a neighborhood U_i of 0 in $\mathbb{R}^n$ such that the image of the map $U_i \cap D_i \to \mathbb{R}^n : v \mapsto x^* + v$ is contained in S, is a generalized approximating cone for S at x^*.*

Proof. This lemma follows immediately from [10, Definition 8]. □

Let S be the particular set $\{x \in \mathbb{R}^n : f(x) < f(x^*)\} \cup \{x^*\}$. The following statement shows how $\mathbf{SG} f(x^*)$ and $\mathbf{SG}^{\infty} f(x^*)$ may lead to a generalized approximating cone for this particular S at x^*.

THEOREM 8.2 (generalized approximating cone). *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x^* \in \mathbb{R}^n$. Let S be the set $\{x \in \mathbb{R}^n : f(x) < f(x^*)\} \cup \{x^*\}$.*

(i) *If $0 \notin \mathbf{SG} f(x^*) \neq \emptyset$, then at least one of the following two cases holds.*

(1) *Good case. $\mathbf{SG} f(x^*)^{\perp}$ is a generalized approximating cone for S at x^*.*

(2) *Bad case. There is a nonzero $\zeta \in$ closed convex cone generated by $\mathbf{SG} f(x^*)$ and a nonzero $\xi \in \mathbf{SG}^{\infty} f(x^*)$ such that* (a) $\zeta + \xi = 0$ *and* (b) $-\zeta \notin$ *closed convex cone generated by $\mathbf{SG} f(x^*)$ or $-\xi \notin \mathbf{SG}^{\infty} f(x^*)$.*

(ii) *If $\mathbf{SG} f(x^*) = \emptyset$, then $\mathbf{SG}^{\infty} f(x^*)^{\perp}$ is a generalized approximating cone for S at x^*.*

Proof. This theorem follows from Lemmas A.6 and A.7 in the appendix. □

We now show how this relates to optimal control. This will provide a possible application of the semigradient and the singular semigradient in optimal control problems with discontinuous costs. The following nonsmooth optimal control problem is

considered: Assume a sufficiently smooth[10] control system $\dot{x} = X(t, x, u)$ ($t \in \mathbb{R}$, $x \in \mathbb{R}^n$, $u \in \mathcal{U} \subseteq \mathbb{R}^s$) and an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$, the endpoint cost. The optimal control problem asks for minimizing $f(\chi(t_1))$ over all trajectories χ of the control system defined on a fixed time interval $[t_0, t_1]$ and starting in a fixed point x_0.

One way to look at this problem is as follows. Pick a reference trajectory χ^* of the control system (corresponding to some open-loop control) defined on $[t_0, t_1]$ starting in x_0. Let $x^* = \chi^*(t_1)$. We would like to test this reference trajectory for optimality. Let S be the set $\{x \in \mathbb{R}^n : f(x) < f(x^*)\} \cup \{x^*\}$. Clearly, a necessary and sufficient condition for χ^* to solve the optimal control problem is that there does not exist a trajectory of the control system defined on $[t_0, t_1]$ starting in x_0 with endpoint in $S \setminus \{x^*\}$. In other words, the set of points reachable from x_0 over the interval $[t_0, t_1]$ and the set S should have no common points other than x^*. Under suitable technical hypotheses, necessary conditions for χ^* to have this property may be found in [10]. These necessary conditions require a generalized approximating cone for S at x^*. It is at this point that $\mathbf{SG}f(x^*)$ and $\mathbf{SG}^{\infty}f(x^*)$ may play a possible role. Indeed, Theorem 8.2 shows how $\mathbf{SG}f(x^*)$ and $\mathbf{SG}^{\infty}f(x^*)$ may lead to such a generalized approximating cone.

Remark 4. Theorem 8.2 provides, under certain conditions, a generalized approximating cone for S at x^*. However, a generalized approximating cone for S at x^* is not a uniquely defined concept. There might be other generalized approximating cones for S at x^* which strictly contain the one provided by Theorem 8.2. These could give stronger results when plugged into the maximum principle of optimal control as stated in [10]. Take, for example, the function f_{12} from Example 8 and $x^* = (0, 0)$. In this case, $S = ((-\infty, 0) \times \mathbb{R}) \cup \{(0, 0)\}$. The set $(-\infty, 0] \times \mathbb{R}$ is a generalized approximating cone for S at $(0, 0)$ that strictly contains $\mathbf{SG}f_{12}(0, 0)^{\perp} = \cup_{\lambda \in [0, \infty)}\{-\lambda\} \times [-\lambda, \lambda]$. This latter is the generalized approximating cone provided by Theorem 8.2. Notice, however, that for the functions f_{11} and f_{13} (Examples 7 and 9) the generalized approximating cone provided by Theorem 8.2 is the best possible one.

9. Conclusion. We have reported some preliminary ideas of a generalized theory of differentiation for arbitrary functions from $\mathbb{R}^n$ to $\mathbb{R}$. The proposed theory associates with an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$ two subsets of the dual space $(\mathbb{R}^n)^*$ which we have called the semigradient and the singular semigradient of f at x. The proposed theory is closely related to and inspired by Clarke's nonsmooth analysis [4] and actually coincides with it when the functions involved are locally Lipschitz.

We have stated two calculus rules and applied the theory to the study of three nonsmooth optimization problems: we considered the problem of optimizing an arbitrary function f from $\mathbb{R}^n$ to $\mathbb{R}$, we have shown how the (singular) semigradient leads to information about those points $\overline{x}$ where $f(\overline{x}) < f(x)$, and finally we have described a possible role for the (singular) semigradient to play in optimal control problems with discontinuous costs.

The present approach differs from other approaches—(regular, general, proximal, Clarke) subgradients—studied in [9] in the following respect: the (regular, general, proximal, Clarke) subgradients can be characterized geometrically in terms of normals to the epigraph of f. This implies in particular that these subgradients only incorporate information of the epigraph of f in the neighborhood of $(x, f(x))$. Therefore,

[10] We omit the precise specification of technical details for the control system, since this will not be needed in the present discussion.

when f is discontinuous, these subgradients do not take into account the full behavior of f in a complete neighborhood of the point x. This is in contrast with the (singular) semigradient, which by definition incorporates information of f in a complete neighborhood of x. Accordingly, the semigradient turns out to be a valuable tool in optimization problems with discontinuous costs, where minima, maxima, infima, and suprema are sought.

Appendix. Technical lemmas.

LEMMA A.1. *Assume two arbitrary functions f and g from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. Then,* $\operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot) = \emptyset$ *iff there is a nonzero* $\zeta \in \mathbf{SG}^{\infty} f(x)$ *and a nonzero* $\xi \in \mathbf{SG}^{\infty} g(x)$ *such that* (a) $\zeta + \xi = 0$ *and* (b) $-\zeta \notin \mathbf{SG}^{\infty} f(x)$ *or* $-\xi \notin \mathbf{SG}^{\infty} g(x)$.

Proof. By [8, Theorem 11.3], $\operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot) = \emptyset$ iff there exists a hyper plane $\boldsymbol{\pi}$ *separating* $\operatorname{dom} f^0(x;\cdot)$ and $\operatorname{dom} g^0(x;\cdot)$ *properly*. This means (a) that $\operatorname{dom} f^0(x;\cdot)$ is contained in one closed half-space associated with $\boldsymbol{\pi}$ and $\operatorname{dom} g^0(x;\cdot)$ in the opposite closed half-space, and (b) that $\operatorname{dom} f^0(x;\cdot)$ and $\operatorname{dom} g^0(x;\cdot)$ are not both contained in $\boldsymbol{\pi}$ itself. Since both $\operatorname{dom} f^0(x;\cdot)$ and $\operatorname{dom} g^0(x;\cdot)$ contain the origin, this hyper plane $\boldsymbol{\pi}$ has to contain the origin. Hence, this is equivalent to the existence of a nonzero $\zeta \in (\operatorname{dom} f^0(x;\cdot))^{\perp}$ and a nonzero $\xi \in (\operatorname{dom} g^0(x;\cdot))^{\perp}$ such that (a) $\zeta + \xi = 0$ and (b) $-\zeta \notin (\operatorname{dom} f^0(x;\cdot))^{\perp}$ or $-\xi \notin (\operatorname{dom} g^0(x;\cdot))^{\perp}$. By the definition of the singular semigradient (Definition 4.1), this establishes the lemma. ∎

LEMMA A.2. *Assume two arbitrary functions f and g from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. If* $\operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot) \neq \emptyset$, *then*

(i) (a) $f^0(x;v) + g^0(x;v) = -\infty$ *for some* $v \in \mathbb{R}^n$ *or* (b) $f^0(x;\cdot)$ *and* $g^0(x;\cdot)$ *are proper convex functions and* $\operatorname{cl} f^0(x;\cdot) + \operatorname{cl} g^0(x;\cdot) \geq \operatorname{cl}(f^0(x;\cdot) + g^0(x;\cdot))$,[11]

(ii) $\operatorname{cl}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{cl}\operatorname{dom} g^0(x;\cdot) \subseteq \operatorname{cl}(\operatorname{dom} f^0(x;\cdot) \cap \operatorname{dom} g^0(x;\cdot))$.

Proof. We first prove (i). If $f^0(x;\cdot)$ or $g^0(x;\cdot)$ is an improper convex function, then $f^0(x;v) + g^0(x;v)$ is defined and equal to $-\infty$ for every $v \in \operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot)$ (by Proposition 3.4). Suppose now that both $f^0(x;\cdot)$ and $g^0(x;\cdot)$ are proper convex functions. We will show that in this case $\operatorname{cl} f^0(x;\cdot) + \operatorname{cl} g^0(x;\cdot) \geq \operatorname{cl}(f^0(x;\cdot)+g^0(x;\cdot))$. First of all, notice that, by the properness of $f^0(x;\cdot)$ and $g^0(x;\cdot)$ and by [8, Theorems 5.2 and 7.4], the left-hand and right-hand sides are defined. Take an arbitrary $w \in \mathbb{R}^n$. Pick a $v \in \operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot)$. For $\lambda \in [0, 1)$, define w_λ as $(1-\lambda)v+\lambda w$. By definition, $f^0(x;\cdot)(w_\lambda)+g^0(x;\cdot)(w_\lambda) = (f^0(x;\cdot)+g^0(x;\cdot))(w_\lambda)$. On the one hand, by [8, Theorem 7.5], the two terms in the left-hand side converge, as $\lambda \uparrow 1$, to, respectively, $\operatorname{cl} f^0(x;\cdot)(w)$ and $\operatorname{cl} g^0(x;\cdot)(w)$. The right-hand side then converges, as $\lambda \uparrow 1$, to $\operatorname{cl} f^0(x;\cdot)(w) + \operatorname{cl} g^0(x;\cdot)(w)$. On the other hand, by [8, p. 52] this limit is $\geq \operatorname{cl}(f^0(x;\cdot) + g^0(x;\cdot))(w)$. This establishes (i).

We prove (ii). Pick an arbitrary $v \in \operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot)$. For each $w \in \operatorname{cl}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{cl}\operatorname{dom} g^0(x;\cdot)$, $(1-\lambda)v + \lambda w \in \operatorname{ri}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{ri}\operatorname{dom} g^0(x;\cdot)$ for all $\lambda \in [0, 1)$ by [8, Theorem 6.1]. Hence w belongs to the closure of $\operatorname{dom} f^0(x;\cdot) \cap \operatorname{dom} g^0(x;\cdot)$. This proves (ii). ∎

LEMMA A.3. *Assume two arbitrary functions f and g from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. If* (a) $f^0(x;v)+g^0(x;v) = -\infty$ *for some* $v \in \mathbb{R}^n$ *or* (b) $f^0(x;\cdot)$ *and* $g^0(x;\cdot)$ *are proper convex functions and* $\operatorname{cl} f^0(x;\cdot) + \operatorname{cl} g^0(x;\cdot) \geq \operatorname{cl}(f^0(x;\cdot) + g^0(x;\cdot))$, *then*

$$\mathbf{SG}(f+g)(x) \subseteq \operatorname{cl}(\mathbf{SG} f(x) + \mathbf{SG} g(x)). \tag{26}$$

[11] The closure of a convex function $c : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$, denoted $\operatorname{cl} c$, is defined as the greatest lower semicontinuous function from $\mathbb{R}^n$ to $\mathbb{R} \cup \{\infty\}$ majorized by c.

Proof. First of all, notice that

$$(f+g)^0(x;v) \leq f^0(x;v) + g^0(x;v) \tag{27}$$

for all $v \in \mathbb{R}^n$ for which the sum in the right-hand side is defined. This follows readily from Proposition 3.2. Consider case (a). If $f^0(x;v)+g^0(x;v) = -\infty$ for some $v \in \mathbb{R}^n$, then $(f+g)^0(x;\cdot)$ is improper by (27) and hence $\mathbf{SG}(f+g)(x) = \emptyset$. In this case the conclusion of the lemma holds trivially. Consider (b). If $f^0(x;\cdot)$ and $g^0(x;\cdot)$ are proper convex functions, then the right-hand side in (27) is defined $\forall\ v \in \mathbb{R}^n$. Taking closures, we get $\operatorname{cl}(f+g)^0(x;\cdot) \leq \operatorname{cl}(f^0(x;\cdot) + g^0(x;\cdot))$, where we have used [8, p. 53 and Theorem 5.2]. We get $\operatorname{cl}(f+g)^0(x;\cdot) \leq \operatorname{cl} f^0(x;\cdot) + \operatorname{cl} g^0(x;\cdot)$. In this equation, the left-hand and right-hand sides are support functions of, respectively, $\mathbf{SG}(f+g)(x)$ and $\mathbf{SG}f(x) + \mathbf{SG}g(x)$ by [8, p. 113]. Using [8, p. 113], the lemma is proved. □

LEMMA A.4. *Assume two arbitrary functions f and g from $\mathbb{R}^n$ to $\mathbb{R}$. Let $x \in \mathbb{R}^n$. If $\operatorname{cl}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{cl}\operatorname{dom} g^0(x;\cdot) \subseteq \operatorname{cl}(\operatorname{dom} f^0(x;\cdot) \cap \operatorname{dom} g^0(x;\cdot))$, then*

$$\mathbf{SG}^{\infty}(f+g)(x) \subseteq \operatorname{cl}(\mathbf{SG}^{\infty}f(x) + \mathbf{SG}^{\infty}g(x)). \tag{28}$$

Proof. As noticed in the proof of the previous lemma, $(f+g)^0(x;v) \leq f^0(x;v) + g^0(x;v)\ \forall\ v \in \mathbb{R}^n$ for which the sum in the right-hand side is defined. Hence, $\operatorname{dom} f^0(x;\cdot) \cap \operatorname{dom} g^0(x;\cdot) \subseteq \operatorname{dom}(f+g)^0(x;\cdot)$ and thus, taking closures, we obtain $\operatorname{cl}(\operatorname{dom} f^0(x;\cdot) \cap \operatorname{dom} g^0(x;\cdot)) \subseteq \operatorname{cl}\operatorname{dom}(f+g)^0(x;\cdot)$. Together with the hypothesis of the lemma, we get $\operatorname{cl}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{cl}\operatorname{dom} g^0(x;\cdot) \subseteq \operatorname{cl}\operatorname{dom}(f+g)^0(x;\cdot)$. Taking polars, it follows that $(\operatorname{cl}\operatorname{dom}(f+g)^0(x;\cdot))^{\perp} \subseteq (\operatorname{cl}\operatorname{dom} f^0(x;\cdot) \cap \operatorname{cl}\operatorname{dom} g^0(x;\cdot))^{\perp}$. Since for two nonempty closed convex cones K and L in $\mathbb{R}^n$, $(K \cap L)^{\perp} = (K^{\perp} + L^{\perp})^{\perp\perp} = \operatorname{cl}(K^{\perp} + L^{\perp})$, we get $(\operatorname{cl}\operatorname{dom}(f+g)^0(x;\cdot))^{\perp} \subseteq \operatorname{cl}((\operatorname{cl}\operatorname{dom} f^0(x;\cdot))^{\perp} + (\operatorname{cl}\operatorname{dom} g^0(x;\cdot))^{\perp})$. Hence, using the fact that for a nonempty convex cone K in $\mathbb{R}^n$ $(\operatorname{cl} K)^{\perp} = K^{\perp}$ and using the definition of the singular semigradient (Definition 4.1), the conclusion of the lemma follows. □

LEMMA A.5. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$. Let C be a convex cone in $\mathbb{R}^n$ such that $f^0(x;v) < 0\ \forall\ v \in C$. Then, for every nonempty closed convex cone $K \subseteq \operatorname{ri} C \cup \{0\}$, there is a neighborhood U of 0 in $\mathbb{R}^n$ with the property*

$$f(\overline{x}) < f(x) \qquad \forall \overline{x} \in \{x\} + (U \cap K \setminus \{0\}). \tag{29}$$

Proof. Without loss of generality, we assume that $x = 0$.

Let $\boldsymbol{\psi}$ be the smallest subspace of $\mathbb{R}^n$ containing K. Clearly, if the dimension of $\boldsymbol{\psi}$ equals 0, then (29) holds trivially. From now on, we suppose that the dimension of $\boldsymbol{\psi} \geq 1$.

Suppose that there does not exist such a neighborhood U. Then there is a sequence $\{x_i\}_{i=1}^{\infty}$ in $K \setminus \{0\}$ converging to 0 with $f(x_i) \geq f(0)\ \forall\ i = 1, 2, \ldots$. Clearly, if the dimension of $\boldsymbol{\psi}$ equals 1, then K is a half line. (K is contained in $C \cup \{0\}$ and $C \cup \{0\}$ does not contain subspaces of $\mathbb{R}^n$ since $f^0(0;\cdot)$ is subadditive.) Then, for every nonzero $v \in K$, $f^0(0;v) \geq 0$, which contradicts the fact that $v \in C$. From now on we assume that the dimension of $\boldsymbol{\psi} \geq 2$. Furthermore, all manipulations below have to be considered as being executed in the space $\boldsymbol{\psi}$. ($\operatorname{ri} K$ then actually coincides with the interior of K.)

Based on compactness arguments, the sequence $\{\frac{x_i}{\|x_i\|}\}_{i=1}^{\infty}$ has a limit point $e \in K$ with $\|e\| = 1$. Since the dimension of $\boldsymbol{\psi} \geq 2$, there is an $e' \in \operatorname{ri} K$ with $\|e'\| = 1$ and $e - e' \neq 0 \neq e + e'$ [8, Theorem 6.2]. The nonzero vector $e - e'$ defines a unique

hyper plane $\boldsymbol{\pi}$ in $\boldsymbol{\psi}$ through the origin. We judiciously choose a $v \in C$ such that the projection of e onto $\boldsymbol{\pi}$ along the line spanned by v is $\in \operatorname{ri} K$ and such that v and e are on the same side of $\boldsymbol{\pi}$. In order to choose this v, let S^1 be the unit circle in the two-dimensional plane spanned by e and e'. The smallest closed circle segment joining e and e' is contained in the closed set $K \cap S^1$ which is contained in the open (relatively with respect to S^1) set $(\operatorname{ri} C) \cap S^1$. From this it follows that the intersection of $C \cap S^1$ and the smallest open circle segment joining e and $-\frac{e+e'}{\|e+e'\|}$ is not empty. Choose a v from this nonempty intersection. The projection of e onto $\boldsymbol{\pi}$ along the line spanned by v is $\lambda(e + e')$ for some $\lambda \in (0, \infty)$ and hence is contained in $\operatorname{ri} K$ [8, p. 50 and Theorem 6.1]. Hence, the inverse of $K \cap \boldsymbol{\pi}$ under the mentioned projection is a closed cone with e in its interior. From this, we see that there exists a cone $L \subseteq K$ which is a neighborhood of e relatively with respect to K and that projects into $K \cap \boldsymbol{\pi}$. Choose this cone L small enough such that $v \notin L$ and $L \setminus \{0\}$ is contained in the open half space associated with $\boldsymbol{\pi}$ containing $e - e'$. This cone L contains a subsequence of $\{x_i\}_{i=1}^{\infty}$ which we, after relabeling, again denote by $\{x_i\}_{i=1}^{\infty}$. The projection of these x_i results in a sequence $\{z_i\}_{i=1}^{\infty}$ in $K \cap \boldsymbol{\pi} \setminus \{0\}$ converging to 0. For all but a finite number of elements of this sequence, $f(z_i)$ has to be $\geq f(0)$. Indeed, suppose that $f(z_i) < f(0)$ for infinitely many elements of the sequence. Then $f(x_i) - f(z_i) > 0$ for infinitely many indices i and thus, by the judicious choice of v and L, $f^0(0; v) \geq 0$. This contradicts the fact that $v \in C$. Hence, there is a subsequence of $\{z_i\}_{i=1}^{\infty}$ which we again denote, after relabeling, by $\{z_i\}_{i=1}^{\infty}$ such that $f(z_i) \geq f(0)\ \forall\ i$.

The situation we have arrived at is the following: there is a sequence $\{z_i\}_{i=1}^{\infty}$ in $K \cap \boldsymbol{\pi} \setminus \{0\}$ converging to 0 with $f(z_i) \geq f(0)\ \forall\ i = 1, 2, \ldots$. This situation is similar to above: $K \cap \boldsymbol{\pi}$ plays the role of K, $\{z_i\}_{i=1}^{\infty}$ plays the role of $\{x_i\}_{i=1}^{\infty}$. But the dimension of the smallest subspace of $\mathbb{R}^n$ containing $K \cap \boldsymbol{\pi}$ is one smaller than the dimension of the smallest subspace of $\mathbb{R}^n$ containing K. Hence, repeating this argument, we eventually arrive at the case of dimension 1, which was dealt with in the beginning of this proof. Hence the lemma is proved by contradiction. □

LEMMA A.6. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x \in \mathbb{R}^n$.*

(i) *If $0 \notin \mathbf{SG} f(x) \neq \emptyset$, then at least one of the following two cases holds.*

(1) *Good case. $f^0(x; v) < 0\ \forall\ v \in \operatorname{ri}(\mathbf{SG} f(x)^{\perp})$.*

(2) *Bad case. There is a nonzero $\zeta \in$ closed convex cone generated by $\mathbf{SG} f(x)$ and a nonzero $\xi \in \mathbf{SG}^{\infty} f(x)$ such that* (a) $\zeta + \xi = 0$ *and* (b) $-\zeta \notin$ *closed convex cone generated by $\mathbf{SG} f(x)$ or $-\xi \notin \mathbf{SG}^{\infty} f(x)$.*

(ii) *If $\mathbf{SG} f(x) = \emptyset$, then $f^0(x; v) = -\infty < 0\ \forall\ v \in \operatorname{ri}(\mathbf{SG}^{\infty} f(x)^{\perp})$.*

(iii) *If $0 \in \mathbf{SG} f(x)$, then $f^0(x; v) \geq 0\ \forall\ v \in \mathbb{R}^n$.*

Proof. We first prove (i). First of all, $v \in \mathbf{SG} f(x)^{\perp}$ iff $\zeta(v) \leq 0\ \forall \zeta \in \mathbf{SG} f(x)$ iff $\operatorname{cl} f^0(x; \cdot)(v) \leq 0$ (Proposition 4.3).

We show that for each $v \in \operatorname{ri}(\mathbf{SG} f(x)^{\perp})$, $\operatorname{cl} f^0(x; \cdot)(v)$ is actually < 0. First, since $0 \notin \mathbf{SG} f(x)$, there is a $\overline{v} \in \mathbb{R}^n$ for which $f^0(x; \overline{v}) < 0$ and thus $\operatorname{cl} f^0(x; \cdot)(\overline{v}) < 0$. Clearly, $\overline{v} \in \mathbf{SG} f(x)^{\perp}$. Second, take an arbitrary $v \in \operatorname{ri}(\mathbf{SG} f(x)^{\perp})$. From [8, Theorem 6.4] there is a $\mu \in (1, \infty)$ such that $w = (1 - \mu)\overline{v} + \mu v \in \mathbf{SG} f(x)^{\perp}$. With this μ and w, we have $v = (1 - \frac{1}{\mu})\overline{v} + \frac{1}{\mu} w$ with $\frac{1}{\mu} \in (0, 1)$. By convexity of $\operatorname{cl} f^0(x; \cdot)$, we get $\operatorname{cl} f^0(x; \cdot)(v) \leq (1 - \frac{1}{\mu}) \operatorname{cl} f^0(x; \cdot)(\overline{v}) + \frac{1}{\mu} \operatorname{cl} f^0(x; \cdot)(w) < 0$.

We now search for sufficient conditions for $\operatorname{cl} f^0(x; \cdot)$ to coincide with $f^0(x; \cdot)$ on $\operatorname{ri}(\mathbf{SG} f(x)^{\perp})$. Notice that, since $\mathbf{SG} f(x) \neq \emptyset$, $f^0(x; \cdot)$ is proper and thus it follows that $\operatorname{cl} f^0(x; \cdot)(v) = f^0(x; v) = \infty\ \forall\ v \in \mathbb{R}^n \setminus \operatorname{cl} \operatorname{dom} f^0(x; \cdot)$. Hence $\operatorname{ri}(\mathbf{SG} f(x)^{\perp}) \subseteq \operatorname{cl} \operatorname{dom} f^0(x; \cdot)$. Based on [8, Corollary 6.5.2] at least one of the following two inclusions holds: $\operatorname{ri}(\mathbf{SG} f(x)^{\perp}) \subseteq \operatorname{ri} \operatorname{dom} f^0(x; \cdot)$ or $\operatorname{ri}(\mathbf{SG} f(x)^{\perp}) \subseteq \operatorname{rbdy} \operatorname{dom} f^0(x; \cdot)$.

If $\mathrm{ri}\,(\mathbf{SG}f(x)^\perp) \subseteq \mathrm{ri}\,\mathrm{dom}\, f^0(x;\cdot)$, then, based on Proposition 4.3, $f^0(x;v) = \mathrm{cl}\, f^0(x;\cdot)(v) < 0\ \forall\ v \in \mathrm{ri}\,(\mathbf{SG}f(x)^\perp)$. If $\mathrm{ri}\,(\mathbf{SG}f(x)^\perp) \subseteq \mathrm{rbdy}\,\mathrm{dom}\, f^0(x;\cdot)$, then $\mathrm{ri}\,(\mathbf{SG}f(x)^\perp) \cap \mathrm{ri}\,\mathrm{dom}\, f^0(x;\cdot) = \emptyset$. As in the proof of Lemma A.1, we see that this is equivalent with the existence of a nonzero $\zeta \in \mathbf{SG}f(x)^{\perp\perp}$ and a nonzero $\xi \in \mathrm{dom}\, f^0(x;\cdot)^\perp$ such that (a) $\zeta + \xi = 0$ and (b) $-\zeta \notin \mathbf{SG}f(x)^{\perp\perp}$ or $-\xi \notin \mathrm{dom}\, f^0(x;\cdot)^\perp$. The following two observations complete the proof of (i): $\mathbf{SG}f(x)^{\perp\perp}$ is the closed convex cone generated by $\mathbf{SG}f(x)$, and $\mathrm{dom}\, f^0(x;\cdot)^\perp = \mathbf{SG}^\infty f(x)$.

We prove (ii). First, since $\mathbf{SG}f(x) = \emptyset$, $\mathrm{cl}\, f^0(x;\cdot)(v) = -\infty\ \forall\ v \in \mathbb{R}^n$. Second, $f^0(x;\cdot)$ and $\mathrm{cl}\, f^0(x;\cdot)$ coincide on $\mathrm{ri}\,\mathrm{dom}\, f^0(x;\cdot)$. Third, $\mathrm{ri}\,\mathrm{dom}\, f^0(x;\cdot) = \mathrm{ri}\,(\mathrm{cl}\,\mathrm{dom}\, f^0(x;\cdot)) = \mathrm{ri}\,(\mathrm{dom}\, f^0(x;\cdot)^{\perp\perp}) = \mathrm{ri}\,(\mathbf{SG}^\infty f(x)^\perp)$. This proves (ii).

Finally, (iii) follows immediately from the definition of $\mathbf{SG}f(x)$. □

Remark 5. The set $\mathrm{ri}\,(\mathbf{SG}f(x)^\perp)$ in Lemma A.6(i) is a nonempty convex cone. The set $\mathrm{ri}\,(\mathbf{SG}^\infty f(x)^\perp)$ in Lemma A.6(ii) is a nonempty convex cone.

LEMMA A.7. *Assume an arbitrary function $f : \mathbb{R}^n \to \mathbb{R}$ and a point $x^* \in \mathbb{R}^n$. Let S be the set $\{x \in \mathbb{R}^n : f(x) < f(x^*)\} \cup \{x^*\}$. Let C be a nonempty convex cone in $\mathbb{R}^n$ such that $f^0(x^*;v) < 0\ \forall\ v \in C$. Then, $\mathrm{cl}\, C$ is a generalized approximating cone for S at x^*.*

Proof. C and hence $\mathrm{ri}\, C$ [8] is a nonempty convex cone. Let $\{v_i\}_{i=1}^\infty$ be a sequence in $\mathrm{ri}\, C$ which is dense in $\mathrm{ri}\, C$. This sequence is also dense in $\mathrm{cl}\,\mathrm{ri}\, C = \mathrm{cl}\, C$. Let L_i be the nonempty closed convex cone $\{v \in \mathbb{R}^n : v = \lambda v_i \text{ for some } \lambda \in [0, \infty)\}$ $(i = 1, 2, \dots)$. Let $K_i = L_1 + \cdots + L_i$ $(i = 1, 2, \dots)$. For each i, K_i is a nonempty closed convex cone contained in $\mathrm{ri}\, C \cup \{0\}$. (Closedness follows from [8, Corollary 9.1.2].) Lemma A.5 ensures, for each i, the existence of a neighborhood U_i of 0 in $\mathbb{R}^n$ such that the image of the map $U_i \cap K_i \to \mathbb{R}^n : v \mapsto x^* + v$ is contained in S. Hence, by Lemma 8.1, $\mathrm{cl}\,(\cup_{i=1}^\infty K_i)$ is a generalized approximating cone for S at x^*. We prove that $\mathrm{cl}\, C = \mathrm{cl}\,(\cup_{i=1}^\infty K_i)$. First, since $K_i \subseteq \mathrm{ri}\, C \cup \{0\}\ \forall\ i$, $\cup_{i=1}^\infty K_i \subseteq \mathrm{ri}\, C \cup \{0\}$. Hence $\mathrm{cl}\,(\cup_{i=1}^\infty K_i) \subseteq \mathrm{cl}\,(\mathrm{ri}\, C \cup \{0\}) = \mathrm{cl}\,\mathrm{ri}\, C = \mathrm{cl}\, C$. Second, take an arbitrary $v \in \mathrm{ri}\, C$. Let $\boldsymbol{\pi}$ be the smallest subspace of $\mathbb{R}^n$ containing $\mathrm{ri}\, C$. Let $n' = \dim \boldsymbol{\pi}$. The following construction has to be considered as being executed in $\boldsymbol{\pi}$. Construct an n' dimensional cube, centered at v, small enough such that it is $\subseteq \mathrm{ri}\, C$. Each of the $2^{n'}$ orthants of this cube contains an element of the sequence $\{v_i\}_{i=1}^\infty$. Let $j_1, \dots, j_{2^{n'}}$ be the indices of these elements. Then, $v \in K_{\max\{j_1,\dots,j_{2^{n'}}\}}$. Hence $\mathrm{ri}\, C \subseteq \cup_{i=1}^\infty K_i$ and, taking closures, $\mathrm{cl}\, C = \mathrm{cl}\,\mathrm{ri}\, C \subseteq \mathrm{cl}\,(\cup_{i=1}^\infty K_i)$. The lemma is proved. □

REFERENCES

[1] V. D. BATUKHTIN, *On solving discontinuous extremal problems*, J. Optim. Theory Appl., 77 (1993), pp. 575–589.

[2] F. H. CLARKE, *Necessary Conditions for Nonsmooth Problems in Optimal Control and the Calculus of Variations*, Ph.D. thesis, University of Washington, Seattle, 1973.

[3] F. H. CLARKE, *Methods of Dynamic and Nonsmooth Optimization*, CBMS-NSF Regional Conf. Ser. Appl. Math. 57, SIAM, Philadelphia, 1989.

[4] F. H. CLARKE, *Optimization and Nonsmooth Analysis*, 2nd ed., Classics in Appl. Math. 5, SIAM, Philadelphia, 1990.

[5] P. D. LOEWEN, *Optimal Control via Nonsmooth Analysis*, CRM Proc. Lecture Notes 2, AMS, Providence, RI, 1993.

[6] L. MOREAU AND D. AEYELS, *Optimization of discontinuous functions and application to optimal control*, in Proceedings of the 13th International Symposium on Mathematical Theory of Networks and Systems, A. Beghi, L. Finesso, and G. Picci, eds., Il Poligrafo, Padova, Italy, 1998, pp. 321-324.

[7] R. T. ROCKAFELLAR, *Convex Functions and Dual Extremum Problems*, Ph.D. thesis, Harvard University, Cambridge, MA, 1963.

[8] R. T. ROCKAFELLAR, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[9] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer, New York, 1998.
[10] H. J. Sussmann, *A strong version of the maximum principle under weak hypotheses*, in Proceedings of the 33rd Conference on Decision and Control, Orlando, FL, IEEE, Piscataway, NJ, 1994, pp. 1950–1956.
[11] J. Warga, *Optimization and controllability without differentiability assumptions*, SIAM J. Control Optim., 21 (1983), pp. 837–855.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 70–86

ON THE RATE OF CONVERGENCE OF OPTIMAL SOLUTIONS OF MONTE CARLO APPROXIMATIONS OF STOCHASTIC PROGRAMS*

ALEXANDER SHAPIRO† AND TITO HOMEM-DE-MELLO‡

Abstract. In this paper we discuss Monte Carlo simulation based approximations of a stochastic programming problem. We show that if the corresponding random functions are convex piecewise linear and the distribution is discrete, then an optimal solution of the approximating problem provides an *exact* optimal solution of the true problem with probability one for sufficiently large sample size. Moreover, by using the theory of large deviations, we show that the probability of such an event approaches one exponentially fast with increase of the sample size. In particular, this happens in the case of linear two- (or multi-) stage stochastic programming with recourse if the corresponding distributions are discrete. The obtained results suggest that, in such cases, Monte Carlo simulation based methods could be very efficient. We present some numerical examples to illustrate the ideas involved.

Key words. two-stage stochastic programming with recourse, Monte Carlo simulation, large deviations theory, convex analysis

AMS subject classifications. 90C15, 90C25

PII. S1052623498349541

1. Introduction.

We discuss in this paper Monte Carlo approximations of stochastic programming problems of the form

$$\operatorname*{Min}_{x\in\Theta}\left\{f(x) := \mathbb{E}_P h(x,\omega)\right\}, \tag{1.1}$$

where P is a probability measure on a sample space $(\Omega, \mathcal{F})$, Θ is a subset of $\mathbb{R}^m$, and $h : \mathbb{R}^m \times \Omega \to \mathbb{R}$ is a real valued function. We refer to the above problem as the "true" optimization problem. By generating an independent identically distributed (i.i.d.) random sample $\omega^1, \ldots, \omega^N$ in $(\Omega, \mathcal{F})$, according to the distribution P, one can construct the corresponding approximating program

$$\operatorname*{Min}_{x\in\Theta}\left\{\hat{f}_N(x) := N^{-1}\sum_{j=1}^{N} h(x,\omega^j)\right\}. \tag{1.2}$$

An optimal solution $\hat{x}_N$ of (1.2) provides an approximation (an estimator) of an optimal solution of the true problem (1.1).

There are numerous publications where various aspects of convergence properties of $\hat{x}_N$ are discussed. Suppose that the true problem has a nonempty set A of optimal solutions. It is possible to show that, under mild regularity conditions, the distance $\operatorname{dist}(\hat{x}_N, A)$, from $\hat{x}_N$ to the set A, converges with probability one (w.p.1) to zero as $N \to \infty$. There is a vast literature in statistics dealing with such consistency

*Received by the editors December 21, 1998; accepted for publication (in revised form) February 11, 2000; published electronically July 25, 2000.
http://www.siam.org/journals/siopt/11-1/34954.html

†School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 (ashapiro@isye.gatech.edu). This work was supported, in part, by grant DMI-9713878 from the National Science Foundation.

‡Department of Industrial, Welding and Systems Engineering, The Ohio State University, Columbus, OH 43210-1271 (homem-de-mello.1@osu.edu).

properties of empirical estimators. In the context of stochastic programming we can mention recent works [9], [14], [17], in which this problem is approached from the point of view of the epiconvergence theory.

It is also possible to give various estimates of the rate of convergence of $\hat{x}_N$ to A. Central limit theorem-type results give such estimates of order $O_p(N^{-1/2})$ for the distance $\mathrm{dist}(\hat{x}_N, A)$ (e.g., [15], [20]), and the large deviations theory shows that one may expect that, for any given $\varepsilon > 0$, the probability of the event $\mathrm{dist}(\hat{x}_N, A) \geq \varepsilon$ approaches zero exponentially fast as $N \to \infty$ (see, e.g., [13], [16], [19]). These are general results and it seems that they describe the situation quite accurately in case the involved distributions are continuous. However, it appears that the asymptotics are completely different if the distributions are *discrete*. We show that in such cases, under rather natural assumptions, the approximating problem (1.2) provides an *exact* optimal solution of the true problem (1.1) for N large enough. That is, $\hat{x}_N \in A$ w.p.1 for sufficiently large N. Even more surprisingly we show that the probability of the event $\{\hat{x}_N \notin A\}$ tends to zero exponentially fast as $N \to \infty$. That is what happens in the case of two-stage stochastic programming with recourse if the corresponding distributions are discrete. This indicates that, in such cases, Monte Carlo simulation based methods could be very efficient.

In order to motivate the discussion, let us consider the following simple example. Let $Y_1, \ldots, Y_m$ be i.i.d. real valued random variables. Consider the following optimization problem:

$$\operatorname*{Min}_{x \in \mathbb{R}^m} \left\{ f(x) := \mathbb{E}\left(\sum_{i=1}^{m} |Y_i - x_i| \right) \right\}. \tag{1.3}$$

This problem is a particular case of two-stage stochastic programming with simple recourse. Clearly the objective function $f(x)$ can be written in the form $f(x) := \sum_{i=1}^{m} f_i(x_i)$, where $f_i(x_i) := \mathbb{E}\{|Y_i - x_i|\}$. Therefore the above optimization problem is separable. It is well known that a minimizer of $f_i(\cdot)$ is given by the median of the distribution of Y_i. Suppose that the distribution of the random variables Y_i is symmetrical around zero. Then $\bar{x} := (0, \ldots, 0)$ is an optimal solution of (1.3).

Now let $Y^1, \ldots, Y^N$ be an i.i.d. random sample of N realizations of the random vector $Y = (Y_1, \ldots, Y_m)$. Consider the following sample average approximation of (1.3):

$$\operatorname*{Min}_{x \in \mathbb{R}^m} \left\{ \hat{f}_N(x) := N^{-1} \sum_{j=1}^{N} h(x, Y^j) \right\}, \tag{1.4}$$

where $h(x, y) := \sum_{i=1}^{m} |y_i - x_i|$ with $x, y \in \mathbb{R}^m$. An optimal solution of the above approximating problem (1.4) is given by $\hat{x}_N := (\hat{x}_{1N}, \ldots, \hat{x}_{mN})$, where $\hat{x}_{iN}$ is the sample median of $Y_i^1, \ldots, Y_i^N$.

Suppose for the moment that $m = 1$, i.e., we are minimizing $\mathbb{E}\{|Y - x|\}$ over $x \in \mathbb{R}$. We assume that the distribution of Y is symmetrical around zero and hence $\bar{x} = 0$ is an optimal solution of the true problem. Suppose now that the distribution of Y is continuous with density function $g(y)$. Then it is well known (e.g., [6]) that the corresponding sample median $\hat{x}_N$ is asymptotically normal. That is, $N^{1/2}(\hat{x}_N - \bar{x})$ converges in distribution to normal with zero mean and variance $[2g(\bar{x})]^{-2}$. For

example, if Y is uniformly distributed on the interval $[-1, 1]$, then $N^{1/2}(\hat{x}_N - \bar{x}) \Rightarrow N(0, 1)$. This means that for $N = 100$ we may expect $\hat{x}_N$ to be in the (so-called confidence) interval $[-0.2, 0.2]$ with probability of about 95%. Now for $m > 1$ we have that the events $\hat{x}_{iN} \in [-0.2, 0.2]$, $i = 1, \ldots, m$, are independent (this is because we assume that Y_i are independent). Therefore the probability that *each* sample median $\hat{x}_{iN}$ will be inside the interval $[-0.2, 0.2]$ is about 0.95^m. For example, for $m = 50$, this probability becomes $0.95^{50} = 0.077$. If we want that probability to be about 0.95 we have to increase the interval to $[-0.3, 0.3]$, which constitutes 30% of the range of the random variable Y. In other words for that sample size and with $m = 50$ our sample estimate will not be accurate.

The situation becomes quite different if we assume that Y has a discrete distribution. Suppose now that Y can take values -1, 0, and 1 with equal probabilities 1/3. In that case the true problem has unique optimal solution $\bar{x} = 0$. The corresponding sample estimate $\hat{x}_N$ can be equal to -1, 0, or 1. We have that the event $\{\hat{x}_N = 1\}$ happens if more than half of the sample points are equal to one. Probability of that is given by $P(X > N/2)$, where X has a binomial distribution $B(N, 1/3)$. If exactly half of the sample points are equal to one, then the sample estimate can be any number in the interval $[0, 1]$. Similar conclusions hold for the event $\{\hat{x}_N = -1\}$. Therefore the probability that $\hat{x}_N = 0$ is at least $1 - 2P(X \geq N/2)$. For $N = 100$, this probability is 0.9992. Therefore the probability that the sample estimate $\hat{x}_N$, given by an optimal solution of the approximating problem (1.4) with the sample size $N = 100$ and the number of random variables $m = 50$, is at least $0.9992^{50} = 0.96$. With the sample size $N = 120$ and the number of random variables $m = 200$ this probability, of $\hat{x}_N = 0$, is about $0.9998^{200} = 0.95$. Note that the number of scenarios for that problem is 3^{200}, which is not small by any standard. And yet with sample size of only 120 the approximating problem produces an estimator which is exactly equal to the true optimal solution with probability of 95%.

The above problem, although simple, illustrates the phenomenon of exponential convergence referred to in the title of the paper. In the above example the corresponding probabilities can be calculated in a closed form, but in the general case of course we cannot expect to do so. The purpose of this paper is to extend this discussion to a class of stochastic programming problems satisfying some assumptions. Our goal is to exhibit some qualitative (rather than quantitative) results. We do not propose an algorithm, but rather show asymptotic properties of Monte Carlo simulation based methods.

The paper is organized as follows. In section 2 we show almost sure (w.p.1) occurrence of the event $\{\hat{x}_N \in A\}$ (recall that A is the set of optimal solutions of the "true" problem). In section 3 we take a step further and, using techniques from large deviations theory, we show that the probability of that event approaches one exponentially fast. In section 4 we discuss the median problem in more detail, and present some numerical results for a two-stage stochastic programming problem with complete recourse. Finally, section 5 presents some conclusions.

2. Almost sure convergence. Consider the "true" stochastic programming problem (1.1). For the sake of simplicity we assume that the corresponding expected value function $f(x) := \mathbb{E}_P h(x, \omega)$ exists (and in particular is finite valued) for all $x \in \mathbb{R}^m$. For example, if the probability measure P has a finite support (i.e., the distribution P is discrete and can take a finite number of different values), and hence the space Ω can be taken to be finite, say $\Omega := \{\omega_1, \ldots, \omega_K\}$, and P is given by the

probabilities $P\{\omega = \omega_k\} = p_k$, $k = 1, \ldots, K$, we have

$$\mathbb{E}_P h(x, \omega) = \sum_{k=1}^{K} p_k h(x, \omega_k). \tag{2.1}$$

We assume that the feasible set Θ is closed and convex and that for every $\omega \in \Omega$, the function $h(\cdot, \omega)$ is convex. This implies that the expected value function $f(\cdot)$ is also convex, and hence the "true" problem (1.1) is convex. Also if P is discrete and the functions $h(\cdot, \omega_k)$, $k = 1, \ldots, K$, are piecewise linear and convex, then $f(\cdot)$ is piecewise linear and convex. That is what happens in two-stage stochastic programming with a finite number of scenarios.

Let $\omega^1, \ldots, \omega^N$ be an i.i.d. random sample in $(\Omega, \mathcal{F})$, generated according to the distribution P, and consider the corresponding approximating program (1.2). Note that, since the functions $h(\cdot, \omega^j)$ are convex, the approximating (sample average) function $\hat{f}_N(\cdot)$ is also convex, and hence the approximating program (1.2) is convex.

We show in this section that, under some natural assumptions which hold, for instance, in the case of two-stage stochastic programming with a finite number of scenarios, w.p.1 for N large enough any optimal solution of the approximating problem (1.2) belongs to the set of optimal solutions of the true problem (1.1). That is, problem (1.2) yields an *exact* optimal solution (w.p.1) when N is sufficiently large.

The statement "w.p.1 for N large enough" should be understood in the sense that for P-almost every $\omega \in \Omega$ there exists $N^* = N^*(\omega)$, such that for any $N \geq N^*$ the corresponding statement holds. The number N^* is a function of ω, i.e., it depends on the random sample and therefore in itself is random. Note also that, since convergence w.p.1 implies convergence in probability, the above statement implies that the probability of the corresponding event to happen tends to one as the sample size N tends to infinity.

We denote by A the set of optimal solutions of the true problem (1.1) and by $f'(x, d)$ the directional derivative of f at x in the direction d. Note that the set A is convex and closed, and since f is a real valued convex function, the directional derivative $f'(x, d)$ exists, for all x and d, and is convex in d. We discuss initially the case when A is a singleton; later we will consider the general setting.

Assumption (A). The true problem (1.1) possesses unique optimal solution $\bar{x}$, i.e., $A = \{\bar{x}\}$, and there exists a positive constant c such that

$$f(x) \geq f(\bar{x}) + c\|x - \bar{x}\| \quad \forall\, x \in \Theta. \tag{2.2}$$

Of course condition (2.2), in itself, implies that $\bar{x}$ is the unique optimal solution of (1.1). In the approximation theory optimal solutions satisfying (2.2) are called sharp minima. It is not difficult to show, since problem (1.1) is convex, that Assumption (A) holds iff

$$f'(\bar{x}, d) > 0 \quad \forall\, d \in T_\Theta(\bar{x}) \setminus \{0\}, \tag{2.3}$$

where $T_\Theta(\bar{x})$ denotes the tangent cone to Θ at $\bar{x}$. In particular, if $f(x)$ is differentiable at $\bar{x}$, then Assumption (A) (or equivalently (2.3)) holds iff $-\nabla f(\bar{x})$ belongs to the interior of the normal cone to Θ at $\bar{x}$. Note that since $f'(\bar{x}, \cdot)$ is a positively homogeneous convex real valued (and hence continuous) function, it follows from (2.3) that $f'(\bar{x}, d) \geq \varepsilon\|d\|$ for some $\varepsilon > 0$ and $\forall\ d \in T_\Theta(\bar{x})$. We refer to a recent paper [4], and references therein, for a discussion of that condition and some of its generalizations.

If the function $f(x)$ is piecewise linear and the set Θ is polyhedral, then problem (1.1) can be formulated as a linear programming problem, and the above assumption (A) always holds provided $\bar{x}$ is the unique optimal solution of (1.1). This happens, for example, in the case of a two-stage linear stochastic programming problem with a finite number of scenarios provided it has a unique optimal solution. Note that Assumption (A) is not restricted to such situations only. In fact, in some of our numerical experiments sharp minima (i.e., Assumption (A)) happened to hold in the case of continuous (normal) distributions. Furthermore, because the problem is assumed to be convex, sharp minima are equivalent to first order sufficient conditions. Under such conditions, first order (i.e., linear) growth (2.2) of $f(x)$ holds *globally*, i.e., for all $x \in \Theta$.

THEOREM 2.1. *Suppose that* (i) *for every* $\omega \in \Omega$ *the function* $h(\cdot, \omega)$ *is convex,* (ii) *the expected value function* $f(\cdot)$ *is well defined and is finite valued,* (iii) *the set* Θ *is closed and convex,* (iv) *Assumption (A) holds. Then w.p.*1 *for* N *large enough the approximating problem* (1.2) *has a unique optimal solution* $\hat{x}_N$ *and* $\hat{x}_N = \bar{x}$.

Proof of the above theorem is based on the following proposition. Results of that proposition (perhaps not exactly in that form) are basically known, but since its proof is simple we give it for the sake of completeness. Denote by $h'_\omega(x, d)$ the directional derivative of $h(\cdot, \omega)$ at the point x in the direction d and by $\mathcal{H}(B, C)$ the Hausdorff distance between sets $B, C \subset \mathbb{R}^m$, that is,

$$\mathcal{H}(B, C) := \max \left\{ \sup_{x \in C} \operatorname{dist}(x, B), \sup_{x \in B} \operatorname{dist}(x, C) \right\}. \tag{2.4}$$

PROPOSITION 2.2. *Suppose that the assumptions* (i) *and* (ii) *of Theorem* 2.1 *are satisfied. Then, for any* $x, d \in \mathbb{R}^m$, *the following holds:*

$$f'(x, d) = \mathbb{E}_P \left\{ h'_\omega(x, d) \right\}, \tag{2.5}$$

$$\lim_{N \to \infty} \sup_{\|d\| \le 1} \left| f'(x, d) - \hat{f}'_N(x, d) \right| = 0 \quad w.p.1, \tag{2.6}$$

$$\lim_{N \to \infty} \mathcal{H}\left(\partial \hat{f}_N(x), \partial f(x) \right) = 0 \quad w.p.1. \tag{2.7}$$

Proof. Since $f(\cdot)$ is convex we have that

$$f'(x, d) = \inf_{t > 0} \frac{f(x + td) - f(x)}{t}, \tag{2.8}$$

and the ratio in the right-hand side of (2.8) decreases monotonically as t decreases to zero, and similarly for the functions $h(\cdot, \omega)$. It follows then by the monotone convergence theorem that

$$f'(x, d) = \mathbb{E}_P \left\{ \inf_{t > 0} \frac{h(x + td, \omega) - h(x, \omega)}{t} \right\}, \tag{2.9}$$

and hence the right-hand side of (2.5) is well defined and the equation follows.

We have that

$$\hat{f}'_N(x, d) = N^{-1} \sum_{j=1}^{N} h'_{\omega^j}(x, d). \tag{2.10}$$

Therefore by the strong form of the law of large numbers it follows from (2.5) that for any $d \in \mathbb{R}^m$, $\hat{f}'_N(x,d)$ converges to $f'(x,d)$ w.p.1 as $N \to \infty$. Consequently for any countable set $D \subset \mathbb{R}^m$ we have that the event "$\lim_{N\to\infty} \hat{f}'_N(x,d) = f'(x,d)\ \forall\ d \in D$" happens w.p.1. Let us take a countable and dense subset D of $\mathbb{R}^m$. Recall that if a sequence of real valued convex functions converges pointwise on a dense subset of $\mathbb{R}^m$, then it converges uniformly on any compact subset of $\mathbb{R}^m$ (e.g., [18, Theorem 10.8]). Therefore, since the functions $\hat{f}'_N(x,\cdot)$ are convex, it follows from the pointwise convergence of $\hat{f}'_N(x,\cdot)$ on D that the convergence is uniform on the unit ball $\{d : \|d\| \le 1\}$. This proves (2.6).

Recall that if g is a real valued convex function, then $g'(x,\cdot)$ coincides with the support function of its subdifferential $\partial g(x)$. Therefore the Hausdorff distance between the subdifferentials of f and $\hat{f}_N$, at x, is equal to the supremum on the left-hand side of (2.6) (see, e.g., [12, Theorem V.3.3.8]). Consequently (2.7) follows from (2.6). □

Proof of Theorem 2.1. As we discussed earlier, Assumption (A) is equivalent to condition (2.3) which, in turn, implies that $f'(\bar{x},d) \ge \varepsilon$ for some $\varepsilon > 0$ and all $d \in T_\Theta(\bar{x}) \cap S^{m-1}$, where

$$S^{m-1} := \{d \in \mathbb{R}^m : \|d\| = 1\}.$$

By (2.6) it follows that w.p.1 for N large enough

$$\hat{f}'_N(\bar{x},d) > 0 \quad \forall\, d \in T_\Theta(\bar{x}) \cap S^{m-1}. \tag{2.11}$$

Since the approximating problem is convex, this implies that $\bar{x}$ is a sharp (and hence unique) optimal solution of the approximating problem. This completes the proof. □

Let us consider now a situation where the true problem (1.1) may have multiple optimal solutions, i.e., the set A is not necessarily a singleton. In that case Theorem 2.1 can be generalized, under stronger assumptions, as follows.

THEOREM 2.3. *Suppose that* (i) *the set* Ω *is finite,* (ii) *for every* $\omega \in \Omega$ *the function* $h(\cdot,\omega)$ *is piecewise linear and convex,* (iii) *the set* Θ *is closed, convex, and polyhedral,* (iv) *the true problem* (1.1) *has a nonempty bounded set* A *of optimal solutions. Then the set* A *is compact, convex, and polyhedral, and w.p.*1 *for* N *large enough the approximating problem* (1.2) *has a nonempty set* A_N *of optimal solutions and* A_N *is a face of the set* A.

Proof of the above theorem is based on the following lemma which may have an independent interest.

LEMMA 2.4. *Suppose that the assumptions* (i) *and* (ii) *of Theorem* 2.3 *are satisfied. Then the following holds:* (a) *There exists a finite number of points* $z_1,\dots,z_r$ *(independent of the sample) such that for every* $x \in \mathbb{R}^m$, *there is* $k \in \{1,\dots,r\}$ *such that* $\partial f(x) = \partial f(z_k)$ *and* $\partial \hat{f}_N(x) = \partial \hat{f}_N(z_k)$ *for any realization of the random sample.* (b) *W.p.*1 *the subdifferentials* $\partial \hat{f}_N(x)$ *converge to* $\partial f(x)$ *uniformly in* $x \in \mathbb{R}^m$, *i.e.,*

$$\lim_{N\to\infty} \sup_{x\in\mathbb{R}^m} \mathcal{H}\left(\partial \hat{f}_N(x), \partial f(x)\right) = 0 \ \ w.p.1. \tag{2.12}$$

(c) *If, in addition, the assumptions* (iii) *and* (iv) *are satisfied, then there exists a finite number of points* $x_1,\dots,x_q$ *(independent of the sample) such that the points* $x_1,\dots,x_\ell$, $\ell < q$, *form the set of extreme points of* A *and if the following condition*

holds,

$$\hat{f}_N(x_i) < \hat{f}_N(x_j) \text{ for any } i \in \{1,\dots,\ell\} \text{ and } j \in \{\ell+1,\dots,q\}, \tag{2.13}$$

then the set A_N is nonempty and forms a face of the set A.

Proof. It follows from the assumptions (i) and (ii) that the expected value function $f(x)$ is piecewise linear and convex, and hence $f(x)$ can be represented as a maximum of a finite number of affine functions $\ell_i(x)$, $i = 1,\dots,n$. Consequently the space $\mathbb{R}^m$ can be partitioned into a union of convex polyhedral sets $C_1,\dots,C_n$ such that $f(x)$, restricted to C_i, coincides with $\ell_i(x)$, $i = 1,\dots,n$.

Let us make the following observations. Suppose that $f(x)$ is affine on a convex polyhedral set C. Then function $h(\cdot,\omega)$ is also affine on C for every $\omega \in \Omega$. Indeed, suppose for a moment that the set C has a nonempty interior and that for some $\omega \in \Omega$ the corresponding function $h(\cdot,\omega)$ is not affine on C. Since $h(\cdot,\omega)$ is piecewise linear and convex, this can happen only if there is a point $\hat{x}$ in the interior of C such that $\partial h(\hat{x},\omega)$ is not a singleton. By the Moreau–Rockafellar theorem (see [18, Theorem 23.8]) we have that $\partial f(\hat{x}) = \sum_{k=1}^K p_k \partial h(\hat{x},\omega_k)$. Therefore if $\partial h(\hat{x},\omega)$ is not a singleton, then $\partial f(\hat{x})$ is also not a singleton. This, however, cannot happen since $f(x)$ is affine on C. In case the interior of C is empty, we can restrict the problem to the linear space generated by C and proceed as above. Now, since the sample average function $\hat{f}_N(x)$ is a linear combination of the functions $h(\cdot,\omega)$, $\omega \in \Omega$, with nonnegative coefficients, it follows that $\hat{f}_N(x)$ is also affine on C for *any* realization of the random sample.

Our second observation is the following. Let $g(x)$ be a convex function taking a constant value over a convex set S. Then $\partial g(x)$ is constant over the relative interior of S (e.g., [3, Lemma 1.115]). By adding an affine function to $g(x)$, we obtain that the same property holds if $g(x)$ is affine over S.

By the above observations we can take points z_i in the relative interior of each face of the sets $C_1,\dots,C_n$. Note that an extreme point of a set C_i is viewed as its face, of dimension zero, and its relative interior coincides with the considered extreme point. Since each set C_i is polyhedral, it has a finite number of faces, and hence the total number of such points will be finite. This completes the proof of the assertion (a). Assertion (b) follows immediately from Proposition 2.2 and assertion (a).

Let us prove (c). Since the function $f(x)$ is piecewise linear, the set A is a convex polyhedral set, and by assumption (iv), A is compact.

Let us observe that by adding a barrier function of the form $\psi(x) := \alpha \operatorname{dist}(x,\Theta)$ to the objective function $f(x)$, for sufficiently large $\alpha > 0$, we can reduce the true problem to the unconstrained problem

$$\operatorname*{Min}_{x\in\mathbb{R}^m} \mathbb{E}_P h^*(x,\omega), \tag{2.14}$$

where $h^*(x,\omega) := h(x,\omega)+\psi(x)$. It is well known that, for α large enough, the optimal solutions of problems (1.1) and (2.14) coincide (see, e.g., [2, Proposition 5.4.1]). Since Θ is convex, the barrier function, and hence the functions $h^*(\cdot,\omega)$, are also convex. Moreover, since by the assumption (iii) the set Θ is polyhedral, the barrier function is also polyhedral if we take distance with respect to the ℓ_1 norm in $\mathbb{R}^m$. Therefore, without loss of generality, we can assume in the subsequent analysis that $\Theta = \mathbb{R}^m$, i.e., that the problem under consideration is *unconstrained*.

Let S be a sufficiently large convex compact polyhedral set (e.g., a cube) such that the set A is included in the interior of the set S. Such a set exists since A is

bounded. Consider the sets $C_i' := C_i \cap S$, $i = 1, \ldots, n$. These sets are polyhedral and compact. We can assume that all these sets are different from each other and that A coincides with the set C_1'. Now let $\{x_1, \ldots, x_q\}$ be the set of all extreme points (vertices) of the sets $C_1', \ldots, C_n'$ such that, for some $\ell < q$, points $x_1, \ldots, x_\ell$ form the set of extreme points of A. Since each set C_i' is polyhedral, there are a finite number of such points. Suppose that condition (2.13) holds, and let C_k', $k \geq 2$, be a set from the above collection such that the intersection of C_k' with A is nonempty. Since $\hat{f}_N(x)$ is linear on C_k' and C_k' is compact, it follows from condition (2.13) that the minimum of $\hat{f}_N(x)$ over C_k' is attained on a nonempty subset of the set A. Consider a collection of such sets C_k' that their union forms a neighborhood of the set A. Then $\hat{f}_N(x)$ attains its minimum over that union on a nonempty subset A_N^* of A. By convexity of $\hat{f}_N(x)$ it follows then that the set A_N coincides with A_N^* and hence is nonempty and is a subset of A. Finally, since $\hat{f}_N(x)$ is linear on A, it follows that A_N is a face of A. □

We give now two proofs of Theorem 2.3, which give a different insight into the problem.

Proof of Theorem 2.3. As was shown in the proof of the above lemma, by adding a barrier function, we can reduce the problem to an unconstrained one. Therefore without loss of generality, we can assume that $\Theta = \mathbb{R}^m$, i.e., that the problem is unconstrained.

It follows from the assumptions (i) and (ii) that the expected value function $f(x)$ is piecewise linear and convex. Therefore the set A of optimal solutions of the true problem is a convex polyhedral and, by (iv), compact set. By the strong law of large numbers we have that w.p.1 the approximating functions $\hat{f}_N(x)$ converge pointwise to $f(x)$. Moreover, by the same arguments as in the proof of Proposition 2.2 we have that this convergence is uniform on any compact subset of $\mathbb{R}^m$. Let V be a compact neighborhood of the set A. Then w.p.1 for N large enough $\hat{f}_N(x)$ has a minimizer over V which is arbitrarily close to A and hence lies in the interior of V. By convexity this minimizer will be a global minimizer of $\hat{f}_N(x)$. This shows that w.p.1 for N large enough the set A_N of optimal solutions of the approximating problem is nonempty.

Since $f(x)$ is piecewise linear and convex, we have that subdifferentials of $f(x)$ are convex compact polyhedral sets and, by Lemma 2.4, it follows that the total number of the extreme points of all subdifferentials $\partial f(x)$ is finite. Moreover, since for any $x \notin A$ we have that $0 \notin \partial f(x)$, it follows that there exists $\varepsilon > 0$ such that the distance from the null vector $0 \in \mathbb{R}^m$ to $\partial f(x)$ is greater than $\varepsilon \ \forall \ x \notin A$. Together with (2.12) this implies that w.p.1 for N large enough, $0 \notin \partial \hat{f}_N(x) \ \forall \ x \notin A$, and hence any $x \notin A$ cannot be an optimal solution of the approximating problem. This shows that w.p.1 for N large enough the inclusion $A_N \subset A$ holds. Finally let us observe that since $f(x)$, and hence $\hat{f}_N(x)$, are linear on A, and A_N is the set of minimizers of $\hat{f}_N(x)$ over A, it follows that A_N is a face of A.

Let us give now the second proof. Let $\{x_1, \ldots, x_q\}$ be the set of points constructed in assertion (c) of Lemma 2.4. Since this set is finite and A is the set of minimizers of $f(x)$, we have that there exists $\varepsilon > 0$ such that $f(x_i) + \varepsilon < f(x_j)$ for any $i \in \{1, \ldots, \ell\}$ and $j \in \{\ell + 1, \ldots, q\}$. By the law of large numbers we have that $\hat{f}_N(x_i)$ converges to $f(x_i)$, w.p.1 as $N \to \infty$, for every $i \in \{x_1, \ldots, x_q\}$. Therefore w.p.1 for N large enough we have that $\hat{f}_N(x_i) < f(x_i) + \varepsilon/2$ for $i \in \{1, \ldots, \ell\}$, and $\hat{f}_N(x_j) > f(x_j) - \varepsilon/2$ for $j \in \{\ell + 1, \ldots, q\}$, and hence condition (2.13) follows. Together with assertion (c) of Lemma 2.4 this proves that A_N is nonempty and forms a face of A. □

Under the assumptions of the above theorem, the set A_N of optimal solutions of the approximating problem is convex and polyhedral. The above theorem shows that w.p.1 for N large enough, every optimal solution of the approximating problem is an optimal solution of the true problem and every vertex of the set of optimal solutions of the approximating problem is a vertex of the set of optimal solutions of the true problem.

In order to see what may happen consider the following example. Let $h(x,\omega) := |x_1 - \omega|$, where $x = (x_1, x_2) \in \mathbb{R}^2$ and $\omega \in \Omega$ with $\Omega := \{-2, -1, 1, 2\} \subset \mathbb{R}$. Suppose that the probability of ω being equal to any of the points of Ω is 0.25 and let $\Theta := \{x \in \mathbb{R}^2 : |x_2| \leq 1\}$. Then the set A of optimal solutions of the corresponding true problem is $A = \{x : |x_1| \leq 1,\ |x_2| \leq 1\}$. On the other hand, for large N, the set of optimal solutions of the approximating problem is given either by the face $\{x : x_1 = -1,\ |x_2| \leq 1\}$ or the face $\{x : x_1 = 1,\ |x_2| \leq 1\}$ of the set A.

3. Exponential rate of convergence. In the previous section we showed that, under appropriate assumptions, the approximating problem (1.2) yields an exact optimal solution of the true problem (1.1) w.p.1 for N large enough. Since convergence w.p.1 implies convergence in probability, it follows that the probability of this event tends to one as N tends to infinity. That result, however, does not say how large the sample size N should be in order for the approximating problem to provide such an exact solution.

Similarly to the example presented in the introduction, it turns out that, in the case under consideration (i.e., when Ω is finite and $h(\cdot,\omega)$ are piecewise linear), the convergence of the corresponding probability to one is *exponentially fast*. A consequence of this somewhat surprising fact is that one does not need a very large sample to find the optimal solution of (1.1), which shows that Monte Carlo approximations techniques can be an effective approach to solve such problems.

In this section we formalize and prove this result. We begin by considering again the case where the true problem (1.1) has a unique optimal solution $\bar{x}$. Suppose that Assumption (A) holds. Recall that S^{m-1} denotes the sphere in $\mathbb{R}^m$, and consider the Banach space $Z := C(S^{m-1})$ of real valued continuous functions defined on S^{m-1} and equipped with the sup-norm. By restricting a positively homogeneous function to S^{m-1}, we can identify Z with the space of continuous positively homogeneous functions on $\mathbb{R}^m$. Denote by Z^* the dual space of Z, i.e., the space of continuous linear functionals defined on Z.

Let $\mathcal{B}$ be the σ-algebra of Borel sets in Z. Consider the function

$$\eta(d,\omega) := h'_\omega(\bar{x}, d), \quad d \in \mathbb{R}^m,\ \omega \in \Omega. \tag{3.1}$$

The function $\eta(\cdot,\omega)$ is convex, and hence continuous, and is positively homogeneous. Therefore it can be considered as an element of Z. Moreover, the mapping $\omega \mapsto \eta(\cdot,\omega)$, from $(\Omega, \mathcal{F})$ into $(Z, \mathcal{B})$, is measurable and hence $\eta(\cdot,\omega)$ can be considered as a random element of $(Z, \mathcal{B})$. Let $\mathbb{P}$ be the probability measure on Z induced by the measure P. Note that $\mathbb{E}_P \eta(d,\omega) = f'(\bar{x}, d)$ and that the measure $\mathbb{P}$ is concentrated on the subset of Z formed by convex positively homogeneous functions.

Assumption (B). There exists a constant $\kappa > 0$ such that

$$\|\eta(\cdot,\omega)\|_Z \leq \kappa, \quad \text{for } P\text{-almost every } \omega.$$

This assumption clearly holds if the set Ω is finite. Note that

$$\|\eta(\cdot,\omega)\|_Z = \sup_{d \in S^{m-1}} |h'_\omega(\bar{x}, d)|.$$

Therefore Assumption (B) means that the subdifferentials $\partial h(\bar{x}, \omega)$ are uniformly bounded for P-almost every ω. Notice that this is what happens in two-stage stochastic programming problems with complete recourse if only the right-hand side is random, since in that case the dual feasibility set does not depend on ω. Complete recourse implies that the dual feasibility set is also bounded. Therefore, in such cases the subdifferentials $\partial h(\bar{x}, \omega)$ are uniformly bounded for all ω.

Let us recall now a few facts about random variables on Banach spaces. Let $\eta_1, \eta_2, \ldots$ be an i.i.d. sequence of random elements of $(Z, \mathcal{B})$, with the common distribution $\mathbb{P}$, and define $\zeta_N := N^{-1} \sum_{j=1}^{N} \eta_j$. Note that Assumption (B) implies that $\int_Z \|z\|_Z \mathbb{P}(dz) < \infty$. Then, by the strong law of large numbers (for Banach spaces) we have that $\zeta_N \to \zeta := \mathbb{E}[\eta]$ w.p.1, where the convergence is in the norm of Z and the expectation operator corresponds to the so-called Bochner integral (see, e.g., Hiai [10]).

Let

$$M(z^*) := \int e^{z^*(z)} \mathbb{P}(dz), \quad z^* \in Z^*,$$

be the moment generating function of $\mathbb{P}$ (i.e., of $\eta(\cdot, \omega)$). A version of Cramér's theorem for Banach spaces (see, e.g., Deuschel and Stroock [8]) can be stated as follows. If for any $\alpha \in [0, \infty)$ we have

$$\int_Z e^{\alpha \|z\|} \mathbb{P}(dz) \; < \; \infty, \tag{3.2}$$

then a large deviations principle (LDP) holds for $\{\zeta_N\}$, i.e., for any $\mathcal{B}$-measurable set $\Gamma \subset Z$ we have that

$$\begin{array}{ll} -\inf_{z \in \operatorname{int}(\Gamma)} I(z) & \leq \liminf_{N \to \infty} N^{-1} \log[P(\zeta_N \in \Gamma)] \\ & \leq \limsup_{N \to \infty} N^{-1} \log[P(\zeta_N \in \Gamma)] \leq -\inf_{z \in \operatorname{cl}(\Gamma)} I(z). \end{array} \tag{3.3}$$

Here $\operatorname{int}(\Gamma)$ and $\operatorname{cl}(\Gamma)$ denote the interior and the topological closure, respectively, of the set $\Gamma \subset Z$, and $I(z)$ is the large deviations rate function, which is given by

$$I(z) := \sup_{z^* \in Z^*} \{z^*(z) - \log M(z^*)\}. \tag{3.4}$$

Notice that (3.2) follows immediately from Assumption (B).

For any $d \in S^{m-1}$ we can define a functional $z_d^* \in Z^*$ as $z_d^*(z) := z(d)$. Let $M_d(t) := M(t z_d^*)$. Note that we can also write

$$M_d(t) = \mathbb{E}_P \left\{ e^{t \eta(d, \omega)} \right\},$$

so we recognize $M_d(t)$ as the moment generating function of the (one-dimensional) random variable $X := \eta(d, \omega)$. Note also that Assumption (B) implies that $M_d(t) < \infty$ $\forall \; t \in \mathbb{R}$. Consider the rate function of $\eta(d, \omega)$, that is,

$$I_d(\alpha) := \sup_{t \in \mathbb{R}} \left[t\alpha - \log M_d(t) \right]. \tag{3.5}$$

By taking z^* in the right-hand side of (3.4) of the form $z^* := t z_d^*$, we obtain that, for any $z \in Z$,

$$I(z) \; \geq \; \sup_{d \in S^{m-1}} \sup_{t \in \mathbb{R}} \left[t z(d) - \log M_d(t) \right] \; = \; \sup_{d \in S^{m-1}} I_d(z(d)). \tag{3.6}$$

Let A_N be the set of optimal solutions of the approximating problem (1.2), and consider the event

$$\mathcal{E}_N := \left\{ \text{ the set } A_N \text{ is nonempty and } A_N = \{\bar{x}\} \right\}. \tag{3.7}$$

The above event $\mathcal{E}_N$ means that the approximating problem possesses a unique optimal solution $\hat{x}_N$ and that $\hat{x}_N = \bar{x}$. Denote by $\mathcal{E}_N^c$ the complement of the event $\mathcal{E}_N$. Note that the probability $P(\mathcal{E}_N)$, of the event $\mathcal{E}_N$, is equal to $1 - P(\mathcal{E}_N^c)$. The following theorem shows that the probability of the event $\mathcal{E}_N^c$ approaches zero exponentially fast.

THEOREM 3.1. *Suppose that the assumptions of Theorem* 2.1 *are satisfied and that Assumption* (B) *holds. Then there exists a constant $\beta > 0$ such that*

$$\limsup_{N\to\infty} \frac{1}{N} \log[P(\mathcal{E}_N^c)] \;\leq\; -\beta. \tag{3.8}$$

Proof. Consider $\zeta_N(\cdot) := N^{-1} \sum_{j=1}^N \eta(\cdot, \omega^j) = \hat{f}'_N(\bar{x}, \cdot)$ and the set

$$F := \left\{ z \in Z : \inf_{d \in T_\Theta(\bar{x}) \cap S^{m-1}} z(d) \leq 0 \right\}. \tag{3.9}$$

Since the topology on Z is that of uniform convergence, it follows that the min-function

$$\phi(z) := \inf_{d \in T_\Theta(\bar{x}) \cap S^{m-1}} z(d)$$

is continuous on the space Z, and hence the set F is closed in Z. By the definition of the set F, we have that if $\zeta_N \notin F$, then $\zeta_N(d) > 0 \;\forall\; d \in T_\Theta(\bar{x}) \cap S^{m-1}$. Consequently, in that case, $\hat{x}_N = \bar{x}$ is the unique optimal solution of the approximating problem. Therefore we have that

$$P(\mathcal{E}_N^c) \leq P(\zeta_N \in F).$$

It follows then by the last inequality of (3.3) that we need only to show that the constant

$$\beta := \inf_{z \in F} I(z) \tag{3.10}$$

is positive.

Consider a fixed direction $d \in T_\Theta(\bar{x}) \cap S^{m-1}$, and let X denote the corresponding random variable $\eta(d, \omega)$. Let $\Lambda(t) := \log M_d(t) = \log \mathbb{E}[e^{tX}]$ be the logarithmic moment generating function of X. By the dominated convergence theorem we have that $M_d(t)$ is differentiable $\forall\; t \in \mathbb{R}$ and $M'_d(t) = \mathbb{E}[Xe^{tX}]$. It follows that $\Lambda'(t) = \mathbb{E}[Xe^{tX}]/\mathbb{E}[e^{tX}]$ and hence, since $|X| \leq \kappa$ by Assumption (B),

$$|\Lambda'(t)| \;\leq\; \frac{\mathbb{E}[|X|e^{tX}]}{\mathbb{E}[e^{tX}]} \leq \kappa \quad \forall\; t \in \mathbb{R}.$$

Similarly, we have

$$|\Lambda''(t)| \;= \left| \frac{\mathbb{E}[X^2 e^{tX}]}{\mathbb{E}[e^{tX}]} - (\Lambda'(t))^2 \right| \leq |\kappa^2 - (\Lambda'(t))^2| \leq\; \kappa^2 \quad \forall\; t \in \mathbb{R}. \tag{3.11}$$

By the mean value theorem, (3.11) implies that $\forall\ t, s \in \mathbb{R}$,

$$|\Lambda'(t) - \Lambda'(s)| \ \leq\ \kappa^2 |t - s|. \tag{3.12}$$

Since the function $\Lambda(\cdot)$ is convex, it follows from a result in convex analysis (e.g., [12, Theorem X.4.2.2]) that the conjugate function $I_d = \Lambda^*$ is *strongly convex* modulus $1/\kappa^2$, that is,

$$I_d(\alpha_2) \ \geq\ I_d(\alpha_1) + I'_d(\alpha_1)(\alpha_2 - \alpha_1) + \frac{1}{2\kappa^2}|\alpha_2 - \alpha_1|^2$$

$\forall\ \alpha_1, \alpha_2 \in \mathbb{R}$. Since at $\bar{\alpha}_d := \mathbb{E}[X] = f'(\bar{x}, d)$ we have that $I_d(\bar{\alpha}_d) = I'_d(\bar{\alpha}_d) = 0$, it follows that

$$I_d(\alpha) \ \geq\ \frac{1}{2\kappa^2}|\alpha - \bar{\alpha}_d|^2 \quad \forall\, \alpha \in \mathbb{R}. \tag{3.13}$$

By Assumption (A) we have that $f'(\bar{x}, d) \geq c \forall\ d \in T_\Theta(\bar{x}) \cap S^{m-1}$, and hence we obtain that

$$I_d(0) \ \geq\ \frac{c^2}{2\kappa^2} \quad \forall\, d \in T_\Theta(\bar{x}) \cap S^{m-1}. \tag{3.14}$$

By the definition of the set F we have that if $z \in F$, then there exists $d \in T_\Theta(\bar{x}) \cap S^{m-1}$ such that $z(d) \leq 0$. It follows then by (3.6) and (3.14) that $I(z) \geq c^2/(2\kappa^2)$ for any $z \in F$. Consequently we obtain

$$\beta \geq\ \frac{c^2}{2\kappa^2}, \tag{3.15}$$

which completes the proof. □

The inequality (3.8) means that the probability that the approximating problem (1.2) has a unique optimal solution which coincides with the optimal solution of the true problem (1.1) approaches one exponentially fast. The inequality (3.15) also gives an estimate of the corresponding exponential constant.

Consider now a situation where the true problem (1.1) may have multiple solutions. As in the case of convergence w.p.1 presented in section 2, stronger assumptions are needed. Let A_N be the set of optimal solutions of the approximating problem (1.2), and consider the event

$$\mathcal{M}_N := \left\{\ \text{the set } A_N \text{ is nonempty and forms a face of the set } A\ \right\}. \tag{3.16}$$

THEOREM 3.2. *Suppose that the assumptions of Theorem* 2.3 *hold. Then there exists a constant $\beta > 0$ such that*

$$\limsup_{N\to\infty} \frac{1}{N} \log[P(\mathcal{M}_N^c)] \ \leq\ -\beta. \tag{3.17}$$

Proof. It is possible to prove this theorem by using arguments of Theorem 3.1 combined with assertions (a) and (b) of Lemma 2.4. The proof becomes even simpler if we use assertion (c) of Lemma 2.4. Let $\{x_1, \ldots, x_q\}$ be the set of points constructed in assertion (c) of Lemma 2.4. Recall that $\{x_1, \ldots, x_\ell\}$ forms the set of extreme points of A, and that $f(x_i) < f(x_j)$ for any $i \in \{1, \ldots, \ell\}$ and $j \in \{\ell + 1, \ldots, q\}$. Note that, by condition (2.13), we have that

$$\mathcal{M}_N^c \subset \left\{\exists\, i \in \{1, \ldots, \ell\}, \exists\, j \in \{\ell + 1, \ldots, q\} \text{ such that } \hat{f}_N(x_i) \geq \hat{f}_N(x_j)\right\}. \tag{3.18}$$

Moreover, there is $\varepsilon > 0$ such that the event in the right-hand side of (3.18) is included in the union of the events $\mathcal{A}_i := \{\hat{f}_N(x_i) \geq f(x_i) + \varepsilon\}$, $i = 1, \ldots, \ell$, and $\mathcal{A}_j := \{\hat{f}_N(x_j) \leq f(x_j) - \varepsilon\}$, $j = \ell + 1, \ldots, q$. It follows that

$$P\left(\mathcal{M}_N^c\right) \;\leq\; \sum_{i=1}^{\ell} P\left(\hat{f}_N(x_i) \geq f(x_i) + \varepsilon\right) + \sum_{j=\ell+1}^{q} P\left(\hat{f}_N(x_j) \leq f(x_j) - \varepsilon\right).$$

Therefore, in order to prove (3.17) it suffices to show that, for any $i \in \{1, \ldots, \ell\}$, there exists $\beta_i > 0$ such that

$$\limsup_{N\to\infty} \frac{1}{N} \log\left[P\left(\hat{f}_N(x_i) \geq f(x_i) + \varepsilon\right)\right] \;\leq\; -\beta_i$$

and, similarly, for any $j \in \{\ell + 1, \ldots, q\}$, there exists $\beta_j > 0$ such that

$$\limsup_{N\to\infty} \frac{1}{N} \log\left[P\left(\hat{f}_N(x_i) \leq f(x_i) - \varepsilon\right)\right] \;\leq\; -\beta_j.$$

Both assertions follow immediately from the LDP (in a unidimensional setting), since $\mathbb{E}[\hat{f}_N(x_i)] = f(x_i)$, $i = 1, \ldots, q$. This completes the proof by taking $\beta := \min_{i\in\{1,\ldots,q\}} \beta_i$. □

4. Examples. In this section we present some examples to illustrate the ideas discussed in sections 2 and 3.

4.1. The median problem, revisited. We begin by analyzing in more detail the median problem (1.3) discussed in the introduction. Let $Y_1, \ldots, Y_m$ be i.i.d. real valued random variables, each one taking values -1, 0, and 1 with equal probabilities $1/3$. Let $\hat{x}_N$ denote an optimal solution of the corresponding approximating problem (1.4). As it was shown in the introduction, $\hat{x}_N$ coincides with the true optimal solution $\bar{x} = 0$ with very high probability, even for small values of N compared to the size of the sample space.

We can approach this problem from the point of view of the large deviations theory. Let X be a binomial random variable $B(N, p)$ with $p = 1/3$. As was discussed in the introduction, the probability of the event $\hat{x}_N = 0$ is at least $1 - 2P(X \geq N/2)$ (more precisely, when N is even this probability is exactly $1-2P(X \geq N/2)+\binom{N}{N/2}p^N$, the last term becoming negligible as N grows). By Cramér's large deviations theorem we have that (see, e.g., [7, Theorem 2.2.3])

$$\begin{aligned} -\inf_{z>1/2} I(z) \;&\leq \liminf_{N\to\infty} \tfrac{1}{N} \log\left[P\left(\tfrac{X}{N} \geq \tfrac{1}{2}\right)\right] \\ &\leq \limsup_{N\to\infty} \tfrac{1}{N} \log\left[P\left(\tfrac{X}{N} \geq \tfrac{1}{2}\right)\right] \leq -\inf_{z\geq 1/2} I(z). \end{aligned}$$

For a binomial distribution $B(N, p)$, the large deviations rate function $I(z)$ is given by

$$I(z) = z \log\left[\frac{(1-p)z}{p(1-z)}\right] - \log\left[1 - p + \frac{(1-p)z}{1-z}\right]. \tag{4.1}$$

Since $I(\cdot)$ is continuous, it follows that

$$\lim_{N\to\infty} \frac{1}{N} \log\left[P\left(\frac{X}{N} \geq \frac{1}{2}\right)\right] = -\inf_{z\geq 1/2} I(z) = -I(0.5),$$

the last equality arising from the fact that the function $I(\cdot)$ is increasing on the interval $[p, \infty)$. From (4.1) we obtain that

$$I(0.5) = \log\left[\frac{(p^{-1}-1)^{1/2}}{2(1-p)}\right]. \tag{4.2}$$

For $p = 1/3$ we have $I(0.5) = \log(\frac{3\sqrt{2}}{4}) = 0.0589$, and hence the probability $P(X/N \geq 1/2)$ converges to zero at the exponential rate $e^{-0.0589N}$. Note that in the considered (one-dimensional) case the upper bound of Cramér's theorem holds for any N (and not just in the limiting sense). It follows that the probability that the sample estimate $\hat{x}_N$ is equal to the true optimal solution is greater than $(1 - 2e^{-0.0589N})^m$, which for large N is approximately equal to $1 - 2m\, e^{-0.0589N}$. Consequently the probability that the sample estimate $\hat{x}_N$ is not equal to the true optimal solution decreases exponentially fast with the sample size N and increases linearly with the number of variables m. For example, for $N = 100$ and $m = 50$ we have, by the above estimate, that the probability of the sample estimate $\hat{x}_N$ being equal to the true optimal solution is at least $(1 - 2e^{-5.89})^{50} = 0.76$. This can be compared with the exact probability of that event, which is about 0.96. This is quite typical for the large deviations estimates. For finite and not too "large" N, the large deviations estimates give poor approximations of the corresponding probabilities. What the large deviations theory provides, of course, is the exponential rate at which the corresponding probabilities converge to zero.

Suppose now that each variable Y_i has the following discrete distribution: it can take values -1, -0.5, 0.5, and 1 with equal probabilities 0.25. In that case the set of optimal solutions of the true problem (1.3) is not a singleton and is given by the cube $\{x : -0.5 \leq x_i \leq 0.5\}$. We have that the probability that the sample estimate $\hat{x}_{iN}$ belongs to the interval $[-0.5, 0.5]$ is at least $1 - 2P(X \geq N/2)$, where $X \sim B(N, 0.25)$. Again we obtain that the probability that $\hat{x}_N$ is an exact optimal solution of the true problem is approaching one exponentially fast with increasing N.

Now let $m = 1$ and suppose that the distribution of Y is discrete with possible values given by an odd number $r = 2\ell + 1$ of points equally spaced on the interval $[-1, 1]$ with equal probabilities of $1/r$. For "large" r we can view this as a discretization of the uniform distribution on the interval $[-1, 1]$. Then by the same arguments as above we obtain that the probability that $\hat{x}_N = 0$ is at least $1 - 2P(X \geq N/2)$, where $X \sim B(N, p)$ with $p = \ell/r$.

An estimate of how fast N grows as a function of the number of variables m and the number of discretization points r can be obtained using again large deviations techniques. Suppose that $m \geq 1$ and that each random variable Y_i, $i = 1, \ldots, m$, has a discrete distribution as above. From (4.2) we have that in this case the constant $\beta := I(0.5)$ is given by

$$\beta = \frac{1}{2}\log\left[\frac{r^2}{r^2-1}\right], \tag{4.3}$$

and hence

$$P(\hat{x}_N = 0) \geq (1 - 2e^{-\beta N})^m \cong 1 - 2me^{-\beta N}.$$

Consequently, for a fixed $\varepsilon > 0$, a (conservative) estimate of the sample size N needed to obtain $P(\hat{x}_N = 0) \geq 1 - \varepsilon$ is given by

$$N = \beta^{-1}\log(2m/\varepsilon) \cong (2r^2 - 1)\log(2m/\varepsilon),$$

so we see that N grows *quadratically* with the number of discretization points and *logarithmically* with the number of random variables.

4.2. A two-stage stochastic programming problem. We now present some numerical results obtained for the capacity expansion problem CEP1 described in [11], which can be modeled as a two-stage stochastic programming problem with complete recourse. The problem has 8 decision variables with 5 constraints (plus bound constraints) on the first stage, and 15 decision variables with 7 constraints (plus lower bound constraints) on the second stage. The random variables, which correspond to demand in the model, appear only on the right-hand side of the second stage. There are 3 i.i.d. random variables, each taking 6 possible values with equal probability, so the sample space has size $6^3 = 216$.

For the sake of verification, we initially solved the problem exactly by solving the equivalent deterministic LP, and we obtained the true minimizer $\bar{x}$. Notice that this optimal solution is unique. We then solved the corresponding Monte Carlo approximations, with sample sizes $N = 2, 5, 10, 15, 20, 35, 50$. For each sample size, we solved the approximating problem 400 times and counted how many times the optimal solution $\hat{x}_N$, of the approximating problem, coincided with the true solution $\bar{x}$. The corresponding proportion $\hat{p}$ is then an estimate of the probability $P(\hat{x}_N = \bar{x})$. Since the generated replications are independent, it follows that an unbiased estimator of the variance of $\hat{p}$ is given by $\hat{p}(1-\hat{p})/399$. From this value we obtain a 95% confidence interval whose half-width is denoted by Δ. The results are displayed in Table 4.1.

TABLE 4.1
Estimated probabilities $P(\hat{x}_N = \bar{x})$.

N	$\hat{p}$	Δ
2	0.463	.049
5	0.715	.044
10	0.793	.040
15	0.835	.036
20	0.905	.029
35	0.958	.020
50	0.975	.015

Notice again the exponential feature of the numbers on the table, i.e., how fast $\hat{p}$ gets close to one. It is interesting to notice that convergence in the CEP1 model is even faster than in the median problem, even though the median problem is much more structured (in particular, the median problem is separable) with a smaller sample space (27 points for 3 random variables, as opposed to 216 points in the CEP1 model). For instance, in the median problem a sample size of 20 gives the true optimal solution with probability 0.544, whereas in the CEP1 problem that probability is approximately 0.9. These results corroborate the ideas presented in the previous sections, showing that convergence can be very fast if there is a sharp minimum such as in the case of the CEP1 model. The results also suggest that the separability inherent in the median problem was not a major factor in the speed of convergence, which encourages us to think that the numerical results reported here can be obtained in more complex problems. Of course, more research is needed to draw any definite conclusions.

5. Conclusions. We presented in this paper some results concerning convergence of Monte Carlo simulation-based approximations for a class of stochastic programming problems. As pointed out in the introduction, the usual approach to con-

vergence analysis found in the literature consists in showing that optimal solutions of approximating problems converge, w.p.1, to optimal solutions of the original problem or in obtaining bounds for the rate of convergence via central limit theorem or large deviations type asymptotics. We show, under some specific assumptions (in particular, under the assumption that the sample space is finite), that the approximating problem provides an *exact* optimal solution w.p.1 for sample size N large enough and, moreover, that the probability of such an event approaches one at an *exponential* rate. This suggests that, in such cases, Monte Carlo simulation based algorithms could be efficient, since one may not need a large sample to find an exact optimal solution.

The median problem presented in section 4 illustrates that point. For a problem with 3^{200} scenarios, an approximating problem which employs only $N = 120$ samples, of a vector of dimension $m = 200$, yields the exact optimal solution approximately 95% of the time. Even more impressively, it is possible to show by the same type of calculations that $N = 150$ samples are enough to obtain the exact optimal solution with probability of about 95% for $m = 1000$ random variables, i.e., for 3^{1000} scenarios. Estimates of the sample size N, which were obtained in section 4 by the large deviations approximations, give slightly bigger values of N (for example, they give $N = 180$ instead of $N = 150$ for $m = 1000$). In either case the required sample size grows as a logarithm of the number m of random variables in that example. Of course, one must take into account the fact that this is a very structured problem, and in a more general case one may not get such drastically fast convergence; in fact, the flatter the objective function is around the optimal solution, the slower the convergence will be. Nevertheless, the CEP1 model studied in section 4 seems to indicate that fast convergence is obtained in more general problems, even in the absence of separability.

One should, however, be cautious about these results, especially with respect to the following aspect. The fact that the convergence is exponential does not necessarily imply that a small sample suffices. Indeed, the constant β in the corresponding exponential rate $e^{-\beta N}$ can be so small that one would need a large sample size N in order to achieve a reasonable precision. The lower bound (3.15) gives us an idea about the exponential constant β. In the median example, with r discretization points for each random variable Y_i, $i = 1, \ldots, m$, we have that we can take $c = 1/r$ and $\kappa = 1$, if we use ℓ_1 norm in the space $\mathbb{R}^m$. This gives us the lower bound $\beta \geq 1/(2r^2)$, which can be compared with the exact value of $\beta = \frac{1}{2}\log[r^2/(r^2-1)] \cong 1/(2r^2-1)$. Note that the estimate $\beta \geq 1/(2r^2)$ does not depend on the number m of random variables. This happens since any multiplicative constant before $e^{-\beta N}$ can be absorbed into the exponential rate as N tends to infinity.

Another remark concerns the assumption of Monte Carlo sampling in our analysis. By doing so, we were able to exploit properties of i.i.d. samples, which we used to derive our results. In practice, however, one might think of implementing *variance reduction techniques* in order to reduce even more the needed sample sizes. The incorporation of such techniques into stochastic optimization algorithms has been shown to be very effective in practice (see, e.g., [1], [5], [21]). Research on specific applications of variance reduction techniques to the type of problems discussed in this paper is underway.

Acknowledgment. We thank the referees for constructive comments which helped to improve the presentation of the paper.

REFERENCES

[1] T.G. Bailey, P.A. Jensen, and D.P. Morton, *Response surface analysis of two-stage stochastic linear programming with recourse*, Naval Res. Logist., 46 (1999), pp. 753–776.

[2] D.P. Bertsekas, *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.

[3] J.F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer-Verlag, New York, 2000.

[4] J.V. Burke and M.C. Ferris, *Weak sharp minima in mathematical programming*, SIAM J. Control Optim., 31 (1993), pp. 1340–1359.

[5] G.B. Dantzig and P.W. Glynn, *Parallel processors for planning under uncertainty*, Ann. Oper. Res., 22 (1990), pp. 1–21.

[6] H.A. David, *Order Statistics*, 2nd ed., Wiley, New York, 1981.

[7] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications*, 2nd ed., Springer-Verlag, New York, 1998.

[8] J.D. Deuschel and D.W. Stroock, *Large Deviations*, Academic Press, Boston, 1989.

[9] J. Dupačová and R.J.-B. Wets, *Asymptotic behavior of statistical estimators and of optimal solutions of stochastic optimization problems*, Ann. Statist., 16 (1988), pp. 1517–1549.

[10] F. Hiai, *Strong laws of large numbers for multivalued random variables*, in Multifunctions and Integrands, A. Dold and B. Eckmann, eds., Springer-Verlag, Berlin, 1984.

[11] J.L. Higle and S. Sen, *Finite master programs in regularized stochastic decomposition*, Math. Programming, 67 (1994), pp. 143–168.

[12] J.-B. Hiriart-Urruty and C. Lemarechal, *Convex Analysis and Minimization Algorithms* I, Springer-Verlag, Berlin, 1993.

[13] Y.M. Kaniovski, A.J. King, and R.J.-B. Wets, *Probabilistic bounds (via large deviations) for the solutions of stochastic programming problems*, Ann. Oper. Res., 56 (1995), pp. 189–208.

[14] A.J. King and Roger J.-B. Wets, *Epi-consistency of convex stochastic programs*, Stochastics, 34 (1991), pp. 83–92.

[15] 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.

[16] G.Ch. Pflug, *Stochastic programs and statistical data*, Ann. Oper. Res., 85 (1999), pp. 59–78.

[17] S.M. Robinson, *Analysis of sample-path optimization*, Math. Oper. Res., 21 (1996), pp. 513–528.

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

[19] W. Römisch and R. Schultz, *Lipschitz stability for stochastic programs with complete recourse*, SIAM J. Optim., 6 (1996), pp. 531–547.

[20] A. Shapiro, *Asymptotic behavior of optimal solutions in stochastic programming*, Math. Oper. Res., 18 (1993), pp. 829–845.

[21] A. Shapiro and T. Homem-de-Mello, *A simulation-based approach to stochastic programming with recourse*, Math. Programming 81 (1998), pp. 301–325.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 87–112

DIFFERENTIAL STABILITY OF TWO-STAGE STOCHASTIC PROGRAMS*

DARINKA DENTCHEVA† AND WERNER RÖMISCH†

Abstract. Two-stage stochastic programs with random right-hand side are considered. Optimal values and solution sets are regarded as mappings of the expected recourse functions and their perturbations, respectively. Conditions are identified implying that these mappings are directionally differentiable and semidifferentiable on appropriate functional spaces. Explicit formulas for the derivatives are derived. Special attention is paid to the role of a Lipschitz condition for solution sets as well as of a quadratic growth condition of the objective function.

Key words. two-stage stochastic programs, sensitivity analysis, directional derivatives, semiderivatives, solution sets

AMS subject classifications. 90C15, 90C31

PII. S1052623499316520

1. Introduction.

Two-stage stochastic programming is concerned with problems that require a here-and-now decision on the basis of given probabilistic information on the random data without making further observations. The costs to be minimized consist of the direct costs of the here-and-now (or first-stage) decision as well as the costs generated by the need of taking a recourse (or second-stage) decision in response to the random environment. Recourse costs are often formulated by means of expected values with respect to the probability distribution of the involved random data. In this way, two-stage models and their solutions depend on the underlying probability distribution. Since this distribution is often incompletely known in applied models, or it has to be approximated for computational purposes, the stability behavior of stochastic programming models when changing the probability measure is important. This problem is studied in a number of papers. We mention here only the surveys [13], [40] and the papers [1], [12], [18], [26], [27], [34], and [35]. The paper [1] contains general results on continuity properties of optimal values and solutions when perturbing the probability measures with respect to the topology of weak convergence. Quantitative continuity results of solution sets to two-stage stochastic programs with respect to suitable distances of probability measures are obtained in [26] and [27]. Asymptotic properties of statistical estimators of values and solutions to stochastic programs are derived in [18], [34], [35]. They are based on directional differentiability properties of the underlying optimization problems with respect to the parameter that carries the randomness [18], [35] or the probability measure [34]. These directional differentiability results for values [35] and solutions [13], [18], [34] lead to asymptotic results via the so-called *delta-method*. For a description of the delta-method we refer to Chapter 6 in [28], [35], to [36] for an up-to-date presentation, and to [16] for a set-valued variant. These papers illuminate the importance

*Received by the editors March 1, 1997; accepted for publication (in revised form) March 6, 2000; published electronically July 25, 2000. This research was supported by the Deutsche Forschungsgemeinschaft.

http://www.siam.org/journals/siopt/11-1/31652.html

†Humboldt-Universität Berlin, Institut für Mathematik, 10099 Berlin, Germany. Current address: Lehigh University, Department of Industrial and Manufacturing Systems Engineering, Bethlehem, PA 15018 (dad8@lehigh.edu, romisch@mathematik.hu-berlin.de).

of the Hadamard directional differentiability (for single-valued functions) and of the semidifferentiability (for set-valued mappings) in the context of asymptotic statistics.

The present paper aims at contributing to this line of differential stability studies. The results in [18], [34] apply to fairly general stochastic optimization models but impose conditions that are rather restrictive in our context. The present paper deals with special two-stage models and, using structural properties, avoids certain assumptions that complicate or even prevent the applicability of the general results to two-stage stochastic programs. Such assumptions are the (local) uniqueness of solutions and differentiability properties of perturbed problems, which are indispensable in [18], [34]. Before discussing this in more detail, let us introduce the class of two-stage stochastic programs we want to consider:

$$\min\{g(x) + Q_\mu(Ax) : x \in C\}, \tag{1.1}$$

where $g : \mathbb{R}^m \to \mathbb{R}$ is a convex function, $C \subseteq \mathbb{R}^m$ is a nonempty closed convex set, A is an (s, m)-matrix, and Q_μ is the expected recourse function with respect to the (Borel) probability measure μ on $\mathbb{R}^s$;

$$Q_\mu(y) = \int_{\mathbb{R}^s} \tilde{Q}(\omega - y)\mu(\mathrm{d}\omega), \tag{1.2}$$

$$\tilde{Q}(t) = \inf\{\langle q, u\rangle : Wu = t, u \geq 0\}, \quad t \in \mathbb{R}^s. \tag{1.3}$$

Here $q \in \mathbb{R}^{\bar{m}}$ are the recourse costs, W is an $(s, \bar{m})$-matrix and called the recourse matrix, and $\tilde{Q}(\omega - Ax)$ corresponds to the value of the optimal second-stage decision for compensating a possible violation of the (random) constraint $Ax = \omega$. To have the problem (1.1)–(1.3) well defined, we assume

(A1) $\operatorname{pos} W = \{Wu : u \in \mathbb{R}^{\bar{m}}_+\} = \mathbb{R}^s$ (complete recourse),

(A2) $M_D = \{t \in \mathbb{R}^s : W^T t \leq q\} \neq \emptyset$ (dual feasibility),

(A3) $\int_{\mathbb{R}^s} \|\omega\|\mu(\mathrm{d}\omega) < \infty$ (finite first moment).

The assumptions (A1) and (A2) imply that $\tilde{Q}$ is finite, convex, and polyhedral on $\mathbb{R}^s$. Due to (A3), Q_μ is also finite and convex on $\mathbb{R}^s$ (cf. [15], [39]). Observe that, in general, an expected recourse function Q_μ may be nondifferentiable on a certain union of hyperplanes in $\mathbb{R}^s$ and that, indeed, differentiability properties of Q_μ depend on the degree of smoothness induced by the measure μ (cf. [15], [21], [38], [39], and Remark 4.10). Another observation is that the uniqueness of solutions to (1.1) is guaranteed only if the constraint set C picks just one element from the relevant level set of $g(\cdot) + Q_\mu(A\,\cdot)$. As the next example shows, this set may be large since $Q_\mu(A\,\cdot)$ is constant on translates of the null space of the matrix A.

Example 1.1. In (1.1)–(1.3), let $m = 3$, $n = 2$, $g(x) = \frac{1}{4}(x_2 - x_3)$, $C = [0, \frac{1}{2}]^3$, $A = \left(\begin{smallmatrix} 1 & 0 & -1 \\ 1 & -1 & 0 \end{smallmatrix}\right)$, $q = (1, 1, 1, 1)$, $W = \left(\begin{smallmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{smallmatrix}\right)$, and μ be the uniform distribution on the square $[-\frac{1}{2}, \frac{1}{2}]^2$ in $\mathbb{R}^2$.

Then we have $\tilde{Q}(t) = |t_1| + |t_2|$ and $Q_\mu(y) = y_1^2 + y_2^2 + \frac{1}{2}$ for $y = (y_1, y_2) \in [-\frac{1}{2}, \frac{1}{2}]^2$. The optimization problem (1.1) and its solution set $\psi(Q_\mu)$ take the form

$$\min\left\{\frac{1}{4}(x_2 - x_3) + (x_1 - x_3)^2 + (x_1 - x_2)^2 + \frac{1}{2} : (x_1, x_2, x_3) \in \left[0, \frac{1}{2}\right]^3\right\},$$

$$\psi(Q_\mu) = \left\{ \left(\frac{1}{8} + u,\, u,\, \frac{1}{4} + u\right) \,:\, u \in \left[0,\, \frac{1}{4}\right] \right\} = \left\{ \left(\frac{1}{8},\, 0,\, \frac{1}{4}\right) + \ker A \right\} \cap C,$$

where $\ker A = \{(u,\, u,\, u) \,:\, u \in \mathbb{R}\}$ is the null space of A.

Proposition 2.1 below provides some more insight into the structure of the solution set to (1.1) and elucidates the role of the set-valued mapping $\sigma(y) := \operatorname{argmin}\{g(x) : x \in C, Ax = y\}$ in this respect.

Note that assumption (A1) could be relaxed by introducing the set $\mathcal{K} = \{y \in \mathbb{R}^s : Q_\mu(y) < +\infty\}$. Then (A2) and (A3) imply that $\mathcal{K}$ is a closed convex polyhedron and that Q_μ is convex and continuous on $\mathcal{K}$ (cf. [39]). Now (A1) can be replaced by the condition $\mathcal{K} \supseteq A(C)$ (relatively complete recourse), and much of the work done in this paper carries over to this more general setting by using spaces of functions defined on $\mathcal{K}$ instead of $\mathbb{R}^s$.

Let K_C denote the set of all convex functions on $\mathbb{R}^s$ which forms a convex cone in the space $C^0(\mathbb{R}^s)$ of all continuous functions on $\mathbb{R}^s$. K_C will serve as the set of possible perturbations of the given expected recourse function $Q_\mu \in K_C$. We define

$$\begin{aligned} \varphi(Q) &:= \inf\{g(x) + Q(Ax) : x \in C\}, \\ \psi(Q) &:= \operatorname{argmin}\{g(x) + Q(Ax) : x \in C\} \end{aligned}$$

and regard φ and ψ as mappings from K_C into the extended reals and the set of all closed convex subsets of $\mathbb{R}^m$, respectively.

In this paper we develop a sensitivity analysis for the mappings φ and ψ at some given function Q_μ. The stochastic programming origin of the model (1.1) takes a back seat, and our results are stated in terms of general conditions on Q_μ and its perturbations Q. We identify conditions such that the value function φ has first- and second-order directional derivatives and the solution-set mapping ψ is directionally differentiable at Q_μ into admissible directions. Here, admissibility means that the direction belongs to the radial tangent cone to K_C at Q_μ, i.e.,

$$T^r(K_C; Q_\mu) = \{\lambda(Q - Q_\mu) : Q \in K_C, \lambda > 0\},$$

ensuring that the difference quotients are well defined. For v belonging to $T^r(K_C; Q_\mu)$ the Gateaux directional derivatives of φ and ψ at Q_μ and $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$, respectively, are defined as

$$\begin{aligned} \varphi'(Q_\mu; v) &= \lim_{t \to 0+} \frac{1}{t}(\varphi(Q_\mu + tv) - \varphi(Q_\mu)), \\ \varphi''(Q_\mu; v) &= \lim_{t \to 0+} \frac{1}{t^2}(\varphi(Q_\mu + tv) - \varphi(Q_\mu) - t\varphi'(Q_\mu; v)), \\ \psi'(Q_\mu, \bar{x}; v) &= \lim_{t \to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - \bar{x}), \end{aligned}$$

if the limits exist. The third limit is understood in the sense of (Painlevé–Kuratowski) set convergence (e.g. [2]). Recall that the lower and upper set limits of a family $(S_t)_{t>0}$ of subsets of a metric space (X, d) are defined as

$$\begin{aligned} \liminf_{t \to 0+} S_t &= \{x \in X : \lim_{t \to 0+} d(x, S_t) = 0\}, \\ \limsup_{t \to 0+} S_t &= \{x \in X : \liminf_{t \to 0+} d(x, S_t) = 0\}. \end{aligned}$$

Both sets are closed and the lower set limit is contained in the upper limit. If both limits coincide, the family $(S_t)_{t>0}$ is said to converge and its limit set is denoted

by $\lim_{t\to 0+} S_t$. For sequences of sets $(S_n)_{n\in\mathbb{N}}$ the definitions of set limits are modified correspondingly.

We also derive conditions implying that the limits defining the directional derivatives exist uniformly with respect to directions v belonging to compact subsets of certain functional spaces. The limits are then called (first- or second-order) Hadamard directional derivatives and semiderivatives for set-valued maps, respectively. The corresponding directional derivatives are defined on tangent cones to the cone of convex functions in certain functional spaces. For more information on concepts of directional differentiability and multifunction differentiability we refer to [4], [33], and to [2], [3], [23], and [25], respectively.

Let us fix some notations used throughout the paper. $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ denote the norm and scalar product, respectively, in some Euclidean space $\mathbb{R}^n$; $B(x,r)$ denotes the open ball around $x\in\mathbb{R}^n$ with radius $r>0$; $d(x,D)$ denotes the distance of $x\in\mathbb{R}^n$ to the set $D\subseteq\mathbb{R}^n$; for a real-valued function f on $\mathbb{R}^n$, ∇f denotes its gradient in $\mathbb{R}^n$ and the (n,n)-matrix $\nabla^2 f$ its Hessian; if f is locally Lipschitzian near $x\in\mathbb{R}^n$, $\partial f(x)$ denotes the Clarke subdifferential of f at x; $f'(x;d)$ denotes the directional derivative of f at x in direction d if it exists; for $x\in C$, $T(C;x)$ denotes the tangent cone to C at x, i.e., $T(C;x)=\liminf_{t\to 0+}\frac{1}{t}(C-x)=\mathrm{cl}\{\lambda(y-x):\ y\in C,\lambda>0\}$, where cl stands for closure; for $x\in C$, $\xi\in T(C;x)$, $T^2(C;x,\xi)$ denotes the second-order tangent set to C at x in direction ξ, i.e., $T^2(C;x,\xi)=\liminf_{t\to 0+}\frac{1}{t^2}(C-x-t\xi)$ (note that $T^2(C;x,\xi)$ is closed and convex; see [10], [6] for further properties).

In our paper, we use the following linear metric spaces of real-valued functions on $\mathbb{R}^s$: The space $C^0(\mathbb{R}^s)$ of continuous functions on $\mathbb{R}^s$ equipped with the distance

$$d_\infty(f,\tilde f)=\sum_{n=1}^{\infty}2^{-n}\frac{\|f-\tilde f\|_{\infty,n}}{1+\|f-\tilde f\|_{\infty,n}},$$

where

$$\|f\|_{\infty,r}=\max_{\|y\|\le r}|f(y)|\ \text{for}\ f,\tilde f\in C^0(\mathbb{R}^s)\ \text{and}\ r>0;$$

the space $C^{0,1}(\mathbb{R}^s)$ of locally Lipschitzian functions on $\mathbb{R}^s$ with the metric

$$d_L(f,\tilde f)=\sum_{n=1}^{\infty}2^{-n}\frac{\|f-\tilde f\|_{\infty,n}+\|f-\tilde f\|_{L,n}}{1+\|f-\tilde f\|_{\infty,n}+\|f-\tilde f\|_{L,n}},$$

where

$$\begin{aligned}\|f\|_{L,r} &= \sup\left\{\frac{|f(y)-f(\tilde y)|}{\|y-\tilde y\|}:\|y\|\le r,\|\tilde y\|\le r,y\ne\tilde y\right\},\\ &= \sup\{\|z\|:z\in\partial f(y),\|y\|\le r\}\ \text{for}\ f,\tilde f\in C^{0,1}(\mathbb{R}^s)\ \text{ and}\ r>0;\end{aligned}$$

the space $C^1(\mathbb{R}^s)$ of continuously differentiable functions on $\mathbb{R}^s$ with the metric $d(f,\tilde f)=d_\infty(f,\tilde f)+d_\infty(\nabla f,\nabla\tilde f)$, $f,\tilde f\in C^1(\mathbb{R}^s)$, and the space $C^{1,1}(\mathbb{R}^s)$ of functions in $C^1(\mathbb{R}^s)$ whose gradients are locally Lipschitzian on $\mathbb{R}^s$ equipped with the distance $d(f,\tilde f)=d_\infty(f,\tilde f)+d_\infty(\nabla f,\nabla\tilde f)+d_L(\nabla f,\nabla\tilde f)$, $f,\tilde f\in C^{1,1}(\mathbb{R}^s)$.

The sensitivity analysis of the mappings φ and ψ is carried out by exploiting structural properties of the optimization model (1.1). We obtain novel differentiability properties of solution sets and extend our earlier results on directional differentiability

of optimal values in [12] considerably. As one might expect, the basic ingredients of our analysis are a Lipschitz continuity result for solution sets with respect to the distance in $C^{0,1}(\mathbb{R}^s)$ (Theorem 2.3) and a quadratic growth condition near solution sets (Theorem 2.7). Both theorems extend earlier results in [27] to more general situations for the first-stage costs g and constraint set C. All results in the paper apply to the *linear-quadratic case*, i.e., to linear- or convex-quadratic g and polyhedral C. Indeed, all results are formulated as generally as possible and most of them are accompanied by illustrative examples. The second-order analysis of φ in section 3 utilizes some ideas from [31] and [32], but its proof is entirely different and its Gateaux differentiability part is valid for nondifferentiable directions (Theorem 3.4). It is also elaborated that the Hadamard directional differentiability properties require the C^0-topology for the first-order result and the C^1-topology for the second-order one (Theorem 3.8), while the $C^{1,1}$-topology is needed for the semidifferentiability of the solution-set mapping ψ (Theorem 4.9). All results on differentiability properties of ψ in section 4 are new and do not follow from recent sensitivity results (e.g., [5], [8], [7], [17], [32]; see also the survey [8] for further references and Remark 4.4 for a more detailed discussion).

The results of sections 3 and 4 have direct implications to asymptotic properties of values and solution sets of two-stage stochastic programs when applying (smooth) nonparametric estimation procedures to approximate Q_μ. For a discussion of some of the related aspects we refer to the brief exposition in Remark 4.11. Further applications to asymptotics are beyond the scope of this paper and will be done elsewhere.

2. Basic directional properties. The first step in our analysis of directional properties consists in establishing results on the lower Lipschitz continuity of ψ and on the directional uniform quadratic growth of the objective near its solution set. Both results become important for our method of deriving directional differentiability properties for the optimal value function φ and the solution set mapping ψ at some given expected recourse function Q_μ. Their proofs are based on a decomposition of the program

$$\min\{g(x) + Q(Ax) : x \in C\}, \tag{2.1}$$

with Q belonging to K_C, into two auxiliary problems. The first one is a convex program with decisions taken from $A(C)$, and the second represents a parametric convex program which does not depend on Q.

PROPOSITION 2.1. *Let $Q \in K_C$, and let $\psi(Q)$ be nonempty. Then we have*

$$\begin{aligned}
\varphi(Q) &= \inf\{\pi(y) + Q(y) : y \in A(C)\} = \pi(Ax) + Q(Ax), \ \textit{for any } x \in \psi(Q), \ \textit{and}\\
\psi(Q) &= \sigma(Y(Q)), \ \textit{where}\\
Y(Q) &:= \operatorname{argmin}\{\pi(y) + Q(y) : y \in A(C)\},\\
\pi(y) &:= \inf\{g(x) : x \in C, Ax = y\}, \ \textit{and}\\
\sigma(y) &:= \operatorname{argmin}\{g(x) : x \in C, Ax = y\}, \quad y \in A(C).
\end{aligned}$$

Moreover, π is convex on $A(C)$ and $\operatorname{dom}\sigma$ is nonempty.

Proof. Let $\bar{x} \in \psi(Q)$. Then we have

$$\varphi(Q) = g(\bar{x}) + Q(A\bar{x}) \geq \pi(A\bar{x}) + Q(A\bar{x}) \geq \inf\{\pi(y) + Q(y) : y \in A(C)\}.$$

For the converse inequality, let $\varepsilon > 0$ and $\bar{y} \in A(C)$ be such that

$$\pi(\bar{y}) + Q(\bar{y}) \leq \inf\{\pi(y) + Q(y) : y \in A(C)\} + \frac{\varepsilon}{2}.$$

Then there exists a $\bar{x} \in C$ such that $A\bar{x} = \bar{y}$ and $g(\bar{x}) \leq \pi(\bar{y}) + \frac{\varepsilon}{2}$. Hence

$$\begin{aligned}\varphi(Q) &\leq g(\bar{x}) + Q(A\bar{x}) \leq \pi(\bar{y}) + Q(\bar{y}) + \frac{\varepsilon}{2} \\ &\leq \inf\{\pi(y) + Q(y) : y \in A(C)\} + \varepsilon.\end{aligned}$$

Since $\varepsilon > 0$ is arbitrary, the first statement has been shown. In particular, $x \in \sigma(Ax)$ and $Ax \in Y(Q)$ for any $x \in \psi(Q)$. Hence, it holds that $\psi(Q) \subseteq \sigma(Y(Q))$. Conversely, let $x \in \sigma(Y(Q))$. Then $x \in \sigma(y)$ for some $y \in Y(Q)$. Thus $Ax = y$ and $g(x) = \pi(y) = \pi(Ax)$, implying

$$\begin{aligned}g(x) + Q(Ax) &= \pi(Ax) + Q(Ax) = \inf\{\pi(y) + Q(y) : y \in A(C)\} \\ &= \varphi(Q) \quad \text{and} \quad x \in \psi(Q).\end{aligned}$$

Since the convexity of π is immediate, the proof is complete. □

In the following, it will turn out that Lipschitzian properties of the solution set mapping $y \mapsto \sigma(y)$ and a quadratic growth property of g near $\sigma(y)$ are essential. For the linear-quadratic case we are in a comfortable situation in this respect. Namely, we have the following proposition.

PROPOSITION 2.2. *Let g be linear or convex-quadratic, let C be convex polyhedral, and assume* $\operatorname{dom}\sigma$ *to be nonempty. Then σ is a polyhedral multifunction which is Hausdorff Lipschitzian on its domain* $\operatorname{dom}\sigma = A(C)$*, i.e., there exists a constant $L > 0$ such that*

$$d_H(\sigma(y), \sigma(\tilde{y})) \leq L\|y - \tilde{y}\| \quad \textit{for all} \quad y, \tilde{y} \in A(C),$$

where d_H denotes the (extended) Hausdorff distance on subsets of $\mathbb{R}^m$.

Moreover, for each $r > 0$ there exists a constant $\eta(r) > 0$ such that

$$g(x) \geq \pi(Ax) + \eta(r)d(x, \sigma(Ax))^2 \quad \textit{for all} \quad x \in C \cap B(0, r).$$

(*Here π and σ are defined as in Proposition* 2.1.)

Proof. The Lipschitz property of σ is shown in [19, Theorem 4.2]. To prove the second statement, let g be of the form $g(x) = \langle Hx, x\rangle + \langle c, x\rangle$, where H is symmetric and positive semidefinite and $c \in \mathbb{R}^m$. For each $y \in A(C)$ we fix some $z(y) \in \sigma(y)$. An elementary characterization of solution sets to convex-quadratic programs with linear constraints yields that

$$\sigma(y) = \{x \in C : Ax = y, Hx = Hz(y), \langle c, x\rangle = \langle c, z(y)\rangle\}.$$

Due to the Lipschitz behavior of convex polyhedra (cf. [37]), there exists a constant $L_\sigma > 0$ such that

$$d(x, \sigma(y)) \leq L_\sigma(\|Hx - Hz(y)\| + |\langle c, x\rangle - \langle c, z(y)\rangle|)$$

for all $y \in A(C)$ and $x \in C$ with $Ax = y$. Using the decomposition $H = H^{\frac{1}{2}}H^{\frac{1}{2}}$, where $H^{\frac{1}{2}}$ denotes the square root of H, and the representation $\langle c, x\rangle - \langle c, z(y)\rangle = g(x) - \pi(y) - \|H^{\frac{1}{2}}x\|^2 + \|H^{\frac{1}{2}}z(y)\|^2$, one arrives at the estimate

$$d(x, \sigma(y)) \leq L_\sigma(\|H^{\frac{1}{2}}\|(1 + \|x\| + \|z(y)\|)\|H^{\frac{1}{2}}(x - z(y))\| + g(x) - \pi(y))$$

for all $y \in A(C)$ and $x \in C$ with $Ax = y$.

Now, let $r > 0$ and let us fix some element $\bar{x} \in C \cap B(0,r)$ and a corresponding $z(A\bar{x}) \in \sigma(A\bar{x})$. For each $y \in A(C)$ we now select $z(y) \in \sigma(y)$ such that $\|z(y) - z(A\bar{x})\| = d(z(A\bar{x}), \sigma(y))$. Since σ is Hausdorff Lipschitzian on $A(C)$, this implies $\|z(y) - z(A\bar{x})\| \leq L\|A\bar{x} - y\|$ for all $y \in A(C)$. Hence, there exists a constant $K(r) > 0$ such that $\|z(Ax)\| \leq K(r)$ for all $x \in C \cap B(0,r)$. Thus our estimate continues to $d(x, \sigma(Ax))^2 \leq \hat{L}(r)(\|H^{\frac{1}{2}}(x - z(Ax))\|^2 + (g(x) - \pi(Ax))^2)$ for all $x \in C \cap B(0,r)$ and some constant $\hat{L}(r) > 0$. Furthermore, the equation

$$g\left(\frac{1}{2}(x + z(y))\right) = \frac{1}{2}g(x) + \frac{1}{2}g(z(y)) - \frac{1}{4}\|H^{\frac{1}{2}}(x - z(y))\|^2$$

implies $\|H^{\frac{1}{2}}(x - z(y))\|^2 \leq 2(g(x) - \pi(y))$ for all $y \in A(C)$, $x \in C$, with $Ax = y$. Therefore, we finally obtain

$$\begin{aligned} d(x, \sigma(Ax))^2 &\leq \hat{L}(r)(2(g(x) - \pi(Ax)) + (g(x) - \pi(Ax))^2) \\ &\leq \hat{L}(r)\max\{2, K(r)\}(g(x) - \pi(Ax)) \end{aligned}$$

for all $x \in C \cap B(0,r)$, where $K(r) := \sup_{x \in C \cap B(0,r)}(g(x) - \pi(Ax))$. □

Due to the above proposition, the main results in this section apply to the linear-quadratic case. Although this case represents the main application of our results, the assumptions of the following theorems are formulated in terms of general conditions on the mapping σ in order to widen the range of applications. The first theorem states (lower) Lipschitz continuity of ψ at Q_μ and supplements Theorem 2.4 in [27].

THEOREM 2.3. *Let $Q_\mu \in K_C$, let $\psi(Q_\mu)$ be nonempty and bounded, and let Q_μ be strongly convex on some open, convex neighborhood of $A\psi(Q_\mu)$. Let $\bar{x} \in \psi(Q_\mu)$ and assume that there exist a constant $L > 0$ and a neighborhood U of $\bar{y}$ with $\{\bar{y}\} = A\psi(Q_\mu)$ such that*

$$d(\bar{x}, \sigma(y)) \leq L\|\bar{y} - y\| \quad \textit{for all} \quad y \in A(C) \cap U.$$

Then there exist constants $\hat{L} > 0$, $\delta > 0$, and $r > 0$ such that

$$d(\bar{x}, \psi(Q)) \leq \hat{L}\|Q - Q_\mu\|_{L,r}$$

whenever $Q \in K_C$ and $\|Q - Q_\mu\|_{L,r} < \delta$.

Proof. We may assume that U is open and convex and that Q_μ is strongly convex on U. Let V be an open, convex, bounded subset of $\mathbb{R}^m$ such that $\psi(Q_\mu) \subset V$ and $A(V) \subset U$. It follows from Proposition 2.3 in [27] (where a slightly different terminology is used) that there exists a constant $\delta > 0$ such that $\emptyset \neq \psi(Q) \subset V$ whenever $Q \in K_C$ and

$$\sup\{\|z\| : z \in \partial(Q - Q_\mu)(y),\, y \in \operatorname{cl} A(V)\} < \delta.$$

Let $r > 0$ be chosen such that $\operatorname{cl} A(V) \subset \bar{B}(0,r)$. Hence, we have $\emptyset \neq \psi(Q) \subset V$ whenever $Q \in K_C$, $\|Q - Q_\mu\|_{L,r} < \delta$. Then Proposition 2.1 yields the relation $\psi(Q) = \sigma(Y(q))$, where $Y(Q) = \operatorname{argmin}\{\pi(y) + Q(y) : y \in A(C)\}$. Since Q_μ is strongly convex on U, there exists a constant $\kappa > 0$ such that

$$\kappa\|y - \bar{y}\|^2 \leq \pi(y) + Q_\mu(y) - (\pi(\bar{y}) + Q_\mu(\bar{y})) \quad \text{for all} \quad y \in U.$$

Let $Q \in K_C$ with $\|Q - Q_\mu\|_{L,r} < \delta$, and let $\tilde{y} \in Y(Q)$. Since y belongs to $A(V) \subset U$, we obtain

$$\begin{aligned} \kappa\|\tilde{y} - \bar{y}\|^2 &\leq \pi(\tilde{y}) + Q_\mu(\tilde{y}) - (\pi(\bar{y}) + Q_\mu(\bar{y})) + \pi(\bar{y}) + Q(\bar{y}) - (\pi(\tilde{y}) + Q(\tilde{y})) \\ &= (Q - Q_\mu)(\bar{y}) - (Q - Q_\mu)(\tilde{y}), \end{aligned}$$

and, hence,

$$\|\tilde{y} - \bar{y}\| \leq \frac{1}{\kappa} \frac{(Q - Q_\mu)(\bar{y}) - (Q - Q_\mu)(\tilde{y})}{\|\bar{y} - \tilde{y}\|} \leq \frac{1}{\kappa} \|Q - Q_\mu\|_{L,r}.$$

The proof can now be completed as follows. Let $Q \in K_C$ be such that $\|Q - Q_\mu\|_{L,r} < \delta$. Then

$$\begin{aligned} d(\bar{x}, \psi(Q)) = d(\bar{x}, \sigma(Y(Q))) &\leq \sup_{y \in Y(Q)} d(\bar{x}, \sigma(y)) \\ &\leq L \sup_{y \in Y(Q)} \|\bar{y} - y\| \leq \frac{L}{\kappa} \|Q - Q_\mu\|_{L,r}. \qquad \square \end{aligned}$$

Remark 2.4. The proof shows that a Lipschitz modulus of ψ can be chosen as the quotient of a Lipschitz constant to σ and a strong convexity constant to Q_μ.

From the proof it is immediate that replacing the local Lipschitz condition on σ by stronger conditions like

$$\begin{aligned} &\sup_{x \in \sigma(\bar{y})} d(x, \sigma(y)) \leq L\|\bar{y} - y\| \quad \text{or} \\ &d_H(\sigma(\bar{y}), \sigma(y)) \leq L\|\bar{y} - y\| \quad \text{for all } \ y \in A(C) \cap U \end{aligned}$$

leads to corresponding stronger Lipschitz continuity properties of solution sets. Because of Proposition 2.2, all of this applies to the linear-quadratic case. However, it is worth mentioning that the theorem also applies to more general problems such that the corresponding solution sets $\sigma(y)$ enjoy Lipschitzian properties. Conditions ensuring Lipschitz behavior of σ can be derived from stability results for the corresponding parametric generalized equation

$$0 \in \nabla L(x, \lambda; y) + N_{C \times \mathbb{R}^s}(x, \lambda), \tag{2.2}$$

which describes the first-order necessary optimality condition. Here $L(x, \lambda; y) := g(x) + \lambda^T(Ax - y)$ is the Lagrangian function, $\nabla L(x, \lambda; y) = \binom{\nabla g(x) + A^T\lambda}{Ax - y}$, where g is assumed to be continuously differentiable, and $N_{C \times \mathbb{R}^s}$ is the normal cone map of convex analysis. Such stability results are presently available for broad classes of parametric generalized equations (e.g., [17], [22], [24]). A typical recent result in this direction, which applies to our situation for twice continuously differentiable g, is Theorem 5.1 in [22]. It says that the solution set mapping of the parametric generalized equation (2.2) is pseudo-Lipschitzian around $(\bar{x}, \bar{\lambda}; \bar{y})$ if the adjoint generalized equation

$$0 \in \nabla^2 L(\bar{x}, \bar{\lambda}; \bar{y}) w^* + D^* N_{C \times \mathbb{R}^s}(\bar{x}, \bar{\lambda}; -\nabla L(\bar{x}, \bar{\lambda}; \bar{y}))(w^*) \tag{2.3}$$

has only the trivial solution $w^* = 0$.

Here $D^* N_{C \times \mathbb{R}^s}(\bar{x}, \bar{\lambda}; -\nabla L(\bar{x}, \bar{\lambda}; \bar{y}))$ is the Mordukhovich coderivative [22] of the normal cone multifunction at the point $(\bar{x}, \bar{\lambda}; -\nabla L(\bar{x}, \bar{\lambda}; \bar{y}))$ belonging to the graph of $N_{C \times \mathbb{R}^s}$. Translating this into our framework, we obtain that the mapping σ is pseudo-Lipschitzian around $(\bar{x}, \bar{y})$ if the following two conditions are satisfied.

(a) There exists an element $\hat{x}$ belonging to the relative interior of C such that $A\hat{x} = \bar{y}$ (Slater condition).

(b) The equations $Aw_1^* = 0$ and $0 \in \nabla^2 g(\bar{x}) w_1^* + A^T w_2^* + D^* N_C(\bar{x}, \bar{\lambda}; -\nabla g(\bar{x}) - A^T\bar{\lambda})(w_1^*)$ have only the trivial solution $w_1^* = 0$, $w_2^* = 0$. (Here $(\bar{x}, \bar{\lambda})$ is a solution of (2.2) for $y = \bar{y}$.)

The next examples show that the theorem applies to instances of two-stage stochastic programs with nonunique solutions and with nonpolyhedral convex constraint sets C.

Example 2.5. We revisit Example 1.1 and obtain with the notations of Proposition 2.1 that $A(C) = [-\frac{1}{2}, \frac{1}{2}]^2$, $\pi(y) = \frac{1}{4}(y_1 - y_2)$, $Y(Q_\mu) = \operatorname{argmin}\{\frac{1}{4}(y_1 - y_2) + y_1^2 + y_2^2 + \frac{1}{2} : y \in A(C)\} = \{(-\frac{1}{8}, \frac{1}{8})\}$, and $\sigma(y) = \{(u, u - y_2, u - y_1) : u \in \mathbb{R}\} \cap C$ for $y \in A(C)$. Hence, $Y(Q_\mu)$ is a singleton, but $\psi(Q_\mu) = \sigma(Y(Q_\mu))$ forms a line segment. Moreover, σ is Hausdorff Lipschitzian on $A(C)$ and Theorem 2.3 applies.

Example 2.6. In (1.1)–(1.3) let $m = 2$, $s = 1$, $g(x) \equiv 0$, $A = (1, 0)$, $q = (1, 1)$, $W = (1, -1)$, μ be the uniform distribution on $[-\frac{1}{2}, \frac{1}{2}]$, and $C = \{(x_1, x_2) \in \mathbb{R}^2 : x_2^2 \le x_1\}$. Then we have

$$\tilde{Q}(t) = |t|, Q_\mu(y) = \int_{\mathbb{R}} |\omega - y| \mu(\mathrm{d}\omega) = \begin{cases} y^2 + \frac{1}{4}, & y \in [-\frac{1}{2}, \frac{1}{2}], \\ |y| & \text{otherwise}, \end{cases}$$

$\psi(Q_\mu) = \{(0, 0)\}$, and Q_μ is strongly convex on $(-\frac{1}{2}, \frac{1}{2})$. For $y \in A(C) = \mathbb{R}_+$ we have

$$\sigma(y) = \{x \in C : Ax = y\} = \{(y, x_2) \in \mathbb{R}^2 : x_2^2 \le y\} = \{y\} \times [-\sqrt{y}, \sqrt{y}],$$

and, hence $d((0, 0), \sigma(y)) = y$ for all $y \in \mathbb{R}_+$. Thus Theorem 2.3 applies for $\bar{x} = (0, 0)$.

Example 2.9 shows that Theorem 2.3 gets lost if Q_μ fails to be strongly convex on some neighborhood of $A\psi(Q_\mu)$. Our next result establishes a sufficient condition for the uniform quadratic growth near solution sets.

THEOREM 2.7. *Let $Q_\mu \in K_C$, let $\psi(Q_\mu)$ be nonempty and bounded, and let Q_μ be strongly convex on some open convex neighborhood U of $A\psi(Q_\mu)$. Assume that there exists a constant $L > 0$ such that*

$$d_H(\sigma(y), \sigma(\tilde{y})) \le L\|y - \tilde{y}\| \quad \textit{for all} \quad y, \tilde{y} \in A(C),$$

and for each $r > 0$ there exists a constant $\eta(r) > 0$ such that

$$g(x) \ge \pi(Ax) + \eta(r) d(x, \sigma(Ax))^2 \quad \textit{for all} \quad x \in C \cap B(0, r).$$

Then, for some open, bounded neighborhood V of $\psi(Q_\mu)$ and each $v \in T^r(K_C; Q_\mu)$, there exist constants $c > 0$ and $\delta > 0$ such that the following uniform growth condition holds:

$$g(x) + (Q_\mu + tv)(Ax) \ge \varphi(Q_\mu + tv) + c d(x, \psi(Q_\mu + tv))^2$$

for all $x \in C \cap V$ and $t \in [0, \delta)$.

Proof. Let $v \in T^r(K_C, Q_\mu)$, and let V be an open, bounded subset of $\mathbb{R}^m$ such that $\psi(Q_\mu) \subset V$ and $A(V) \subseteq U$. As in Theorem 2.3 we choose $\delta > 0$ such that $\emptyset \ne \psi(Q_\mu + tv) \subset V$ and, in addition, that $Q_\mu + tv$ is strongly convex on U for all $t \in [0, \delta)$ (with a uniform constant $\kappa > 0$). For each $t \in [0, \delta)$ Proposition 2.1 then yields that $\psi(Q_\mu + tv) = \sigma(y_t)$, where y_t is the unique minimizer of the strongly convex function $\pi + Q_\mu + tv$ on $A(C)$ and, moreover, we have $\kappa\|y - y_t\|^2 \le \pi(y) + (Q_\mu + tv)(y) - \varphi(Q_\mu + tv)$ for all $y \in A(C) \cap U$. Now, we choose $r > 0$ such that

$V \subseteq B(0,r)$ and continue for each $x \in C \cap V$ and $t \in [0,\delta)$ as follows:

$$\begin{aligned} d(x,\psi(Q_\mu + tv))^2 &= d(x,\sigma(y_t))^2 \\ &\le 2(d(x,\sigma(Ax))^2 + d_H(\sigma(Ax),\sigma(y_t))^2) \\ &\le 2\left(\frac{1}{\eta(r)}(g(x)-\pi(Ax)) + L^2\|Ax-y_t\|^2\right) \\ &\le 2\left(\frac{1}{\eta(r)}(g(x)-\pi(Ax)) + \frac{L^2}{\kappa}(\pi(Ax)+(Q_\mu+tv)(Ax)-\varphi(Q_\mu+tv))\right) \\ &\le 2\max\left\{\frac{1}{\eta(r)},\frac{L^2}{\kappa}\right\}(g(x)+(Q_\mu+tv)(Ax)-\varphi(Q_\mu+tv)). \end{aligned}$$

Putting $c^{-1} = 2\max\{\frac{1}{\eta(r)},\frac{L^2}{\kappa}\}$ completes the proof. □

The following examples show that the quadratic growth condition gets lost even for the original problem, i.e., $t = 0$, if either the Lipschitz condition for σ or the strong convexity property for Q_μ are violated.

Example 2.8. Consider again the set-up of Example 2.6. Since it holds that $d_H(\sigma(y),\sigma(0)) = (y^2+y)^{\frac{1}{2}}$ for all $y \in \mathbb{R}_+ = A(C)$, σ is *not* Hausdorff Lipschitzian on $A(C)$. Supposed there exists a neighborhood V of $\psi(Q_\mu) = \{(0,0)\}$ and a constant $\varrho > 0$ such that the growth condition

$$\varrho\, d(x,\psi(Q_\mu))^2 = \varrho\|x\|^2 \le Q_\mu(x_1) - \varphi(Q_\mu) = x_1^2 \quad \text{for all} \quad x \in C \cap V$$

is satisfied. Since the sequence $((\frac{1}{n},\frac{1}{\sqrt{n}}))$ belongs to $C \cap V$ for sufficiently large $n \in \mathbb{N}$, this would imply $\varrho(\frac{1}{n^2}+\frac{1}{n}) \le \frac{1}{n^2}$ for large n, which is a contradiction.

Example 2.9. In (1.1)–(1.3) let $m = s = 1$, $g(x) \equiv 0$, $A = 1$, $C = \mathbb{R}$, $q = (1,1)$, $W = (1,-1)$, and let μ be the probability distribution on $\mathbb{R}$ having the density

$$f_\mu(z) = \begin{cases} |z|, & z \in [-1,1], \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$Q_\mu(y) = \int_{\mathbb{R}} |\omega - y|\mu(\mathrm{d}\omega) = \begin{cases} \frac{1}{3}|y|^3 + \frac{2}{3}, & y \in [-1,1] \\ |y|, & \text{otherwise,} \end{cases}$$

$\psi(Q_\mu) = \{0\}$, and there is *no* neighborhood of $\psi(Q_\mu)$ where Q_μ is strongly convex. It is clear that the quadratic growth condition fails to hold, since the inequality $\varrho x^2 \le Q_\mu(x) - \varphi(Q_\mu) = \frac{1}{3}|x|^3$ cannot be true for some $\varrho > 0$ and all x belonging to some neighborhood of $x = 0$.

With the linear function $v(x) = -x$ ($x \in \mathbb{R}$) we obtain for all $t \in [0,1]$ that $\psi(Q_\mu + tv) = \{\sqrt{t}\}$ (cf. Example 3.7). Hence, the lower Lipschitz property of ψ fails to hold as well. Since the strong convexity and later also the strict convexity of the expected recourse function Q_μ (on certain convex subsets of $\mathbb{R}^s$) form essential conditions in most of our results, we recall a theorem (Theorem 2.2 in [30]) that provides a handy criterion to check these properties for problem (1.1)–(1.3).

PROPOSITION 2.10. *Let $V \subset \mathbb{R}^s$ be open convex, and assume* (A1) *and* (A3). *Consider the following conditions.*

(A2)* $\operatorname{int} M_D = \{t \in \mathbb{R}^s : W^T t < q\} \neq \emptyset$.

(A4) μ *is absolutely continuous on* $\mathbb{R}^s$.

(A4)* μ *satisfies* (A4) *and there exist a density* f_μ *for* μ *and a constant* $\delta > 0$ *such that* $f_\mu(z) \geq \delta$ *whenever* $d(z, V) \leq \delta$.

Then (A2)* *and* (A4) *imply that* Q_μ *is strictly convex on* V *if* V *is a subset of the support of* μ, *and* (A2)* *and* (A4)* *imply that* Q_μ *is strongly convex on* V.

In addition, it is shown in [30] that under (A1)–(A4) the condition (A2)* is also necessary for the strict convexity of Q_μ. For extended simple recourse models (i.e., $W = (H, -H)$ with some nonsingular (s, s)-matrix H) (A2)* is equivalent to $q^+ + q^- > 0$ (componentwise), where $q = (q^+, q^-)$ and $q^+, q^- \in \mathbb{R}^s$. This may be used to check strict or strong convexity properties in the Examples 2.6 and 2.9.

3. Directional derivatives of optimal values. In this section, we study first- and second-order directional differentiability properties of the optimal value function φ on its domain K_C. We begin with the first-order analysis and show that φ as a mapping from K_C to the extended reals is Hadamard directionally differentiable at some given expected recourse function $Q_\mu \in K_C$. Here K_C is regarded as a subset of $C^0(\mathbb{R}^s)$. Recall that φ is Hadamard directionally differentiable at Q_μ on K_C iff for all sequences (v_n) converging to some v in $C^0(\mathbb{R}^s)$ and all sequences $t_n \to 0+$ such that the elements $Q_\mu + t_n v_n$ belong to K_C the limit

$$\varphi'(Q_\mu; v) = \lim_{n\to\infty} \frac{1}{t_n}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu))$$

exists. Since the condition $Q_\mu + t_n v_n \in K_C$ means that $v_n = \frac{1}{t_n}(Q_n - Q_\mu)$ for some $Q_n \in K_C$, the limit v belongs to the tangent cone $T(K_C; Q_\mu)$ to K_C at Q_μ in $C^0(\mathbb{R}^s)$. In [35], [36] this property is also called Hadamard directional differentiability tangentially to K_C.

PROPOSITION 3.1. *Let $Q_\mu \in K_C$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Then φ is Hadamard directionally differentiable at Q_μ on K_C, and it holds for all $v \in T(K_C; Q_\mu)$ that*

$$\varphi'(Q_\mu; v) = \min\{v(Ax) : x \in \psi(Q_\mu)\}.$$

If, in addition, Q_μ is strictly convex on some open convex neighborhood of $A\psi(Q_\mu)$, we have

$$\varphi'(Q_\mu; v) = v(\bar{y}), \quad \text{where} \quad \{\bar{y}\} = A\psi(Q_\mu).$$

Proof. Arguing similarly as in the proof of Proposition 2.1 in [26] there exists a neighborhood $\mathcal{N}$ of Q_μ in $C^0(\mathbb{R}^s)$ such that $\psi(Q)$ is nonempty for all $Q \in K_C \cap \mathcal{N}$. Let (t_n) and (v_n) be sequences such that $t_n \to 0+$, $v_n \to v$ in $C^0(\mathbb{R}^s)$, and $Q_\mu + t_n v_n$ belongs to K_C for all $n \in \mathbb{N}$. Then $Q_\mu + t_n v_n \in K_C \cap \mathcal{N}$ for sufficiently large $n \in \mathbb{N}$. Let $x_n \in \psi(Q_\mu + t_n v_n)$ for those $n \in \mathbb{N}$. Since ψ is Berge upper-semicontinuous at Q_μ [26], the sequence (x_n) has an accumulation point $x \in \psi(Q_\mu)$, and we obtain

$$\begin{aligned}\limsup_{n\to\infty} &\frac{1}{t_n}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu)) \\ &\geq \limsup_{n\to\infty} \frac{1}{t_n}(g(x_n) + (Q_\mu + t_n v_n)(Ax_n) - g(x_n) - Q_\mu(Ax_n)) \geq v(Ax),\end{aligned}$$

where the last inequality follows from the uniform convergence of (v_n) to v on bounded subsets of $\mathbb{R}^s$. In order to show the reverse inequality for lim inf, let $x \in \psi(Q_\mu)$. Then

$$\liminf_{n\to\infty} \frac{1}{t_n}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu)) \\ \leq \liminf_{n\to\infty} \frac{1}{t_n}(g(x) + (Q_\mu + t_n v_n)(Ax) - g(x) - Q_\mu(Ax)) = v(Ax).$$

This completes the proof of the first part. The second part is an immediate conclusion, since $A\psi(Q_\mu)$ is a singleton whenever Q_μ is strictly convex on some of its open, convex neighborhoods. □

The preceding result can also be proved by using the methodology of Theorem 6.4.1 in [28]. There the compactness of the constraint set is assumed, and Gateaux directional differentiability of φ at Q_μ together with its Lipschitz continuity is shown. Here we prefer a direct two-sided argument, which will also be used in the subsequent second-order analysis of φ. Namely, we will first derive an upper bound for the second-order Hadamard directional derivative of φ at some $Q_\mu \in K_C$, where K_C is equipped with the $C^{0,1}$ topology. Second, we identify conditions implying that the upper bound coincides with the Gateaux directional derivative of φ at Q_μ for all directions taken from $T^r(K_C; Q_\mu)$.

LEMMA 3.2. *Let $y \in \mathbb{R}^s$, $Q_\mu \in K_C$, $t_n \to 0+$, (Q_n) be a sequence in K_C such that $v_n := \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^{0,1}(\mathbb{R}^s)$, and let (ξ_n) be a sequence converging to ξ in $\mathbb{R}^s$. Then we have* $\limsup_{n\to\infty} \frac{1}{t_n}(v_n(y + t_n\xi_n) - v_n(y)) \leq \max_{\zeta\in\partial v(y)} \langle \zeta, \xi\rangle$.

Proof. Each function v_n is locally Lipschitzian on $\mathbb{R}^s$ and, hence, Lebourg's mean value theorem for Clarke's subdifferential [9] implies the existence of elements $\tilde{y}_n$ belonging to the segments $[y, y + t_n\xi_n]$ such that

$$\frac{1}{t_n}(v_n(y + t_n\xi_n) - v_n(y)) \in \{\langle \zeta, \xi_n\rangle : \zeta \in \partial v_n(\tilde{y}_n)\}.$$

The convergence $v_n \to v$ in $C^{0,1}(\mathbb{R}^s)$ implies that

$$\sup\{\|\zeta\| : \zeta \in \partial(v_n - v)(y), \|y\| \leq r\} \underset{n\to\infty}{\longrightarrow} 0$$

holds for any $r > 0$. This yields

$$d_H(\partial v_n(\tilde{y}_n), \partial v(\tilde{y}_n)) \leq \sup\{\|\zeta\| : \zeta \in \partial(v_n - v)(\tilde{y}_n)\} \underset{n\to\infty}{\longrightarrow} 0.$$

Here d_H denotes the Hausdorff distance, and the inequality is a consequence of general properties of the subdifferential (cf. Lemma 2.1 in [27]). Hence, there exist elements $\tilde{\zeta}_n$ belonging to $\partial v(\tilde{y}_n)$ such that

$$\frac{1}{t_n}(v_n(y + t_n\xi_n) - v_n(y)) \leq \|\xi_n\| d_H(\partial v_n(\tilde{y}_n), \partial v(\tilde{y}_n)) + \langle \tilde{\zeta}_n, \xi_n\rangle$$

and, for some $\tilde{\zeta} \in \partial v(y)$,

$$\limsup_{n\to\infty} \frac{1}{t_n}(v_n(y + t_n\xi_n) - v_n(y)) \leq \limsup_{n\to\infty} \langle \tilde{\zeta}_n, \xi_n\rangle = \langle \tilde{\zeta}, \xi\rangle \leq \max_{\zeta\in\partial v(y)} \langle \zeta, \xi\rangle,$$

where the upper semicontinuity of $\partial v(\cdot)$ is used. This completes the proof. □

PROPOSITION 3.3. *Let $Q_\mu \in K_C$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Let g be twice continuously differentiable, and let Q_μ be strictly convex on*

some open convex neighborhood of $A\psi(Q_\mu)$ and twice continuously differentiable at $\bar{y}$, where $\{\bar{y}\} = A\psi(Q_\mu)$. Let $\bar{x} \in \psi(Q_\mu)$, $t_n \to 0+$, and (Q_n) be a sequence in K_C such that $v_n := \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^{0,1}(\mathbb{R}^s)$. Then

$$\begin{aligned}\limsup_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n)) \\ \leq \inf\{\langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x}), \xi, \xi\rangle \\ + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \max_{\zeta\in\partial v(\bar{y})} \langle \zeta, A\xi\rangle : \xi \in S(\bar{x}), z \in T^2(C; \bar{x}, \xi)\},\end{aligned}$$

where $S(\bar{x}) := \{\xi \in T(C; \bar{x}) : \langle \nabla g(\bar{x}), \xi\rangle + \langle \nabla Q_\mu(\bar{y}), A\xi\rangle = 0\}$, $T(C; \bar{x})$ is the tangent cone to C at $\bar{x}$, and $T^2(C; \bar{x}, \xi)$ is the second-order tangent set to C at $\bar{x}$ in direction ξ.

Proof. Let $\xi \in S(\bar{x})$ and $z \in T^2(C; \bar{x}, \xi)$. Then there exists a sequence (z_n) such that $z_n \to z$ and $\bar{x} + t_n\xi + t_n^2 z_n \in C$ for all $n \in \mathbb{N}$. Using Proposition 3.1, this allows for the following estimate:

$$\begin{aligned}&\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n) \\ &\quad\leq g(\bar{x} + t_n\xi + t_n^2 z_n) + Q_\mu(A(\bar{x} + t_n\xi + t_n^2 z_n)) + t_n v_n(A(\bar{x} + t_n\xi + t_n^2 z_n)) \\ &\qquad - g(\bar{x}) - Q_\mu(A\bar{x}) - t_n v_n(A\bar{x}) \\ &\quad= [g(\bar{x} + t_n\xi + t_n^2 z_n) - g(\bar{x}) - t_n\langle \nabla g(\bar{x}), \xi\rangle] \\ &\qquad + [Q_\mu(A(\bar{x} + t_n\xi + t_n^2 z_n)) - Q_\mu(A\bar{x}) - t_n\langle \nabla Q_\mu(A\bar{x}), A\xi\rangle] \\ &\qquad + t_n[v_n(A(\bar{x} + t_n\xi + t_n^2 z_n)) - v_n(A\bar{x})].\end{aligned}$$

After dividing by t_n^2 and using Lemma 3.2, the limes superior as $n \to \infty$ of the right-hand side can be bounded above by

$$\langle \nabla g(\bar{x}), z\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \langle \nabla Q_\mu(A\bar{x}), Az\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(A\bar{x})A\xi, A\xi\rangle + \max_{\zeta\in\partial v(A\bar{x})} \langle \zeta, A\xi\rangle.$$

Taking the infimum on the right-hand side yields the assertion. ∎

We notice that the upper second-order Hadamard directional derivative $\limsup_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n))$ is nonpositive, since φ is concave on K_C and, hence, the inequality $\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) = \varphi(Q_n) - \varphi(Q_\mu) \leq \varphi'(Q_\mu; Q_n - Q_\mu) = t_n\varphi'(Q_\mu; v_n)$ is valid. We also note that the upper bound is nonpositive, since $(0, 0)$ belongs to $S(\bar{x}) \times T^2(C; \bar{x}, 0) = S(\bar{x}) \times T(C; \bar{x})$. Next we consider particular perturbations Q_n of Q_μ, namely, $Q_n := Q_\mu + \lambda t_n(Q - Q_\mu)$ for some $Q \in K_C$, $\lambda > 0$, and sufficiently large $n \in \mathbb{N}$. Then $v_n = \lambda(Q - Q_\mu) \in T^r(K_C; Q_\mu)$. The next result provides conditions implying that the second-order (Gateaux) directional derivative exists and coincides with the upper bound of the previous proposition. To state the result we need the notion of *second-order regularity* (cf. [6]). The constraint set C is called second-order regular at $\bar{x} \in C$ if for any direction $\xi \in T(C; \bar{x})$ and any sequence $x_n \in C$ of the form $x_n = \bar{x} + t_n\xi + t_n^2 r_n$ where, $t_n \to 0+$ and r_n being a sequence in $\mathbb{R}^m$ satisfying $t_n r_n \to 0$, it holds that $\lim_{n\to\infty} d(r_n, T^2(C; \bar{x}, \xi)) = 0$. For example, C is second-order regular at $\bar{x} \in C$ if $0 \in T^2(C; \bar{x}, \xi)$ for every $\xi \in T(C; \bar{x})$ (cf. [6]). In particular, a polyhedral (convex) set C is second-order regular at any $\bar{x} \in C$.

THEOREM 3.4. *Let $Q_\mu \in K_C$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Let g be twice continuously differentiable, and let Q_μ be strictly convex on some*

open convex neighborhood of $A\psi(Q_\mu)$ and twice continuously differentiable at $\bar{y}$, where $\{\bar{y}\} = A\psi(Q_\mu)$. Let $\bar{x} \in \psi(Q_\mu)$, $v \in T^r(K_C; Q_\mu)$, and assume that

(i) *$d(\bar{x}, \psi(Q_\mu + tv)) = O(t)$ for small $t > 0$, and*

(ii) *C is second-order regular at $\bar{x}$.*

Then the second-order Gateaux directional derivative of φ at Q_μ in direction v exists, and it holds that

$$\varphi''(Q_\mu; v) = \lim_{t\to 0+} \frac{1}{t^2}(\varphi(Q_\mu + t_n v) - \varphi(Q_\mu) - t\varphi'(Q_\mu; v))$$

$$= \inf\left\{\frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + v'(\bar{y}; A\xi) + b(\xi) : \xi \in S(\bar{x})\right\}, \tag{3.1}$$

where $b(\xi) = \inf\{\langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle : z \in T^2(C; \bar{x}, \xi)\}$ is nonnegative and convex on $S(\bar{x})$. Moreover, the infimum in (3.1) is attained at some $\bar{\xi} \in S(\bar{x})$ having the property that $\varphi''(Q_\mu; v) = \frac{1}{2}v'(\bar{y}; A\bar{\xi}) + \frac{1}{2}b(\bar{\xi})$.

(Here $S(\bar{x})$ and $T^2(C; \bar{x}, \xi)$ are defined as in the previous result, $v'(\bar{y}; \eta)$ is the directional derivative of v at $\bar{y}$ in direction η, and $O(t)$ denotes a real quantity such that $\frac{1}{t}|O(t)|$ is bounded as $t \to 0+$.)

Proof. (i) implies that there exist constants $L > 0$, $\delta > 0$, and elements $x(t) \in \psi(Q_\mu + tv)$ such that $\|x(t) - \bar{x}\| \le Lt$ for all $t \in (0, \delta)$. Now take a sequence (t_n) tending to $0+$ in such a way that

$$\liminf_{t\to 0+} \frac{1}{t^2}(\varphi(Q_\mu + tv) - \varphi(Q_\mu) - t\varphi'(Q_\mu; v))$$
$$= \lim_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v))$$

and that $\xi_n := \frac{1}{t_n}(x(t_n) - \bar{x}) \underset{n\to\infty}{\longrightarrow} \bar{\xi}$. The latter is possible since $\|\frac{1}{t_n}(x(t_n) - \bar{x})\| \le L$ for $n \in \mathbb{N}$ sufficiently large. Then $\bar{\xi} \in T(C; \bar{x})$ and Proposition 3.1 yields

$$v(A\bar{x}) = \varphi'(Q_\mu; v) = \lim_{n\to\infty} \frac{1}{t_n}(\varphi(Q_\mu + t_n v) - \varphi(Q_\mu))$$
$$= \lim_{n\to\infty} \frac{1}{t_n}(g(\bar{x} + t_n\xi_n) + (Q_\mu + t_n v)(A(\bar{x} + t_n\xi_n)) - g(\bar{x}) - Q_\mu(A\bar{x}))$$
$$= \langle \nabla g(\bar{x}), \bar{\xi}\rangle + \langle \nabla Q_\mu(A\bar{x}), A\bar{\xi}\rangle + v(A\bar{x}).$$

This implies $\bar{\xi} \in S(\bar{x})$. We put $r_n = \frac{1}{t_n}(\xi_n - \bar{\xi})$ and $x_n = x(t_n) = \bar{x} + t_n\bar{\xi} + t_n^2 r_n$. By expanding g and Q_μ and using Proposition 3.1, we obtain

$$\varphi(Q_\mu + t_n v) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v)$$
$$= g(x_n) + Q_\mu(Ax_n) + t_n v(Ax_n) - g(\bar{x}) - Q_\mu(A\bar{x}) - t_n v(A\bar{x})$$
$$= \langle \nabla g(\bar{x}), x_n - \bar{x}\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})(x_n - \bar{x}), x_n - \bar{x}\rangle$$
$$+ \langle \nabla Q_\mu(A\bar{x}), Ax_n - \bar{x})\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(A\bar{x})(A(x_n - \bar{x})), A(x_n - \bar{x})\rangle$$
$$+ t_n(v(Ax_n) - v(A\bar{x})) + o(\|x_n - \bar{x}\|^2)$$
$$= t_n^2(\langle \nabla g(\bar{x}), r_n\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle) + t_n^2(\langle \nabla Q_\mu(A\bar{x}), Ar_n)\rangle$$
$$+ \frac{1}{2}\langle \nabla^2 Q_\mu(A\bar{x})A\bar{\xi}, A\bar{\xi}\rangle) + t_n(v(Ax_n) - v(A\bar{x})) + o(t_n^2).$$

Here we used that $o(\|x_n - \bar{x}\|^2) = o(t_n^2)$, where $o(t^k)$ denotes a real quantity having the property $\frac{1}{t^k} o(t) \to 0$ as $t \to 0+$ $(k \in \mathbb{N})$.

Since C is second-order regular at $\bar{x}$, there exists a sequence $z_n \in T^2(C; \bar{x}, \bar{\xi})$ such that $\lim_{n\to\infty} \|r_n - z_n\| = 0$, and we get from the previous chain of equalities

$$\begin{aligned}
&\frac{1}{t_n^2}(\varphi(Q_\mu + t_n v) - \varphi(Q_\mu) - t_n \varphi'(Q_\mu; v)) \\
&\quad = \langle \nabla g(\bar{x}), z_n \rangle + \langle \nabla Q_\mu(\bar{y}), A z_n \rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle \\
&\qquad + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\bar{\xi}, A\bar{\xi}\rangle + \frac{1}{t_n}(v(\bar{y} + t_n A\xi_n) - v(\bar{y})) + o(1) \\
&\quad \geq b(\bar{\xi}) + \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\bar{\xi}, A\bar{\xi}\rangle + \frac{1}{t_n}(v(\bar{y} + t_n A\xi_n) - v(\bar{y})) + o(1).
\end{aligned}$$

Using the fact that v is Hadamard directionally differentiable and Clarke regular [9], i.e., $v'(\bar{y}; \eta) = \max_{\zeta \in \partial v(\bar{y})} \langle \zeta, \eta \rangle$, we obtain

$$\begin{aligned}
&\liminf_{t\to 0+} \frac{1}{t^2}(\varphi(Q_\mu + tv) - \varphi(Q_\mu) - t\varphi'(Q_\mu; v)) \\
&\qquad \geq \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\bar{\xi}, A\bar{\xi}\rangle + v'(\bar{y}; A\bar{\xi}) + b(\bar{\xi}) \\
&\qquad \geq \inf\left\{ \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\xi, A\xi\rangle + v'(\bar{y}; A\xi) + b(\xi) : \xi \in S(\bar{x}) \right\}.
\end{aligned}$$

Proposition 3.3 implies that this lower bound for $\liminf_{t\to 0+}$ is also an upper bound for $\limsup_{t\to 0+}$. Hence, the limit $\lim_{t\to 0+} \frac{1}{t^2}(\varphi(Q_\mu + tv) - \varphi(Q_\mu) - t\varphi'(Q_\mu; v))$ exists and is equal to the infimum subject to $\xi \in S(\bar{x})$. Moreover, this infimum is attained at $\bar{\xi} \in S(\bar{x})$.

The nonnegativity of b is due to the fact that the necessary optimality condition for (1.1) at $\bar{x}$ yields

$$\langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle \geq 0 \quad \text{for all} \quad z \in T^2(C; \bar{x}, \xi), \xi \in S(\bar{x}).$$

The convexity of b follows from the property $T^2(C; \bar{x}, \lambda\xi + (1-\lambda)\tilde{\xi}) \supseteq \lambda T^2(C; \bar{x}, \xi) + (1-\lambda)T^2(C; \bar{x}, \tilde{\xi})$ for all $\xi, \tilde{\xi} \in T(C; \bar{x})$, and $\lambda \in [0, 1]$.

For the remainder of the proof we put $a(\xi) := v'(\bar{y}; A\xi)$ and

$$B(\xi) := \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\xi, A\xi\rangle \quad \text{for all} \quad \xi \in \mathbb{R}^m.$$

Since $S(\bar{x})$ is a (convex) cone, we have $S(\bar{x}) = \lambda S(\bar{x})$ for any $\lambda > 0$. Moreover, it holds that $T^2(C; \bar{x}, \lambda\bar{\xi}) = \lambda T^2(C; \bar{x}, \bar{\xi})$ and thus that $b(\lambda\bar{\xi}) = \lambda b(\bar{\xi})$ for any $\lambda > 0$. Hence, we obtain

$$\begin{aligned}
0 \leq f(\lambda) &:= B(\lambda\bar{\xi}) + a(\lambda\bar{\xi}) + b(\lambda\bar{\xi}) - B(\bar{\xi}) - a(\bar{\xi}) - b(\bar{\xi}) \\
&= \lambda^2 B(\bar{\xi}) + (\lambda - 1)(a(\bar{\xi}) + b(\bar{\xi})) - B(\bar{\xi}) \quad \text{for all} \quad \lambda > 0.
\end{aligned}$$

In the case of $B(\bar{\xi}) > 0$, the quadratic function f vanishes at $\lambda = 1$ with the property $f'(1) = 2B(\bar{\xi}) + a(\bar{\xi}) + b(\bar{\xi}) = 0$, and the final assertion is shown. If $B(\bar{\xi}) = 0$, the fact that $0 \leq f(\lambda) = (\lambda - 1)(a(\bar{\xi}) + b(\bar{\xi}))$ holds for any $\lambda > 0$ implies $a(\bar{\xi}) + b(\bar{\xi}) = 0$. Thus $\varphi''(Q_\mu; v) = 0 = \frac{1}{2}(a(\bar{\xi}) + b(\bar{\xi}))$, and the proof is complete. □

The theorem extends our earlier work in [12], where essentially polyhedrality of C is assumed. Compared to [12], the additional term $b(.)$ enters the formula for $\varphi''(Q_\mu; v)$. The convex function $b(.)$ reflects second-order properties of the constraint set C and vanishes if C is polyhedral. Next we state a more handy criterion implying that $\varphi''(Q_\mu; v)$ exists for any direction $v \in T^r(K_C; Q_\mu)$.

COROLLARY 3.5. *Let $Q_\mu \in K_C$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Let g be twice continuously differentiable, and let Q_μ be strongly convex on some open convex neighborhood of $A\psi(Q_\mu)$ and twice continuously differentiable at $\bar{y}$, where $\{\bar{y}\} = A\psi(Q_\mu)$. Let $\bar{x} \in \psi(Q_\mu)$ and assume that*

(i)′ *there exist a constant $L > 0$ and a neighborhood U of $\bar{y}$ such that*
$d(\bar{x}, \sigma(y)) \le L\|\bar{y} - y\|$ *for all $y \in A(C) \cap U$, where*
$\sigma(y) := \operatorname{argmin}\{g(x) : x \in C, Ax = y\}$, *$y \in A(C)$, and*

(ii) *C is second-order regular at $\bar{x}$.*

Then the second-order Gateaux directional derivative of φ at Q_μ exists for any direction $v \in T^r(K_C; Q_\mu)$, and the formula for $\varphi''(Q_\mu; v)$ in Theorem 3.4 holds true. Moreover, conditions (i)′ *and* (ii) *are satisfied for any $\bar{x} \in \psi(Q_\mu)$ if C is polyhedral and g is linear or (convex) quadratic.*

Proof. Let $v \in T^r(K_C; Q_\mu)$. Theorem 2.3 then says that there exist constants $\hat{L} > 0$, $\delta > 0$, and $r > 0$ such that

$$d(\bar{x}, \psi(Q_\mu + tv)) \le \hat{L}\|v\|_{L,r}\, t \quad \text{whenever} \quad \|v\|_{L,r}\, t < \delta.$$

Hence, the strong convexity of Q_μ and condition (i)′ imply that condition (i) of the previous theorem is satisfied and that the first part of the assertion is shown. If C is polyhedral and g is linear or (convex) quadratic, (ii) is satisfied and Proposition 2.2 implies (i)′ to hold for any $\bar{x} \in \psi(Q_\mu) = \sigma(\bar{y})$. □

Let us consider two illustrative examples to provide some insight into the benefit and limits of the previous results.

Example 3.6. We revisit Example 2.6 and know that the general assumptions of Corollary 3.5 and condition (i)′ are satisfied for $\bar{x} = (0, 0)$. Furthermore, it holds that $T(C; \bar{x}) = \mathbb{R}_+ \times \mathbb{R}$ and

$$T^2(C; \bar{x}, \xi) = \begin{cases} \mathbb{R}^2, & \xi_1 > 0, \\ \{x_1 \in \mathbb{R} : x_1 \ge \xi_2^2\} \times \mathbb{R}, & \xi_1 = 0, \end{cases} \quad \text{for any } \xi \in T(C; \bar{x}).$$

Moreover, C is second-order regular at $\bar{x}$ (as can be seen from Proposition 4.1 in [6]) and it holds that $b(\xi) = 0$ for all $\xi \in \mathbb{R}^2$. Hence, Corollary 3.5 implies that $\varphi''(Q_\mu; v)$ exists for any $v \in T^r(K_C; Q_\mu)$ and that $\varphi''(Q_\mu; v) = \frac{1}{2} v'(0, \bar{\xi}_1)$, where $\bar{\xi} = (\bar{\xi}_1, \bar{\xi}_2) \in \operatorname{argmin}\{\xi_1^2 + v'(0, \xi_1) : (\xi_1, \xi_2) \in \mathbb{R}_+ \times \mathbb{R}\}$.

Example 3.7. Here we revisit Example 2.9 and have

$$Q_\mu(y) = \frac{1}{3}|y|^3 + \frac{2}{3} \quad \text{for all } |y| \le 1, \text{ and } \psi(Q_\mu) = \{0\},\ \varphi(Q_\mu) = \frac{2}{3}.$$

For the function $v(x) = -x$ ($x \in \mathbb{R}$) and $t \in [0, 1)$ we obtain

$$\varphi(Q_\mu + tv) = \inf\{Q_\mu(x) - tx : x \in \mathbb{R}\} = \frac{2}{3}(1 - t^{\frac{3}{2}}),$$
$$\psi(Q_\mu + tv) = \operatorname{argmin}\{Q_\mu(x) - tx : x \in \mathbb{R}\} = \{\sqrt{t}\}.$$

Then $\varphi'(Q_\mu; v) = 0$ and $\frac{1}{t^2}(\varphi(Q_\mu + tv) - \varphi(Q_\mu) - \varphi'(Q_\mu; v)) = -\frac{2}{3} t^{-\frac{1}{2}}$. Hence, φ has no second-order directional derivative at Q_μ in direction v. Note that there is no neighborhood of $\bar{x} = 0$ where Q_μ is strongly convex.

Finally, we aim at showing that φ is even second-order Hadamard directionally differentiable at Q_μ when equipping K_C with a suitable topology. To this end we need a certain counterpart of Lemma 3.2 for the corresponding limes inferior. Since such a bound does not exist for nonsmooth functions, it is a natural idea to consider the space $C^1(\mathbb{R}^s)$, to restrict φ to the subset $K_C \cap C^1$, and to equip $K_C \cap C^1$ with the C^1 topology. Then we are able to show that the assumptions of Corollary 3.5 imply the second-order Hadamard directional differentiability of φ at Q_μ.

THEOREM 3.8. *Let $Q_\mu \in K_C \cap C^1$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Let g be twice continuously differentiable, and let Q_μ be strongly convex on some open convex neighborhood of $A\psi(Q_\mu)$ and twice continuously differentiable at $\bar{y}$, where $\{\bar{y}\} = A\psi(Q_\mu)$. Let $\bar{x} \in \psi(Q_\mu)$ and assume the conditions* (i)′ *and* (ii) *of Corollary* 3.5 *to hold. Then the second order Hadamard directional derivative of φ at Q_μ exists in any direction v belonging to the tangent cone $T(K_C \cap C^1; Q_\mu)$ in $C^1(\mathbb{R}^s)$, i.e., for any such v, and all sequences $t_n \to 0+$ and (Q_n) in K_C such that $v_n := \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^1(\mathbb{R}^s)$ the limit*

$$\varphi''(Q_\mu; v) = \lim_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n))$$

exists, and it holds that

$$\varphi''(Q_\mu; v) = \inf\left\{\frac{1}{2}\langle\nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle\nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle\nabla v(\bar{y}), A\xi\rangle + b(\xi) : \xi \in S(\bar{x})\right\}.$$

Proof. Let $v \in T(K_C \cap C^1; Q_\mu)$, $t_n \to 0+$, and (Q_n) be a sequence in K_C such that $v_n = \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^1(\mathbb{R}^s)$. Condition (i)′ together with Theorem 2.3 then imply that there exist constants $L > 0$, $r > 0$, $n_0 \in \mathbb{N}$, and elements $x_n \in \psi(Q_\mu + t_n v_n)$ such that

$$\|x_n - \bar{x}\| \le L t_n \|v_n\|_{L,r} \quad \text{for all } \ n \in \mathbb{N}, n \ge n_0.$$

Since the sequence (v_n) converges in $C^1(\mathbb{R}^s)$, the norms $\|v_n\|_{L,r}$ are uniformly bounded and we have $\|x_n - \bar{x}\| = O(t_n)$. As in the proof of Theorem 3.4 we select a subsequence of (t_n), which is again denoted by (t_n), tending to $0+$ such that $\xi_n := \frac{1}{t_n}(x_n - \bar{x}) \underset{n\to\infty}{\longrightarrow} \bar{\xi} \in S(\bar{x})$. Analogously, we obtain for sufficiently large n:

$$\begin{aligned}&\frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n))\\ &\quad \ge b(\bar{\xi}) + \frac{1}{2}\langle\nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle\nabla^2 Q_\mu(\bar{y})A\bar{\xi}, A\bar{\xi}\rangle + \frac{1}{t_n}(v_n(\bar{y} + t_n A\xi_n) - v_n(\bar{y})) + o(1).\end{aligned}$$

Using the mean value theorem for v_n we may continue with some $\bar{y}_n \in [\bar{y}, \bar{y} + t_n A\xi_n]$ as follows:

$$\begin{aligned}&\frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n))\\ &\ge \frac{1}{2}\langle\nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle\nabla^2 Q_\mu(\bar{y})A\bar{\xi}, A\bar{\xi}\rangle + \langle\nabla v_n(\bar{y}_n), A\xi_n\rangle + b(\bar{\xi}) + o(1).\end{aligned}$$

Arguing as in the proof of Theorem 3.4 and using $v_n \to v$ in $C^1(\mathbb{R}^s)$, we arrive at the estimate

$$\begin{aligned}&\liminf_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n\varphi'(Q_\mu; v_n))\\ &\ge \frac{1}{2}\langle\nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle\nabla^2 Q_\mu(\bar{y})A\bar{\xi}, A\bar{\xi}\rangle + \langle\nabla v(\bar{y}), A\bar{\xi}\rangle + b(\bar{\xi})\end{aligned}$$

and, using Proposition 3.3, we arrive at the desired result. □

Let us finally note that all minimization problems appearing as bounds or formulas for second-order directional derivatives represent convex programs. Those in the results Theorem 3.4, Corollary 3.5, and Theorem 3.8 have convex cone constraints, which are polyhedral if C is polyhedral. Moreover, the solution sets of the convex minimization problems in Theorem 3.4, Corollary 3.5, and Theorem 3.8 are nonempty. Indeed, we show next that these solution sets represent certain derivatives of the set-valued mapping ψ at the pair $(Q_\mu, \bar{x})$.

4. Differentiability of solution sets. It is well known that second-order differentiability properties of optimal values in perturbed optimization are intrinsic for establishing the differentiability of solutions (see, e.g., [8]). We also pursue this approach and derive conditions implying directional differentiability properties of the solution set mapping by exploiting the results of the previous section. Our first results in this direction concern Gateaux directional differentiability and complement Theorem 3.4 and its corollary.

THEOREM 4.1. *Assume that the general conditions on g, Q_μ, and C of Theorem* 3.4 *are satisfied. Let $\bar{x} \in \psi(Q_\mu)$, $v \in T^r(K_C; Q_\mu)$, and suppose the conditions* (i) *and* (ii) *of Theorem* 3.4 *to be satisfied. In addition, assume that*

(iii) *there exist a neighborhood V of $\psi(Q_\mu)$ and constants $c > 0$, $\delta > 0$ such that the uniform growth condition*

$$g(x) + (Q_\mu + tv)(Ax) \geq \varphi(Q_\mu + tv) + cd(x, \psi(Q_\mu + tv))^2$$

for all $x \in C \cap V$ and $t \in [0, \delta)$ is satisfied.

Then the Gateaux directional derivative of ψ at the pair $(Q_\mu, \bar{x})$ into direction v exists, and it holds that

$$\begin{aligned}\psi'(Q_\mu, \bar{x}; v) &= \lim_{t \to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - \bar{x}) \\ &= \operatorname{argmin}\left\{ \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + v'(\bar{y}; A\xi) + b(\xi) : \xi \in S(\bar{x}) \right\}.\end{aligned}$$

Proof. Let $M(\bar{x}; v)$ denote the solution set in the assertion. First we show that $\limsup_{t \to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - \bar{x}) \subseteq M(\bar{x}; v)$.

Let $\xi \in \limsup_{t \to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - \bar{x})$. Then there exists a sequence (t_n, ξ_n) converging to $(0+, \xi)$ such that $\xi_n \in \frac{1}{t_n}(\psi(Q_\mu + t_n v) - \bar{x})$ and, thus, $\bar{x} + t_n\xi_n \in \psi(Q_\mu + t_n v)$ for all $n \in \mathbb{N}$. Analogously to the proof of Theorem 3.4 we show that ξ belongs to $S(\bar{x})$ and that $\varphi''(Q_\mu; v) = \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + v'(\bar{y}; A\xi) + b(\xi)$. Hence $\xi \in M(\bar{x}; v)$.

In the second step we demonstrate that

$$M(\bar{x}; v) \subseteq \liminf_{t \to 0+} \frac{1}{t}(\psi(Q_\mu + tv) - \bar{x}),$$

or, equivalently, that it holds for any $\xi \in M(\bar{x}, v)$ that

$$\lim_{t \to 0} \frac{1}{t}\, d(\bar{x} + t\xi, \psi(Q_\mu + tv)) = 0.$$

Let $\xi \in M(\bar{x}; v)$ and (t_n) be a sequence with $t_n \to 0+$. We have to show that $\lim_{n \to \infty} \frac{1}{t_n}\, d(\bar{x} + t_n\xi, \psi(Q_\mu + t_n v)) = 0$.

Let $\varepsilon > 0$ be given, and let $z \in T^2(C; \bar{x}, \xi)$ be such that $\langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle \leq b(\xi) + \varepsilon$. Then there exists a sequence (z_n) converging to z with $x_n = \bar{x} + t_n\xi + t_n^2 z_n \in C$ for all $n \in \mathbb{N}$. Hence, it suffices to show that

$$\lim_{n\to\infty} \frac{1}{t_n}\, d(\bar{x} + t_n\xi + t_n^2 z_n, \psi(Q_\mu + t_n v)) = 0.$$

Condition (iii) implies the following estimate for all sufficiently large $n \in \mathbb{N}$:

$$\begin{aligned} c d(\bar{x} + t_n\xi + t_n^2 z_n, \psi(Q_\mu + t_n v))^2 \\ \leq g(\bar{x} + t_n\xi + t_n^2 z_n) + (Q_\mu + t_n v)(A(\bar{x} + t_n\xi + t_n^2 z_n)) - \varphi(Q_\mu + t_n v). \end{aligned}$$

By expanding g and Q_μ as in the proof of Theorem 3.4 and using the fact that ξ belongs to $S(\bar{x})$, we may express the right-hand side as

$$\begin{aligned} & t_n^2 \langle \nabla g(\bar{x}), z_n\rangle + \frac{1}{2} t_n^2 \langle \nabla^2 g(\bar{x})(\xi + t_n z_n), \xi + t_n z_n\rangle \\ & + t_n^2 \langle \nabla Q_\mu(\bar{y}), Az_n\rangle + \frac{1}{2} t_n^2 \langle \nabla^2 Q_\mu(\bar{y})(A(\xi + t_n z_n)), A(\xi + t_n z_n)\rangle \\ & - (\varphi(Q_\mu + t_n v) - \varphi(Q_\mu) - t_n \varphi'(Q_\mu; v)) \\ & + t_n (v(A(\bar{x} + t_n\xi + t_n^2 z_n)) - v(A\bar{x})) + o(t_n^2 \|\xi + t_n z_n\|^2). \end{aligned}$$

After dividing by t_n^2 and taking the $\limsup_{n\to\infty}$, on both sides of the latter inequality, we obtain

$$\begin{aligned} & \limsup_{n\to\infty} \frac{c}{t_n^2}\, d(\bar{x} + t_n\xi + t_n^2 z_n, \psi(Q_\mu + t_n v))^2 \\ & \leq \langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle \\ & + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y}) A\xi, A\xi\rangle - \varphi''(Q_\mu; v) + v'(\bar{y}; A\xi) \leq \varepsilon, \end{aligned}$$

where we made use of the choice of z, $\xi \in M(\bar{x}; v)$, and Theorem 3.4. This completes the proof. □

Complementing Corollary 3.5, we provide a result on the directional differentiability of ψ at Q_μ into any direction $v \in T^r(K_C; Q_\mu)$.

THEOREM 4.2. *Assume that the general conditions on g, Q_μ, and C of Corollary 3.5 are satisfied. Let $\bar{x} \in \psi(Q_\mu)$, and assume the following.*

(i)″ *There exists a constant $L > 0$ such that*

$$d_H(\sigma(y), \sigma(\tilde{y})) \leq L\|y - \tilde{y}\| \quad \text{for all } y, \tilde{y} \in A(C),$$

and, for each $r > 0$, there exists a constant $\eta(r) > 0$ such that

$$g(x) \geq \pi(Ax) + \eta(r) d(x, \sigma(Ax))^2 \quad \text{for all } x \in C \cap B(0, r),$$

where $\pi(y) = \inf\{g(x) : x \in C, Ax = y\}$ and

$$\sigma(y) = \operatorname{argmin}\{g(x) : x \in C, Ax = y\},\ y \in A(C).$$

(ii) *C is second-order regular at $\bar{x}$.*

Then the Gateaux directional derivative $\psi'(Q_\mu, \bar{x}; v)$ of ψ at the pair $(Q_\mu, \bar{x})$ exists for any direction $v \in T^r(K_C; Q_\mu)$ and satisfies the formula in Theorem 4.1. *Moreover, conditions* (i)$''$ *and* (ii) *are satisfied if C is polyhedral and g is linear- or (convex-) quadratic.*

Proof. Let $v \in T^r(K_C; Q_\mu)$. Since Q_μ is strongly convex on some open convex neighborhood of $A\psi(Q_\mu)$, we infer from condition (i)$''$ and Theorem 2.7 that condition (iii) of Theorem 4.1 is satisfied. Moreover, condition (i)$''$ implies (i)$'$, and thus, Corollary 3.5 says that the second-order directional derivative $\varphi''(Q_\mu; v)$ exists. Hence, the first part of the assertion follows from the proof of the previous theorem. Condition (ii) is satisfied if C is polyhedral, and if, in addition, g is convex-quadratic, Proposition 2.2 implies condition (i)$''$ holds. □

We note that Example 3.7 shows that, in general, the directional differentiability property of ψ gets lost at pairs $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$, where Q_μ is not strongly convex on some neighborhood of $A\psi(Q_\mu)$. Our next example demonstrates that Theorem 4.2 applies to situations where the solution set and its Gateaux directional derivatives are not singletons.

Example 4.3. We revisit the Examples 1.1 and 2.5 and observe that the assumptions of Theorem 4.2 are satisfied for any $\bar{x} \in \psi(Q_\mu)$. Hence, the Gateaux directional derivative $\psi'(Q_\mu, \bar{x}; v)$ exists at any pair $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$ and any direction $v \in T^r(K_C; Q_\mu)$. Since it holds that $\nabla^2 g(\bar{x}) = 0$, $\langle \nabla g(\bar{x}), \xi \rangle + \langle \nabla Q_\mu(A\bar{x}), A\xi \rangle = 0$ for all $\xi \in \mathbb{R}^3$, and $\nabla^2 Q_\mu(A\bar{x}) = 2 \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, it takes the form $\psi'(Q_\mu, \bar{x}; v) = \operatorname{argmin}\{\|A\xi\|^2 + v'(A\bar{x}; A\xi) : \xi \in T(C; \bar{x})\}$. Since the function $y \mapsto \|y\|^2 + v'(A\bar{x}; y)$ is strongly convex on $A(T(C; \bar{x}))$, it has a unique minimizer $\bar{y}(v) \in A(T(C; \bar{x}))$. Hence, there exists an element $\bar{\xi}(v) \in T(C; \bar{x})$ such that $A\bar{\xi}(v) = \bar{y}(v)$ and $\psi'(Q_\mu, \bar{x}; v) = (\bar{\xi}(v) + \ker A) \cap T(C; \bar{x})$. In particular, the Gateaux directional derivative $\psi'(Q_\mu, \bar{x}; .)$ is a set-valued mapping of the direction.

Remark 4.4. The approach we followed for deriving Gateaux directional differentiability of solution sets to (1.1) into directions $v \in T^r(K_C; Q_\mu)$ is based on lower and upper estimates for the optimal value function. Compared to the work in [5], [8], and [32], where this approach is developed and reviewed, we assume neither that the data of the perturbed problems $\min\{g(x) + Q(Ax) : x \in C\}$ is differentiable nor that solutions to (1.1) are unique. The (set-valued) Gateaux directional derivatives $\psi'(Q_\mu, \bar{x}; v)$ in the previous results are valid for the case $v = Q - Q_\mu$ with a general $Q \in K_C$. Hence, the results complement earlier work on contaminated distributions (e.g., [13], [14]). They apply to situations where Q is an expected recourse function with respect to a Dirac measure with unit mass placed at ω_*, i.e., $Q(y) = \tilde{Q}(\omega_* - y)$, and, hence, are relevant to study the influence of a specific scenario on changes of solution sets.

Another prominent approach to sensitivity analysis of optimization problems is based on the perturbation analysis of first-order necessary optimality conditions written as generalized equations (e.g., [17], [22], [24]). Applying this technique to study sensitivity of (1.1) requires C^1-properties of perturbed expected recourse functions Q. In the case of (1.1) and $Q \in C^1$, the parametric generalized equation reads

$$0 \in \nabla g(x) + A^T \nabla Q(Ax) + N_C(x),$$

where $N_C(x)$ denotes the normal cone to C at x and Q plays the role of a parameter. Relevant conditions in this context implying Lipschitz and differentiability properties of solutions at some $(Q_\mu, \bar{x})$ are the *strong regularity* of the generalized equation at parameter Q_μ [24], and the *subinvertibility* of the set-valued mapping

$F(x) = \nabla g(x) + A^T \nabla Q_\mu(Ax) + N_C(x)$ [17] together with the single-valuedness of the inverse of the contingent derivative of F at $(\bar{x}, 0)$ (cf., [2]), respectively. To see that both conditions are violated in general, we consider the linear case (i.e., g is linear and C is polyhedral). Then both conditions are equivalent if $Q_\mu \in C^2$ (Theorem 6.1 in [17]). The contingent derivative of F at $(\bar{x}, 0)$ has the form $DF(\bar{x}, 0)(u) = A^T \nabla^2 Q_\mu(A\bar{x})Au + DN_C(\bar{x}, -\nabla g(\bar{x}) - A^T \nabla Q_\mu(A\bar{x}))(u)$ (cf., Section 5.1 in [2]), where the contingent derivative DN_C is again a polyhedral multifunction. Since the first summand remains constant on translates of the null space of the matrix A, single-valuedness of the inverse of $DF(\bar{x}, 0)(u)$ fails to hold in general. This is essentially due to the same structural property, which leads to multiple solutions in Example 1.1 and to set-valued Gateaux directional derivatives in Example 4.3.

Finally, we turn to directional differentiability properties of ψ where the derivatives exist uniformly with respect to directions taken from compact sets of certain functional spaces. For our first result we consider the space $C^1(\mathbb{R}^s)$ and equip the set $K_C \cap C^1$ with the C^1-topology.

PROPOSITION 4.5. *Let $Q_\mu \in K_C \cap C^1$ and assume that the general conditions on g, Q_μ, and C in Proposition* 3.3 *are satisfied. In addition, we suppose condition* (ii) *of Theorem* 3.4 *to be satisfied. Let $\bar{x} \in \psi(Q_\mu)$, $t_n \to 0+$, and let (Q_n) be a sequence in K_C such that $v_n := \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^1(\mathbb{R}^s)$.*

Then the upper set limit of the sequence $(\frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$ of closed convex subsets in $\mathbb{R}^m$, i.e., $\limsup_{n\to\infty} \frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x}))$, is contained in the closed convex set

$$\operatorname{argmin}\left\{ \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle \nabla v(\bar{y}), A\xi\rangle + b(\xi) : \xi \in S(\bar{x}) \right\}.$$

Proof. Let $D_n := \frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$ for all $n \in \mathbb{N}$, and let $\bar{\xi}$ belong to the upper set limit $\limsup_{n\to\infty} D_n$. Then there exist a subsequence (again denoted by (D_n)) and elements $\xi_n \in D_n$ such that $\xi_n \to \bar{\xi}$. Since $\bar{x} + t_n\xi_n \in \psi(Q_\mu + t_n v_n) \subseteq C$, we have that $\bar{\xi} \in T(C; \bar{x})$, and as in the proof of Theorem 3.4, we deduce that $\bar{\xi} \in S(\bar{x})$. By expanding g and Q_μ as in the proof of Theorem 3.4 we obtain analogously

$$\begin{aligned}
&\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n \varphi'(Q_\mu; v_n) \\
&\quad = g(\bar{x} + t_n\xi_n) + Q_\mu(A(\bar{x} + t_n\xi_n)) + t_n v_n(A(\bar{x} + t_n\xi_n)) - g(\bar{x}) - Q_\mu(A\bar{x}) \\
&\qquad - t_n v_n(A\bar{x}) \\
&\quad \geq t_n^2 b(\bar{\xi}) + \frac{1}{2}t_n^2\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}t_n^2\langle \nabla^2 Q_\mu(A\bar{x})A\bar{\xi}, A\bar{\xi}\rangle \\
&\qquad + t_n(v_n(A(\bar{x} + t_n\xi_n)) - v_n(A\bar{x})) + o(t_n^2).
\end{aligned}$$

After dividing by t_n^2 and taking the $\limsup_{n\to\infty}$ on both sides of the inequality, we obtain, as in the proof of Theorem 3.8,

$$\begin{aligned}
&\limsup_{n\to\infty} \frac{1}{t_n^2}(\varphi(Q_\mu + t_n v_n) - \varphi(Q_\mu) - t_n \varphi'(Q_\mu; v_n)) \\
&\qquad \geq \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi}, \bar{\xi}\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(A\bar{x})A\bar{\xi}, A\bar{\xi}\rangle + \langle \nabla v(A\bar{x}), A\bar{\xi}\rangle + b(\bar{\xi}).
\end{aligned}$$

Hence, we may conclude from Proposition 3.3 that $\bar{\xi}$ belongs to the set

$$\operatorname{argmin}\left\{ \frac{1}{2}\langle \nabla^2 \mathrm{g}(\bar{\mathrm{x}})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 \mathrm{Q}_\mu(\bar{\mathrm{y}})\mathrm{A}\xi, \mathrm{A}\xi\rangle + \langle \nabla \mathrm{v}(\bar{\mathrm{y}}), \mathrm{A}\xi\rangle + \mathrm{b}(\xi) : \xi \in \mathrm{S}(\bar{\mathrm{x}}) \right\},$$

and we are done. ∎

Remark 4.6. The upper limit of the sequence $(\frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$ in Proposition 4.5 is nonempty if the mapping $d(\bar{x}, \psi(\cdot))$ from K_C into the extended reals has the Lipschitzian property of Theorem 2.3 at Q_μ. Indeed, we may select $x_n \in \psi(Q_\mu + t_n v_n)$ for large $n \in \mathbb{N}$ such that for some constants $\hat{L} > 0$ and $r > 0$, $\|\bar{x} - x_n\| = d(\bar{x}, \psi(Q_\mu + t_n v_n)) \le \hat{L} t_n \|v_n\|_{L,r}$. Hence, the sequence $(\frac{1}{t_n}(x_n - \bar{x}))$ is bounded and has a convergent subsequence whose limit belongs to $\limsup_{n\to\infty} \frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$. If the Lipschitz property of $d(\bar{x}, \psi(\cdot))$ is violated, the upper set limit may be empty. This is illustrated by Example 3.7, in which we have $\bar{x} = 0$, $\psi(Q_\mu + t_n v) = \{\sqrt{t_n}\}$, and, thus, $\frac{1}{t_n}(\psi(Q_\mu + t_n v) - \bar{x}) = \{t_n^{-\frac{1}{2}}\}$.

In order to establish the semidifferentiability of ψ at a pair $(Q_\mu, \bar{x})$ belonging to the graph of ψ, it remains to show, according to Proposition 4.5, that the solution set

$$\operatorname{argmin}\left\{\frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle \nabla v(\bar{y}), A\xi\rangle + b(\xi) : \xi \in S(\bar{x})\right\}$$

is contained in the lower set limit $\liminf_{n\to\infty} \frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$, where $v_n := \frac{1}{t_n}(Q_n - Q_\mu)$, $Q_n \in K_C$, for all $n \in \mathbb{N}$, and (v_n) converges to v. To this end, a uniform quadratic growth condition of the objective functions $g(\cdot) + (Q_\mu + t_n v_n)(A\,\cdot)$ for large $n \in \mathbb{N}$ is significant. In view of Theorem 2.7, the uniform strong convexity of Q_μ and its approximations Q_n for large $n \in \mathbb{N}$ is decisive for the growth condition. The next example and the following result show that the approximations Q_n do not maintain the strong convexity property of Q_μ in general if the sequence (Q_n) converges to Q_μ in $C^1(\mathbb{R}^s)$, but that the situation is much more advantageous when considering the $C^{1,1}$-topology.

Example 4.7. Let $Q_\mu(y) = y^2$ for all $y \in \mathbb{R}$ and let Q_n be the following differentiable convex function:

$$Q_n(y) := \max\left\{0, -y - \frac{1}{n}\right\}^2 + \max\left\{0, y - \frac{1}{n}\right\}^2 \quad \text{for all } \ y \in \mathbb{R}, n \in \mathbb{N}.$$

Note that $Q_n(y) = 0$ for all $y \in [-\frac{1}{n}, \frac{1}{n}]$, and Q_n is not strongly convex for each $n \in \mathbb{N}$, but (Q_n) converges to Q_μ in $C^1(\mathbb{R}^s)$.

LEMMA 4.8. *Let $Q_\mu \in K_C \cap C^{1,1}(\mathbb{R}^s)$ be strongly convex on some bounded convex set $U \subseteq \mathbb{R}^s$ (with some constant $\kappa > 0$). Then there exists a neighborhood $\mathcal{N}$ of Q_μ in $C^{1,1}(\mathbb{R}^s)$ such that each function Q belonging to $\mathcal{N}$ is strongly convex on U with constant $\frac{\kappa}{2}$.*

Proof. The strong convexity of Q_μ on U (with constant $\kappa > 0$) is equivalent to the condition $\langle \nabla Q_\mu(y) - \nabla Q_\mu(\tilde{y}), y - \tilde{y}\rangle \ge \kappa \|y - \tilde{y}\|^2$ for all $y, \tilde{y} \in U$. Let $r > 0$ be chosen such that $\operatorname{cl} U \subseteq B(0, r)$, and let $\mathcal{N}$ be a neighborhood of Q_μ in $C^{1,1}(\mathbb{R}^s)$ having the property $\|\nabla(Q_\mu - Q)\|_{L,r} \le \frac{\kappa}{2}$ for all $Q \in \mathcal{N}$. Let $y, \tilde{y} \in U$, with $y \ne \tilde{y}$. Then we obtain for any $Q \in \mathcal{N}$

$$\begin{aligned}
\kappa &\le \frac{\langle \nabla Q_\mu(y) - \nabla Q_\mu(\tilde{y}), y - \tilde{y}\rangle}{\|y - \tilde{y}\|^2} \\
&= \frac{\langle \nabla Q(y) - \nabla Q(\tilde{y}), y - \tilde{y}\rangle}{\|y - \tilde{y}\|^2} + \frac{\langle \nabla(Q_\mu - Q)(y) - \nabla(Q_\mu - Q)(\tilde{y}), y - \tilde{y}\rangle}{\|y - \tilde{y}\|^2} \\
&\le \frac{\langle \nabla Q(y) - \nabla Q(\tilde{y}), y - \tilde{y}\rangle}{\|y - \tilde{y}\|^2} + \frac{\|\nabla(Q_\mu - Q)(y) - \nabla(Q_\mu - Q)(\tilde{y})\|}{\|y - \tilde{y}\|^2} \\
&\le \frac{\langle \nabla Q(y) - \nabla Q(\tilde{y}), y - \tilde{y}\rangle}{\|y - \tilde{y}\|^2} + \|\nabla(Q_\mu - Q)\|_{L,r},
\end{aligned}$$

and, hence,

$$\frac{\kappa}{2}\|y-\tilde{y}\|^2 \le \langle \nabla Q(y) - \nabla Q(\tilde{y}), y-\tilde{y}\rangle.$$

This means that Q is strongly convex on U with constant $\frac{\kappa}{2}$. □

Now we are able to show that the solution set mapping ψ is semidifferentiable on $K_C \cap C^{1,1}$ at some pairs $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$, into any direction v from the tangent cone $T(K_C \cap C^{1,1}; Q_\mu)$ to $K_C \cap C^{1,1}(\mathbb{R}^s)$ at Q_μ in $C^{1,1}(\mathbb{R}^s)$. The assumptions are essentially the same as in Theorem 4.2.

THEOREM 4.9. *Let $Q_\mu \in K_C \cap C^{1,1}$, and assume that $\psi(Q_\mu)$ is nonempty and bounded. Let g be twice continuously differentiable, and let Q_μ be strongly convex on some open convex neighborhood U of $A\psi(Q_\mu)$ and twice continuously differentiable at $\bar{y}$, where $\{\bar{y}\} = A\psi(Q_\mu)$. Assume that condition* (i)″ *of Theorem* 4.2 *is satisfied.*

Then the solution set mapping ψ from $K_C \cap C^{1,1}$ into $\mathbb{R}^m$ is semidifferentiable at any pair $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$, such that C is second-order regular at $\bar{x}$, and into any direction $v \in T(K_C \cap C^{1,1}; Q_\mu)$, i.e., for any such $\bar{x}$ and v, $t_n \to 0+$, and (Q_n) in $K_C \cap C^{1,1}$ with $v_n = \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^{1,1}(\mathbb{R}^s)$ the set limit

$$D\psi(Q_\mu, \bar{x}; v) = \lim_{n\to\infty} \frac{1}{t_n}(\psi(Q_\mu + t_n v_n) - \bar{x})$$

exists. The semiderivative $D\psi(Q_\mu, \bar{x}; v)$ is equal to the set

$$\operatorname{argmin}\left\{ \frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle \nabla v(\bar{y}), A\xi\rangle + b(\xi) : \xi \in S(\bar{x}) \right\}.$$

Moreover, ψ is semidifferentiable at any pair $(Q_\mu, \bar{x})$, $\bar{x} \in \psi(Q_\mu)$, into any direction $v \in T(K_C \cap C^{1,1}; Q_\mu)$ if C is polyhedral. Condition (i)″ *is satisfied if C is polyhedral and g is linear- or (convex-) quadratic.*

Proof. Let $\bar{x} \in \psi(Q_\mu)$ be such that C is second-order regular at $\bar{x}$, $v \in T(K_C \cap C^{1,1}; Q_\mu)$, and $v_n = \frac{1}{t_n}(Q_n - Q_\mu) \to v$ in $C^{1,1}(\mathbb{R}^s)$, where $t_n \to 0+$ and (Q_n) is a sequence in $K_C \cap C^{1,1}$. We may assume that the neighborhood U is bounded. Since (Q_n) converges to Q_μ in $C^{1,1}(\mathbb{R}^s)$, we obtain from Lemma 4.8 that there exists an $n_0 \in \mathbb{N}$ such that Q_n is strongly convex on U for each $n \ge n_0$ with a uniform constant $\kappa > 0$. Moreover, we choose n_0 sufficiently large such that $\psi(Q_n)$ is nonempty for each $n \ge n_0$. Arguing as in the proof of Theorem 2.7, we obtain a constant $c > 0$ and a neighborhood V of $\psi(Q_n)$ such that the growth condition

$$g(x) + Q_n(Ax) \ge \varphi(Q_n) + cd(x, \psi(Q_n))^2$$

holds for all $x \in C \cap V$ and $n \ge n_0$.

Let $\bar{\xi} \in S(\bar{x})$ be a minimizer of the function $\frac{1}{2}\langle \nabla^2 g(\bar{x})\xi, \xi\rangle + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle \nabla v(\bar{y}), A\xi\rangle + b(\xi)$ subject to $\xi \in S(\bar{x})$. Because of Proposition 4.5 it remains to show that $\bar{\xi}$ belongs to the lower limit $\liminf_{n\to\infty} \frac{1}{n}(\psi(Q_\mu + t_n v_n) - \bar{x}) = \liminf_{n\to\infty} \frac{1}{t_n}(\psi(Q_n) - \bar{x})$. To this end we argue as in the proof of Theorem 4.1. Let $\varepsilon > 0$ be given, and let $z \in T^2(C; \bar{x}, \bar{\xi})$ be such that $\langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle \le b(\bar{\xi}) + \varepsilon$. Then there exists a sequence (z_n) converging to z with $x_n = \bar{x} + t_n\xi + t_n^2 z_n \in C$ for all $n \in \mathbb{N}$. Then it suffices to show that

$$\lim_{n\to\infty} \frac{1}{t_n} d(\bar{x} + t_n\bar{\xi} + t_n^2 z_n, \psi(Q_n)) = 0.$$

By using the above growth condition and by expanding the function g and Q_μ, we obtain, similar to the proof of Theorem 4.1, that

$$\begin{aligned}\limsup_{n\to\infty} \frac{c}{t_n^2}\, d(\bar{x} + t_n\bar{\xi} + t_n^2 z_n, \psi(Q_n))^2 \\ \leq \langle \nabla g(\bar{x}), z\rangle + \langle \nabla Q_\mu(\bar{y}), Az\rangle + \frac{1}{2}\langle \nabla^2 g(\bar{x})\bar{\xi},\bar{\xi}\rangle \\ + \frac{1}{2}\langle \nabla^2 Q_\mu(\bar{y})A\bar{\xi}, A\bar{\xi}\rangle - \varphi''(Q_\mu; v) + \langle \nabla v(\bar{y}), A\bar{\xi}\rangle \leq \varepsilon\,.\end{aligned}$$

This implies $\bar{\xi} \in \liminf_{n\to\infty} \frac{1}{t_n}(\psi(Q_n) - \bar{x})$ and the semidifferentiability of ψ at $(Q_\mu, \bar{x})$ in direction v is shown. The remaining part of the assertion follows as in the proof of Theorem 4.2. □

For the linear-quadratic case, the essential assumptions in Theorem 4.9 are the strong convexity of Q_μ, and the smoothness properties of Q_μ and its perturbations Q, respectively. While criteria for strong convexity were already discussed in section 2, we now add some comments on $C^{1,1}$ and C^2 properties of expected recourse functions. Later we close by indicating some conclusions of the results of sections 3 and 4 on asymptotic properties of statistical estimators of optimal values and solution sets.

Remark 4.10. Assume (A1)–(A3) and μ to have a density with respect to the Lebesgue measure on $\mathbb{R}^s$. Then the function Q_μ in (1.2) is continuously differentiable on $\mathbb{R}^s$ and its gradient is of the form $\nabla Q_\mu(y) = \sum_{i=1}^{\ell} d_i \mu(y + B_i(\mathbb{R}^s_+))$ for all $y \in \mathbb{R}^s$, where B_i, $i = 1, \ldots, \ell$, are certain basis submatrices of the recourse matrix W such that the simplicial cones $B_i(\mathbb{R}^s_+)$, $i = 1, \ldots, \ell$, are linearity regions of $\tilde{Q}$ and $-d_i$ is the gradient of $\tilde{Q}$ on int $B_i(\mathbb{R}^s_+)$, $i = 1, \ldots, \ell$ (cf., [15], [39]). Denoting by F_μ the distribution function of μ and using the formula

$$\mu(y + B(\mathbb{R}^s_+)) = F_{\mu\circ(-B)}(-B^{-1}y) \quad \text{for all } \; y \in \mathbb{R}^s,$$

for any nonsingular (s,s)-matrix B, $C^{1,1}$ and C^2 properties of Q_μ may thus be formulated in terms of Lipschitz and differentiability properties of the distribution functions $F_{\mu\circ(-B_i)}$ to the linear transforms $\mu \circ (-B_i)$, $i = 1, \ldots, \ell$, of the measure μ.

The distribution function F_μ of a probability measure μ on $\mathbb{R}^s$ is locally Lipschitzian if all one-dimensional marginal distribution functions of μ are locally Lipschitzian (cf. [26], [38]). F_μ is continuously differentiable if μ has a continuous density function and all one-dimensional marginal distribution functions of μ are continuously differentiable (cf. [21], [38]). If μ has a continuous density function, then $\mu \circ B$ has a continuous density for any nonsingular (s,s)-matrix B, too. Hence, we may conclude, for instance, that Q_μ belongs to $C^{1,1}(\mathbb{R}^s)$ (and $C^2(\mathbb{R}^s)$) if μ has a (continuous) density and the above-mentioned conditions on the one-dimensional marginal distribution functions for $F_{\mu\circ B}$ belonging to $C^{0,1}(\mathbb{R}^s)$ (and $C^1(\mathbb{R}^s)$, respectively) are satisfied for any nonsingular (s,s)-matrix B. This criterion is particularly useful for probability distributions μ which have the property that all one-dimensional marginal distributions of μ and all linear transforms $\mu \circ B$ for all nonsingular matrices B belong to the same class of measures. For instance, all multivariate normal and all logarithmic concave probability measures (e.g., [15]) form classes having this property.

Remark 4.11. We consider a sequence (Q_n) of nonparametric estimators of Q_μ and assume that each Q_n is a random variable with values in some linear metric (function) space Z and in K_C. Furthermore, we assume that a central limit result of the form

$$\tau_n^{-1}(Q_n - Q_\mu) \to_d \zeta$$

is satisfied for some sequence of positive numbers (τ_n) decreasing to 0 and for some random variable ζ taking values in a separable subset of Z. Here, we denote by $\to_d$ the convergence in distribution of Z-valued random variables. Then versions of the delta-method (see, e.g., [36]) together with the second-order Hadamard differentiability of the optimal value φ at Q_μ (Theorem 3.8 and $Z = C^1(\mathbb{R}^s)$) and the semidifferentiability of the solution set ψ at Q_μ (Theorem 4.9 and $Z = C^{1,1}(\mathbb{R}^s)$) lead to central limit formulas for the sequence $(\varphi(Q_n))$ of real random variables and the sequence of random sets $(\psi(Q_n))$, respectively. In particular, we obtain from Theorem 3.8 and a second-order version of the delta-method that

$$\tau_n^{-2}(\varphi(Q_n)-\varphi(Q_\mu)-\varphi'(Q_\mu;Q_n-Q_\mu)) = \tau_n^{-2}(\varphi(Q_n)-g(\bar{x})-Q_n(A\bar{x})) \to_d \varphi''(Q_\mu;\zeta),$$

where $\bar{x} \in \psi(Q_\mu)$ and $\to_d$ refer to convergence in distribution of real-valued random variables. Theorem 4.9 and a set-valued version of the delta-method [16], [20] imply

$$\tau_n^{-1}(\psi(Q_n) - \bar{x}) \to_d D\psi(Q_\mu, \bar{x}; \zeta),$$

where $\bar{x} \in \psi(Q_\mu)$ and $\to_d$ refer to convergence in distribution of closed-valued measurable multifunctions in $\mathbb{R}^m$ (cf. [29]). The asymptotic distributions in both central limit results are the probability distributions of the optimal value and of the solution set, respectively, of the random convex program that consists in minimizing the (random) objective $\frac{1}{2}\langle\nabla^2 g(\bar{x})\xi,\xi\rangle + \frac{1}{2}\langle\nabla^2 Q_\mu(\bar{y})A\xi, A\xi\rangle + \langle\nabla\zeta(\bar{y}), A\xi\rangle + b(\xi)$ subject to ξ satisfying the (deterministic) constraints $\xi \in T(C;\bar{x})$ and $\langle\nabla g(\bar{x}),\xi\rangle + \langle\nabla Q_\mu(\bar{y}), A\xi\rangle = 0$. Furthermore, in the linear-quadratic case the set-valued central limit result may be complemented by limit theorems for selections forming a Castaing representation of ψ (cf. [11]).

Acknowledgments. The authors wish to thank Alexander Shapiro (Georgia Institute of Technology, Atlanta) and René Henrion (WIAS, Berlin) for beneficial discussions. Moreover, the comments and suggestions of the associate editor and of a referee are gratefully acknowledged.

REFERENCES

[1] Z. Artstein and R. J.-B. Wets, *Stability results for stochastic programs and sensors, allowing for discontinuous objective functions*, SIAM J. Optim., 4 (1994), pp. 537–550.
[2] J.-P. Aubin and H. Frankowska, *Set-Valued Analysis*, Birkhäuser, Boston, 1990.
[3] A. Auslender and R. Cominetti, *A comparative study of multifunction differentiability with applications in mathematical programming*, Math. Oper. Res., 16 (1991), pp. 240–258.
[4] A. Ben-Tal and J. Zowe, *Directional derivatives in nonsmooth optimization*, J. Optim. Theory Appl., 47 (1985), pp. 483–490.
[5] J. F. Bonnans and R. Cominetti, *Perturbed optimization in Banach spaces* I: *A general theory based on a weak directional constraint qualification*, SIAM J. Control Optim., 34 (1996), pp. 1151–1171.
[6] J. F. Bonnans, R. Cominetti, and A. Shapiro, *Second order optimality conditions based on parabolic second order tangent sets*, SIAM J. Optim., 9 (1999), pp. 466–492.
[7] J. F. Bonnans and A. D. Ioffe, *Quadratic growth and stability in convex programming problems with multiple solutions*, J. Convex Anal., 2 (1995), pp. 41–57.
[8] J. F. Bonnans and A. Shapiro, *Optimization problems with perturbations: A guided tour*, SIAM Rev., 40 (1998), pp. 228–264.
[9] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley, New York, 1983.
[10] R. Cominetti, *Metric regularity, tangent sets and second order optimality conditions*, Appl. Math. Optim., 21 (1990), pp. 265–287.
[11] D. Dentcheva, *Differentiable selections and Castaing representations of multifunctions*, J. Math. Anal. Appl., 223 (1998), pp. 371–396.

[12] D. Dentcheva, W. Römisch, and R. Schultz, *Strong convexity and directional derivatives of marginal values in two-stage stochastic programming*, in Stochastic Programming, K. Marti and P. Kall, eds., Lecture Notes in Econom. and Math. Systems 423, Springer-Verlag, Berlin, 1995, pp. 8–21.
[13] J. Dupačová, *Stability and sensitivity analysis for stochastic programming*, Ann. Oper. Res., 27 (1990), pp. 115–142.
[14] J. Dupačová, *Stability in stochastic programming with recourse. Contaminated distributions*, Math. Programming Stud., 27 (1986), pp. 133–144.
[15] P. Kall, *Stochastic Linear Programming*, Springer-Verlag, Berlin, 1976.
[16] A. J. King, *Generalized delta theorems for multivalued mappings and measurable selections*, Math. Oper. Res., 14 (1989), pp. 720–736.
[17] A. J. King and R. T. Rockafellar, *Sensitivity analysis for nonsmooth generalized equations*, Math. Programming, 55 (1992), pp. 193–212.
[18] 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.
[19] D. Klatte and G. Thiere, *Error bounds for solutions of linear equations and inequalities*, ZOR—Math. Methods Oper. Res., 41 (1995), pp. 191–214.
[20] P. Lachout, *On multifunction transforms of probability measures*, Ann. Oper. Res., 56 (1995), pp. 241–249.
[21] K. Marti, *Approximationen der Entscheidungsprobleme mit linearer Ergebnisfunktion und positiv homogener, subadditiver Verlustfunktion*, Zeitschrift Wahrscheinlichkeitstheorie und verwandte Gebiete, 31 (1975), pp. 203–233.
[22] B. S. Mordukhovich, *Stability theory for parametric generalized equations and variational inequalities via nonsmooth analysis*, Trans. AMS, 343 (1994), pp. 609–657.
[23] J.-P. Penot, *Differentiability of relations and differential stability of perturbed optimization problems*, SIAM J. Control Optim., 22 (1984), pp. 529–551.
[24] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.
[25] R. T. Rockafellar, *Proto-differentiability of set-valued mappings and its applications in optimization*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 6 (1989), Suppl., pp. 449–482.
[26] W. Römisch and R. Schultz, *Stability of solutions for stochastic programs with complete recourse*, Math. Oper. Res., 18 (1993), pp. 590–609.
[27] W. Römisch and R. Schultz, *Lipschitz stability for stochastic programs with complete recourse*, SIAM J. Optim., 6 (1996), pp. 531–547.
[28] R. Y. Rubinstein and A. Shapiro, *Discrete Event Systems. Sensitivity Analysis and Stochastic Optimization by the Score Function Method*, Wiley, Chichester, UK, 1993.
[29] 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.
[30] R. Schultz, *Strong convexity in stochastic programs with complete recourse*, J. Comput. Appl. Math., 56 (1994), pp. 3–22.
[31] A. Seeger, *Second order directional derivatives in parametric optimization problems*, Math. Oper. Res., 13 (1988), pp. 628–645.
[32] A. Shapiro, *Sensitivity analysis of nonlinear programs and differentiability properties of metric projections*, SIAM J. Control Optim., 26 (1988), pp. 628–645.
[33] A. Shapiro, *On concepts of directional differentiability*, J. Optim. Theory Appl., 66 (1990), pp. 477–487.
[34] A. Shapiro, *On differential stability in stochastic programming*, Math. Programming, 47 (1990), pp. 107–116.
[35] A. Shapiro, *Asymptotic analysis of stochastic programs*, Ann. Oper. Res., 30 (1991), pp. 169–186.
[36] A. W. van der Vaart and J. A. Wellner, *Weak Convergence and Empirical Processes*, Springer Ser. Statist., Springer-Verlag, New York, 1996.
[37] D. Walkup and R. J.-B. Wets, *A Lipschitzian characterization of convex polyhedra*, Proc. AMS, 23 (1969), pp. 167–173.
[38] J. Wang, *Distribution sensitivity analysis for stochastic programs with complete recourse*, Math. Programming, 31 (1985), pp. 286–297.
[39] R. J.-B. Wets, *Stochastic programs with fixed recourse: The equivalent deterministic program*, SIAM Rev., 16 (1974), pp. 309–339.
[40] R. J.-B. Wets, *Stochastic programming*, in Handbooks in Operations Research and Management Science, 1, Optimization, G. L. Nemhauser, A. H. G. Rinnoy Kan, and M. J. Todd, eds., North-Holland, Amsterdam, 1989, pp. 573–629.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 113–132

A NEW QP-FREE, GLOBALLY CONVERGENT, LOCALLY SUPERLINEARLY CONVERGENT ALGORITHM FOR INEQUALITY CONSTRAINED OPTIMIZATION*

HOU-DUO QI† AND LIQUN QI†

Abstract. In this paper, we propose a new QP-free method, which ensures the feasibility of all iterates, for inequality constrained optimization. The method is based on a nonsmooth equation reformulation of the KKT optimality condition, by using the Fischer–Burmeister nonlinear complementarity problem function. The study is strongly motivated by recent successful applications of this function to the complementarity problem and the variational inequality problem. The method we propose here enjoys some advantages over similar methods based on the equality part of the KKT optimality condition. For example, without assuming isolatedness of the accumulation point or boundedness of the Lagrangian multiplier approximation sequence, we show that every accumulation point of the iterative sequence generated by this method is a KKT point if the linear independence condition holds. And if the second-order sufficient condition and the strict complementarity condition hold, the method is superlinearly convergent. Some preliminary numerical results indicate that this new QP-free method is quite promising.

Key words. QP-free method, linear independence, strict complementarity, global convergence, superlinear convergence

AMS subject classifications. 90C30, 65K10

PII. S1052623499353935

1. Introduction.

This paper is concerned with finding a solution of the inequality constrained optimization problem

$$\begin{array}{ll} \min & f(x) \\ \text{subject to (s.t.)} & x \in \mathcal{F} := \{x \in \mathbb{R}^n | g(x) \le 0\}, \end{array} \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ are continuously differentiable. A pair $(x, \lambda) \in \mathbb{R}^{n+m}$ with $x \in \mathcal{F}$ is called a stationary point of (1.1) if it satisfies

$$\begin{aligned} \nabla_x L(x, \lambda) &= 0, \\ \lambda_i g_i(x) &= 0, \quad i = 1, \ldots, m, \end{aligned} \tag{1.2}$$

where $L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i g_i(x)$ is the Lagrangian function of (1.1). If furthermore $\lambda \ge 0$, (x, λ) is called a KKT point. Sometimes, we also call $x \in \mathcal{F}$ a stationary point or a KKT point of (1.1) if there exists $\lambda \in \mathbb{R}^m$ such that (x, λ) is a stationary point or a KKT point of (1.1).

Our study here is encouraged by recent successful numerical methods for nonlinear complementarity problems; for a survey see [6]. The function most often used in these methods is probably the Fischer–Burmeister nonlinear complementarity problem (NCP) function, which has a very simple structure:

$$\phi(a, b) = \sqrt{a^2 + b^2} - a - b, \qquad a, b \in \mathbb{R}.$$

*Received by the editors March 27, 1999; accepted for publication (in revised form) March 7, 2000; published electronically July 25, 2000. This work was supported by the Australian Research Council.

http://www.siam.org/journals/siopt/11-1/35393.html

†School of Mathematics, The University of New South Wales, Sydney 2052, NSW, Australia (hdqi@maths.unsw.edu.au, L.Qi@unsw.edu.au).

The function enjoys several interesting, indeed somewhat surprising, properties, among which are

- $\phi(a,b)=0 \iff a\geq 0,\ b\geq 0,\ ab=0.$
- The square of ϕ is continuously differentiable.
- ϕ is twice continuously differentiable everywhere except at the origin, but it is strongly semismooth at the origin.

So the KKT condition of (1.1) can be equivalently reformulated as

$$\Phi(x,\lambda)=\begin{bmatrix}\nabla_x L(x,\lambda)\\ \phi(-g_1(x),\lambda_1)\\ \vdots\\ \phi(-g_m(x),\lambda_m)\end{bmatrix}=0, \tag{1.3}$$

and the semismooth Newton method [20, 24] can be applied to the function Φ. The differentiability of the norm square $\psi=\|\Phi\|^2$ of Φ justifies its role as a merit function in the globalized procedure of the semismooth Newton method; see [2, 27] for excellent examples of this idea. Numerical experiments on nonlinear complementarity problems show advantages of methods based on the Fischer–Burmeister function over existing ones [2, 1]. Encouraged by the success on nonlinear complementarity problems, the methods have been carried over to solve variational inequality problems (VIP) [4, 15], which include the KKT system of (1.1) as a special case. However, such methods are only able to find a stationary point of ψ, which is a KKT point under additional conditions. For other semismooth equation reformulations of the KKT condition, see the paper by Qi and Jiang [23]. We note that, in a very recent paper [8], Ferris and Sinapiromsaran used the Fischer–Burmeister function to reformulate the KKT system of (1.1) as a mixed complementarity problem. The function serves as the merit function in their PATH solver. The numerical results reported there are very competitive over known problem solvers for nonlinear constrained optimization problems. We also note that a QP-free method based on the Fischer–Burmeister function is studied in [15]. The preliminary numerical results reported there show the promise of the method. For the advantage of QP-free methods for nonlinear constrained optimization problems, see the paper [5] by Facchinei and Lucidi.

The feasibility issue for (1.1) is always important because some real-life applications such as in engineering design and economics [12, 18] require the data only defined in the feasible region. The QP-free method proposed in [15] does not ensure the feasibility of all iterates. A QP-free method for (1.1), which also ensures the feasibility of all iterates, was proposed about 10 years ago by Panier, Tits, and Herskovits [19]. Letting (x^k,μ^k) with $x^k\in\mathcal{F}$ be an approximation to a KKT point (x^*,λ^*), their algorithm first calculates a descent direction d^{k0} by solving the following system, which is derived from (1.2):

$$\begin{bmatrix} H_k, & \nabla g(x^k)\\ \operatorname{diag}(\mu^k)\nabla g^T(x^k), & \operatorname{diag}(g_k)\end{bmatrix}\begin{bmatrix} d\\ \lambda\end{bmatrix}=\begin{bmatrix}-\nabla f(x^k)\\ 0\end{bmatrix}, \tag{1.4}$$

where $H_k\in\mathbb{R}^{n\times n}$ is a positive definite matrix, $g_k:=g(x^k)$, $\nabla g(x^k)$ denotes the transposed Jacobian of g at x^k, and for a vector $\mu\in\mathbb{R}^m$ $\operatorname{diag}(\mu)$ denotes the diagonal matrix whose ith diagonal element is μ_i. If (x^k,μ^k) is a KKT point, then the coefficient matrix in (1.4) is nonsingular if and only if the linear independence condition and the strict complementarity condition hold. This observation is crucial to their convergence analysis. In order to guarantee the feasibility of the next iterate, i.e., $x^{k+1}\in\mathcal{F}$, they continue to calculate a direction d^{k1} by solving a perturbed system of (1.4):

$$\begin{bmatrix} H_k, & \nabla g(x^k) \\ \operatorname{diag}(\mu^k)\nabla g^T(x^k), & \operatorname{diag}(g_k) \end{bmatrix} \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x^k), \\ -\|d^{k0}\|^\nu \operatorname{diag}(\mu^k)e \end{bmatrix}, \tag{1.5}$$

where $\nu > 2$ and e is the vector of all ones in an appropriate dimensional space (here $e \in \mathbb{R}^m$). Globally, the search direction is a convex combination of the two directions, namely,

$$d^k = (1 - \rho_k)d^{k0} + \rho_k d^{k1},$$

where ρ_k is calculated explicitly. To avoid the Maratos effect, locally the search direction should be bent slightly by a relatively small amount of direction, which is also a solution of a linear system.

On the one hand, as Panier, Tits, and Herskovits observed [19, p. 810] that "linear system (1.4), (1.5) may become very ill-conditioned if some multiplier μ_i corresponding to a nearly active constraint g_i becomes very small. This may occur close to a solution of problem (1.1) at which the strict complementarity conditions are not satisfied," their QP-free method is very sensitive to the parameters chosen. The ill-conditionedness of the matrix may force the multiplier approximation sequence to diverge. On the other hand, they could show only that any accumulation point of the iterates is a stationary point. Under additional assumption of isolatedness of the stationary point, this point is shown to be a KKT point. The algorithm was later improved by Gao, He, and Wu [11] in the sense that every accumulation point of the iterates is a KKT point. To achieve this, they solve an extra linear system, which is a slight perturbation of (1.4) on the right-hand side. However, they assume that the multiplier approximation sequence remains bounded, which is unlikely as the matrix in (1.4) becomes very ill-conditioned.

Regarding the recent development on QP-free methods for constrained optimization problems, we note that the method of Panier–Tits–Herskovits was recently converted to a primal-dual logarithmic barrier interior-point method, whose convergence properties are also established under the conditions for the original one; see [26] for details. We also note that a quite general feasible direction approach for the constrained optimization problem was studied by Herskovits in [13]. The convergence of this approach is established under a set of conditions similar to [19]. Based on this approach, first-order, Newton, and quasi-Newton algorithms can be obtained, depending on what information about functions was used.

In this paper, we propose a new QP-free method for (1.1) based on the Fischer–Burmeister function. Our method always ensures the strict feasibility of all iterates, and the property in turn ensures the differentiability of Φ. If the linear independence condition is assumed in $\mathcal{F}$, then the matrix sequence used in our algorithm is uniformly nonsingular. We stress that we do not need the strict complementarity condition for the uniform nonsingularity. This, we believe, is an outstanding advantage of our method over the ones proposed in [19, 11]. Some ideas are borrowed from [19, 11]. For example, to ensure the descentness of the search direction and the feasibility of the iterates we adopt the idea of "bending" in the paper [19]; and to ensure the convergence to a KKT point, we also solve an extra linear system. Without assuming the boundedness of the multiplier approximation sequence or the isolatedness of the limit point, we establish that every accumulation point of the iterates is a KKT point of (1.1). Under mild conditions, we establish superlinear convergence of the algorithm. The paper is organized as follows. In section 2, we present our algorithm and develop

some properties for later use. The global convergence of this algorithm to a KKT point of (1.1) is established in section 3, whereas we prove local fast convergence in section 4. Section 5 is devoted to some preliminary numerical results. Some conclusions are drawn in section 6.

2. Algorithm. Let $(x^k, \mu^k) \in \mathbb{R}^{n+m}$ be given with x^k being strictly feasible in the sense that $g_i(x^k) < 0$ for all $i = 1, \ldots, m$. We define three vectors η^k, θ^k, and $\delta^k \in \mathbb{R}^m$ as follows:

$$\eta_i^k = \frac{g_i(x^k)}{\sqrt{g_i^2(x^k) + (\mu_i^k)^2}} + 1, \qquad \theta_i^k = \left(1 - \frac{\mu_i^k}{\sqrt{g_i^2(x^k) + (\mu_i^k)^2}}\right)^{1/2},$$

and

$$\delta_i^k = \left(\frac{1}{\sqrt{g_i^2(x^k) + (\mu_i^k)^2}(\mu_i^k + \sqrt{g_i^2(x^k) + (\mu_i^k)^2})}\right)^{1/2}.$$

Since x^k is strictly feasible, η^k, θ^k, and δ^k are well defined. It is well known [9, 10] that

$$(\eta_i^k)^2 + (\theta_i^k)^4 \geq 3 - 2\sqrt{2}. \tag{2.1}$$

Also it is easy to check that

$$\theta_i^k = -\delta_i^k g_i(x^k). \tag{2.2}$$

Let $A_k = \nabla g(x^k)$, $\Gamma_k = \operatorname{diag}(\eta^k)$, and $\Theta_k = -\sqrt{2}\operatorname{diag}(\theta^k)$. If f, $g_i, i = 1, \ldots, m$, are twice continuously differentiable, then the function defined in (1.3) is continuously differentiable at (x^k, μ^k) and the Jacobian of Φ at (x^k, μ^k) is given by

$$\Phi'(x^k, \mu^k) = \begin{bmatrix} \nabla_x^2 L(x^k, \mu^k) & A_k \\ \Gamma_k A_k^T & -\frac{1}{2}(\Theta_k)^2 \end{bmatrix}.$$

In order to apply quasi-Newton methods to solve (1.1), we replace $\nabla_x^2 L(x^k, \mu^k)$ by a symmetric positive definite matrix H_k. And in order to achieve superlinear convergence of our algorithm, we use the matrix

$$V_k = \begin{bmatrix} H_k & A_k \\ \Gamma_k A_k^T & \Theta_k \end{bmatrix}$$

instead of the Jacobian matrix of Φ in our algorithm. The use of Θ_k in V_k, instead of one half of the negative of its square, is very important to our superlinear convergence result. Now we formally state our algorithm.

ALGORITHM 2.1 (new QP-free method).

(S.0) *Initialization. Choose parameters:* $\alpha \in (0, 1/2)$, $\theta \in (0, 1)$, $\beta \in (0, 1)$, $\nu > 2$, $\tau \in (2, 3), \kappa \in (0, 1)$, $\bar{\mu} > 0$.
Choose data: x^0, *a strictly feasible point in* $\mathcal{F}$, $H_0 \in \mathbb{R}^{n \times n}$, *a symmetric positive definite matrix;* μ_i^0, *a scalar satisfying* $0 < \mu_i^0 \leq \bar{\mu}, i = 1, \ldots, m$. *Let* $w^0 = (x^0, \mu^0)$ *and set* $k := 0$.

(S.1) *Computation of the search direction.*
(i) *Compute* d^{k0} *and* λ^{k0} *by solving the following linear system in* (d, λ)*:*

$$V_k \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x^k) \\ 0 \end{bmatrix}. \tag{2.3}$$

If $d^{k0} = 0$*, stop. Otherwise go to* (ii) *below.*
(ii) *Compute* d^{k1} *and* λ^{k1} *by solving the linear system in* (d, λ)

$$V_k \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x^k) \\ \Gamma_k(\lambda_-^{k0})^3 \end{bmatrix}, \tag{2.4}$$

where the i*th element of* $(\lambda_-^{k0})^3$ *is defined by* $(\min\{\lambda_i^{k0}, 0\})^3$.
(iii) *Compute* d^{k2} *and* λ^{k2} *by solving the linear system in* (d, λ)

$$V_k \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x^k) \\ \Gamma_k(\lambda_-^{k0})^3 - \|d^{k1}\|^\nu \Gamma_k e \end{bmatrix}. \tag{2.5}$$

(iv) *Compute the search direction* d^k *and the approximate multiplier vector* λ^k *according to*

$$\begin{bmatrix} d^k \\ \lambda^k \end{bmatrix} = (1-\rho_k) \begin{bmatrix} d^{k1} \\ \lambda^{k1} \end{bmatrix} + \rho_k \begin{bmatrix} d^{k2} \\ \lambda^{k2} \end{bmatrix}, \tag{2.6}$$

where

$$\rho_k = (\theta - 1) \frac{\langle \nabla f(x^k), d^{k1} \rangle}{1 + |\sum_{i=1}^{m} \lambda_i^{k0}| \|d^{k1}\|^\nu}. \tag{2.7}$$

(v) *Compute a correction* $\hat{d}^k$*, solution of the linear square problem in* d

$$\min \frac{1}{2}\langle d, H_k d \rangle \quad \text{s.t. } g_i(x^k + d^k) + \langle \nabla g_i(x^k), d \rangle = -\psi_k \quad \text{for all } i \in I_k, \tag{2.8}$$

where $I_k = \{i \mid g_i(x^k) \geq -\lambda_i^k\}$ *and*

$$\psi_k = \max \left\{ \|d^k\|^\tau, \max_{i \in I_k} \left| \frac{\eta_i^k}{\sqrt{2\delta_i^k \lambda_i^k}} - 1 \right|^\kappa \|d^k\|^2 \right\}.$$

If (2.8) *has no solution or if* $\|\hat{d}^k\| > \|d^k\|$*, set* $\hat{d}^k = 0$.
(S.2) *Line search. Compute* t_k*, the first number* t *of the sequence* $\{1, \beta, \beta^2, \ldots\}$ *satisfying*

$$\begin{cases} f(x^k + td^k + t^2\hat{d}^k) \leq f(x^k) + \alpha t \langle \nabla f(x^k), d^k \rangle, \\ g_i(x^k + td^k + t^2\hat{d}^k) < 0, \quad i = 1, \ldots, m. \end{cases} \tag{2.9}$$

(S.3) *Update. Compute a new symmetric definite positive approximation* H_{k+1} *to the Hessian matrix. Set*

$$x^{k+1} = x^k + t_k d^k + t_k^2 \hat{d}^k, \quad \mu^{k+1} = \min\{\max\{\lambda^{k0}, \|d^k\|e\}, \bar{\mu}e\}. \tag{2.10}$$

Let $w^{k+1} = (x^{k+1}, \mu^{k+1})$*, and set* $k = k + 1$*. Go to* (S.1).

Remarks: The main purpose of system (2.3) is to enforce the direction d^{k0} to be a descent direction of f. However, this direction may converge to zero with a negative multiplier; thus a deep descent direction is supplied by solving (2.4), which is a slight perturbation of (2.3) by adding to the right side the negative part of the multipliers associated with (2.3). The purpose of (2.5) is to ensure the feasibility of the next iterate.

The rest of the section is devoted to showing that Algorithm 2.1 is well defined. To this end, we first show that if it terminates at (S.1)(i), the current point is an unconstrained stationary point (hence a trivial KKT point) of (1.1). Then we continue to show that the matrices involved in the linear systems (2.3)–(2.5) are nonsingular if the algorithm does not terminate at x^k. Finally we show that the direction d^k is actually a descent direction of the merit function f so that the line search is always possible and hence the algorithm is well defined.

LEMMA 2.2. *If $d^{k0} = 0$, then $\nabla f(x^k) = 0$, i.e., x^k is an unconstrained stationary point of f.*

Proof. If $d^{k0} = 0$, the system (2.3) reduces to

$$\begin{aligned} &\sum_{i=1}^{m} \lambda_i^{k0} \nabla g_i(x^k) + \nabla f(x^k) = 0, \\ &\theta_i^k \lambda_i^{k0} = 0, \quad i = 1, \ldots, m. \end{aligned} \tag{2.11}$$

Since $g_i(x^k) < 0$ for all $i = 1, \ldots, m$, $\theta_i^k > 0$ by its definition. It follows from (2.11) that $\nabla f(x^k) = 0$. □

Without loss of generality, we assume that the algorithm never terminates at (S.1)(i), i.e., $d^{k0} \neq 0$. Since x^k is strictly feasible and $\mu^k > 0$, we have $\eta^k > 0$ and $\theta^k > 0$. We now show the matrix V_k is nonsingular. Let $u \in \mathbb{R}^n$ and $v \in \mathbb{R}^m$ be a solution of

$$H_k u + A_k v = 0, \tag{2.12}$$

$$\Gamma_k A_k^T u + \Theta_k v = 0. \tag{2.13}$$

It follows from (2.13) that

$$v = -\Theta_k^{-1} \Gamma_k A_k^T u. \tag{2.14}$$

Substituting (2.14) into (2.12) and multiplying (2.12) by u^T, we have

$$u^T H_k u - u^T A_k \Theta_k^{-1} \Gamma_k A_k^T u = 0. \tag{2.15}$$

Since H_k is positive definite and the matrix $(-A_k \Theta_k^{-1} \Gamma_k A_k^T)$ is positive semidefinite, we have $u = 0$ from (2.15). It follows (2.14) that $v = 0$. Hence V_k is nonsingular. So the quantities $d^{k1}, d^{k2}, \lambda^{k1}, \lambda^{k2}$ are well defined. The nonsingularity of V_k implies that the matrix $D_k = \Gamma_k A_k^T H_k^{-1} A_k - \Theta_k$ is also nonsingular. Let

$$B_k = H_k^{-1} A_k D_k^{-1} \quad \text{and} \quad Q_k = H_k^{-1}(I - A_k \Gamma_k B_k^T).$$

The following relations are important to our analysis.

$$\begin{cases} d^{k0} = -Q_k \nabla f(x^k), \quad \lambda^{k0} = -\Gamma_k B_k^T \nabla f(x^k), \\ d^{k1} = d^{k0} + B_k \Gamma_k (\lambda_-^{k0})^3, \quad \lambda^{k1} = \lambda^{k0} - D_k^{-1} \Gamma_k (\lambda_-^{k0})^3, \\ d^{k2} = d^{k1} - \|d^{k1}\|^\nu B_k \Gamma_k e, \quad \lambda^{k2} = \lambda^{k1} + \|d^{k1}\|^\nu D_k^{-1} \Gamma_k e, \\ d^k = d^{k1} - \rho_k \|d^{k1}\|^\nu B_k \Gamma_k e, \quad \lambda^k = \lambda^{k1} + \rho_k \|d^{k1}\|^\nu D_k^{-1} \Gamma_k e. \end{cases} \tag{2.16}$$

It is also easy to see that (d^k, λ^k) satisfies the linear system

$$V_k \begin{bmatrix} d^k \\ \lambda^k \end{bmatrix} = \begin{bmatrix} -\nabla f(x^k) \\ \Gamma_k(\lambda_-^{k0})^3 - \rho_k \|d^{k1}\|^\nu \Gamma_k e \end{bmatrix}. \tag{2.17}$$

The following results show d^{k1} is a very deep descent direction of f over d^{k0}.

LEMMA 2.3. (i) $\langle \nabla f(x^k), d^{k0} \rangle \le -\langle d^{k0}, H_k d^{k0} \rangle$.

(ii)

$$\langle \nabla f(x^k), d^{k1} \rangle = \langle \nabla f(x^k), d^{k0} \rangle - \sum_{i:\lambda_i^{k0}<0} (\lambda_i^{k0})^4.$$

(iii) $\langle \nabla f(x^k), d^k \rangle \le \theta \langle \nabla f(x^k), d^{k1} \rangle$.

Proof. (i) can be proved similarly to [19, Lemma 3.2]. It follows from (2.16) that

$$\begin{aligned} \langle \nabla f(x^k), d^{k1} \rangle &= \langle \nabla f(x^k), d^{k0} \rangle + \langle \nabla f(x^k), B_k \Gamma_k (\lambda_-^{k0})^3 \rangle \\ &= \langle \nabla f(x^k), d^{k0} \rangle + \langle \Gamma_k B_k^T \nabla f(x^k), (\lambda_-^{k0})^3 \rangle \\ &= \langle \nabla f(x^k), d^{k0} \rangle - \langle \lambda^{k0}, (\lambda_-^{k0})^3 \rangle \\ &= \langle \nabla f(x^k), d^{k0} \rangle - \sum_{i:\lambda_i^{k0}<0} (\lambda_i^{k0})^4. \end{aligned}$$

This establishes (ii). Now we prove (iii). Straight calculation gives

$$\begin{aligned} &\langle \nabla f(x^k), d^k \rangle \\ &= \langle \nabla f(x^k), d^{k1} \rangle - \rho_k \|d^{k1}\|^\nu \nabla f(x^k)^T B_k \Gamma_k e \\ &= \langle \nabla f(x^k), d^{k1} \rangle + (\theta - 1) \frac{\langle \nabla f(x^k), d^{k1} \rangle}{1 + \|d^{k1}\|^\nu |\sum_{i=1}^m \lambda_i^{k0}|} \|d^{k1}\|^\nu \sum_{i=1}^m \lambda_i^{k0} \\ &\le \theta \langle \nabla f(x^k), d^{k1} \rangle. \end{aligned}$$

The last inequality above used the fact that $\langle \nabla f(x^k), d^{k1} \rangle < 0$. ∎

It is easy by following standard analysis of [19, Proposition 3.3] to show that there exists a t_k, the first number of the sequence $\{1, \beta, \beta^2, \ldots\}$, which satisfies the line search (2.9), so that we are able to get the next iterate x^{k+1} from the current iterate x^k, and $\mu^{k+1} > 0$. Therefore we claim as follows.

PROPOSITION 2.4. *Algorithm* 2.1 *is well defined.*

3. Global convergence. To study the convergence behavior of the algorithm, let $\{w^k = (x^k, \mu^k)\}$ be a sequence generated by Algorithm 2.1. In addition to existence of a strictly feasible point x^0, we need the following assumptions.

(H1) For all $x \in \mathcal{F}$, the linear independence condition holds at x, i.e., the set of vectors $\{\nabla g_i(x) : i \in I(x)\}$ is linearly independent, where $I(x) = \{i : g_i(x) = 0\}$.

(H2) The set $\mathcal{F} \cap \{x |\ f(x) \le f(x^0)\}$ is compact.

(H3) There exist positive numbers σ_1 and σ_2 such that $\sigma_1 \|d\|^2 \le \langle d, H_k d \rangle \le \sigma_2 \|d\|^2$ for all k and all $d \in \mathbb{R}^n$.

We have assumed that $d^{k0} \ne 0$ for all $k = 0, 1, \ldots$. It follows from Lemma 2.3 that $d^k \ne 0$. Hence the updating rule in (2.10) implies that $\mu^{k+1} > 0$, which in turn

implies $\eta^{k+1} > 0$ and $\theta^{k+1} > 0$. Therefore V_k is nonsingular for all k. Moreover, we have a strong result on the nonsingularity of V_k.

LEMMA 3.1. *The sequence* $\{\|V_k^{-1}\|\}$ *is bounded for* $k = 0, 1, \ldots$.

Proof. It is already known that V_k is nonsingular for all k. Now suppose to the contrary that there exists a subsequence $\{H_k, x^k, \mu^k\}_K$ such that

$$\|V_k^{-1}\| \to \infty \quad \text{as} \quad k \in K \to \infty. \tag{3.1}$$

Since $\{\eta^k\}$ and $\{\theta^k\}$ are bounded, without loss of generality, we may assume that

$$\eta^k \to \eta^*, \qquad \theta^k \to \theta^*.$$

By (H2), without loss of generality, we may assume that $x^k \to x^* \in \mathcal{F}$. It is easy to see that $\theta_i^* = 0$ only if $g_i(x^*) = 0$ and $\eta_i^* = 1$. Moreover, the relation (2.1) is observed in the limit, i.e., for all i,

$$(\eta_i^*)^2 + (\theta_i^*)^4 \geq 3 - 2\sqrt{2}.$$

By (H3), without loss of generality, we may assume that $H_k \to H_*$, a positive definite matrix in $\mathbb{R}^{n\times n}$. Putting all the limits together, we have

$$V_k \to V_* = \begin{bmatrix} H_*, & \nabla g(x^*) \\ \operatorname{diag}(\eta^*)\nabla g^T(x^*), & -\sqrt{2}\operatorname{diag}(\theta^*) \end{bmatrix}.$$

Now we show that V_* is nonsingular, which contradicts the assumption at the beginning of the proof and hence establishes the result. Let $(u, v) \in \mathbb{R}^{n+m}$ be a solution of the linear system

$$H_* u + \nabla g(x^*) v = 0, \tag{3.2}$$

$$\operatorname{diag}(\eta^*)\nabla g^T(x^*) u - \sqrt{2}\operatorname{diag}(\theta^*) v = 0. \tag{3.3}$$

We show $(u, v) = (0, 0)$. First we consider such i for which $\eta_i^* = 0$; it follows from the relation (2.1) that $\theta_i^* > 0$, hence $v_i = 0$ by (3.3). For i such that $\eta_i^* \neq 0$, multiplying (3.2) by u^T we get

$$u^T H_* u + \sqrt{2} \sum_{i:\eta_i^* \neq 0} \frac{\theta_i^*}{\eta_i^*} v_i^2 = 0.$$

Since H_* is positive definite, we must have

$$u = 0 \quad \text{and} \quad \theta_i^* v_i = 0 \quad \text{for all } i \text{ such that} \quad \eta_i^* \neq 0.$$

If $\theta_i^* = 0$ for some i, then $g_i(x^*) = 0$. Hence (3.2) implies

$$\sum_{i:g_i(x^*)=0} \nabla g_i(x^*) v_i = 0.$$

We note that the linear independence condition holds, which implies $v_i = 0$ for all $i \in I(x^*)$. This proves $(u, v) = (0, 0)$ and hence V_* is nonsingular, which contradicts the assumption (3.1). Hence the lemma holds. □

The following two corollaries are some important consequences of Lemma 3.1.

COROLLARY 3.2. *The sequences of* $\{(d^{k0}, \lambda^{k0})\}$, $\{(d^{k1}, \lambda^{k1})\}$, *and* $\{(d^{k2}, \lambda^{k2})\}$ *are all bounded on* $k = 0, 1 \ldots$.

Proof. The matrix sequence $\{V_k^{-1}\}$ is uniformly bounded as proved in Lemma 3.1; $\{x^k\}$ is bounded due to the assumption (H2). The solvability of system (2.3) implies that $\{(d^{k0}, \lambda^{k0})\}$ is bounded, which implies the boundedness of the right-hand side of (2.4). Hence $\{(d^{k1}, \lambda^{k1})\}$ is also bounded. The boundedness of $\{d^{k1}\}$ implies the boundedness of the right-hand side of system (2.5), and thus the sequence $\{(d^{k2}, \lambda^{k2})\}$ is also bounded. □

COROLLARY 3.3. *There exists* $\kappa_1 > 0$ *such that for all* $k = 0, 1, \ldots,$

$$\|d^k - d^{k1}\| \leq \kappa_1 \|d^{k1}\|^\nu.$$

Proof. Since $\{d^{k1}\}$ and $\{\lambda^{k0}\}$ are bounded, the definition of ρ_k yields the boundedness of the sequence $\{\rho_k\}$. Let

$$\kappa_1 = 2 \sup\{\rho_k\} \sup\{\|V_k^{-1}\|\}.$$

It is from Lemma 3.1 that κ_1 is a finite scalar. Now define

$$\Delta d^k = d^k - d^{k1} \quad \text{and} \quad \Delta \lambda^k = \lambda^k - \lambda^{k1}.$$

Then the vector $(\Delta d^k, \Delta \lambda^k)$ is the solution of

$$V_k \begin{bmatrix} \Delta d^k \\ \Delta \lambda^k \end{bmatrix} = \begin{bmatrix} 0 \\ -\rho_k \|d^{k1}\|^\nu \Gamma_k e \end{bmatrix}. \tag{3.4}$$

It is easy to see that $\eta_i^k \leq 2$, (3.4) yield

$$\|(\Delta d^k, \Delta \lambda^k)\| \leq \kappa_1 \|d^{k1}\|^\nu$$

so that

$$\|\Delta d^k\| \leq \kappa_1 \|d^{k1}\|^\nu. \qquad \square$$

LEMMA 3.4. *Let* x^* *be an accumulation point of* $\{x^k\}$ *and suppose that* $\{x^k\}_K \to x^*$. *If* $\{d^k\}_K \to 0$, *then* x^* *is a KKT point of* (1.1) *and* $\{\lambda^{k0}\}_K \to \lambda^*$, *where* λ^* *is the unique multiplier vector associated with* x^*.

Proof. It follows from Lemma 2.3 and the assumption (H3) that

$$\langle \nabla f(x^k), d^k \rangle \leq -\sigma_1 \theta \|d^{k0}\|^2 - \theta \sum_{i:\lambda_i^{k0}<0} (\lambda_i^{k0})^4.$$

Hence $\{d^k\}_K \to 0$ implies that

$$\{d^{k0}\}_K \to 0 \qquad \text{and} \qquad \sum_{i:\lambda_i^{k0}<0} (\lambda_i^{k0})^4 \to 0, \ k \in K. \tag{3.5}$$

Since $\{\lambda^{k0}\}$ and $\{\mu^k\}$ are bounded, there exist $\lambda^*, \mu^* \in \mathbb{R}^m$, and a subset K' of K such that (λ^*, μ^*) is the limit of the subsequence $\{(\lambda^{k0}, \mu^k)\}_{K'}$. We now show that (x^*, λ^*) is a KKT point of (1.1). First it follows from (3.5) that $\lambda^* \geq 0$, i.e., the limit

of the multipliers is nonnegative. Now we consider that for $i \notin I(x^*)$, θ_i^k converges on K' and satisfies

$$\lim_{k\in K'} \theta_i^k = \theta_i^* = \left(1 - \frac{\mu_i^*}{\sqrt{g_i^2(x^*) + (\mu_i^*)^2}}\right)^{1/2} > 0. \tag{3.6}$$

Taking the limits in both sides of (2.3) on K', we obtain by noting $d^{k0} \to 0$ as $k \in K \to \infty$

$$\sum_{i=1}^{m} \lambda_i^* \nabla g_i(x^*) = -\nabla f(x^*), \tag{3.7}$$

$$\lim_{k\in K} \theta_i^k \lambda_i^* = 0, \quad i = 1, \ldots, m. \tag{3.8}$$

Note that θ_i^k does not necessarily converge for $i \in I(x^*)$. (3.8) and (3.6) imply

$$\lambda_i^* = 0 \qquad \text{for } \ i \notin I(x^*).$$

Therefore (x^*, λ^*) satisfies

$$\nabla f(x^*) + \sum_{i\in I(x^*)} \lambda_i^* \nabla g_i(x^*) = 0.$$

Furthermore, $x^* \in \mathcal{F}$ because $\{x^k\} \subseteq \mathcal{F}$. Hence (x^*, λ^*) is a KKT point of (1.1). □

The proof of the following result is the same as [19, Lemma 3.8] by using Lemma 3.4.

LEMMA 3.5. *Let x^* be an accumulation point of $\{x^k\}$ and $\{x^k\}_K \to x^*$. If*

$$\{d^{k-1}\}_K \to 0, \tag{3.9}$$

then x^* is a KKT point.

The following result addresses the case where the condition (3.9) does not hold on a subsequence.

LEMMA 3.6. *Let x^* be an accumulation point of $\{x^k\}$ and $\{x^k\}_K \to x^*$. If*

$$\inf\{\|d^{k-1}\|\}_K > 0,$$

then x^* is a KKT point.

Proof. What we proved in Lemma 3.4 is the following: If $\langle \nabla f(x^k), d^{k1}\rangle \to 0$, then x^* must be a KKT point. Now we suppose that x^* is not a KKT point of (1.1). Then there must exist $\gamma > 0$ satisfying

$$\langle \nabla f(x^k), d^{k1}\rangle \le -\gamma \qquad \text{for all } k \in K, \tag{3.10}$$

which implies that there exists $\delta > 0$

$$\liminf_{k\in K} \|d^{k1}\| > \delta.$$

A direct consequence of (3.10) is that there exists $\bar{\rho} > 0$ such that $\rho_k \geq \bar{\rho}$ for all $k \in K'$. This can be proved by employing the boundedness of $\{d^{k1}\}$, $\{\lambda^{k0}\}$ and (3.10).

We now show that there exists $\bar{t} > 0$ such that for all $t \in [0, \bar{t}]$ and $k \in K$ large enough, two line search conditions in (2.9) are satisfied. It follows from (3.10) that, for $k \in K$ large enough,

$$\langle \nabla f(x^k), d^k \rangle \leq -\theta\gamma. \tag{3.11}$$

From the proof of [19, Proposition 3.9], we have following basic relations (because of the differentiability of the functions) for the functions $f, g_i, i = 1, \ldots, m$, and $k \in K$ large enough

$$\begin{aligned} & f(x^k + td^k + t^2\hat{d}^k) - f(x^k) - \alpha t \langle \nabla f(x^k), d^k \rangle \\ & \leq t \Bigg\{ \sup_{\xi \in [0,1]} \|\nabla f(x^k + t\xi d^k + t^2\xi^2 \hat{d}^k) - \nabla f(x^k)\| \|d^k\| \\ & \qquad + 2t \sup_{\xi \in [0,1]} \|\nabla f(x^k + t\xi d^k + t^2\xi^2 \hat{d}^k)\| \|\hat{d}^k\| - (1-\alpha)\theta\gamma \Bigg\}, \end{aligned} \tag{3.12}$$

$$g_i(x^k + td^k + t^2\hat{d}^k) \leq g_i(x^k) + t\{u_i^k(t) + \langle \nabla g_i(x^k), d^k \rangle\}, \tag{3.13}$$

where for $i = 1, \ldots, m$

$$\begin{aligned} u_i^k(t) = & \sup_{\xi \in [0,1]} \|\nabla g_i(x^k + t\xi d^k + t^2\xi^2 \hat{d}^k) - \nabla g_i(x^k)\| \|d^k\| \\ & + 2t \sup_{\xi \in [0,1]} \|\nabla g_i(x^k + t\xi d^k + t^2\xi^2 \hat{d}^k)\| \|\hat{d}^k\|. \end{aligned}$$

We note that $\{\hat{d}^k\}$ is also bounded due to the relation $\|\hat{d}^k\| \leq \|d^k\|$. Hence (3.12) implies that there exists $\bar{t}_f > 0$, independent of k, such that, for $k \in K$ large enough,

$$f(x^k + td^k + t^2\hat{d}^k) - f(x^k) - \alpha t \langle \nabla f(x^k), d^k \rangle \leq 0 \quad \text{for all } t \in [0, \bar{t}_f]. \tag{3.14}$$

Since $\{\|d^{k-1}\|\}$ is bounded away from zero on K, the definition in (2.10) implies that there exists $\hat{\mu} > 0$ such that $\mu^k \geq \hat{\mu}$ for all $k \in K$ large enough. It follows from (2.17) that for all $i = 1, \ldots, m$

$$\langle \nabla g_i(x^k), d^k \rangle = (\lambda_-^{k0})_i^3 + \frac{\sqrt{2}\lambda_i^k}{\eta_i^k}\theta_i^k - \rho_k \|d^{k1}\|^\nu.$$

For these indices $i \notin I(x^*)$, there exists $\gamma_1 > 0$ such that

$$g_i(x^k) \leq -\gamma_1$$

for all $k \in K$ large enough. It is easy to see from the boundedness of the directions d^k and $\hat{d}^k$ that for these indices, there exists $\bar{t}_i$, $i \notin I(x^*)$ independent of k such that

$$g_i(x^k + td^k + t^2\hat{d}^k) < 0$$

for all $t \in (0, \bar{t}_i]$. For $i \in I(x^*)$, we have for $k \in K$ sufficiently large

$$g_i(x^k) \to 0, \quad \eta_i^k \to 1,$$

and

$$\theta_i^k = \left(\frac{g_i^2(x^k)}{\sqrt{(\mu_i^k)^2 + g_i^2(x^k)}(\mu_i^k + \sqrt{(\mu_i^k)^2 + g_i^2(x^k)})} \right)^{1/2} \leq \frac{1}{\sqrt{2}\hat{\mu}} |g_i(x^k)|.$$

The boundedness of $\{\lambda_i^k\}$, the continuity of $u_i^k(t)$ as a function of t, and $u_i^k(0) = 0$ imply that there exists $\bar{t}_i > 0$, independent of k, such that, for $t \in [0, \bar{t}_i]$, $i \in I(x^*)$, and $k \in K$ large enough,

$$u_i^k(t) + \langle \nabla g_i(x^k), d^k \rangle \leq u_i^k(t) + (\lambda_-^{k0})_i^3 - \frac{|\lambda_i^k|}{\hat{\mu}\eta_i^k} g_i(x^k) - \bar{\rho}\delta^\nu < 0.$$

This fact, together with (3.13) and $g_i(x^k) < 0$, implies

$$g_i(x^k + td^k + t^2\hat{d}^k) < 0 \qquad \text{for all } t \in [0, \bar{t}_i].$$

Let

$$\bar{t} = \min\{\bar{t}_f, \bar{t}_1, \ldots, \bar{t}_m\}.$$

The line search rule gives $t_k \geq \beta\bar{t}$ for all $k \in K$ sufficiently large. It follows from (3.11) and (3.14) that

$$f(x^k + t_k d^k + t_k^2 \hat{d}^k) - f(x^k) \leq -\alpha\beta\bar{t}\theta\gamma,$$

which drives $f(x^k) \to -\infty$, a contradiction to (H2). Hence x^* is a KKT point. □

Putting the results in Lemmas 3.4–3.6 together, we obtain our global convergence result.

THEOREM 3.7. *If (x^*, λ^*) is a limit point of the sequence $\{(x^k, \lambda^{k0})\}$ generated by Algorithm* 2.1*, then (x^*, λ^*) is a KKT point of* (1.1).

4. Local convergence. Beside the assumptions of (H1)–(H3), we also need the following assumptions in our superlinear convergence analysis. Let $w^* = (x^*, \lambda^*)$ be an accumulation point of the sequence $\{(x^k, \lambda^{k0})\}$. Then, according to Theorem 3.7, w^* is a KKT point.

(H4) The strict complementarity condition holds at w^*, i.e., $\lambda^* - g(x^*) > 0$.

(H5) The second-order sufficient condition holds at w^*, i.e., the Hessian $\nabla_x^2 L(x^*, \lambda^*)$ is positive definite on the space $\{u : \langle \nabla g_i(x^*), u \rangle = 0, i \in I(x^*)\}$.

(H6) The scalar $\bar{\mu} > 0$ is sufficiently large such that $\lambda^* \leq \bar{\mu}$.

The next result is on the convergence of the whole sequence to an isolated point. The original version of the result is due to Moré and Sorensen [17]; here we cite a slightly different version of the result from [15, Proposition 5.4].

PROPOSITION 4.1. *Assume that $w^* \in \mathbb{R}^t$ is an isolated accumulation point of a sequence $\{w^k\} \subset \mathbb{R}^t$ such that, for every subsequence $\{w^k\}_K$ converging to w^*, there is an infinite subset $\tilde{K} \subseteq K$ such that $\{\|w^{k+1} - w^k\|\}_{\tilde{K}} \to 0$. Then the whole sequence $\{w^k\}$ converges to w^*.*

The proposition is used in the following result.

LEMMA 4.2. *Under the stated assumptions, the whole sequence $\{x^k\}$ converges to x^*, and $\{\lambda^{k0}\}$ converges to the unique multiplier λ^*.*

Proof. The second-order sufficient condition and the linear independence condition mean that x^* is an isolated accumulation point of $\{x^k\}$; see [25]. Let $\{x^k\}_K$ be

a subsequence converging to x^*. It is enough, according to Proposition 4.1, to show that there exists a subsubsequence of $\{x^k\}_{K'}$ with $K' \subseteq K$ satisfying

$$\|x^{k+1} - x^k\| \to 0 \quad \text{as} \quad k \in K' \to \infty.$$

Since

$$\|x^{k+1} - x^k\| \le \|d^k\| + \|\hat{d}^k\| \le 2\|d^k\|,$$

it is sufficient to show that

$$\|d^k\| \to 0 \quad \text{as} \quad k \in K' \to \infty. \tag{4.1}$$

Assume to the contrary that there does not exist such a subsequence K' such that (4.1) holds, i.e.,

$$\liminf_{k \in K} \|d^k\| > 0.$$

Corollary 3.3 implies that

$$\liminf_{k \in K} \|d^{k1}\| > 0. \tag{4.2}$$

Without loss of generality, we assume that

$$\eta^k \to \eta^*, \ \theta^k \to \theta^*, \ H_k \to H_* \quad \text{as} \quad k \in K \to \infty.$$

(H3) implies that H_* is positive definite. Hence

$$V_k \to V_* = \begin{bmatrix} H_*, & \nabla g(x^*) \\ \operatorname{diag}(\eta^*)\nabla g^T(x^*), & -\sqrt{2}\operatorname{diag}(\theta^*) \end{bmatrix}.$$

Following a similar proof of Lemma 3.1, we can prove that V_* is nonsingular. It is proved in Lemma 3.4 that $\lambda^{k0} \to \lambda^*$. Hence we must have $\lambda_-^{k0} \to 0$. By the continuity properties of $V_k \to V_*$ and of the function $\nabla f(x)$, the nonsingularity of V_* implies (d^{k1}, λ^{k1}) converges to the unique solution of the linear system

$$V_* \begin{bmatrix} d \\ \lambda \end{bmatrix} = \begin{bmatrix} -\nabla f(x^*) \\ 0 \end{bmatrix}. \tag{4.3}$$

It is already known that (x^*, λ^*) is a KKT point and $(0, \lambda^*)$ is the unique solution of the above system. So we must have $d^{k1} \to 0$, which contradicts with (4.2). Hence there must exist a subsequence K' such that (4.1) is satisfied. Proposition 4.1 implies x^k converges to x^*. The convergence of λ^{k0} to λ^* is a direct consequence of Lemma 3.4. This completes the proof. □

We stress that we do not use the strict complementarity condition in proving the convergence of $\{x^k\}$ to x^*. The following result is an easy consequence of Lemma 4.2.

COROLLARY 4.3. *For k large enough, $I_k = I(x^*)$ and*

(i) $d^k \to 0$, $d^{k0} \to 0$, $d^{k1} \to 0$.

(ii) $\lambda^k \to \lambda^*$, $\lambda^{k1} \to \lambda^*$.

(iii) $\mu^k \to \lambda^*$.

Proof. We first prove that $d^{k0} \to 0$. We note that $\{d^{k0}\}$ is bounded. Let d^{*0} be an accumulation point of some subsequence $\{d^{k0}\}_K$. It is already shown that $\lambda^{k0} \to \lambda^*$. Hence (d^{*0}, λ^*) must be the unique solution of the linear system (4.3), where V_* is a limit of $\{V_k\}_K$ and V_* must be nonsingular by Lemma 3.1. It is already known that (x^*, λ^*) is a KKT point and $(0, \lambda^*)$ is the unique solution of (4.3). So we must have $\{d^{k0}\}_K \to 0$. Hence $d^{k0} \to 0$ as $k \to \infty$. $d^k \to 0$ and $d^{k1} \to 0$ follow from Lemma 2.3. This establishes (i). Just as we show that $\{V_k\}$ is uniformly nonsingular in Lemma 3.1, we can show that the matrix sequence $\{D_k\}$ is uniformly nonsingular. (ii) follows from the relations in (2.16). (H6) and the updating rule (2.10) give (iii). It is well known that if the strict complementarity condition holds at x^*, then the set I_k detects $I(x^*)$ completely whenever (x^k, λ^k) is close to (x^*, λ^*). □

The following simple observation is very useful in the analysis later.

LEMMA 4.4. *For $i \in I(x^*)$,*

(i) $\eta_i^k \to 1,\ \theta_i^k \to 0.$

(ii) $\lambda_i^k \theta_i^k / g_i(x^k) \to -1/\sqrt{2}, \quad \lambda_i^k \delta_i^k \to 1/\sqrt{2}.$

For $i \notin I(x^*)$, we have (iii) $\eta_i^k \to 0,\ \theta_i^k \to 1.$

Proof. Straightforward calculation on η_i^k and θ_i^k gives the results in (i) and (iii). For (ii), we note that

$$(\theta_i^k)^2 = 1 - \frac{\mu_i^k}{\sqrt{g_i^2(x^k) + (\mu_i^k)^2}} = \frac{g_i^2(x^k)}{(\mu_i^k + \sqrt{g_i^2(x^k) + (\mu_i^k)^2})\sqrt{g_i^2(x^k) + (\mu_i^k)^2}}.$$

The limits in Corollary 4.3 and the strict complementarity condition yield the results in (ii). □

Now we consider the case $i \notin I(x^*)$. It follows from (2.3) that d^{k0} satisfies

$$\eta_i^k \langle \nabla g_i(x^k), d^{k0} \rangle - \sqrt{2}\theta_i^k \lambda_i^{k0} = 0.$$

Lemma 4.4 implies that

$$\lambda_i^{k0} = o(\|d^{k0}\|) \quad \text{for all } i \notin I(x^*). \tag{4.4}$$

So it follows from (2.16) that

$$d^k = d^{k0} + o(\|d^{k0}\|^2) \quad \text{and} \quad d^{k1} = d^{k0} + o(\|d^{k0}\|^2). \tag{4.5}$$

In order that the steplength of unit is accepted eventually, and hence the superlinear convergence is obtained, we assume that

(H7) The sequence of matrices $\{H_k\}$ satisfies

$$\frac{\|P_k(H_k - \nabla_x^2 L(x^*, \lambda^*))d^k\|}{\|d^k\|} \to 0,$$

where

$$P_k = I - N_k(N_k^T N_k)^{-1} N_k^T \quad \text{and} \quad N_k = [\nabla g_i(x^k) :\ i \in I(x^*)].$$

We now develop two lemmas, which are the correspondence of Lemmas 4.3 and 4.4 in [19] and are essential for the steplength of one to be accepted.

LEMMA 4.5. *For k large enough, the direction d^k can be decomposed into*

$$d^k = P_k d^k + \tilde{d}^k$$

with

$$\|\tilde{d}^k\| = O\left(\sum_{i\in I(x^*)} g_i^2(x^*)\right)^{1/2} + o(\|d^{k0}\|^2).$$

Proof. It follows from (2.17) that d^k satisfies, for $i = 1, \ldots, m$,

$$\eta_i^k \langle \nabla g_i(x^k), d^k\rangle - \sqrt{2}\theta_i^k \lambda_i^k = \eta_i^k (\min\{\lambda_i^{k0}, 0\})^3 - \eta_i^k \rho_k \|d^{k1}\|^\nu.$$

In particular,

$$N_k^T d^k = h^k,$$

where h^k is a $|I(x^*)|$-vector whose components are the numbers

$$\frac{\sqrt{2}\theta_i^k \lambda_i^k}{\eta_i^k} - (\min\{\lambda_i^{k0}, 0\})^3 - \rho_k \|d^{k1}\|^\nu, \qquad i \in I(x^*).$$

It follows from Lemma 4.4 and (4.5) that

$$\|h^k\| = O\left(\sum_{i\in I(x^*)} g_i^2(x^*)\right)^{1/2} + o(\|d^{k0}\|^2).$$

It is easy to know that

$$\tilde{d}^k = N_k (N_k^T N_k)^{-1} h^k.$$

This establishes the result. □

It follows from the relation (2.2) and the system (2.17) that for $i \in I(x^*)$

$$-\psi_k - g_i(x^k + d^k) = -\psi_k + \left(\frac{\eta_i^k}{\sqrt{2}\delta_i^k \lambda_i^k} - 1\right) \langle \nabla g_i(x^k), d^k\rangle + O(\|d^k\|^2).$$

Following almost the same argument as in [19, Lemma 4.4], we have the following result, regarding the direction $\hat{d}^k$. We omit the proof.

LEMMA 4.6. *For k large enough, the direction $\hat{d}^k$ is obtained as the solution of* (2.8) *and it satisfies*

$$\|\hat{d}^k\| = O\left(\max\left\{\|d^k\|^2, \max_{i\in I(x^*)} \left|\frac{\eta_i^k}{\sqrt{2}\delta_i^k \lambda_i^k} - 1\right| \|d^k\|\right\}\right) = o(\|d^k\|).$$

Almost all preparation is obtained to ensure the steplength of one to be eventually accepted. The rest our of analysis is based on Taylor expansion and we omit the detailed proof. Two good examples for the proof of the next result can be found in [19, Proposition 4.5] and [11, Theorem 4.1].

PROPOSITION 4.7. *For k large enough, the step $t_k = 1$ is accepted by the line search.*

Now we consider the superlinear convergence of $\{x^k\}$. Note that the iteration can be cast as

$$x^{k+1} = x^k + d^{k0} + o(\|d^{k0}\|). \tag{4.6}$$

We first consider the following iterate. Let (d_f^k, λ_f^k) be a solution of the linear system

$$\begin{bmatrix} H_k & N_k \\ N_k^T & 0 \end{bmatrix} \begin{bmatrix} d_f^k \\ \lambda_f^k \end{bmatrix} = - \begin{bmatrix} \nabla f(x^k) \\ g_{I_k}(x^k) \end{bmatrix}, \tag{4.7}$$

where the elements of $g_{I_k}(x^k)$ are given by $g_i(x^k), i \in I_k$. The linear system (4.7) is well studied in a neighborhood of x^* and much is known about its convergence properties under stated conditions. We cite several facts from the literature. The symbol $\sim$ below means that the ratio of the expression on the left-hand side to the right-hand side is both bounded above and bounded away from zero, as $k \to \infty$.

LEMMA 4.8. (i) [22, *Lemma* 4] $\|d_f^k\| \sim \|g_{I_k}(x^k)\| + \|P_k \nabla f(x^k)\|$.

(ii) [16, *Lemma* 21] $\|x^k - x^*\| \sim \|g_{I_k}(x^k)\| + \|P_k \nabla f(x^k)\|$.

(iii) [5, *Theorem* 5.2] $\|x^k + d_f^k - x^*\| = o(\|x^k - x^*\|)$.

An immediate consequence of Lemma 4.8 is

$$\|x^k + d_f^k + o(\|d_f^k\|) - x^*\| = o(\|x^k - x^*\|). \tag{4.8}$$

To establish superlinear convergence of the iterate (4.6), we prove

$$d^{k0} = d_f^k + o(\|d^{k0}\|). \tag{4.9}$$

By definition, the direction d^{k0} satisfies

$$H_k d^{k0} + \sum_{i=1}^{m} \lambda_i^{k0} \nabla g_i(x^k) = -\nabla f(x^k), \tag{4.10}$$

$$\eta_i^k \langle \nabla g_i(x^k), d^{k0} \rangle + \sqrt{2} \lambda_i^{k0} \delta_i^k g_i(x^k) = 0, \quad i = 1, \ldots, m. \tag{4.11}$$

We have from (4.4) and (4.10)

$$H_k d^{k0} + \sum_{i \in I(x^*)} \lambda_i^{k0} \nabla g_i(x^k) = -\nabla f(x^k) + o(\|d^{k0}\|). \tag{4.12}$$

Since $\lambda^{k0} \to \lambda^*$ from Lemma 4.2, it is easy to calculate $\lambda_i^{k0} \delta_i^k = 1/\sqrt{2}$ for $i \in I(x^*)$, as we have proved for λ^k in Lemma 4.4. Hence (4.11) yields

$$\langle \nabla g_i(x^k), d^{k0} \rangle + g_i(x^k) = o(\|d^{k0}\|), \quad i \in I(x^*). \tag{4.13}$$

We note that $I_k = I(x^*)$ for k sufficiently large, and the matrix in (4.7) is nonsingular in a sufficiently small neighborhood of x^*. Comparing (4.12) and (4.13) with (4.7), we have the relation (4.9) because of the nonsingularity of the matrix involved. Putting the facts (4.6), (4.8), and (4.9) together, we obtain superlinear convergence of Algorithm 2.1.

THEOREM 4.9. *Under the stated assumptions, we have*

$$\|x^{k+1} - x^*\| = o(\|x^k - x^*\|).$$

5. Numerical results. Algorithm 2.1 was implemented in MATLAB and tested on a DEC George Server 8200 over a set of problems from [14]. The details about the implementation are described as follows.

(a) The termination criterion. By Lemma 2.2, $\|d^{k0}\| = 0$ only ensures that x^k is an unconstrained stationary point of f. Hence we replaced the termination criterion in Algorithm 2.1 by

$$\|\Phi(x^k, \lambda^{k0})\| \le \text{tol}, \tag{5.1}$$

where Φ is defined by (1.3) and tol is the tolerance. It follows from the properties of the Fischer–Burmeister function that the final iterate must be an approximate KKT point of (1.1). This termination criterion worked quite well for our test problems.

(b) BFGS update. The initial Lagrangian Hessian estimate is $H_0 = I$, and H_k is updated by the damped BFGS formula described in [21]. In particular, we set

$$H_{k+1} = H_k - \frac{H_k s_k s_k^T H_k}{s_k^T H_k s_k} + \frac{y_k y^T}{s_k y_k},$$

where

$$y_k = \begin{cases} \hat{y}_k, & \hat{y}_k^T s_k \ge 0.2 s_k^T H_k s_k, \\ \theta_k \hat{y}_k + (1-\theta_k) H_k s_k & \text{otherwise}, \end{cases}$$

and

$$\begin{cases} s_k = x^{k+1} - x^k, \\ \hat{y}_k = \nabla f(x^{k+1}) - \nabla f(x^k) + (\nabla g(x^{k+1}) - \nabla g(x^k))\lambda^{k0}, \\ \theta_k = 0.8 s_k^T H_k s_k / (s_k^T H_k s_k - s_k^T \hat{y}_k). \end{cases}$$

We note that λ^{k0} is used in defining $\hat{y}_k$ because λ^{k0} converges to a KKT multiplier of (1.1) according to Theorem 3.7.

(c) Computing the correction direction. In order to save computation, evaluation of the correction $\hat{d}^k$ should be calculated only when the iterate is close to a solution of problem (1.1). In our implementation, $\hat{d}^k$ is calculated when

$$\|\Phi(x^k, \lambda^k)\| \le 1 \quad \text{and} \quad \|d^k\| \le 0.1.$$

Doing so, we need to estimate the active set at x^k. Instead of using the traditional strategy stated in (S.1)(v) of Algorithm 2.1, we define

$$I_k := \left\{ i |\ -g_i(x^k) \le \sqrt{\|R(x^k, \lambda^k)\|} \right\}, \tag{5.2}$$

where

$$R(x, \lambda) = \begin{pmatrix} \nabla_x L(x, \lambda) \\ \min(-g(x), \lambda) \end{pmatrix}.$$

It follows from [3, Theorem 3.7] that I_k defined by (5.2) is able to identify the actual active set I_* without requiring the strict complementarity condition. We note that such a replacement does not affect the convergence of Algorithm 2.1 because the strict complementarity condition is required in our local fast convergence analysis.

TABLE 5.1
Numerical results for Algorithm 2.1.

Problem	It	Nf	Ng	$\|\Phi\|$	FV
1	40	66	66	1.5e-07	1.0884e-17
3	12	17	23	1.3e-08	1.2778e-08
4	4	9	11	7.5e-09	2.6667e+00
5	6	11	11	2.2e-06	-1.9132e+00
12	7	15	17	1.2e-06	-3.0000e+01
24	11	19	24	1.8e-13	-1.0000e+00
25	0	1	1	1.8e-08	3.2835e+01
29	8	15	18	3.4e-06	-2.2627e+01
30	7	10	14	2.5e-09	1.0000e+00
31	10	37	41	1.7e-07	6.0000e+00
33	10	28	34	1.3e-09	-4.0000e+00
34	23	68	78	2.3e-11	-8.3403e-01
35	7	12	15	9.9e-06	1.1111e-01
36	13	72	74	3.6e-13	-3.3000e+03
37	17	79	85	4.8e-06	-3.4560e+03
43	12	25	30	1.5e-06	-4.4000e+01
44	17	39	42	1.1e-11	-1.5000e+01
76	10	18	23	8.9e-09	-4.6818e+00
100	15	39	45	1.5e-08	6.8063e+02
113	22	50	58	5.6e-06	2.4306e+01

(d) Test problems and the parameter setting. A total of 20 problems are selected from [14]. These problems have inequality constraints only and the starting points provided are strictly feasible. The parameters used in our implementation are

$$\alpha = 0.3,\ \beta = 0.5,\ \theta = 0.8,\ \nu = 3,\ \tau = 2.5,\ \kappa = 0.9,\ \bar{\mu} = +\infty,\ \text{tol} = 10^{-5}.$$

In Table 5.1, which presensts results of the numerical experiments, we use the following notation:

Problem: number of the problem in [14],
It: number of iterations,
Nf: number of objective function evaluations,
Ng: number of constraint function evaluations,
$\|\Phi\|$: value of $\|\Phi(\cdot,\cdot)\|$ at the final iterate (x^k, λ^{k0}),
FV: objective function value at the final iterate.

We note that the number of Jacobian evaluations is one more than the number of iterations.

The results in Table 5.1 indicate that Algorithm 2.1 is quite promising. In general, the number of iterations and the function evaluations are very small. Moreover, the results compare well with those given in [26]. For almost all problems except problem 25, our algorithm can find the solution. The starting point for problem 25 already satisfies the criterion (5.1). We note that the number of the constraint function evaluations is relatively larger than the number of the objection function evaluations. The reason is that we need to calculate the correction direction $\hat{d}^k$ whenever the iterate is close to the solution. The calculation needs evaluation of constraint functions at an additional point, namely, $x^k + d^k$. Finally, we stress that the behavior of the algorithm appeared to be relatively insensitive to changes in the values of the algorithm parameters.

6. Conclusions. A new feasible QP-free method is proposed for the inequality constrained optimization problem. The method is based on a nonsmooth equation

reformulation of the KKT optimality condition. Compared with the one in [19], which is based on the equality part of the KKT optimality condition, our method enjoys several advantages, among which are the nonsingularity of the iteration matrices without assuming the strict complementarity condition (see Lemma 3.1); the fact that every limit point of the iterative sequence is a KKT point of the original problem without assuming boundedness of the Lagrangian multiplier approximation sequence as required in [11]; and the uniform boundedness of the direction sequences. All of those merits come from the one simple fact that the Jacobian matrix of the reformulated nonsmooth equation is nonsingular under mild assumptions. The efficiency of the new method is verified with a subset of problems from [14]. The numerical results confirmed our feeling that the Fischer–Burmeister function can be successfully used for the solution of the inequality constrained optimization problems, not only for complementarity problems.

An interesting problem which remains open is whether the strict complementarity condition can be removed from the analysis of the local fast convergence. It seems hopeful by incorporating the identification technique recently studied by Facchinei, Fischer, and Kanzow in [3] into our algorithm. However, it is extremely hard since we are unable to ensure that the unit steplength is accepted eventually without the strict complementarity condition.

Acknowledgments. We would like to thank both referees for their valuable comments and suggestions, which greatly improved the presentation of the paper. In particular, one referee drew our attention to references [13, 26]. We are also grateful to the associate editor for his suggestion on implementation issues.

REFERENCES

[1] B. Chen, X. Chen, and C. Kanzow, *A penalized Fischer–Burmeister NCP function: Theoretical investigation and numerical results*, Math. Programming, to appear.

[2] 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.

[3] F. Facchinei, A. Fischer, and C. Kanzow, *On the accurate identification of active constraints*, SIAM J. Optim., 9 (1999), pp. 14–32.

[4] F. Facchinei, A. Fischer, C. Kanzow, and J. M. Peng, *A simply constrained optimization reformulation of KKT systems arising from variational inequalities*, Appl. Math. Optim., 40 (1999), pp. 19–37.

[5] F. Facchinei and S. Lucidi, *Quadratically and superlinearly convergent algorithms for the solution of inequality constrained minimization problems*, J. Optim. Theory Appl., 85 (1995), pp. 265–289.

[6] M. C. Ferris and C. Kanzow, *Complementarity and related problems: A survey*, in Handbook of Applied Optimization, P. M. Pardalos and M. G. C. Resende, eds., Oxford University Press, to appear.

[7] M. C. Ferris, C. Kanzow, and T. S. Munson, *Feasible descent algorithms for mixed complementarity problems*, Math. Programming, 86 (1999), pp. 475–497.

[8] M. C. Ferris and K. Sinapiromsaran, *Formulating and solving nonlinear programs as mixed complementarity problems*, in Ninth BFG Conference on Optimization, J. J. Strodiot, ed., Springer-Verlag, New York, 1999.

[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 Advance in Nonsmooth Optimization, D. Du, L. Qi, and R. Womersley, eds., World Scientific Publishers, Singapore, 1995, pp. 88–105.

[11] Z. Gao, G. He, and F. Wu, *Sequential systems of linear equations algorithm for nonlinear optimization problems with general constraints*, J. Optim. Theory Appl., 95 (1997), pp. 371–397.

[12] J. Herskovits, *A two-stage feasible direction algorithm for nonlinear constrained optimization*, Math. Programming, 36 (1986), pp. 19–38.

[13] J. Herskovits, *Feasible direction interior-point technique for nonlinear optimization*, J. Optim. Theory Appl., 99 (1998), pp. 121–146.
[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 and H.-D. Qi, *A QP-free constrained Newton-type method for variational inequality problems*, Math. Programming, 85 (1999), pp. 81–106.
[16] C. T. Lawrence and A. L. Tits, *A Computationally Efficient Feasible Sequential Quadratic Programming Algorithm*, TR 98-46, Department of Electrical Engineering and Institute for Systems Research, University of Maryland, College Park, 1998.
[17] J. J. Moré and D. C. Sorensen, *Computing a trust region step*, SIAM J. Sci. Statist. Comput., 4 (1983), pp. 553–52.
[18] 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.
[19] E. R. Panier, A. L. Tits, and J. N. Herskovits, *A QP-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization*, SIAM J. Control Optim., 36 (1988), pp. 788–811.
[20] J.-S. Pang and L. Qi, *Nonsmooth equations: Motivation and algorithms*, SIAM J. Optim., 3 (1993), pp. 443–465.
[21] M. J. D. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, in Numerical Analysis, Proceedings, Biennial Conference, Dundee 1977, Lecture Notes in Math. 630, G. A. Watson, ed., Springer-Verlag, Berlin, New York, 1978, pp. 144–157.
[22] 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.
[23] L. Qi and H. Jiang, *Semismooth Karush-Kuhn-Tucker equations and convergence analysis of Newton and quasi-Newton methods for solving these equations*, Math. Oper. Res., 22 (1997), pp. 301–325.
[24] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–368.
[25] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.
[26] T. Urban, A. L. Tits, and C. T. Lawrence, *A Primal-Dual Interior-Point Method for Nonconvex Optimization with Multiple Logarithmic Barrier Parameters and with Strong Convergence Properties*, TR 98-27, Department of Electrical Engineering and Institute for Systems Research, University of Maryland, College Park, 1998.
[27] N. Yamashita and M. Fukushima, *Modified Newton methods for solving semismooth reformulations of monotone complementarity problems*, Math. Programming, 76 (1997), pp. 469–491.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 133–148

CONVERGENCE ANALYSIS OF INEXACT INFEASIBLE-INTERIOR-POINT ALGORITHMS FOR SOLVING LINEAR PROGRAMMING PROBLEMS*

JANOS KORZAK†

Abstract. In this paper we present a convergence analysis for some inexact (polynomial) variants of the infeasible-interior-point algorithm of Kojima, Megiddo, and Mizuno. For this analysis we assume that the iterates are bounded. The new variants allow the use of search directions that are calculated from the defining linear system with only moderate accuracy, e.g., via the use of Krylov subspace methods like CG or QMR. Furthermore, some numerical results for the proposed methods are given.

Key words. linear programming, infeasible-interior-point method, inexact search direction

AMS subject classifications. 90C05, 65K05, 90C06

PII. S1052623497329993

1. Introduction. Considering the fact that the primal-dual algorithm of Kojima, Megiddo, and Mizuno [3] (henceforth called the KMM algorithm) does not use a predictor-corrector approach, it is surprising that it is an efficient algorithm for solving linear programming problems in practice. Of course, this efficiency follows not only from the use of the Newton search direction, which is clearly inferior to the predictor-corrector direction proposed by Mehrotra [7], but also from the fact that the KMM algorithm features some other useful properties that are missing in most other infeasible-interior-point algorithms: It allows the use of arbitrary starting points and long step sizes that can be different in the primal and dual subspaces. Moreover, because of the simple structure of the KMM algorithm, it can easily be modified to handle inexact search directions.

In this paper we give a variant of the KMM algorithm that allows the use of search directions that are calculated only to moderate accuracy (*inexact search directions*) and we prove its convergence behavior under the assumption that the iterates are bounded. This is different from the analysis of the exact algorithm in [3], which gives some information about the infeasibility of the given problem if the iterates are unbounded. This (theoretical) information cannot be obtained with the analysis presented here.

The use of inexact search directions is a major difference to most interior-point algorithms, whose convergence is proved under the assumption that the search directions are calculated exactly. Algorithms featuring similar search directions were proposed by Freund, Jarre, and Mizuno (see [1], [2], and [8]).

After some basic notes and definitions (section 2) we give a motivation for the use of inexact search directions and state our inexact variant of the KMM algorithm (section 3). In section 4 we show that the (polynomial) convergence of the new variant can be proved in almost the same way as the convergence of the original algorithm. After that we give a short analysis of the behavior of the algorithm if unsolvable

*Received by the editors November 11, 1997; accepted for publication (in revised form) February 4, 2000; published electronically July 25, 2000.
http://www.siam.org/journals/siopt/11-1/32999.html
†Fachbereich Mathematik, Universität Wuppertal, Gaußstr. 20, D-42097 Wuppertal, Germany (korzak@math.uni-wuppertal.de).

problems are processed. In section 5 we give a method to incorporate the predictor-corrector direction of Mehrotra in the inexact framework of this paper. Finally, we state some numerical results in section 6.

Throughout the paper we use the following notation: If x^k and z^k are elements of $\mathbb{R}^n$, then X^k and Z^k denote the diagonal matrices $X^k = \operatorname{diag}(x^k)$ and $Z^k = \operatorname{diag}(z^k)$. By e we denote the vector $e = (1, \ldots, 1)^T \in \mathbb{R}^n$ and by 0 and I the zero, resp., the identity, matrix with sizes apparent from the context. As usual, the notation $x > 0$ ($x \geq 0$) means that every component of x is greater then zero (nonnegative). For a matrix $A \in \mathbb{R}^{m\times n}$ with $\operatorname{rank}(A) > 0$ we denote by $\sigma_{\max}(A)$ ($\sigma_{\min}(A)$) the largest (smallest positive) singular value of A, and by $\kappa_2(A) := \sigma_{\max}(A)/\sigma_{\min}(A)$ we denote the spectral condition number of A.

2. The problem. In this paper we consider the linear program

$$(\mathrm{PD}): \left\{ \begin{array}{l} \text{minimize } x^T z, \\ (x,y,z) \in \mathcal{PD}, \end{array} \right. \quad \text{where } \mathcal{PD} := \left\{ \begin{array}{l} (x,y,z) \in \mathbb{R}^{n+m+n}, \\ Ax = b, \\ A^T y + z = c, \\ x \geq 0, \\ z \geq 0. \end{array} \right.$$

A is an $m \times n$ matrix with $\operatorname{rank}(A) = m$, $b \in \mathbb{R}^m$, and $c \in \mathbb{R}^n$. The set $\mathcal{PD}$ is called the set of all feasible points. The basic duality theorem states that (x^*, y^*, z^*) is a solution of (PD) iff

$$(x^*, y^*, z^*) \in \{(x,y,z) \in \mathbb{R}^{n+m+n} : \ x \geq 0,\ z \geq 0,\ x^T z = 0,\ Ax = b,\ A^T y + z = c\}.$$

For given $\varepsilon > 0$, $\varepsilon_p > 0$, and $\varepsilon_d > 0$ the infeasible-interior-point algorithms introduced here try to calculate an element of the set

$$\{(x,y,z) \in \mathbb{R}^{n+m+n} : \ x \geq 0,\ z \geq 0,\ x^T z \leq \varepsilon,\ \|Ax - b\|_2 \leq \varepsilon_p,\ \|A^T y + z - c\|_2 \leq \varepsilon_d\}$$

and take it as an approximation to a solution of (PD) (a so-called $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution).

In order to ensure convergence toward an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution, we force the iterates to lie within a neighborhood of the central path. In this paper we will use the following wide neighborhood proposed by Kojima, Megiddo, and Mizuno [3].

DEFINITION 2.1. *Let $\gamma \in (0,1)$, $\gamma_p > 0$, $\gamma_d > 0$, $\varepsilon_p > 0$, and $\varepsilon_d > 0$. The neighborhood $\mathcal{N} = \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d)$ is defined by*

$$\begin{array}{rl} \mathcal{N} = \{(x,y,z) \in \mathbb{R}^{n+m+n} : & x > 0,\ z > 0, \\ & x_i z_i \geq \gamma x^T z/n \ (i = 1, \ldots, n), \\ & x^T z \geq \gamma_p \|Ax - b\|_2 \vee \|Ax - b\|_2 \leq \varepsilon_p, \\ & x^T z \geq \gamma_d \|A^T y + z - c\|_2 \vee \|A^T y + z - c\|_2 \leq \varepsilon_d\}. \end{array}$$

The following trivial lemma gives a connection between $\mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d)$ and an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution.

LEMMA 2.2. *If $(x,y,z) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d)$ and $x^T z \leq \min\{\varepsilon, \varepsilon_p \gamma_p, \varepsilon_d \gamma_d\}$, then (x,y,z) is an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution of* (PD).

3. An inexact infeasible-interior-point algorithm. To give a motivation for the use of inexact search directions, we review how the KMM algorithm tries to calculate an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution. Let's assume (just for this motivation) that the set

of strictly feasible points $\overset{\circ}{\mathcal{PD}} := \{(x,y,z) \in \mathcal{PD} : x > 0,\ z > 0\}$ is not empty. Thus for all $\mu > 0$,

$$F_\mu(x,y,z) = \begin{pmatrix} Ax - b \\ A^T y + z - c \\ Xz - \mu e \end{pmatrix}$$

has a unique zero (x_μ, y_μ, z_μ) with $x_\mu > 0$ and $z_\mu > 0$, and $\lim_{\mu \to 0}(x_\mu, y_\mu, z_\mu)$ is a solution of (PD). The KMM algorithm therefore calculates a decreasing sequence μ^k with $\lim_{k\to\infty}\mu^k = 0$ and determines in each iteration k an approximation $(x^{k+1}, y^{k+1}, z^{k+1})$ of the central point $(x_{\mu^k}, y_{\mu^k}, z_{\mu^k})$ via a damped Newton step: For the given vector $(x^k, y^k, z^k) \in \mathcal{N}$ the linear system

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \begin{pmatrix} b - Ax^k \\ c - A^T y^k - z^k \\ \mu^k e - X^k z^k \end{pmatrix} \tag{3.1}$$

is solved and $\alpha_p^k \in (0,1]$ and $\alpha_d^k \in (0,1]$ are chosen such that the new iterate

$$(x^{k+1}, y^{k+1}, z^{k+1}) = (x^k + \alpha_p^k \Delta x^k, y^k + \alpha_d^k \Delta y^k, z^k + \alpha_d^k \Delta z^k)$$

is an element of $\mathcal{N}$ (and satisfies an additional descent condition). This procedure together with the fact that the search direction cannot be calculated exactly in practice leads us to the consideration that it should be sufficient to calculate an approximation of $(\Delta x^k, \Delta y^k, \Delta z^k)$ and then proceed as stated before. In this paper we use search directions that can be calculated without any knowledge of $(\Delta x^k, \Delta y^k, \Delta z^k)$ from (3.1): We accept $(\Delta x^k, \Delta y^k, \Delta z^k)$ as an *inexact (Newton) search direction*, if

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \begin{pmatrix} b - Ax^k \\ c - A^T y^k - z^k \\ \mu^k e - X^k z^k \end{pmatrix} + \begin{pmatrix} r^k \\ s^k \\ t^k \end{pmatrix}, \tag{3.2}$$

where the "residual components" satisfy

$$\begin{cases} \|r^k\|_2 & \le & (1-\tau_1)\|Ax^k - b\|_2, \\ \|s^k\|_2 & \le & (1-\tau_2)\|A^T y^k + z^k - c\|_2, \\ \|t^k\|_\infty & \le & \tau_3 \frac{(x^k)^T z^k}{n} \end{cases} \tag{3.3}$$

and $\tau_1 \in (0,1]$, $\tau_2 \in (0,1]$, and $\tau_3 \in [0,1)$ are some appropriately chosen constants.

Note that (3.3) can be interpreted as a condition of the *relative* exactness on each of the three components individually. Therefore, as was pointed out by one of the referees, we cannot guarantee that an iterate calculated via some iterative solver applied to system (3.1) will eventually satisfy condition (3.3), even if the iteration is known to converge. However, (3.3) will be satisfied if we perform an iteration on an appropriately reduced system. This will be explained in detail at the beginning of section 6, where we report on our numerical experiments.

As the new "inexact" algorithm follows the central path in a less rigorous way than the "exact" KMM algorithm, we expect an increase in the number of iterations. But the use of inexact search directions can nevertheless result in a decrease of the total processing time, because inexact search directions can sometimes be calculated very efficiently (see section 6).

We are now ready to state our inexact variant of the KMM algorithm.

ALGORITHM 1.

1. Choose $\varepsilon > 0$, $\varepsilon_p > 0$, $\varepsilon_d > 0$, $\gamma \in (0,1)$, $\gamma_p > 0$, $\gamma_d > 0$, $\omega \geq 1$, and $(x^0, y^0, z^0) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d)$ with $\|(x^0, z^0)\|_1 \leq \omega$. Set $k = 0$ and choose $0 < \beta_1 < \beta_2 < \beta_3 < 1$, $\tau_1 \in (0,1]$, $\tau_2 \in (0,1]$, and $\tau_3 \in [0,1)$ with

   (a) $\delta_1 := (1-\gamma)\beta_1 - (1+\gamma)\tau_3 > 0$, (b) $\delta_2 := \beta_1 + \tau_1 - \tau_3 - 1 > 0$,
   (c) $\delta_3 := \beta_1 + \tau_2 - \tau_3 - 1 > 0$, (d) $\delta_4 := \beta_2 - \beta_1 - \tau_3 > 0$.

2. If (x^k, y^k, z^k) is an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution, stop.
3. Set $\mu^k = \beta_1 (x^k)^T z^k / n$ and calculate a search direction $(\Delta x^k, \Delta y^k, \Delta z^k)$ that satisfies (3.2) and (3.3).
4. Calculate

$$\begin{aligned} \alpha_p^{*,k} &= \sup\{\alpha \in \mathbb{R} : \ x^k + \alpha \Delta x^k \geq 0\}, \\ \alpha_d^{*,k} &= \sup\{\alpha \in \mathbb{R} : \ z^k + \alpha \Delta z^k \geq 0\}, \\ \alpha^{*,k} &= \min\{\alpha_p^{*,k}, \alpha_d^{*,k}\}. \end{aligned}$$

   If $(x^k, y^k, z^k) + \alpha^{*,k}(\Delta x^k, \Delta y^k, \Delta z^k)$ is an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution, stop.
5. Let $\overline{\alpha}^k$ be the maximum of all $\tilde{\alpha} \in [0,1]$ restricted to: All $\alpha \in [0, \tilde{\alpha}]$ are satisfying

$$\begin{aligned} &(x^k, y^k, z^k) + \alpha(\Delta x^k, \Delta y^k, \Delta z^k) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d), \\ &(x^k + \alpha \Delta x^k)^T (z^k + \alpha \Delta z^k) \leq (1 - \alpha(1-\beta_2))(x^k)^T z^k. \end{aligned}$$

6. Choose $\alpha_p^k \in [0,1]$ and $\alpha_d^k \in [0,1]$ in such a way that the new iterate $(x^{k+1}, y^{k+1}, z^{k+1}) = (x^k + \alpha_p^k \Delta x^k, y^k + \alpha_d^k \Delta y^k, z^k + \alpha_d^k \Delta z^k)$ satisfies

$$\begin{aligned} &(x^{k+1}, y^{k+1}, z^{k+1}) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d), \\ &(x^{k+1})^T z^{k+1} \leq (1 - \overline{\alpha}^k(1-\beta_3))(x^k)^T z^k. \end{aligned}$$

7. If $\|(x^{k+1}, z^{k+1})\|_1 \geq \omega$, stop. Otherwise set $k = k+1$ and go to step 2.

Remark 3.1.

1. For $\tau_1 = 1$, $\tau_2 = 1$, and $\tau_3 = 0$ Algorithm 1 reduces to the "exact" KMM algorithm.
2. $||(x^0, z^0)||_1$ is generally quite large, hence ω should be chosen to be large too.
3. The conditions (a)–(d) in step 1 are met if $\beta_1 = 0.1$, $\beta_2 = 0.99995$, $\tau_1 = 0.95$, $\tau_2 = 0.95$, $\tau_3 = 0.049$, and $\gamma < 51/149$.
4. Every vector $(\Delta x^k, \Delta y^k, \Delta z^k)$ which satisfies the conditions (3.2) and (3.3) satisfies $(\Delta x^k, \Delta z^k) \not\geq 0$, hence $\alpha^{*,k} \in (0, \infty)$.
5. Since $(x^k, y^k, z^k) + \overline{\alpha}^k(\Delta x^k, \Delta y^k, \Delta z^k) \in \mathcal{N}$ and

$$(x^k + \overline{\alpha}^k \Delta x^k)^T (z^k + \overline{\alpha}^k \Delta z^k) \leq (1 - \overline{\alpha}^k(1-\beta_3))(x^k)^T z^k,$$

   it is always possible to choose $\alpha_p^k = \alpha_d^k = \overline{\alpha}^k$.
6. Usually it is not necessary to calculate $\overline{\alpha}^k$. The conditions in step 6 are met for a given vector $(x^{k+1}, y^{k+1}, z^{k+1}) \in \mathcal{N}$ if

$$(x^{k+1})^T z^{k+1} \leq (1 - \alpha^{*,k}(1-\beta_3))(x^k)^T z^k,$$

   because we have (by Theorem 4.2 below)

$$(1 - \alpha^{*,k}(1-\beta_3))(x^k)^T z^k \leq (1 - \overline{\alpha}^k(1-\beta_3))(x^k)^T z^k.$$

4. Convergence results. We now modify the ideas of Kojima, Megiddo, and Mizuno [3] to prove the global convergence of Algorithm 1. We first state some basic properties of Algorithm 1, then give a lower bound for $\overline{\alpha}^k$ (Lemma 4.3), and finally prove global convergence (Theorem 4.4). We start by defining

$$\begin{aligned}
f_i^k(\alpha) &= (x_i^k + \alpha\Delta x_i^k)(z_i^k + \alpha\Delta z_i^k) - \gamma(x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k)/n \\
&\quad (i = 1, \ldots, n), \\
g_p^k(\alpha) &= (x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k) - \gamma_p\|A(x^k + \alpha\Delta x^k) - b\|_2, \\
g_d^k(\alpha) &= (x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k) - \gamma_d\|A^T(y^k + \alpha\Delta y^k) + z + \alpha\Delta z^k - c\|_2, \\
h^k(\alpha) &= (1 - \alpha(1 - \beta_2))(x^k)^T z^k - (x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k)
\end{aligned}$$

and $\dot{\alpha}^k$ as the maximum of all $\tilde{\alpha} \in [0, 1]$ for which

$$\left\{\begin{array}{l}
f_i^k(\alpha) \geq 0, \\
g_p^k(\alpha) \geq 0 \ \text{ or } \ \|A(x^k + \alpha\Delta x^k) - b\|_2 \leq \varepsilon_p, \\
g_d^k(\alpha) \geq 0 \ \text{ or } \ \|A^T(y^k + \alpha\Delta y^k) + z^k + \alpha\Delta z^k - c\|_2 \leq \varepsilon_d, \\
h^k(\alpha) \geq 0
\end{array}\right. \tag{4.1}$$

hold for all $\alpha \in [0, \tilde{\alpha}]$.

The following lemma can be proved easily with the help of steps 3 and 6 of Algorithm 1 and its proof is therefore omitted.

LEMMA 4.1. *At step* 2 *of iteration* k

1. $(x^k, y^k, z^k) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d)$,
2. $\|(x^k, z^k)\|_1 \leq \omega$,
3. $(x^k)^T z^k \leq (1 - \overline{\alpha}^{k-1}(1 - \beta_3))(x^{k-1})^T z^{k-1}$ *(if* $k > 0$*).*

At step 4 *of iteration* k *we have for* $\alpha \in [0, 1]$

4. $\|A(x^k + \alpha\Delta x^k) - b\|_2 \leq (1 - \alpha\tau_1)\|Ax^k - b\|_2$,
5. $\|A^T(y^k + \alpha\Delta y^k) + z^k + \alpha\Delta z^k - c\|_2 \leq (1 - \alpha\tau_2)\|A^T y^k + z^k - c\|_2$,
6. (a) $(x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k) \leq (1 + \alpha(\beta_1 + \tau_3 - 1))(x^k)^T z^k + \alpha^2(\Delta x^k)^T\Delta z^k$,
   (b) $(x^k + \alpha\Delta x^k)^T(z^k + \alpha\Delta z^k) \geq (1 + \alpha(\beta_1 - \tau_3 - 1))(x^k)^T z^k + \alpha^2(\Delta x^k)^T\Delta z^k$,
7. $(x_i^k + \alpha\Delta x_i^k)(z_i^k + \alpha\Delta z_i^k) \geq (1 - \alpha)x_i^k z_i^k + \alpha(\beta_1 - \tau_3)(x^k)^T z^k/n + \alpha^2\Delta x_i^k\Delta z_i^k$ *for* $i = 1, \ldots, n$.

THEOREM 4.2. *If Algorithm* 1 *does not terminate at step* 4 *of iteration* k*, then* $\overline{\alpha}^k = \dot{\alpha}^k < \alpha^{*,k}$.

Proof. Suppose that $\dot{\alpha}^k \geq \alpha^{*,k}$. The conditions (4.1) therefore hold for $\alpha^{*,k}$. Because of the definition of $\alpha^{*,k}$, there exists an index i with $(x_i^k + \alpha^{*,k}\Delta x_i^k)(z_i^k + \alpha^{*,k}\Delta z_i^k) = 0$, hence

$$f_i^k(\alpha^{*,k}) = -\gamma(x^k + \alpha^{*,k}\Delta x^k)^T(z^k + \alpha^{*,k}\Delta z^k)/n \geq 0,$$

or, equivalently, $(x^k + \alpha^{*,k}\Delta x^k)^T(z^k + \alpha^{*,k}\Delta z^k) \leq 0$. Using $(x^k + \alpha^{*,k}\Delta x^k) \geq 0$ and $(z^k + \alpha^{*,k}\Delta z^k) \geq 0$ we have $(x^k + \alpha^{*,k}\Delta x^k)^T(z^k + \alpha^{*,k}\Delta z^k) = 0$.

If $g_p^k(\alpha^{*,k}) < 0$, then $\|A(x^k + \alpha^{*,k}\Delta x^k) - b\|_2 \leq \varepsilon_p$; otherwise

$$g_p^k(\alpha^{*,k}) = -\gamma_p\|A(x^k + \alpha^{*,k}\Delta x^k) - b\|_2 \geq 0,$$

or, equivalently, $\|A(x^k + \alpha^{*,k}\Delta x^k) - b\|_2 = 0$. In the same way we can show that $\|A^T(y^k + \alpha^{*,k}\Delta y^k) + z^k + \alpha^{*,k}\Delta z^k - c\|_2 \leq \varepsilon_d$, hence $(x^k, y^k, z^k) + \alpha^{*,k}(\Delta x^k, \Delta y^k, \Delta z^k)$ is an $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution of (PD). This is a contradiction, because Algorithm 1 would have stopped in step 4 of iteration k. So we have shown $\dot{\alpha}^k < \alpha^{*,k}$. Using the definitions of $\dot{\alpha}^k$ and $\overline{\alpha}^k$, one also sees that $\overline{\alpha}^k = \dot{\alpha}^k < \alpha^{*,k}$ holds. ∎

LEMMA 4.3. *Let $\eta > 0$, $|\Delta x_i^k \Delta z_i^k - \gamma(\Delta x^k)^T \Delta z^k/n| \le \eta$ for $i = 1, \ldots, n$ and $|(\Delta x^k)^T \Delta z^k| \le \eta$. At step 5 of iteration k we have*

$$\overline{\alpha}^k \ge \tilde{\alpha}^k := \min\left\{1, \frac{\delta_1 (x^k)^T z^k}{n\eta}, \min\{\delta_2, \delta_3, \delta_4\} \cdot \frac{(x^k)^T z^k}{\eta}\right\}.$$

Proof. Using parts 7, 6(a), and 1 of Lemma 4.1 and the definition of δ_1 in step 1, we have for $\alpha \in [0, \tilde{\alpha}^k]$ and $i = 1, \ldots, n$

$$\begin{aligned}
f_i^k(\alpha) &= (x_i^k + \alpha \Delta x_i^k)(z_i^k + \alpha \Delta z_i^k) - \gamma(x^k + \alpha \Delta x^k)^T (z^k + \alpha \Delta z^k)/n \\
&\ge (1-\alpha) x_i^k z_i^k + \alpha(\beta_1 - \tau_3)(x^k)^T z^k/n + \alpha^2 \Delta x_i^k \Delta z_i^k \\
&\quad - \frac{\gamma}{n}(1 + \alpha(\beta_1 + \tau_3 - 1))(x^k)^T z^k - \frac{\gamma}{n}\alpha^2 (\Delta x^k)^T \Delta z^k \\
&= [\Delta x_i^k \Delta z_i^k - \gamma(\Delta x^k)^T \Delta z^k/n]\alpha^2 + (1-\alpha)[x_i^k z_i^k - \gamma(x^k)^T z^k/n] \\
&\quad + \left((\beta_1 - \tau_3 - \gamma\beta_1 - \gamma\tau_3)\frac{(x^k)^T z^k}{n}\right)\alpha \\
&\ge -\eta\alpha^2 + \left(((1-\gamma)\beta_1 - (1+\gamma)\tau_3)\frac{(x^k)^T z^k}{n}\right)\alpha \\
&= \left(\frac{\delta_1 (x^k)^T z^k}{n} - \eta\alpha\right)\alpha \\
&\ge \left(\frac{\delta_1 (x^k)^T z^k}{n} - \eta\tilde{\alpha}^k\right)\alpha \\
&\ge 0.
\end{aligned}$$

If $g_p^k(0) = (x^k)^T z^k - \gamma_p ||Ax^k - b||_2 \ge 0$, we have by parts 4 and 6(b) of Lemma 4.1 for all $\alpha \in [0, \tilde{\alpha}^k]$

$$\begin{aligned}
g_p^k(\alpha) &= (x^k + \alpha \Delta x^k)^T (z^k + \alpha \Delta z^k) - \gamma_p \|A(x^k + \alpha \Delta x^k) - b\|_2 \\
&\ge (1 + \alpha(\beta_1 - \tau_3 - 1))(x^k)^T z^k + \alpha^2 (\Delta x^k)^T \Delta z^k - \gamma_p(1 - \alpha\tau_1)\|Ax^k - b\|_2 \\
&\ge -\eta\alpha^2 + (1 + \alpha(\delta_2 - \tau_1))(x^k)^T z^k - \gamma_p(1 - \alpha\tau_1)\|Ax^k - b\|_2 \\
&= -\eta\alpha^2 + [\delta_2 (x^k)^T z^k]\alpha + (1 - \alpha\tau_1) g_p^k(0) \\
&\ge [\delta_2 (x^k)^T z^k - \eta\tilde{\alpha}^k]\alpha \\
&\ge 0.
\end{aligned}$$

If $g_p^k(0) < 0$, we have $\|Ax^k - b\|_2 \le \varepsilon_p$ since $(x^k, y^k, z^k) \in \mathcal{N}$. Using part 4 of Lemma 4.1, we therefore obtain for $\alpha \in [0, \tilde{\alpha}^k]$

$$\|A(x^k + \alpha \Delta x^k) - b\|_2 \le (1 - \alpha\tau_1)\|Ax^k - b\|_2 \le (1 - \alpha\tau_1)\varepsilon_p \le \varepsilon_p.$$

Similiarly we can prove that for $\alpha \in [0, \tilde{\alpha}^k]$

1. $g_d^k(\alpha) \ge 0$ or $\|A^T(y^k + \alpha \Delta y^k) + z^k + \alpha \Delta z^k - c\|_2 \le \varepsilon_d$,
2. $h^k(\alpha) \ge 0$;

hence $\dot{\alpha}^k \ge \tilde{\alpha}^k$. So the lemma follows from applying Theorem 4.2. ∎

THEOREM 4.4. *Algorithm 1 terminates after a finite number of iterations.*

Proof. Suppose that Algorithm 1 does not terminate. From $(x^k, y^k, z^k) \in \mathcal{N}$, Lemma 2.2, and part 2 of Lemma 4.1 it follows for all $k \ge 0$

$$(x^k)^T z^k \ge \varepsilon^* := \min\{\varepsilon, \varepsilon_p\gamma_p, \varepsilon_d\gamma_d\}, \quad \|(x^k, z^k)\|_1 \le \omega, \quad \text{and} \tag{4.2}$$

$$(x^k, z^k) \in M := \left\{ (x, z) \ \in \mathbb{R}^{n+n} : \ \frac{\gamma \varepsilon^*}{n\omega} \le x_i, z_i \le \omega \text{ for } i = 1, \ldots, n \right\}.$$

Using parts 4 and 5 of Lemma 4.1, $(x^k, z^k) \in M$, $\omega \ge 1$, and step 3 of Algorithm 1, we have

$$\begin{aligned} \|b - Ax^k + r^k\|_2 &\le \|Ax^k - b\|_2 + \|r^k\|_2 \\ &\le \|Ax^k - b\|_2 + (1 - \tau_1)\|Ax^k - b\|_2 \\ &= (2 - \tau_1)\|Ax^k - b\|_2 \\ &\le (2 - \tau_1) \left(\prod_{i=0}^{k-1} (1 - \alpha_p^i \tau_1) \right) \|Ax^0 - b\|_2 \\ &\le (2 - \tau_1)\|Ax^0 - b\|_2, \end{aligned}$$

$$\|c - A^T y^k - z^k + s^k\|_2 \le (2 - \tau_2)\|A^T y^0 + z^0 - c\|_2,$$

$$\|\mu^k e - X^k z^k + t^k\|_2 \le (\beta_1 + 1 + \tau_3)\sqrt{n}\omega^2.$$

Hence the search direction $(\Delta x^k, \Delta y^k, \Delta z^k)$ is the solution of a linear system

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \begin{pmatrix} \overline{h} \\ \overline{i} \\ \overline{j} \end{pmatrix}, \tag{4.3}$$

where

$$\begin{aligned} \|\overline{h}\|_2 &\le (2 - \tau_1)\|Ax^0 - b\|_2, \\ \|\overline{i}\|_2 &\le (2 - \tau_2)\|A^T y^0 + z^0 - c\|_2, \\ \|\overline{j}\|_2 &\le (\beta_1 + 1 + \tau_3)\sqrt{n}\omega^2. \end{aligned}$$

We note that by $(x^k, z^k) \in M$ and the fact that M is a compact set, there exists a compact set K which contains the matrix of (4.3) for all k and every matrix $B \in K$ is regular. Since the inverses of all $B \in K$ are bounded, $(\Delta x^k, \Delta y^k, \Delta z^k)$ is bounded for all $k \ge 0$. It follows that there exists a fixed $\eta > 0$ with

$$|\Delta x_i^k \Delta z_i^k - \gamma (\Delta x^k)^T \Delta z^k / n| \le \eta \text{ and } |(\Delta x^k)^T \Delta z^k| \le \eta$$

for $i = 1, \ldots, n$ and all $k \ge 0$. Using Lemma 4.3 and (4.2) we have

$$\overline{\alpha}^k \ge \tilde{\alpha}^k \ge \alpha^* := \min \left\{ 1, \frac{\delta_1 \varepsilon^*}{n\eta}, \min\{\delta_2, \delta_3, \delta_4\} \cdot \frac{\varepsilon^*}{\eta} \right\} \in (0, 1];$$

hence applying part 3 of Lemma 4.1 yields

$$(x^k)^T z^k \le (1 - \alpha^*(1 - \beta_3))^k (x^0)^T z^0.$$

This implies $\lim_{k \to \infty} (x^k)^T z^k = 0$, which contradicts (4.2). □

Remark 4.5. If ε, ε_p, ε_d, γ, γ_p, γ_d, ω, δ_1, δ_2, δ_3, δ_4, β_1, β_2, β_3, τ_1, τ_2, and τ_3 are chosen independently of the problem, then it is possible to prove (see [4]) that the quantity η in the proof of Theorem 4.4 satisfies

$$\eta = \mathcal{O}\left(\kappa_2^2(AA^T) \cdot \left(1 + (2 - \tau_2)\|A^T y^0 + z^0 - c\|_2 + \frac{(2 - \tau_1)\|Ax^0 - b\|_2}{\sigma_{\max}(A)} \right)^2 \cdot n^{6.5} \right).$$

Because of this relation it is easy to show that Algorithm 1 terminates after at most

$$\mathcal{O}\left(\kappa_2^2(AA^T)\cdot\left(1+(2-\tau_2)\|A^Ty^0+z^0-c\|_2+\frac{(2-\tau_1)\|Ax^0-b\|_2}{\sigma_{\max}(A)}\right)^2\cdot n^{7.5}\cdot L\right)$$

iterations if L is defined as $L=\max\{0,\lceil\ln(\frac{(x^0)^Tz^0}{\min\{\varepsilon,\varepsilon_p\gamma_p,\varepsilon_d\gamma_d\}})\rceil\}$. We have in particular that the KMM algorithm terminates after at most

$$\mathcal{O}\left(\kappa_2^2(AA^T)\cdot\left(1+\|A^Ty^0+z^0-c\|_2+\frac{\|Ax^0-b\|_2}{\sigma_{\max}(A)}\right)^2\cdot n^{7.5}\cdot L\right)$$

iterations.

We finish this section with some notes on the stopping criterion given in step 7 of Algorithm 1: If the algorithm terminates in step 7 of iteration j, the search direction is the exact Newton direction in iterations s to j, and ω is chosen large enough, then there exists no strictly feasible point in a certain subregion of $\mathbb{R}^n$ (see Theorem 4.3 in [3]). Therefore a termination in step 7 can be viewed as a hint that $\overset{\circ}{\mathcal{P}}\mathcal{D}=\emptyset$ or, equivalently (see Robinson [10]), that (PD) is not solution/functional-stable. In this sense we can say that the KMM algorithm either calculates an $(\varepsilon,\varepsilon_p,\varepsilon_d)$-solution or detects that (PD) is probably unstable/unsolvable.

It does not seem possible to establish a variant of this statement for inexact search directions. Therefore, if one wants to detect instability or unsolvability of a problem, we propose to restart Algorithm 1 with the current iterate as the starting point, $\tau_1=1$, $\tau_2=1$, $\tau_3=0$, and a larger ω if the algorithm terminates in step 7. (One can also use an exact algorithm, e.g., the algorithm of Mizuno and Jarre [9], that guarantees the calculation of an $(\varepsilon,\varepsilon_p,\varepsilon_d)$-solution, if ω is large enough and (PD) is solvable.) Although this is a theoretical disadvantage, this is not important in practice: First, exact search directions cannot be calculated in practice.

Secondly, even for simple unsolvable problems the exact KMM algorithm behaves in the following way (see [4]; the search directions are calculated with high accuracy and treated as exact): After a few iterations (≤ 10) the norm of the search direction becomes very large and the algorithm is forced to use very small step sizes to stay in $\mathcal{N}$. At this point, $\|(x^k,z^k)\|_1$ is usually quite small ($\leq 10^{20}$) compared with ω ($\approx 10^{40}$), and $\|(x^k,z^k)\|_1$ is increased by only approximately 100 in each iteration. Since this means that the stopping criterion in step 7 will not be met in a reasonable time, the exact KMM algorithm is unable to detect the instability or the unsolvability of a problem in practice.

Since Algorithm 1 is a variant of the KMM algorithm, it is not surprising that Algorithm 1 behaves in a similar way when being applied to unsolvable problems. It is therefore natural not to restart Algorithm 1, but to terminate in step 7 with the statement that (PD) is probably unstable or unsolvable.

Moreover, because of the behavior of the norm of the search direction, it seems reasonable to use the following stopping criterion: Stop, if $\|(\Delta x^k,\Delta z^k)\|_\infty>\omega$. If this stopping criterion is used by Algorithm 1, it is easy to prove (see [4]) that under the assumptions of Remark 4.5, Algorithm 1 terminates after at most $\mathcal{O}(n^2L)$ iterations.

5. Inexact predictor-corrector methods. We now give a variant of Algorithm 1 that allows the use of a whole class of inexact search directions. This class includes an inexact variant of the predictor-corrector search direction of Mehrotra [7],

which is of one of the most efficient search directions in practice (see, e.g., Lustig, Marsten, and Shanno [5], [6]). The convergence of the given variant is ensured in a simple way: If the current search direction does not allow for sufficiently large step sizes, we use an inexact Newton search direction instead. After we state our algorithm, we therefore give only one remark that states the convergence of Algorithm 2.

ALGORITHM 2.

The steps 2, 4, and 6 are identical to steps 2, 4, and 6 of Algorithm 1.

1. Choose all quantities as in step 1 of Algorithm 1 and $\alpha_c \in (0,1]$. Set $N^0 = 0$.
3. (a) If $N^k = 1$, calculate $(\Delta x^k, \Delta y^k, \Delta z^k)$ as in step 3 of Algorithm 1.
   (b) If $N^k = 0$, calculate a search direction $(\Delta x^k, \Delta y^k, \Delta z^k)$ which satisfies

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \begin{pmatrix} b - Ax^k \\ c - A^T y^k - z^k \\ j^k \end{pmatrix} + \begin{pmatrix} r^k \\ s^k \\ t^k \end{pmatrix},$$

where

$$\begin{array}{lll} ||r^k||_2 & \leq & (1-\tau_1)||Ax^k - b||_2, \\ ||s^k||_2 & \leq & (1-\tau_2)||A^T y^k + z^k - c||_2, \\ \text{and } j^k & \in \mathbb{R}^n \text{ and } t^k & \in \mathbb{R}^n \text{ are arbitrary.} \end{array}$$

   If $(\Delta x^k, \Delta z^k) \geq 0$, set $N^k = 1$ and go to step 3.

5. (a) This step is identical to step 5 of Algorithm 1.
   (b) If $N^k = 0$ and $\overline{\alpha}^k < \alpha_c$, set $N^k = 1$ and go to step 3.
7. If $\|(x^{k+1}, z^{k+1})\|_1 \geq \omega$, stop. Otherwise set $k = k+1$ and $N^k = 0$, and go to step 2.

Remark 5.1. If α_c is chosen independently of the problem, then Remark 4.5 can be applied to Algorithm 2.

Remark 5.2. We can modify Algorithm 2 in the following way: If $N^k = 0$, we attempt to find a vector

$$(x^{k+1}, y^{k+1}, z^{k+1}) = (x^k + \alpha_p^k \Delta x^k, y^k + \alpha_d^k \Delta y^k, z^k + \alpha_d^k \Delta z^k)$$

which satisfies

$$\begin{array}{l} (x^{k+1}, y^{k+1}, z^{k+1}) \in \mathcal{N}(\gamma, \gamma_p, \gamma_d, \varepsilon_p, \varepsilon_d), \\ (x^{k+1})^T z^{k+1} \leq (1 - \alpha_c(1-\beta_3))(x^k)^T z^k, \end{array}$$

and we reject the search direction only if no appropriate vector can be found through some trials (e.g., $\alpha_p^k = \beta_4 \alpha_p^{*,k}$ and $\alpha_d^k = \beta_4 \alpha_d^{*,k}$, or $\alpha_p^k = \alpha_d^k = \beta_4 \alpha^{*,k}$ with $\beta_4 \in (0,1)$).

An inexact variant of the search direction of Mehrotra can now be incorporated in the following way:

A. Calculate a predictor direction $(\Delta x_a^k, \Delta y_a^k, \Delta z_a^k)$ which satisfies

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x_a^k \\ \Delta y_a^k \\ \Delta z_a^k \end{pmatrix} = \begin{pmatrix} b - Ax^k \\ c - A^T y^k - z^k \\ -X^k z^k \end{pmatrix} + \begin{pmatrix} r_a^k \\ s_a^k \\ t_a^k \end{pmatrix},$$

where

(a) $||r_a^k||_2 \leq (1-\tau_1)||Ax^k - b||_2$,
(b) $||s_a^k||_2 \leq (1-\tau_2)||A^T y^k + z^k - c||_2$,

(c) $||t_a^k||_\infty \leq \tau_3 \frac{(x^k)^T z^k}{n}$.

B. If $(x^k)^T z^k < 1$, $\Delta x_a^k \geq 0$, or $\Delta z_a^k \geq 0$, define $\mu^k = \frac{(x^k)^T z^k}{\Phi(n)}$, where

$$\Phi(n) = \begin{cases} n^2, & n \leq 5000, \\ n^{1.5}, & n > 5000. \end{cases}$$

Otherwise set

$$\tilde{\alpha}_p = \min\left\{ -\frac{x_i^k}{(\Delta x_a^k)_i} : \ i \in \{1, \ldots, n\} \text{ and } (\Delta x_a^k)_i < 0 \right\},$$

$$\tilde{\alpha}_d = \min\left\{ -\frac{z_i^k}{(\Delta z_a^k)_i} : \ i \in \{1, \ldots, n\} \text{ and } (\Delta z_a^k)_i < 0 \right\},$$

$\tilde{\alpha}_p^* = 0.99995\tilde{\alpha}_p$, $\tilde{\alpha}_d^* = 0.99995\tilde{\alpha}_d$, and

$$\mu^k = \left(\frac{(x^k + \tilde{\alpha}_p^* \Delta x_a^k)^T (z^k + \tilde{\alpha}_d^* \Delta z_a^k)}{(x^k)^T z^k} \right)^2 \cdot \left(\frac{(x^k + \tilde{\alpha}_p^* \Delta x_a^k)^T (z^k + \tilde{\alpha}_d^* \Delta z_a^k)}{n} \right).$$

C. Calculate a search direction $(\Delta x^k, \Delta y^k, \Delta z^k)$ which satisfies

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^T & I \\ Z^k & 0 & X^k \end{pmatrix} \cdot \begin{pmatrix} \Delta x^k \\ \Delta y^k \\ \Delta z^k \end{pmatrix} = \begin{pmatrix} b - Ax^k \\ c - A^T y^k - z^k \\ \mu^k e - X^k z^k - \Delta X_a^k \Delta z_a^k \end{pmatrix} + \begin{pmatrix} r^k \\ s^k \\ t^k \end{pmatrix},$$

where

(a) $||r^k||_2 \leq (1 - \tau_1)||Ax^k - b||_2$,

(b) $||s^k||_2 \leq (1 - \tau_2)||A^T y^k + z^k - c||_2$,

(c) $||t^k||_\infty \leq \tau_3 \frac{(x^k)^T z^k}{n}$.

This approach has the following drawback: One of the main reasons for the efficiency of the exact search direction of Mehrotra is the fact that it can be calculated with the help of only one matrix factorization. But if we determine "inexact" directions $(\Delta x_a^k, \Delta y_a^k, \Delta z_a^k)$ and $(\Delta x^k, \Delta y^k, \Delta z^k)$ with the help of Krylov subspace methods, we have to do the Krylov iteration for two linear systems. The calculation of a search direction with Krylov subspace methods can therefore be more time-consuming than the calculation via direct methods. We make some more notes on this topic in the following section.

6. Numerical results. We now give some numerical results that were obtained with the algorithms of this paper. Inexact search directions can be calculated in several ways (see [1], [2], and [4]), but in this paper we give results for only two methods. We note that the exact Newton search direction can be calculated via (with $D^k = X^k (Z^k)^{-1}$ and (see (4.3)) appropriate $\overline{h}$, $\overline{i}$, and $\overline{j}$)

$$\begin{cases} \Delta y^k &= (AD^k A^T)^{-1}(AD^k(-(X^k)^{-1}\overline{j} + \overline{i}) + \overline{h}), \\ \Delta z^k &= \overline{i} - A^T \Delta y^k, \\ \Delta x^k &= -D^k(\Delta z^k - (X^k)^{-1}\overline{j}). \end{cases} \tag{6.1}$$

For the calculation of an inexact Newton search direction we therefore define $M = AD^k A^T$ and $\overline{b} = AD^k(-(X^k)^{-1}\overline{j} + \overline{i}) + \overline{h}$ and solve $M\Delta y^k = \overline{b}$ with the help of

Method A. the sparse Cholesky decomposition of the (minimum-degree reordered) matrix M,

Method B. the preconditioned CG algorithm.

The CG algorithm was implemented to use the simple Jacobi preconditioner and to terminate if the CG iterate $\Delta\tilde{y}^k$ satisfies

$$\|M\Delta\tilde{y}^k - \bar{b}\|_2 \leq (1-\tau_1)\|Ax^k - b\|_2.$$

Δy^k is then set to $\Delta\tilde{y}^k$, and for both methods Δx^k and Δz^k are then calculated as stated in (6.1). Note that in the case of Method B the vector $(\Delta x^k, \Delta y^k, \Delta z^k)$ satisfies

$$\begin{aligned}\|r^k\|_2 &= \|A\Delta x^k - \bar{h}\|_2 \\ &= \|A((Z^k)^{-1}\bar{j} - D^k\Delta z^k) - \bar{h}\|_2 \\ &= \|A(Z^k)^{-1}\bar{j} - AD^k(\bar{i} - A^T\Delta y^k) - \bar{h}\|_2 \\ &= \|M\Delta\tilde{y}^k - \bar{b}|_2 \\ &\leq (1-\tau_1)\|Ax^k - b\|_2.\end{aligned}$$

Furthermore, we have $\|s^k\|_2 = \|t^k\|_\infty = 0$; hence $(\Delta x^k, \Delta y^k, \Delta z^k)$ satisfies (3.3). This means that $(\Delta x^k, \Delta y^k, \Delta z^k)$ is a valid inexact search direction in the sense of this paper.

The search direction which is calculated with the help of Method A is usually treated as an "exact" search direction, although, due to the effect of round-off, it is sometimes not even an inexact search direction in the sense of this paper. Nevertheless, we always treat the differences in the results of Method A and Method B as being caused by the use of inexact search directions in Method B. Note that Algorithm 1 reduces to the KMM algorithm if the search direction is calculated with Method A.

Algorithm 2 always tried to use the inexact predictor-corrector search direction of the preceding section, which were calculated analogous to inexact Newton search directions. Note that Algorithm 2 reduces to a variant of the algorithm of Mehrotra (see, e.g., Lustig, Marsten, and Shanno [5], [6]) if the search direction is calculated with Method A.

Before we state results, we give some details of the implementation:

1. The algorithms were programmed with MATLAB 5.3 on a Sun UltraSparc 60.
2. The starting points are calculated as proposed by Lustig, Marsten, and Shanno [5].
3. The parameters in step 1 of the algorithms are chosen as follows ($\mathcal{N}$ is enlarged, if necessary): $\varepsilon_p = 1e-08$, $\varepsilon_d = 1e-08$, $\gamma = 1e-08$, $\gamma_p = 1e-08$, $\gamma_d = 1e-08$, $\omega = 1e{+}40$, $\alpha_c = 1e-10$, $\beta_1 = 0.1$, $\beta_2 = 0.99995$, $\beta_3 = 0.99997$, $\beta_4 = 0.99995$, $\tau_1 = 0.95$, $\tau_2 = 0.95$ and $\tau_3 = 0.049$. ε was set to $1e-08, 1e-07$, or $5e-03$.
4. In each iteration the algorithms try to use the large step sizes $\alpha_p^k = \beta_4\alpha_p^{*,k}$ and $\alpha_d^k = \beta_4\alpha_d^{*,k}$ or the step sizes $\alpha_p^k = \alpha_d^k = \beta_4\alpha^{*,k}$. If the calculated vector does not satisfy the conditions in step 6 of the algorithms, $\alpha_p^k = \alpha_d^k = \overline{\alpha}^k$ is used instead. The use of large step sizes is one of the reasons for the efficiency of the exact variants of Algorithm 1 and Algorithm 2, because the use of this step sizes results in a low number of iterations. But after a few iterations it also results in a very high condition number of M, because some components of x and z become very small. This forces the CG algorithm to use a large number of iterations for the determination of Δy^k. We nevertheless use the stated step sizes, because we mainly want to detect differences in the number of iterations until convergence.

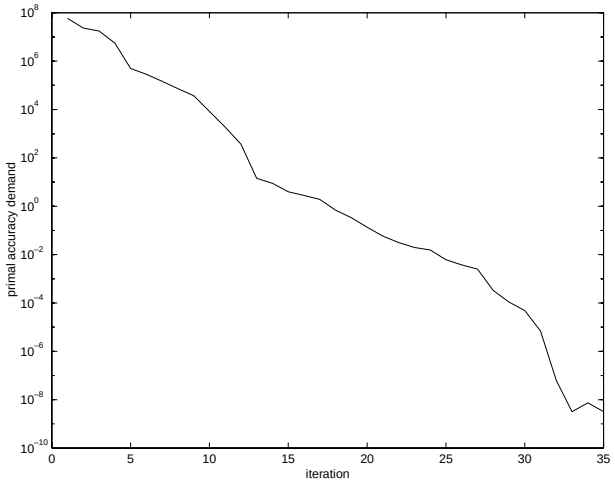


FIG. 1.

5. We use the (scaled) stopping criterion used by Lustig, Marsten, and Shanno [5], that is, the algorithms terminate if (x^k, y^k, z^k) satisfies

$$\frac{(x^k)^T z^k}{1+|b^T y^k|} \leq \varepsilon, \qquad \frac{\|Ax^k - b\|_2}{1+\|x^k\|_2} \leq \varepsilon_p, \qquad \frac{\|A^T y^k + z^k - c\|_2}{1+\|z^k\|_2} \leq \varepsilon_d.$$

First we have a look at some results obtained with the small problem `israel` of the NETLIB set. More results for 38 problems of the NETLIB set and the problems of the NETGEN set can be found in [4]. The first table gives the number of iterations and the times (in seconds) that the algorithms needed for the calculation of a (scaled) $(\varepsilon, \varepsilon_p, \varepsilon_d)$-solution ($\varepsilon$ was set to $1e-08$).

Results for problem israel				
	Method A		Method B	
	iterations	seconds	iterations	seconds
Algorithm 1	35	3.60	34	31.87
Algorithm 2	23	2.56	23	27.63

We notice that the number of iterations for Method B can be compared with the number of iterations needed by Method A, but the processing time is higher by a factor of 8.9 and 10.8. We already stated that the reason for this is the high number of iterations of the CG algorithm within each step of Algorithm 1 and Algorithm 2, which in turn is caused by the high demands on the accuracy of the search direction combined with the high condition number of M for large k. Figures 1 and 2 show the primal accuracy demand $(1-\tau_1)\|Ax^k - b\|_2$ and the calculation time for Algorithm 1 and Method B.

We can see that the time needed for the calculation of the search direction indeed increases for higher k. Because in the final iterations the calculation time is 30 times as high as the calculation time in the first iteration, and because the calculation of the search direction via Method A needs a constant time (only 0.09 seconds), it makes sense to use Method B only in the first few iterations and switch to

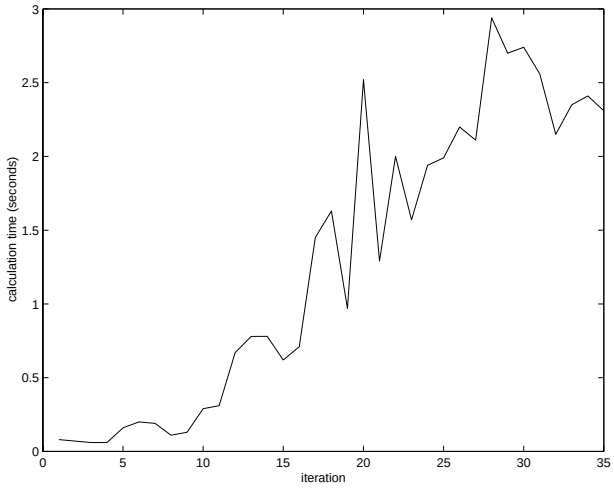


FIG. 2.

Method A if Method B needs more time than Method A (this method is henceforth called Method C). Using Method C we obtain the following results: Algorithm 1 (Algorithm 2) needs 34 (23) iterations and 3.42 (2.41) seconds, with a total of only 6 (3) search directions calculated via Method B. This result (a small increase in the number of iterations, a small increase or decrease in the total processing time, and very few search directions calculated with Method B) is typical for nearly all problems of the NETLIB set (see [4]) and can be explained by the fact that the sparse Cholesky decomposition can be calculated very efficiently (in MATLAB) for those small problems. Even for the few NETLIB problems for which Method C calculated at least two search directions with Method B the difference in the processing time (and the number of iterations) of Method A and Method C is nearly negligible ($\leq 7\%$).

In order to show the performance of our algorithms on large problems, we give results for some slightly modified problems of the NETGEN set (the upper bounds on x were removed). The matrices of these large problems ($n = 15.325, 27.887, 52.809,\ m = 5.000$) have a sparsity of less than 0.06%; the calculation of the Cholesky decomposition of the reordered matrix M takes 41–251 seconds. Since, on the other hand, an iteration of the CG algorithm takes 0.01–0.08 seconds, we expect that the use of Method C will result in a substantial decrease of the needed calculation time. The following tables give the results for $\varepsilon = 1e-07$ or $5e-03$. (Smaller values for ε produced numerical problems when calculating the Cholesky decomposition of M.)

We notice two facts. First, the use of Method C results in an increase in the number of iterations, but because of the high number of search directions that are calculated with Method B a speed-up between 1.72 and 7.13 (usually ≈ 3) is reached. Secondly, the shortest running time is reached with Algorithm 2 and Method C. This is somewhat surprising, because in most iterations it is necessary to use the CG algorithm for the solution of two linear systems. In Figure 3, we finally have a look at the time that Algorithm 2 needs to calculate a search direction via Method A and Method B for the problem `NETGEN 103`.

The plot in Figure 3 shows that in the first iterations Method B is much faster than Method A. Because the number of iterations of Algorithm 2 increases by only two if Method C is used instead of Method A, the use of Method C results in a huge decrease in processing time.

Algorithm 1: Results for NETGEN problems, $\varepsilon = 1e-07$						
	Method A		*Method C*			
	iterations	*seconds*	*iterations*		*seconds*	*Speed-up*
				total	*Method B*	
101	38	7589	43	38	2315	3.28
102	40	8024	45	32	3111	2.58
103	39	7978	46	38	2647	3.01
105	37	7863	44	34	2979	2.64
108	49	12453	56	30	6887	1.81
109	54	12461	44	31	3397	3.67

Algorithm 1: Results for NETGEN problems, $\varepsilon = 5e-03$						
	Method A		*Method C*			
	iterations	*seconds*	*iterations*		*seconds*	*Speed-up*
				total	*Method B*	
104	24	4932	36	36	1163	4.24
106	22	920	26	21	433	2.08
107	32	7848	42	42	1995	3.93
110	23	943	26	20	500	1.89

Algorithm 2: Results for NETGEN problems, $\varepsilon = 1e-07$						
	Method A		*Method C*			
	iterations	*seconds*	*iterations*		*seconds*	*Speed-up*
				total	*Method B*	
101	20	4262	25	21	1494	2.85
102	25	4891	27	19	1767	2.77
103	23	4796	25	21	1640	2.92
105	20	4317	23	20	1492	2.89
108	28	7366	30	23	2504	2.94
109	26	6341	31	24	2427	2.61

Algorithm 2: Results for NETGEN problems, $\varepsilon = 5e-03$						
	Method A		*Method C*			
	iterations	*seconds*	*iterations*		*seconds*	*Speed-up*
				total	*Method B*	
104	15	3220	23	20	1081	2.98
106	13	597	21	16	348	1.72
107	17	4480	22	22	628	7.13
110	14	617	16	14	255	2.42

7. Concluding remarks. In this paper we proved the (polynomial) complexity of a class of inexact infeasible-interior-point algorithms. This class includes inexact variants of some practically efficient infeasible-interior-point algorithms, in particular variants of the algorithms of Kojima et al. and Mehrotra. The theory developed in this paper usually justifies the use of the Cholesky decomposition for determining a search direction, because the calculated search direction, which is afflicted with rounding errors, is in most cases an inexact search direction in the sense of this paper. Furthermore, we have seen that the use of Krylov subspace methods results in an increase in the number of iterations, and for large problems, in a huge decrease of the

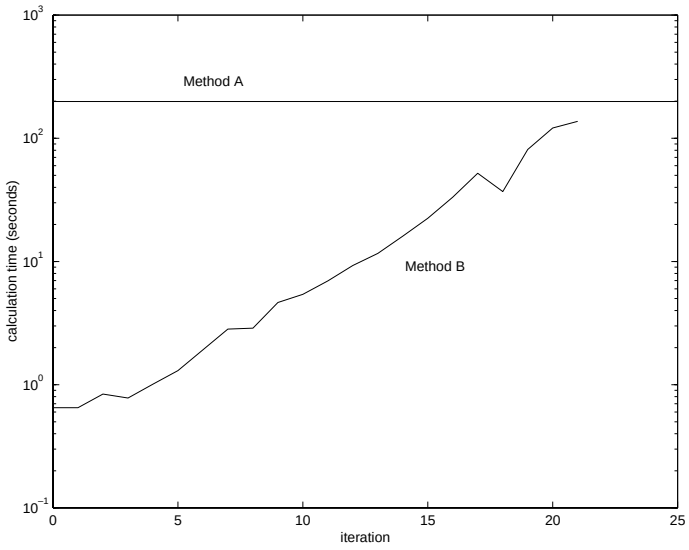


FIG. 3.

processing time. To make this kind of calculation more time efficient, it seems necessary to use a more sophisticated preconditioner and to calculate the search directions with the help of stable linear systems if $(x^k)^T z^k$ approaches zero (e.g., one can try to use $(M + \mu I)$ for some $\mu > 0$). Another approach is to use smaller step sizes, because the used long step sizes are at least partially the reason for the high condition number of M even for small k. The use of smaller step sizes will result in an increase of the number of iterations of Algorithm 1 and Algorithm 2, but if the step sizes are chosen carefully, this can nevertheless lead to a decrease in the total processing time. Some numerical evidence for this is that we can achieve a decrease of 22% in processing time for problem `israel` and Algorithm 1 with Method B if we set $\beta_4 = 0.99$ instead of $\beta_4 = 0.99995$.

Acknowledgment. I am grateful to the referees for suggestions and improvements upon the earlier versions of this manuscript.

REFERENCES

[1] R.W. FREUND AND F. JARRE, *A QMR-based interior-point algorithm for solving linear programs*, Math. Programming, 76 (1997), pp. 183–210.

[2] R.W. FREUND, F. JARRE, AND S. MIZUNO, *Convergence of a class of inexact interior-point algorithms for linear programs*, Math. Oper. Res., 24 (1999), pp. 50–71.

[3] M. KOJIMA, N. MEGIDDO, AND S. MIZUNO, *A primal-dual infeasible-interior-point algorithm for linear programming*, Math. Programming, 61 (1993), pp. 263–280.

[4] J. KORZAK, *Primal-Duale pfadfolgende inexakte Äußere-Punkte-Verfahren zur Lösung linearer Optimierungsaufgaben*, Dissertation, Fachbereich Mathematik, Bergische Universität - Gesamthochschule Wuppertal, 1997.

[5] 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.

[6] I.J. LUSTIG, R.E. MARSTEN, AND D.F. SHANNO, *Computational experience with a globally convergent primal-dual predictor-corrector algorithm for linear programming*, Math. Programming, 66 (1994), pp. 123–135.

[7] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.
[8] S. Mizuno and F. Jarre, *Global and polynomial-time convergence of an infeasible-interior-point algorithm using inexact computation*, Math. Program., 84 (1999), pp. 357–373.
[9] S. Mizuno and F. Jarre, *An infeasible-interior-point algorithm using projections onto a convex set*, Ann. Oper. Res., 62 (1996), pp. 59–80.
[10] S.M. Robinson, *A characterization of stability in linear programming*, Oper. Res., 25 (1977), pp. 435–447.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 149–159

AUCTION ALGORITHMS FOR SHORTEST HYPERPATH PROBLEMS*

R. DE LEONE† AND D. PRETOLANI†

Abstract. The auction-reduction algorithm is a strongly polynomial version of the auction method for the shortest path problem. In this paper we extend the auction-reduction algorithm to different types of shortest hyperpath problems in directed hypergraphs. The results of preliminary computational experiences show that the auction-reduction method is comparable to other known methods for specific classes of hypergraphs.

Key words. directed hypergraphs, hyperpaths, shortest paths, auction algorithms

AMS subject classifications. 90C35, 90B10, 05C65

PII. S1052623498343477

1. Introduction.

The shortest hyperpath problem is the extension to directed hypergraphs [12] of the classical shortest path problem (SPT) in directed graphs. Though not as pervasive as SPT, shortest hyperpaths have several relevant applications. In particular, they are at the core of traffic assignment algorithms for transit networks [15, 16, 20]. Shortest hyperpath models have been constructed for the SPT problem in stochastic and time-dependent networks [19] and for production planning in assembly lines [13]. Moreover, shortest hyperpath algorithms are used as building blocks of enumerative algorithms for hard combinatorial problems [11]. As a consequence, there is a growing interest for efficient shortest hyperpath algorithms. This provides motivations for further investigating known methods [12, 17], both from a theoretical and a practical point of view, and for developing new ones.

Auction algorithms were first proposed by Bertsekas [1, 2] for the assignment problem and later extended to general transportation problems [5, 6]. A survey of the auction algorithms for network optimization problems is contained in [4, Chapter 4]. Auction algorithms for shortest path problems on graphs were proposed in [3]. For the single-origin single-destination case the method can be viewed as an application of the auction method (with $\epsilon = 0$) to a specifically constructed assignment problem, and finite termination of the procedure can be established. Furthermore, the algorithm is a dual coordinate ascent method. Strongly polynomial versions of the auction method were proposed by Pallottino and Scutellà [18], who define a pruning procedure that reduces the graph to the shortest path tree. Further improvements to this method are given in [7], where the pruning method is strengthened and the structure of the reduced graph is exploited to obtain a better time complexity. A variant of the auction algorithm with pruning is proposed in [8].

In this paper, we devise an auction method for shortest hyperpaths with nonnegative hyperarc weights, by slightly modifying the SPT algorithm given in [7]. Our method can be tailored to solve several types of shortest hyperpath problems; for the various cases, we provide a worst case complexity bound. Finally, we report the results of a preliminary computational experience.

*Received by the editors August 11, 1998; accepted for publication (in revised form) December 2, 1999; published electronically July 25, 2000. This work was partly supported by Fondo Ricerca di Ateneo, Università di Camerino.

http://www.siam.org/journals/siopt/11-1/34347.html

†Dipartimento di Matematica e Fisica, Università di Camerino, Via Madonna delle Carceri, 62032 Camerino (MC), Italy (deleone@camserv.unicam.it, pretola@campus.unicam.it).

In section 2 we give the basic definitions of hypergraphs and shortest hyperpaths. The proposed auction method for shortest hyperpaths is presented in section 3. Computational results and conclusions are presented in sections 4 and 5, respectively.

2. Shortest hyperpaths in directed hypergraphs. A *directed hypergraph* $\mathcal{H}$ is a pair $(\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of *nodes* and $\mathcal{E}$ is a set of *directed hyperarcs*; a hyperarc is a pair $e = (T(e), h(e))$, where $T(e) \subset \mathcal{V}$ is the *tail* of e, and $h(e) \in \mathcal{V} \backslash T(e)$ is its *head*. A detailed introduction to directed hypergraphs can be found in [12], where a more general definition of hypergraphs is introduced; the particular class of hypergraphs considered in this paper is called *B-graphs* in [12]. The *size* $|e|$ of hyperarc e is the number of nodes it contains in its tail and head:

$$|e| = |T(e)| + 1.$$

The hyperarc e is an *arc* if $|e| = 2$, and a *proper* hyperarc if $|e| > 2$. Denote by m_a and m_h the number of arcs and proper hyperarcs, respectively; let $m = m_a + m_h = |\mathcal{E}|$ and $n = |\mathcal{N}|$. The size of $\mathcal{H}$ is the sum of the cardinalities of its hyperarcs:

$$size(\mathcal{H}) = \sum_{e \in \mathcal{E}} |e|.$$

Given a node u, the *forward star* of u, $FS(u)$, is the set of hyperarcs e such that $u \in T(e)$ and the *backward star* of u, $BS(u)$, is the set of hyperarcs e such that $u = h(e)$.

A *path* P_{st}, of length q, in the hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ is a sequence:

$$P_{st} = (v_1 = s, e_1, v_2, e_2, \ldots, v_{q+1} = t),$$

where, for $1 \leq i \leq q$, $e_i \in \mathcal{E}$, $v_i \in T(e_i)$, and $v_{i+1} = h(e_i)$. The nodes s and t are the *origin* and the *destination* of the path P_{st}, respectively. We say that node t is connected to node s in $\mathcal{H}$ if a path P_{st} exists in $\mathcal{H}$. If $t \in T(e_1)$, then the path P_{st} is a *cycle*. A path is *cycle-free* if it does not contain any subpath which is a cycle, i.e., $v_i \in T(e_j) \Rightarrow j \geq i$, for $1 \leq i \leq q + 1$.

Given a hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$ and two nodes $s, t \in \mathcal{V}$, a *hyperpath* π_{st} is a minimal hypergraph (with respect to deletion of nodes and hyperarcs) $\mathcal{H}_\pi = (\mathcal{V}_\pi, \mathcal{E}_\pi)$ such that

1. $\mathcal{E}_\pi \subseteq \mathcal{E}$;
2. $s, t \in \mathcal{V}_\pi = \bigcup_{e \in \mathcal{E}_\pi} (T(e) \cup \{h(e)\})$;
3. $u \in \mathcal{V}_\pi$, $u \neq s \implies u$ is connected to s in $\mathcal{H}_\pi$ by a cycle-free path.

Observe that for each $u \in \mathcal{V}_\pi \setminus \{s\}$ there exists a unique hyperarc $e \in \mathcal{E}_\pi$ such that $h(e) = u$; e is the *predecessor hyperarc* of u in π and is denoted by $e_\pi(u)$. We say that node t is *hyperconnected* to s in $\mathcal{H}$ if there exists a hyperpath from s to t in $\mathcal{H}$.

Given a hyperarc a, we say that a hyperarc a_r is *contained* in a, or is a *reduction* of a, if $h(a_r) = h(a)$, and $\emptyset \neq T(a_r) \subseteq T(a)$. Note that a is contained in itself, and a_r is *strictly contained* in a if $T(a_r) \subset T(a_r)$. Given a and $u \in T(a)$, we denote by $a \setminus u$ the reduction of a obtained deleting u from $T(a)$. We say that a hypergraph is *full* when it contains all the possible reductions of each of its proper hyperarcs. A full hypergraph can be represented by its *support* hypergraph $\mathcal{H}_s$, obtained by deleting all the strictly contained hyperarcs. Conversely, given any hypergraph $\mathcal{H}$, we can obtain the corresponding full hypergraph $\mathcal{H}_f$ by adding all the strictly contained hyperarcs.

2.1. Shortest hyperpaths. A *weighted hypergraph* is such that each hyperarc e is assigned a nonnegative real weight $w(e)$. The weight of a hyperpath in a weighted hypergraph can be defined in several ways. It is known that some definitions lead to intractable shortest hyperpath problems [10]. Here we restrict ourselves to definitions that are known to be tractable [12].

A *weighting function* is a node function that, given a hyperpath $\pi_{st} = (\mathcal{V}_\pi, \mathcal{E}_\pi)$, assigns a value $W_\pi(u)$ to its nodes, depending on the weights of its hyperarcs. The value $W_\pi(t)$ is the weight of π under the chosen weighting function. An *additive* weighting function satisfies the properties that $W_\pi(s) = 0$, and for $u \neq s$, $W_\pi(u)$ is a function of the predecessor $e_\pi(u)$ only. Formally, an additive weighting function can be defined by means of the following recursive equations:

$$W_\pi(u) = \begin{cases} 0 & \text{if } u = s, \\ w(e_\pi(u)) + F(e_\pi(u)) & \text{if } u \in \mathcal{V}_\pi / \{s\}, \end{cases} \tag{2.1}$$

where $F(e)$ is a nondecreasing function of the weights of the nodes in $T(e)$. Clearly, many different additive weighting functions can be defined (see [12]); consider first the *value* function, which is obtained by defining $F(e)$ in (2.1) as follows:

$$F(e) = \sum_{u \in T(e)} W_\pi(u).$$

The *minimum value* (shortest hyperpath) problem consists of finding a set of minimum value hyperpaths from the origin node s to each node $u \neq s$ hyperconnected to s. We denote by $V(u)$ the minimum value of a hyperpath π_{su} in $\mathcal{H}$; we assume $V(s) = 0$, and $V(u) = +\infty$ if u is not hyperconnected to s.

The *distance* function is obtained from (2.1) defining $F(e)$ as follows:

$$F(e) = \max\{W_\pi(u) : \ u \in T(e)\}.$$

The *minimum distance* problem asks for the minimum distance hyperpaths from origin s. Denote by D the vector of minimum distances, where again $D(s) = 0$, and $D(u) = +\infty$ if u is not hyperconnected to s.

Since arc weights are nonnegative, the minimum value and minimum distance problems can be solved efficiently by procedure *SBT* [12]. A computational analysis of several variants of *SBT* can be found in [17].

The *value* function considered here is a particular case of the more general function defined in [14, 12], where $F(e)$ is a generic weighted sum with nonnegative weights. A linear programming formulation of the minimum value problem can be given in terms of *flows in hypergraphs* [14]. This formulation cannot be extended to the *distance* function, for which only a *bounded MIP-representation* has been given [11].

2.1.1. Minimum time problems in transit hypergraphs. The problem of finding the passenger's expected travel time is at the core of several *urban transit networks* models. This problem has been formulated in terms of hyperpaths in transit networks [15] and in *F-graphs* [12, 16]. Here, introducing the *time* weighting function in *transit hypergraphs*, we define a particular shortest hyperpath problem that, though using a different (and slightly more general) terminology, is equivalent to the formulations found in [15, 16].

A *transit* hypergraph is a weighted support hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where a positive parameter ϕ_u is associated with each node $u \in \mathcal{V}$. Let $\mathcal{H}_f = (\mathcal{V}, \mathcal{E}_f)$ be the

full hypergraph represented by $\mathcal{H}$. Consider an $e \in \mathcal{E}_f$ contained in a proper hyperarc $a \in \mathcal{H}$: the *time* weighting function is obtained from (2.1) by defining the weight $w(e)$ and the function $F(e)$ as follows:

$$w(e) = 1/\Phi(e),$$
$$F(e) = \sum_{u \in T(e)} \frac{\phi_u}{\Phi(e)} W_\pi(u),$$

where

$$\Phi(e) = \sum_{u \in T(e)} \phi_u.$$

In practice, $F(e)$ is the weighted average (with weights $\phi.$) of the values $W_\pi(\cdot)$ in $T(e)$, while $w(e)$ is the inverse of the sum of the weights $\phi.$ in $T(e)$. For an arc $e = (\{u\}, v) \in \mathcal{E}_f$ corresponding to an arc $a \in \mathcal{E}$ the function $F(e)$ is defined in the same way, which gives $F(e) = W_\pi(u)$; however, in this case $w(e) = w(a)$ can be any nonnegative value.

The *minimum time* problem consists of finding the minimum time hyperpaths, from a given origin s, in the full hypergraph $\mathcal{H}_f$. Note that $\mathcal{H}_f$ may be considerably larger than its support $\mathcal{H}$: in practice, solving the minimum time problem efficiently requires one to work directly on $\mathcal{H}$. This is the aim of the following observations.

Denote by E the vector of minimum times, where $E(s) = 0$ and $E(u) = \infty$ if u is not hyperconnected to s in $\mathcal{H}_f$. For each $e \in \mathcal{E}_f$ denote by $t(e)$ the value of the *time* weighting function for e with respect to E, i.e.,

$$t(e) = w(e) + \sum_{u \in T(e)} \frac{\phi_u}{\Phi(e)} E(u).$$

Consider a proper hyperarc $a \in \mathcal{E}$, and let $R(a) \subset \mathcal{E}_f$ be the set of the reductions of a. We are interested in finding a reduction of a yielding minimum time, i.e., a hyperarc $e(a) \in R(a)$ such that

$$t(e(a)) = \min_{e \in R(a)} t(e).$$

Consider the nodes in $T(a)$ in increasing order of E, i.e., let $T(a) = \{u_1, \ldots, u_k\}$, with $E(u_i) \le E(u_{i+1})$, $1 \le i < k$. For $1 \le i \le k$ let a_i be the reduction of a such that $T(a_i) = \{u_1, \ldots, u_i\}$. The following property holds:

$$t(e(a)) = \min_{e \in R(a)} t(e) = \min_{1 \le i \le k} t(a_i). \tag{2.2}$$

In order to see why (2.2) is true, consider two different reductions e, e' of a, where $T(e') = T(e) \cup \{u\}$; it follows from the definition of *time* that

$$t(e') - t(e) = \frac{t(e)\Phi(e) + E(u)\phi_u - t(e)\Phi(e')}{\Phi(e')} = \frac{\phi_u}{\Phi(e')}\Big(E(u) - t(e)\Big).$$

In other words, $T(e')$ is smaller (greater) than $T(e)$ if and only if $E(u) < t(e)$ ($E(u) > t(e)$, respectively). This implies the following relations [15, Proposition 6]:

$$\begin{aligned} &t(e(a)) \ge E(w) \qquad \forall w \in T(e(a)), \\ &t(e(a)) \le E(w) \qquad \forall w \in T(a) \setminus T(e(a)), \end{aligned} \tag{2.3}$$

which in turn imply (2.2).

According to the previous observations, we can find the hyperarc $e(a)$ without considering the whole set of reductions of a. This can be done by processing the nodes in $T(a)$ in the order $u_1, u_2, \ldots, u_k$. For each $u_i \in T(a)$, compute the value $t(a_i)$: if $E(u_i) < t(a_i)$, then $u_i \in T(e(a))$, otherwise, $e(a) = a_{i-1}$. This technique has been used to compute expected travel times efficiently [15, 16] and will be adopted in our auction algorithm for the minimum time problem.

2.1.2. Reductions and shortest hyperpaths. A *hyperarc reduction* operation on a proper hyperarc a consists of replacing a by a hyperarc a_r contained in a, returning a *reduced* hypergraph. Clearly, if the hypergraph is weighted, a nonnegative weight $w(a_r)$ must be assigned to a_r. The following propositions show that by suitably choosing the weight on a_r a reduction operation does not modify the optimal solution of the shortest hyperpath problem. Proofs are rather straightforward and are omitted.

Suppose we are given a weighted hypergraph $\mathcal{H}$, and the corresponding vectors of optimal solutions V, D for the value and distance weighting functions. Given a proper hyperarc a and $u \in T(a)$, consider replacing a by $a_r = a \setminus u$.

PROPOSITION 2.1. *If $w(a_r) = w(a) + V(u)$, V is the vector of optimal values in the reduced hypergraph.*

PROPOSITION 2.2. *If $D(u) \leq \max\{D(v) : v \in T(a_r)\}$ and $w(a_r) = w(a)$, D is the vector of optimal distances in the reduced hypergraph.*

Now suppose we are given a transit hypergraph $\mathcal{H}$, the corresponding full hypergraph $\mathcal{H}_f$, and the optimum times E. Let e be a proper hyperarc in $\mathcal{H}_f$, with $t(e) > E(u)$ for each $u \in T(e)$, and consider replacing e by an arc $e_r = (\{u\}, h(a))$, with $u \in T(e)$ as follows.

PROPOSITION 2.3. *If $w(e_r) = t(e) - E(u)$, E is the vector of optimal times in the reduced full hypergraph.*

3. Auction algorithms for shortest hyperpaths. In this section we propose an auction method for the minimum value problem and discuss the adaptation to other weighting functions. Before introducing our approach, we briefly recall some relevant features of the auction algorithms for SPT; the reader is referred to the cited literature for further details.

The auction algorithm for shortest path problems on graphs maintains a path P (the *candidate* path) starting at the origin s and a set of *dual* node prices p satisfying the following *complementary slackness* (CS) conditions:

$$\begin{array}{rcll} p(i) & \leq & c_{ij} + p(j) & \forall (i,j), \\ p(i) & = & c_{ij} + p(j) & \forall (i,j) \in P, \end{array} \tag{3.1}$$

where c_{ij} is the cost of arc (i, j). The algorithm consists of three basic operations: path *extension*, path *contraction*, and dual *price raise*. At each iteration, the candidate path P is possibly extended, by adding a new node at the end of the path without violating (3.1). When no extensions are possible, the dual price of the terminal node i in P is raised, and if $i \neq s$ the path is contracted by deleting node i. For the single-origin single-destination case the algorithm terminates when the destination node is reached; several variants have been devised, also for the multiple-destination case.

Consider the case of nonnegative costs and dual prices initially set to zero. At the *first scan* of a node i (i.e., when node i becomes the last node in P for the first time) the optimal distance of node i is determined; indeed, it is equal to $p(s)$. As a consequence, since $p(s)$ is never decreased during the algorithm, the sequences of

first scan operations ranks the nodes in increasing order of distance from the origin s. Based on the above property, the *auction-reduction* method [18] introduces the following *reduction* operation: at the first scan of node i, delete each arc entering i, except the last arc in P. By means of these reduction operations, the graph is transformed into the shortest path tree, and a strongly polynomial time complexity can be obtained. Further reduction operations, including deletion of nodes, have been proposed in [7], improving the complexity bound.

The following observation is at the core of our auction shortest hyperpath method.

OBSERVATION 3.1. *According to the definition of value, distance, and time weighting functions, for an arc* $a = (\{u\}, v)$ *we have* $F(a) = W_\pi(u)$.

In other words, the weighting functions above define a standard SPT problem if the hypergraph is a directed graph. This suggests the following technique:

- apply the auction-reduction SPT algorithm to the arcs of the hypergraph;
- at the first scan of node i, apply hyperarc reduction operations according to Properties (2.1)–(2.3), possibly generating new arcs.

Note that the hypergraph is modified during the algorithm; each step is applied to the *current* hypergraph, as returned by the previous reduction operations. In practice, during the execution the proper hyperarcs lie in the not yet explored part of the hypergraph, and they are not considered until they are replaced by arcs as a result of successive reductions.

Our auction algorithm for the minimum value problem is described in procedure *MinValue*. Remark that procedure *MinValue* applied to a graph becomes the *auction algorithm with graph reduction* described in [7].

PROCEDURE MINVALUE($\mathcal{H}, s$)

Step 0 (initialization)
for each $u \in \mathcal{V}$: $pred(u) := \emptyset$, $p(u) := 0$, $V(u) := l(u) := +\infty$;
$l(s) := 0$, $P := \{s\}$;

Step 1 $i := last(P)$; if $i = s$ and $FS_A(s) = \emptyset$ return V;
if $V(i) = +\infty$: perform steps (a)…(d) (first scan of i)
(a) (set value) $V(i) := l(i)$;
(b) (delete $BS(i)$) $\mathcal{E} := \mathcal{E} \setminus BS(i) \cup pred(i)$;
(c) (reduce hyperarcs) for each $a \in FS_H(i)$:
$a := a \setminus i$, $w(a) := w(a) + V(i)$;
(d) (update labels, delete arcs) for each $a = (i, j) \in FS_A(i)$:
if $l(i) + w(a) < l(j)$:
$\mathcal{E} := \mathcal{E} \setminus pred(j)$, $pred(j) := \{a\}$, $l(j) := l(i) + w(a)$;
otherwise, $\mathcal{E} := \mathcal{E} \setminus \{a\}$;

Step 2 (node deletion; contraction or expansion)
if $FS_A(i) = \emptyset$: $\mathcal{V} := \mathcal{V} \setminus \{i\}$, $\mathcal{E} := \mathcal{E} \setminus pred(i)$, contract P, go to Step 1;
if $p(i) = \min\{w(a) + p(j) |\ a = (i, j) \in FS_A(i)\}$: go to Step 4;

Step 3 (contraction) contract P; set

$$p(i) := \min_{a=(i,j)\in FS_A(i)} \{w(a) + p(j)\};$$

go to Step 1;

Step 4 (expansion) expand P by node j_i, where:

$$j_i := \arg \min_{a=(i,j)\in FS_A(i)} \{w(a) + p(j)\};$$

go to Step 1.

For each node i, the predecessor *pred*(i) gives the last arc in the best s-i path determined so far; for notational convenience, we consider *pred*(i) as a set; initially, *pred*$(i) = \emptyset$. The label $l(i)$ is the minimum s-i hyperpath value determined so far, which becomes the optimum s-i hyperpath value $V(i)$ at the first scan of node i. We denote by $FS_A(i)$ and $FS_H(i)$ the arcs and proper hyperarcs in $FS(i)$, respectively; thus $FS(i) = FS_A(i) \cup FS_H(i)$. Replacing hyperarc a by its reduction $a \setminus i$ is denoted by $a := a \setminus i$; note that $a \setminus i$ may be an arc. The last node in P is denoted by *last*(P).

During the execution of the algorithm, the *contained graph* $\mathcal{H}_A = (\mathcal{V}_A, \mathcal{E}_A)$ is the directed graph defined by the nodes and arcs in the current hypergraph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, i.e., $\mathcal{V}_A = \mathcal{V}$ and

$$\mathcal{E}_A = \bigcup_{i \in \mathcal{V}} FS_A(i).$$

PROPOSITION 3.1. *At each step of the algorithm, for each node $i \in \mathcal{V}_A$ such that $V(i) < +\infty$ the quantity $V(i)$ gives the shortest s-i path length in the current contained graph $\mathcal{H}_A$.*

Proof. The property follows from the correctness of the auction algorithm for SPT, observing that a new arc (i, j) is created only before the first scan of nodes i and j. □

THEOREM 3.2. *The vector V determined by the algorithm gives the minimum hyperpath values in the original hypergraph.*

Proof. The theorem can be proved by induction considering nodes in order of first scan, that is, in nondecreasing order of value $V(\cdot)$. The claim is clearly true at the beginning because to node s is assigned $V(s) = 0$. Assume that all the previously assigned V are correct at the first scan of node i. It follows from Proposition 3.1 that $l(i)$ is a lower bound on the length of any path in $\mathcal{H}_A$ from node s to each node j such that $p(j) = 0$. Therefore, in the current hypergraph, the value of any hyperpath containing a proper hyperarc cannot be less than $l(i)$. This implies that $V(i) = l(i)$ is correct; as a consequence, Step 1(c) does not change the optimal solution (Proposition 2.1). □

3.1. Other weighting functions. The auction algorithm for minimum value can be easily adapted to the minimum distance problem. To this aim, it suffices to skip the weight update $w(a) := w(a) + V(i)$ in Step 1(c). This follows from Property (2.2) since, when a is replaced by $a \setminus i$, $V(i) \leq V(j)$ for each node $j \in T(a \setminus i)$. At the end of the algorithm, $D = V$ gives the vector of optimal distances. The proof of correctness for the distance function is similar to the one of Theorem 3.2.

The situation is slightly more complex for travel times. Recall that our goal is to work with the support transit hypergraph, thus we must deal with the corresponding full hypergraph implicitly. To this aim, we replace hyperarc reductions by *arc insertion* operations, as described below.

Consider a proper hyperarc a in the support, with $T(a) = \{u_1, \ldots, u_k\}$ and $E(u_i) \leq E(u_{i+1})$, $1 \leq i < k$. We know that it suffices to consider the k reductions a_i, $1 \leq i \leq k$, with $T(a_i) = \{u_1, \ldots, u_i\}$ (see section 2.1.1). At the first scan of node $u_i \in T(a)$, we compute the value $t(a_i)$; if $t(a_i) > V(u_i)$, we generate an arc $(u_i, h(a))$ with weight $t(a_i) - V(u_i)$, according to Proposition 2.3. Otherwise, i.e., if $t(a_i) \leq V(u_i)$, we conclude that a_{i-1} is the reduction $e(a)$ of a yielding minimum time, and we delete a. If necessary, a is removed at the first scan of node u_k. In conclusion, for a proper hyperarc a, up to $|T(a)|$ arcs can be generated.

In order to compute each $t(a_i)$ efficiently, for each proper hyperarc a in the support hypergraph we keep two values, initially set to zero:

$$\sigma(a) = \sum_{\substack{u \in T(a) \\ V(u) < +\infty}} \phi_u V(u);$$
$$\varphi(a) = \sum_{\substack{u \in T(a) \\ V(u) < +\infty}} \phi_u.$$

At first scan of u_i, it is $t(a_i) = \sigma(a)/\varphi(a)$. We also keep a counter $k(a)$ of visited nodes in $T(a)$. We rewrite Step 1(c) as follows.

Step 1(c) (reduce hyperarcs) for each $a \in FS_H(i)$: $\sigma(a) := \sigma(a) + \phi_i V(i)$,

$\varphi(a) := \varphi(a) + \phi_i$, $t := \sigma(a)/\varphi(a)$;
if $t < l(h(a))$: add arc $a' = (i, h(a))$, $w(a') := t - V(i)$;
$k(a) := k(a) + 1$; if $t \le V(i)$ or $k(a) = |T(a)|$: delete a.

Observe that a new arc is added only if it can be used to improve the label of $h(a)$. In this case, the current predecessor *pred*$(h(a))$ will be deleted in Step 1(d); therefore, at most one arc generated from a belongs to the current contained graph at the end of Step 1.

The correctness of the auction algorithm for minimum times can be proved by induction, as we did for Theorem 3.2. The induction step requires a slightly more complicated analysis, given in the following lemma.

LEMMA 3.3. *If the values $V(\cdot)$ assigned before the first scan of node i are correct, then the value $V(i)$ assigned at the first scan of node i is correct.*

Proof. Let S be the set of nodes in the current hypergraph whose first scan occurred before first scan of node i. We know that, for $u \in S$, $V(u)$ gives the SPT distance from s in the contained graph; moreover, $V(i)$ is a lower bound on the SPT distance for each node $u \notin S$. Consider a generic proper hyperarc a in the current support hypergraph; note that $h(a) \notin S$ and $l(h(a)) \ge V(i)$. The reductions of a containing nodes in S only have been already considered by the algorithm, possibly adding new arcs; for each such reduction e, we have $t(e) \ge V(i)$. It follows from (2.3) that $V(i)$ is a lower bound on $t(e(a))$, and therefore, $V(i)$ is a lower bound on the minimum time $E(u)$ for each node $u \notin S$ in the current hypergraph. The thesis follows. ∎

3.2. Computational complexity. The auction-reduction algorithm presented in [7] solves the SPT problem on a graph $G = (V, E)$ in $O(|V| \min\{|E|, |V| \log |V|\})$ time. It is easy to see that the maximum number of arcs generated during the execution of *MinValue* is m, for the value and distance weighting functions, and $O(size(\mathcal{H}))$ for the time function. Moreover, the total time spent in first scans (Step 1) is $O(size(\mathcal{H}))$. Therefore, we can state the following proposition.

PROPOSITION 3.4. *The running time of the auction shortest hyperpaths algorithm is $O(size(\mathcal{H}) + n \min\{m, n \log n\})$, for value and distance, while for the time function it is $O(size(\mathcal{H}) + n \min\{size(\mathcal{H}), n \log n\})$.*

Two techniques for improving the running time of the auction-reduction method are presented in [7]: *path scanning* and *multipath restructuring*. The resulting complexity is $O(|V|^2 + |E|)$. In fact, the total computation time between two successive first scan operations is $O(|V|)$, and clearly there are at most $|V|$ first scans. The above techniques can be easily applied within our shortest hyperpath algorithm; the next proposition follows.

PROPOSITION 3.5. *The auction shortest hyperpath algorithm with path scanning or multipath restructuring takes $O(size(\mathcal{H}) + n^2)$ time, for the value, distance, and time functions.*

4. Computational results. In this section we present the preliminary computational results for auction methods for shortest hyperpath problems. Our main goal here is to compare a few variants of the basic method; a complete experimental evaluation of auction shortest hyperpath methods would require a much larger effort.

Our basic shortest hyperpath algorithm, denoted by HAR, is an implementation of procedure *MinValue* where we used the *last* data structure [7]. A variant of this algorithm, denoted by HAR2, makes use of the "second best" device [4, Chapter 4] too. We implemented a third version, denoted by HARn, where the "second best" device is not used, and a *node contraction* operation is introduced. A node contraction deletes a node k with indegree and outdegree equal to one: the arcs incident with node k, say (i,k) and (k,j), are replaced by an arc (i,j), where $w(i,j) = w(i,k) + w(k,j)$. Node contractions simplify the current graph and may help in keeping the current path shorter. A similar technique was introduced in [9].

We compared our auction algorithms to an implementation of procedure *SBT-heap* [12], denoted by SBTh. All algorithms were coded in C language, and run on an IBM RISC-6000 P43 workstation, with 64Mb RAM, using the AIX operating system.

In general, devising a reasonable experimental setup for shortest hyperpaths is not a trivial task, since hypergraphs show many more degrees of freedom than graphs (see, e.g., [17]). Here, we restricted ourselves to one weighting function, namely the distance, and we considered two different hypergraph topologies: *random* and *grid*.

Random hypergraphs do not show any special structures, except that the origin s is a distinguished node, and $FS(s)$ contains only arcs. The size of proper hyperarcs is chosen randomly in the interval $[d_{\min}, d_{\max}]$. In our experiments, we set $d_{\min} = 3$ and $|FS(s)| = 125$, and we defined five classes of random hypergraphs with different values of $d_{\max}$, n, m_h, and m_a. For proper hyperarcs, and for arcs exiting the root, weights were generated randomly in the interval $[0, \frac{1}{10}]$; for the remaining arcs, weights belong to $[\frac{1}{10}, 1]$. This choice has been motivated by the attempt to increase the relevance of hyperarcs.

The results for random hypergraphs are shown in Table 4.1. For each class, the value δ is the expected size of $FS(u)$ for $u \neq s$. Execution times are given in milliseconds; each entry is the (rounded) average of 20 instances.

In a grid hypergraph nodes are arranged in a $b \times h$ grid; a node is identified by its coordinates (x,y), $1 \leq x \leq b$, $1 \leq y \leq h$. Nodes with the same x coordinate form a *level*; for each pair (x,y) and (x,y'), with $y' = y + 1 \bmod h$, there are two *vertical* arcs $((x,y),(x,y'))$ and $((x,y'),(x,y))$. Hyperarcs connect nodes in successive levels; for each (x,y) with $1 < x \leq b$ there exists a hyperarc

$$\Big(\big\{(x-1,y'),(x-1,y'')\big\},(x,y)\Big),$$

where $y' = y + 1 \bmod h$, and $y'' = y - 1 \bmod h$. In addition, there is an origin node s, and arcs $(s,(1,y))$ for each $1 \leq y \leq h$.

We generated three classes of grid hypergraphs: *square*, where $b = h$, *long*, where $b \gg h$, and *high*, where $h \gg b$. Parameters b and h were chosen in order to have the same number of nodes in the three classes. Hyperarc weights lay in the interval $[0,1]$; vertical arcs weights lay in $[1,2]$; weights of arcs leaving s lay in $[0, \frac{1}{10}]$.

Execution times are reported in Table 4.2. Each entry is the (rounded) average of five instances; times are given in seconds.

TABLE 4.1
Random hypergraphs.

n	$4,000$	$2,000$	$1,000$	$1,000$	$1,000$
m_a	$2n$	$25n$	$2n$	$25n$	$50n$
m_h	$2n$	$2n$	$25n$	$25n$	$50n$
$d_{\max}$	5	5	7	7	7
δ	8	31	102	125	250
HAR	171	117	131	144	273
HAR2	144	98	126	135	257
HARn	176	123	135	150	277
SBT	65	90	95	125	270

TABLE 4.2
Grid hypergraphs.

	High		*Square*		*Long*	
b	80	100	400	500	$2,000$	$2,500$
h	$2,000$	$2,500$	400	500	80	100
HAR	44	88	86	172	370	785
HAR2	34	67	87	177	388	798
HARn	44	85	86	176	370	783
SBT	2.22	3.85	1.18	2.06	.65	1.15

Though clearly incomplete, the above results allow us to draw some conclusions. For what concerns random hypergraphs, our auction algorithms are comparable to standard label-setting methods, that are the most efficient for this class of hypergraphs [17]. Auction methods become more and more competitive as the density increases; in one case, HAR2 gives the best results. On the other hand, auction methods do not seem to be suitable for large grid hypergraphs. This result (that matches the computational results for auction methods for long grid graphs) is not surprising, since the auction algorithm must maintain a long current path P in order to connect nodes in the last layers.

The "second best" data structure gives the best results for random hypergraphs, and for high grids, but it is not suitable for square and long grids. Again, this result is not surprising, since in a grid hypergraph there exist at most two hyperarcs (plus two vertical arcs) leaving each node; it is conceivable that the good results for high grids are due to savings obtained when scanning the origin node.

On the contrary, the node contraction operation is almost useless, also for grid hypergraphs. This result is rather disappointing, since in some preliminary experiments this operation proved to be very effective on some classes of grid graphs. A possible explanation may be the following: if a node has the highest distance in the tail of a hyperarc, it is likely to have the highest distance also in the tail of the other hyperarc it belongs to; in this case, hyperarc reduction may create two arcs leaving the node, so that node contraction cannot be applied. This observation may suggest some guidelines for improving our algorithms.

5. Conclusions. In this paper, we propose an auction method for shortest hyperpath problems, that can be adapted to several types of weighting functions. Our method is derived, with minor changes, from the auction-reduction SPT algorithm. Indeed, an appealing feature of our approach is that several techniques originally developed for graphs could be easily exported to hypergraphs.

From a practical point of view, auction shortest hyperpath methods are comparable to other known methods, at least in favorable cases. As one would expect, their

behavior can be dramatically affected by the structure of the underlying hypergraph; however, this seems to resemble closely what happens for graphs.

We may conclude that auction shortest hyperpath methods are an interesting topic for further research, both on the theoretical and the practical side. A possible direction could be adapting some of the variants proposed in the literature, such as the price raise technique devised in [8]. In particular, we are currently investigating the *forward-reverse* approach, which proved to be quite effective for single-origin single-destination (or few destinations) [4, Chapter 4].

REFERENCES

[1] D. Bertsekas, *A Distributed Algorithm for the Assignment Problem*, Tech. report, Laboratory for Information and Decision System, Massachusetts Institute of Technology, Cambridge, MA, 1979.

[2] D. Bertsekas, *The auction algorithm: A distributed relaxation method for the assignment problems*, Ann. Oper. Res., 14 (1988), pp. 105–123.

[3] D. Bertsekas, *An auction algorithm for shortest paths*, SIAM J. Optim., 1 (1991), pp. 425–447.

[4] D. Bertsekas, *Linear Network Optimization: Algorithms and Codes*, The MIT Press, Cambridge, MA, 1991.

[5] D. Bertsekas and D. Castanon, *The Auction Algorithm for the Minimum Cost Network Flow Problem*, Tech. report LIDS-P-1925, Laboratory for Information and Decision System, Massachusetts Institute of Technology, Cambridge, MA, 1989.

[6] D. Bertsekas and D. Castanon, *The auction algorithm for transportation problems*, Ann. Oper. Res., 20 (1989), pp. 67–96.

[7] D. Bertsekas, S. Pallottino, and M. Scutellà, *Polynomial auction algorithms for shortest paths*, Comput. Optim. Appl., 2 (1995), pp. 99–125.

[8] R. Cerulli, R. De Leone, and G. Piacente, *A modified auction algorithm for the shortest path problem*, Optim. Methods Softw., 4 (1994), pp. 209–224.

[9] R. Cerulli, P. Festa, and G. Raiconi, *Graph Collapsing in Shortest Path Auction Algorithms*, Tech. report TR 6/97, Dipartimento di Informatica e Applicazioni, Università di Salerno, Salerno, Italy, 1997.

[10] G. G. Ausiello, G. Italiano, and U. Nanni, *Dynamic maintenance of directed hypergraphs*, Theoret. Comput. Sci., 72 (1990), pp. 97–117.

[11] G. Gallo, C. Gentile, D. Pretolani, and G. Rago, *Max Horn SAT and the minimum cut problem in directed hypergraphs*, Math. Programming, 80 (1998), pp. 213–237.

[12] G. Gallo, G. Longo, S. Nguyen, and S. Pallottino, *Directed hypergraphs and applications*, Discrete Appl. Math., 42 (1993), pp. 177–201.

[13] G. Gallo and M. G. Scutellà, *Minimum Makespan Assembly Plans*, Tech. report TR-98-10, Dipartimento di Informatica, Università di Pisa, Pisa, Italy, 1998.

[14] R. Jeroslow, K. Martin, R. Rardin, and J. Wang, *Gainfree Leontief substitution flow problems*, Math. Programming, 57 (1992), pp. 375–414.

[15] S. Nguyen and S. Pallottino, *Equilibrium traffic assignment for large scale transit networks*, European J. Oper. Res., 37 (1988), pp. 176–186.

[16] S. Nguyen, S. Pallottino, and M. Gendreau, *Implicit enumeration of hyperpaths in logit models for transit networks*, Transport. Sci., 32 (1998), pp. 54–64.

[17] S. Nguyen and D. Pretolani, *A Computational Study of Shortest Hyperpath Algorithms*, Tech. report CRT-95-34, Centre de Recherche sur les Transports, Université de Montréal, Montréal, Canada, 1995.

[18] S. Pallottino and M. Scutellà, *Strongly polynomial auction algorithms for shortest paths*, Ricerca Operativa, 60 (1991), pp. 33–54.

[19] D. Pretolani, *A directed hypergraph model for random time dependent shortest paths*, European J. Oper. Res., 123 (2000), pp. 89–98.

[20] J. Wu and M. Florian, *A simplicial decomposition method for the transit equilibrium assignment problem*, Ann. Oper. Res., 44 (1993), pp. 245–260.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 160–178

CALM MINIMA IN PARAMETERIZED FINITE-DIMENSIONAL OPTIMIZATION*

A. B. LEVY†

Abstract. Calmness is a restricted form of local Lipschitz continuity where one point of comparison is fixed. We study the calmness of solutions to parameterized optimization problems of the form

$$\min\{f(x,w)\} \text{ over all } x \in \mathbb{R}^n,$$

where the extended real-valued objective function f is continuously prox-regular in x with compatible parameterization in w. This model covers most finite-dimensional optimization problems, though we focus particular attention here on the case of parameterized nonlinear programming. We give a second-order sufficient condition for there to exist unique optimal solutions that are calm with respect to the parameter. We also characterize a slightly stronger stability property in terms of the same second-order condition, thus clarifying the gap between our sufficient condition and the calmness property. In the case of nonlinear programming, our results complement a long study of the stability properties of optimal solutions: for instance, one consequence of our results is that the Mangasarian–Fromovitz constraint qualification when paired with a new (and relatively weak) second-order condition ensures the calmness of solutions to parameterized nonlinear programs.

Key words. parameterized optimization, nonlinear programming, stability, calmness, Lipschitz continuity, continuously prox-regular function

AMS subject classification. 90C31

PII. S1052623498340783

1. Introduction.

In this paper, we study the stability with respect to the parameter $w \in \mathbb{R}^d$ of solutions to the optimization problems

$$\min\{f(x,w)\} \text{ over all } x \in \mathbb{R}^n, \tag{1}$$

where the extended real-valued objective function f is continuously prox-regular in x with compatible parameterization by w. To measure solution stability, we use a weak form of Lipschitz continuity called "calmness."

DEFINITION 1.1. *We say that $\bar{x}$ gives a* calm local minimum for f *if there exists a $\delta > 0$ and a neighborhood $W \in \mathbb{R}^d$ of 0 such that for each parameter $w \in W$, there exists a unique solution $x(w)$ to*

$$\min\{f(x,w)\} \text{ over all } x \in \mathbb{R}^n \text{ with } |x - \bar{x}| \le \delta$$

and that $x(w)$ satisfies

$$|x(w) - \bar{x}| \le K|w| \quad \text{for all } w \in W \tag{2}$$

for some fixed $K > 0$.

Notice that the calmness bound (2) is slightly weaker than the usual local Lipschitz continuity since the base point $\bar{x}$ is required in (2) to always be one of the two points considered for comparison. Nonetheless, calmness is obviously a very useful and important stability property since it gives a Lipschitz bound on the distance

*Received by the editors June 22, 1998; accepted for publication (in revised form) February 23, 2000; published electronically July 25, 2000.
http://www.siam.org/journals/siopt/11-1/34078.html
†Department of Mathematics, Bowdoin College, Brunswick, ME 04011 (alevy@bowdoin.edu).

of perturbed solutions from the unperturbed solution. Note also that the calmness bound (2) itself, without the existence and uniqueness assertions, is essentially the property of local upper Lipschitz continuity studied widely (for example, in [21], [11], [3], and [8]).

The class of continuously prox-regular functions was introduced in [17] where it was shown to include all $\mathcal{C}^{1,1}$ functions, all lower semicontinuous, proper, convex functions, all lower-$\mathcal{C}^2$ functions, all primal-lower-nice functions (see [16]), and all "strongly amenable functions" (convex functions composed with $\mathcal{C}^2$ mappings; see [22], for example). This list covers most of the objective functions in finite-dimensional optimization, including constrained optimization where constraints are incorporated into the objective via infinite penalties. In this paper, we study continuously prox-regular parameterized functions which are continuously prox-regular in a uniform way with respect to the parameterization, and we call such functions *continuously prox-regular with compatible parameterization* (see the following section for more details). One important example in parameterized constrained optimization on which we will focus particular attention is the case when f represents the essential objective associated with the nonlinear program:

$$\min\{g_0(x,w)\} \text{ over all } x \in C(w), \tag{3}$$

where g_0 is of class $\mathcal{C}^2$ and the constraint set is defined by $\mathcal{C}^2$ functions g_i as follows:

$$C(w) := \{x \in \mathbb{R}^n : g_i(x,w) \le 0 \text{ if } i = 1,\dots,s \text{ and } g_i(x,w) = 0 \text{ if } i = s+1,\dots,m\}.$$

The essential objective f in this case is defined for pairs (x,w) satisfying $x \in C(w)$ by $f(x,w) = g_0(x,w)$ and by $f(x,w) = \infty$ otherwise.

Our main result is a sufficient condition for $\bar{x}$ to give a calm local minimum for a continuously prox-regular function f with compatible parameterization. We assume that 0 is a partial subgradient with respect to x of f at $\bar{x}$, denoted $0 \in \partial_x f(\bar{x},0)$, since this is a necessary condition for local optimality. We also assume the following constraint qualification:

$$(0,y) \in \partial^\infty f(\bar{x},0) \Rightarrow y = 0, \tag{4}$$

in terms of the set $\partial^\infty f(\bar{x},0)$ of horizon subgradients of f at $(\bar{x},0)$. Our result is stated in terms of two different generalized second-order derivatives: a strong partial outer graphical derivative $\tilde{D}^2_{xx} f(\bar{x},0|0)$, and a partial outer graphical derivative $D^2_x f(\bar{x},0|0)$ (see the following section for details on partial subgradients, horizon subgradients, continuous prox-regularity, and the generalized second-derivatives).

THEOREM 1.1. *Consider a function $f : \mathbb{R}^{n+d} \to \mathbb{R} \cup \{\infty\}$ that is continuously prox-regular in x at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ with compatible parameterization in w at 0. If the constraint qualification* (4) *holds, and the strong partial outer graphical derivative $\tilde{D}^2_{xx} f(\bar{x},0|0)$ is positive-definite in the sense that*

$$v' \in \tilde{D}^2_{xx} f(\bar{x},0|0)(x') \Rightarrow \langle v', x' \rangle > 0 \text{ unless } x' = 0, \tag{5}$$

then there exists a $\delta > 0$ and a neighborhood $W \in \mathbb{R}^d$ of 0 such that for each parameter $w \in W$, there exists a unique solution $x(w)$ to

$$\min\{f(x,w)\} \text{ over all } x \in \mathbb{R}^n \text{ with } |x - \bar{x}| \le \delta.$$

If, in addition, the partial outer graphical derivative $D^2_x f(\bar{x},0|0)$ satisfies the kernel condition

$$0 \in D^2_x f(\bar{x},0|0)(x',0) \Rightarrow x' = 0, \tag{6}$$

then the point $\bar{x}$ gives a calm local minimum for f (2).

The gap between the sufficient conditions (5) and (6) and the calmness property is shown in Theorem 3.1 to include only functions whose local minima are not Lipschitz stable with respect to linear perturbations (a property here called "tilt stability"). In the special case of parameterized nonlinear programming (3), the constraint qualification (4) is just the usual Mangasarian–Fromovitz constraint qualification, and the second-order conditions (5) and (6) are both weaker than the general strong second-order sufficient condition defined in [4]. It follows that our result improves the previous sufficient conditions for calm local minima in nonlinear programming (see the final section for more details on this).

The stability properties of solutions to parameterized constrained optimization problems have been studied widely. Most of this study has focused on particular models such as nonlinear programs, but there has also been work on more general constrained optimization models. A very general result in the same spirit as our Theorem 1.1 is found in [14], where a sufficient condition is obtained for the Lipschitz continuity of optimal solutions to problems with continuously prox-regular objectives. However, this stronger stability property is obtained in [14] through very different generalized derivative objects, and consequently no application is made to nonlinear programs. Other results about calmness include a recent survey [1] of stability in constrained optimization covering a generalized nonlinear program where the constraints are of the form $G(x, w) \in K$ and containing sufficient conditions for the calmness of optimal solution selections associated with perturbations of the parameter along a fixed direction. Another paper [7] also considers general constrained optimization models and gives conditions for calmness of solution selections when the unperturbed problem can have multiple solutions. Various stability properties of parameterized nonlinear programs are studied in [4], [3], [9], [21], [19], [15], [5], [6], and [23], among others.

The organization of this paper illustrates how our approach is distinguished from many of the previous results in this area. Instead of using a particular optimization model from the beginning, we first derive sufficient conditions for calmness of solutions to the general model (1). With this basic theorem in hand, we can derive sufficient conditions for calmness of solutions to particular optimization problems merely by computing the appropriate generalized second derivatives, which we do for the case of nonlinear programming.

2. Continuous prox-regularity and generalized second derivatives. In dealing with subgradients, we follow the notation and terminology of the book [22]. For a function $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ and a point $x \in \mathbb{R}^n$, a vector $v \in \mathbb{R}^n$ is a *regular subgradient* of f at x if $f(x)$ is finite and $f(x + w) \geq f(x) + \langle v, w\rangle + o(|w|)$. It is a (*general*) *subgradient* at x if $f(x)$ is finite and there exist sequences $\{x^\nu\}_{\nu=1}^\infty$ and $\{v^\nu\}_{\nu=1}^\infty$ with v^ν a regular subgradient of f at x^ν, such that $v^\nu \to v$, $x^\nu \to x$, and $f(x^\nu) \to f(x)$. The set of all such (general) subgradients of f at x includes the regular subgradients at x and is denoted by $\partial f(x)$. A set-valued *subgradient mapping* $\partial f : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ is thereby defined, which is empty-valued outside of $\{x : f(x) < \infty\}$. The *graph* of ∂f is the set $\operatorname{gph} \partial f \subset \mathbb{R}^n \times \mathbb{R}^n$ consisting of the pairs (x, v) such that $v \in \partial f(x)$.

Also of use to us will be the concept of v being a *horizon subgradient* of f 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 f at x^ν, such that $x^\nu \to x$, $f(x^\nu) \to f(x)$, 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 f at x is

denoted by $\partial^\infty f(x)$.

We call a lower semicontinuous function $f : \mathbb{R}^{n+d} \to \mathbb{R} \cup \{\infty\}$ *continuously prox-regular in x at $\bar{x}$ for $\bar{v} \in \partial_x f(\bar{x}, 0)$ with compatible parameterization in w at* 0 if it is prox-regular and subdifferentially continuous in the sense of [22] and if these properties extend uniformly with respect to w in the following way: There exists a neighborhood $W \in \mathbb{R}^d$ of 0 such that

(i) (uniform subdifferential continuity) for any positive scalar η, there is a $\delta > 0$ such that whenever $w \in W$ and x_1, x_2 satisfy $|x_i - \bar{x}| \le \delta$ for $i = 1, 2$ with $v_i \in \partial_x f(x_i, w)$ satisfying $|v_i - \bar{v}| \le \delta$ for $i = 1, 2$, the function f satisfies $|f(x_1, w) - f(x_2, w)| \le \eta$; and

(ii) (uniform prox-regularity) there exist constants $\delta > 0$ and $r > 0$ such that for any $w \in W$, the function f satisfies

$$f(x_1, w) - \langle v, x_1\rangle \ge f(x_2, w) - \langle v, x_2\rangle - \frac{r}{2}|x_1 - x_2|^2 \tag{7}$$

whenever $|x_i - \bar{x}| \le \delta$ for $i = 1, 2$ and $v \in \partial_x f(x_2, w)$ with $|v - \bar{v}| \le \delta$.

For this paper, one important example of a continuously prox-regular function with compatible parameterization is the essential objective associated with the nonlinear program (3) under the Mangasarian–Fromovitz constraint qualification (cf. [14, Proposition 2.2]).

The generalized second-derivatives that we use here are obtained as outer graphical derivatives of the partial subdifferential multifunction $\partial_x f$. For any multifunction $T : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$, the *outer graphical derivative at x for $v \in T(x)$* is the multifunction $DT(x|v) : \mathbb{R}^m \rightrightarrows \mathbb{R}^n$ that is the outer graphical limit as $\tau \downarrow 0$ of the family of difference quotient multifunctions $\Delta_\tau T(x|v) : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ defined by

$$\Delta_\tau T(x|v)(x') := \frac{T(x + \tau x') - v}{\tau}.$$

Recall that the *outer graphical limit as $\tau \downarrow 0$* of a family of multifunctions $\Delta_\tau : \mathbb{R}^m \to \mathbb{R}^n$ is the multifunction $\Delta : \mathbb{R}^m \to \mathbb{R}^n$ whose graph $\operatorname{gph} \Delta$ agrees with the collection of all cluster points obtained from elements chosen from the graphs $\operatorname{gph} \Delta_{\tau_n}$ for sequences $\tau_n \downarrow 0$.

One generalized derivative of f that we use is the multifunction $D_x^2 f(x, w|v) : \mathbb{R}^{n+d} \rightrightarrows \mathbb{R}^n$ that is the outer graphical derivative at (x, w) for $v \in \partial_x f(x, w)$ of the multifunction $(x, w) \mapsto \partial_x f(x, w)$. This derivative is a generalization of the Hessian mapping

$$(x', w') \mapsto \nabla_{xx} f(x, w) \cdot x' + \nabla_{xw} f(x, w) \cdot w'.$$

Another generalized derivative that we use is the multifunction $D_{xx}^2 f(x, w|v) : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ that is the outer graphical derivative at x for $v \in \partial_x f(x, w)$ of the multifunction $x \mapsto \partial_x f(x, w)$. It follows that $D_{xx}^2 f(x, w|v)$ is a generalization of the Hessian mapping $x' \mapsto \nabla_{xx} f(x, w) \cdot x'$. (See [22] for more on graphical derivatives defined in this way.) In the present paper, we "strengthen" this latter generalized second derivative by taking a final outer graphical limit. We use the notation $\tilde{D}_{xx}^2 f(\bar{x}, 0|\bar{v})$ to denote the *strong partial outer graphical derivative of f at $\bar{x}$ for $\bar{v} \in \partial_x f(\bar{x}, 0)$* that is defined as the outer graphical limit as $x \to \bar{x}$, $w \to 0$, and $v \to \bar{v}$ of the sequence of partial outer graphical derivative multifunctions $D_{xx}^2 f(x, w|v)$.

3. Tilt stability and the proof of Theorem 1.1. In order to study the calmness of the optimal solutions to (1), it turns out to be useful to consider a slightly modified optimization problem

$$\min\{f(x,w) - \langle v, x\rangle\} \text{ over all } x \text{ satisfying } |x - \bar{x}| \le \delta, \tag{8}$$

where the graph of the objective function is "tilted" by the parameter v. Associated with this problem is the local solution mapping

$$P_\delta(w,v) := \operatorname*{argmin}_{x\in\mathbb{B}(\bar{x};\delta)}\{f(x,w) - \langle v, x\rangle\}, \tag{9}$$

where $\mathbb{B}(\bar{x};\delta)$ denotes the closed ball of radius δ about $\bar{x}$.

Definition 3.1. *We say that $\bar{x}$ gives a* tilt stable local minimum for f locally uniformly in w *if there exists a $\delta > 0$ such that the local solution mapping* (9) *satisfies $P_\delta(0,0) = \{\bar{x}\}$ and is single-valued on some neighborhood $W \times V \subseteq \mathbb{R}^{d+n}$ of $(0,0)$ with the mappings $v \mapsto P_\delta(w,v)$ Lipschitz continuous on V with the same modulus $K > 0$ for all fixed $w \in W$.*

Remark. A related concept of tilt stability is studied in [14], but without the requirement that such stability persist for a neighborhood of w. The locally uniform version of tilt stability that we study here is thus a stronger condition than the one considered in [14].

Tilt stability for unparameterized prox-regular functions $f(x)$ was first studied in [18], where this property was characterized by the positive-definiteness of a different generalized second derivative than either of those used in the present paper. This concept has also been seen before in other contexts; in particular it is related to Robinson's strong regularity [20] applied to optimal solutions (see [1], [2], and [3] for more on strong regularity).

To study tilt stability for parameterized functions $f(x,w)$, it is useful to consider "localizations" T_w on $\mathbb{B}(\bar{x};\delta) \times V$ of the partial subgradient mappings $\partial_x f(x,w)$. Recall that the *localization of $x \mapsto \partial_x f(x,w)$ on $\mathbb{B}(\bar{x};\delta) \times V$* is the mapping $T_w : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ defined by

$$T_w(x) := \begin{cases} \partial_x f(x,w) \cap V & \text{for } x \in \mathbb{B}(\bar{x};\delta), \\ \emptyset & \text{otherwise.} \end{cases}$$

Proposition 3.1. *Consider a function $f : \mathbb{R}^{n+d} \to \mathbb{R} \cup \{\infty\}$ that is continuously prox-regular in x at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ with compatible parameterization in w at 0, and consider any ρ greater than the modulus r of prox-regularity. If the constraint qualification* (4) *holds and $\bar{x}$ gives an isolated minimum for the unperturbed problem, then there exists a $\delta > 0$ and a neighborhood $W \times V \subseteq \mathbb{R}^{d+n}$ of $(0,0)$ such that the localization T_w of $x \mapsto \partial_x f(x,w)$ on $\mathbb{B}(\bar{x};\delta) \times V$ is such that the mapping $F_w := (T_w + \rho I)^{-1}$ is single-valued, monotone, and Lipschitzian with modulus $1/(\rho - r)$ on $V + \rho\bar{x}$.*

Proof. We consider the auxiliary minimization problem

$$\min_{x\in\mathbb{B}(\bar{x};\delta)} \left\{ f(x,w) + \frac{\rho}{2}|x - \bar{x}|^2 - \langle v, x\rangle \right\} \tag{10}$$

and note that the assumption $P_\delta(0,0) = \{\bar{x}\}$ ensures that $\bar{x}$ is the only solution to (10) with $(w,v) = (0,0)$. It follows that we can apply [14, Proposition 3.5] to the function $(x,w) \mapsto f(x,w) + \rho/2\,|x - \bar{x}|^2$ to obtain a neighborhood $W \times V \subseteq \mathbb{R}^{d+n}$

of $(0,0)$ such that the minimum in (10) for $(w,v) \in W \times V$ must be achieved in the interior of $\mathbb{B}(\bar{x};\delta)$. Thus any point giving this minimum is also a stationary point.

For any $w \in W$, if T_w is the localization of $x \mapsto \partial_x f(x,w)$ on $\mathbb{B}(\bar{x};\delta) \times V$, we know (cf. [17, Theorem 3.2]) that the mapping $T_w + rI$ is monotone. We define the mapping $F_w := (T_w + \rho I)^{-1}$, let $\lambda = 1/(\rho - r)$, and note that according to [22, Theorem 12.12], the mapping

$$\big(I + \lambda(T_w + rI)\big)^{-1} = \big(\lambda F_w^{-1}\big)^{-1}$$

is monotone and nonexpansive. This implies that F_w itself is monotone and that any elements (v_1,x_1) and (v_2,x_2) in the graph of F_w satisfy $|x_1 - x_2| \le \lambda |v_1 - v_2|$. In particular, it follows from this that F_w has a single-valued image at any point in its domain. Since the image of F_w at $v + \rho\bar{x}$ is the set of the stationary points associated with the minimization problem (10), it follows from the above discussion that the domain of F_w includes the set $V + \rho\bar{x}$, so that F_w is actually Lipschitzian with modulus λ on this set. □

We can now prove a characterization of tilt stability in terms of the positive-definiteness of the strong partial outer graphical derivative. In places, this proof is essentially a parametric version of the argument supporting [18, Theorem 1.3], though special accommodations must be made for the entirely different type of generalized derivative used here (generalized "coderivatives" were used in [18]).

PROPOSITION 3.2. *Consider a function $f : \mathbb{R}^{n+d} \to \mathbb{R} \cup \{\infty\}$ that is continuously prox-regular in x at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ with compatible parameterization in w at 0. If the constraint qualification (4) holds, then the following are equivalent:*

(i) *The point $\bar{x}$ gives a tilt stable local minimum for f locally uniformly in w at 0.*

(ii) *The strong partial outer graphical derivative $\tilde{D}^2_{xx} f(\bar{x},0|0)$ is positive-definite in the sense of (5).*

(iii) *There exist constants $\delta > 0$ and $K > 0$, and a neighborhood $W \times V \subseteq \mathbb{R}^{d+n}$ of $(0,0)$ such that for every $w \in W$ and the stationary point mapping defined by*

$$S_\delta(w,v) := \{x \in \mathbb{B}(\bar{x};\delta) | v \in \partial_x f(x,w)\},$$

the mapping $v \mapsto S_\delta(v,w)$ is single-valued, monotone, and Lipschitzian on V with modulus K.

Moreover, under any of these equivalent conditions, the optimal solution mapping satisfies $P_\delta(w,v) = S_\delta(w,v)$ on $W \times V$.

Proof. (ii) ⇒ (i) In order to prove this implication, we consider the mapping $R : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ that is defined exactly as the strong partial outer graphical derivative $\tilde{D}^2_{xx} f(\bar{x},0|0)$, but where the final graphical outer limit is taken with respect only to sequences $x \to \bar{x}$, $w \to 0$, and $v \to 0$ for which the partial outer graphical derivatives $D^2_{xx} f(x,w|v)$ satisfy

$$D^2_{xx} f(x,w|v)(x') = \left\{ v' \;\middle|\; \begin{array}{l} \exists\, \epsilon > 0 \text{ and } v(t) \in \partial_x f\big(x(t),w\big) \text{ for } t \in [0,\epsilon] \text{ such} \\ \text{that } \Big[\big(x(t),v(t)\big) - (x,v)\Big]/t \to (x',v') \text{ as } t \downarrow 0 \end{array} \right\} \tag{11}$$

and $\operatorname{gph} D^2_{xx} f(x,w|v)$ is an n-dimensional subspace of $\mathbb{R}^{2n}$.

Clearly, under assumption (ii), the mapping R is also positive-definite since its image set $R(x')$ is always contained in the image set $\tilde{D}^2_{xx} f(\bar{x},0|0)(x')$.

According to [18], when $D^2_{xx}f(x,w|v)$ satisfies the two properties (11), it can be expressed in terms of symmetric matrices $A_{x,w,v} \in \mathbb{R}^{n\times n}$ and subspaces $M_{x,w,v} \subseteq \mathbb{R}^n$ as

$$D^2_{xx}f(x,w|v)(x') = \left\{ \begin{array}{ll} A_{x,w,v}(x') + \big(M_{x,w,v}\big)^{\perp} & \text{if } x' \in M_{x,w,v}, \\ \emptyset & \text{otherwise} \end{array} \right\}. \tag{12}$$

The bound

$$\langle A_{x,w,v}(x'), x'\rangle \geq -r|x'|^2 \qquad \text{for all } x' \in M_{x,w,v} \tag{13}$$

follows immediately from the continuous prox-regularity of f since (via [17, Theorem 3.2]) the mappings $x \mapsto \partial_x f(x,w) + rx$ are locally monotone. We use the expression (12) to show that the inequality $|A_{x,w,v}(x')| \geq \epsilon > 0$ holds for all unit vectors $x' \in N_{x,w,v}$. Suppose that this is not the case and that for some sequence $\epsilon \downarrow 0$, there are sequences $w_\epsilon \to 0$, $x_\epsilon \to \bar{x}$, and $v_\epsilon \to 0$, and a sequence of unit vectors $x'_\epsilon \in M_{x_\epsilon,w_\epsilon,v_\epsilon}$ with $|A_{x_\epsilon,w_\epsilon,v_\epsilon}(x'_\epsilon)| < \epsilon$. Taking subsequences of $\{x'_\epsilon\}$ and $\{v'_\epsilon\}$ if necessary, we conclude that $v'_\epsilon \to 0$ and that there exists a unit vector x' such that $x'_\epsilon \to x'$. By the definition above of the mapping R, this implies that $0 \in R(x')$ which contradicts the positive-definiteness assumption. It follows that there exist scalars $\delta > 0$ and $\epsilon > 0$ such that for any triple $(x,w,v) \in \mathbb{B}((\bar{x},0,0);\delta)$ for which $D^2_{xx}f(x,w|v)$ satisfies the two conditions (11), the following holds:

$$|A_{x,w,v}(x')| \geq \epsilon|x'| \qquad \text{for all } x' \in M_{x,w,v}. \tag{14}$$

It follows from (14) that all the eigenvalues of the $A_{x,w,v}$ have absolute value greater than ϵ, and we claim that these eigenvalues are actually positive.

To prove this claim, we suppose that for some sequence $\epsilon \downarrow 0$, there are sequences $w_\epsilon \to 0$, $x_\epsilon \to \bar{x}$, and $v_\epsilon \to 0$, and a sequence of nonpositive eigenvalues λ_ϵ with their corresponding sequence of unit eigenvectors $x'_\epsilon \in M_{x_\epsilon,w_\epsilon,v_\epsilon}$. There are two cases to consider. In one case, the λ_ϵ are bounded in absolute value, so a subsequence must converge to some λ. Since the eigenvectors are all unit length, a subsubsequence of them converges to some unit vector x'. It follows that $\lambda x' \in R(x')$, which by the positive-definiteness assumption implies that λ is positive. This contradicts the assumption that the eigenvalues λ_ϵ are nonpositive. The only alternative is that the λ_ϵ diverge to $-\infty$ so that the inner product $\lambda_\epsilon|x'_\epsilon|^2 = \langle A_{x_\epsilon,w_\epsilon,v_\epsilon}(x'_\epsilon), x'_\epsilon\rangle$ also diverges to $-\infty$. But this contradicts the bound (13), so neither can this case occur. It follows that the eigenvalues of the $A_{x,w,v}$ are all positive, which means that the bound (14) translates into the fact that for any $(x,w,v) \in \mathbb{B}((\bar{x},0,0);\delta)$, the graphs of the partial outer graphical derivatives $D^2_{xx}f(x,w|v)$ satisfying (11) contain only pairs (x',v') which satisfy $\langle x',v'\rangle \geq \epsilon|x'|^2$, which implies that the mappings $D^2_{xx}f(x,w|v) - \epsilon I$ are monotone.

We wish to apply Proposition 3.1 in this situation, so we must first verify that $\bar{x}$ gives an isolated local minimum for the unperturbed problem. To prove this, we construct the function

$$\tilde{f}(x) := \left\{ \begin{array}{ll} f(x+\bar{x},0) - f(\bar{x},0) & \text{for } x \in \mathbb{B}(0;\delta), \\ \infty & \text{otherwise.} \end{array} \right.$$

It follows from the prox-regularity of $x \mapsto f(x,0)$ at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ that $\tilde{f}$ is prox-regular at 0 for $0 \in \partial\tilde{f}(0)$. From the monotonicity of $D^2_{xx}f(x,w|v) - \epsilon I$ obtained above, it follows from [17, Proposition 5.7] that there exists some $\tilde{\delta} > 0$ such that the

localization of $\partial\tilde{f}$ on $\mathbb{B}((0,0);\tilde{\delta})$ is strongly monotone with modulus ϵ. According to [17, Proposition 5.5] then, there are positive constants λ and c for which the function $e_\lambda - c|\cdot|^2$ is convex near 0, where e_λ represents the Moreau envelope

$$e_\lambda(x) := \min_{x'} \left\{ \tilde{f}(x') + \frac{1}{2\lambda}|x' - x|^2 \right\}.$$

Since $\tilde{f}(0) = 0$ and $\tilde{f}$ is prox-regular at 0 for $0 \in \partial\tilde{f}(0)$, we know that the Moreau envelope satisfies $e_\lambda(0) = 0$, so that the strong convexity above translates into the inequality $e_\lambda(x) \geq c\,|x|^2$ for x near 0. By the definition of e_λ, this implies that $\tilde{f}(x) \geq c\,|x|^2$ for x near 0, which translates into the bound

$$f(x,0) \geq c\,|x - \bar{x}|^2 + f(\bar{x},0) \text{ for } x \text{ near } \bar{x}.$$

This final bound ensures that we can shrink δ if necessary to obtain $P_\delta(0,0) = \{\bar{x}\}$.

We can now apply Proposition 3.1 for any $\rho > r$ to obtain a new $\delta > 0$ and a neighborhood $W \times V \in \mathbb{R}^{d+n}$ of $(0,0)$ such that for every $w \in W$ the localization T_w of $x \mapsto \partial_x f(x,w)$ on $\mathbb{B}(\bar{x};\delta) \times V$ has the property that the mapping $F_w := \big(T_w + \rho I\big)^{-1}$ is single-valued, monotone, and Lipschitz continuous on $V + \rho\bar{x}$ (we adjust $\delta > 0$ if necessary to be the smallest of the δ's encountered so far, and we shrink the neighborhoods W and V to include the balls of radius $\delta > 0$ about 0). By its definition, F_w is differentiable at $v + \rho\bar{x}$ exactly when the graph of $D^2_{xx} f(F_w(v+\rho\bar{x}), w|v)$ is an n-dimensional subspace. Since the outer graphical derivative of the inverse is the inverse of the outer graphical derivative (see [22]), and since the outer graphical derivative of F_w at $v + \rho\bar{x}$ is the same as the Jacobian $\nabla F_w(v + \rho\bar{x})$ when F_w is differentiable (see [22]), it follows that for any $w \in W$ and any $v' \in \mathbb{R}^n$ the following inclusion holds:

$$v' - \rho\nabla F_w(v + \rho\bar{x})v' \in D^2_x f(F_w(v + \rho\bar{x}), w|v)(\nabla F_w(v + \rho\bar{x})v').$$

By the definition of F_w, the element $F_w(v + \rho\bar{x})$ is contained in the set $\mathbb{B}(\bar{x};\delta)$ for any $v \in V$, so the monotonicity of the mappings $D^2_{xx} f(x, w|v) - \epsilon I$ proved above then implies that the inequality

$$\langle v', \nabla F_w(v + \rho\bar{x})v' \rangle \geq (\epsilon + \rho)|\nabla F_w(v + \rho\bar{x})v'|^2$$

holds, which via [17, Lemma 5.6] implies that $F_w^{-1} - (\epsilon + \rho)I = T_w - \epsilon I$ is monotone.

Applying [14, Proposition 3.5] (and shrinking W and V if necessary), we have that S_δ is nonempty on $W \times V$, and since for any $w \in W$ the image of the mapping $v \mapsto S_\delta(w,v)$ is the inverse of the mapping $x \in \mathbb{B}(\bar{x};\delta) \mapsto \partial_x f(x,w)$, the monotonicity of $T_w - \epsilon I$ ensures that the mapping $v \mapsto S_\delta(w,v)$ is single-valued, monotone, and Lipschitzian on V (with modulus $K = 1/\epsilon$). Finally, [14, Proposition 3.5] ensures that these same properties hold for the optimal solution mapping P_δ.

(iii) $\Rightarrow$ (ii) We again consider the mapping $R : \mathbb{R}^n \rightrightarrows \mathbb{R}^n$ whose graph is the outer limit as $x \to \bar{x}$, $w \to 0$, and $v \to 0$ of the $D^2_{xx} f(x, w|v)$ satisfying (11). Since the mappings $v \mapsto S_\delta(w,v)$ are monotone on V, it follows that any pair (x', v') in the graph of the partial outer graphical derivative $D^2_{xx} f(x, w|v)$ also satisfies $\langle x', v' \rangle \geq 0$, and thus so does every pair (x', v') in the graph of R. Moreover, the only point $x' \in \mathbb{R}^n$ with $0 \in R(x')$ is $x' = 0$. This follows since $0 \in R(x')$ if and only if there exist sequences $\tau \downarrow 0$, $x'_n \to x'$, $v'_n \to 0$, $v_n \to 0$, $w_n \to 0$, and $x_n \to \bar{x}$ satisfying $x_n \in S_\delta(w_n, v_n)$ and $x_n + \tau x'_n \in S_\delta(w_n, v_n + \tau v'_n)$. Since for every $w \in W$, the mapping $v \mapsto S_\delta(w,v)$ is Lipschitzian on V with modulus $K > 0$, it follows that $|x'_n| \leq K|v'_n|$, and since $v'_n \to 0$, x'_n must converge to 0 also.

We can now apply an argument similar to that in (ii) $\Rightarrow$ (i) to show that there exist scalars $\delta > 0$ and $\epsilon > 0$ together with neighborhoods W of 0 and V of 0, such that for any triple $(x, w, v) \in \mathbb{B}((\bar{x}, 0, 0); \delta)$ for which $D^2_{xx} f(x, w|v)$ satisfies (11), the following holds:

$$|A_{x,w,v}(x')| \geq \epsilon |x'| \qquad \text{for all } x' \in M_{x,w,v}. \tag{15}$$

It follows from (15) that all the eigenvalues of the $A_{x,w,v}$ have absolute value greater than ϵ. However, any eigenvalue $\lambda_{x,w,v}$ of $A_{x,w,v}$ satisfies $\lambda_{x,w,v}|x'|^2 = \langle A_{x,w,v}(x'), x' \rangle$ where x' is the corresponding eigenvector, and we already showed above that the inner product on the right was nonnegative. It follows that these eigenvalues are all positive and greater than ϵ, so that the graphs of the $D^2_{xx} f(x, w|v)$ satisfying (11) contain only pairs (x', v') which satisfy $\langle x', v' \rangle \geq \epsilon |x'|^2$. From this last inequality, the positive-definiteness of the mapping R follows immediately.

Having established the positive-definiteness of the mapping R, we apply Proposition 3.1 (as in the proof of (ii) $\Rightarrow$ (i)) to obtain $\delta > 0$, $\rho > r$, and a neighborhood $W \times V \subseteq \mathbb{R}^{d+n}$ of $(0, 0)$ such that for every $w \in W$, the localization T_w of $x \mapsto \partial_x f(x, w)$ on $\mathbb{B}(\bar{x}; \delta) \times V$ has the property that the mapping $F_w := \big(T_w + \rho I\big)^{-1}$ is single-valued, monotone, and Lipschitzian on $V + \rho \bar{x}$, and moreover $F_w^{-1} - (\epsilon + \rho) I = T_w - \epsilon I$ is monotone for some $\epsilon > 0$. This means that any two triples (x, w, v) and (x', u, v') close enough to $(\bar{x}, 0, 0)$ and with $v \in \partial_x f(x, w)$ and $v' \in \partial_x f(x', w)$ satisfy

$$\langle x - x', v - v' \rangle \geq \epsilon |x - x'|^2.$$

This last inequality implies that the partial outer graphical derivatives $D^2_{xx} f(x, w|v)$ satisfy

$$v' \in D^2_{xx} f(x, w|v)(x') \Rightarrow \langle v', x' \rangle \geq \epsilon |x'|^2$$

which implies that the strong partial outer graphical derivative $\tilde{D}^2_{xx} f(\bar{x}, 0|0)$ is positive-definite.

(i) $\Rightarrow$ (iii) For $w \in W$ we define the function

$$g_w(v) := \max_{x \in \mathbb{B}(\bar{x}; \delta)} \{\langle x, v \rangle - f(x, w)\}$$

which is the convex conjugate of the function $x \mapsto f(x, w) + \delta_{\mathbb{B}(\bar{x}; \delta)}$. Thus defined, $g_w : \mathbb{R}^n \to \mathbb{R}$ is proper, lower semicontinuous, convex, and finite-valued. We also define the mapping

$$G(w, v) := \operatorname*{argmax}_{x \in \mathbb{B}(\bar{x}; \delta)} \{\langle x, v \rangle - f(x, w)\}$$

which is the same as $P_\delta(w, v)$ so that under our assumptions, G is single-valued on $W \times V$. It is easy to show that $G(w, \cdot)$ is monotone on V, and since $G(w, \cdot)$ is single-valued and Lipschitz on V (thus maximal monotone on V), it is easy to verify that $G(w, v) = \partial g_w(v)$ for any $v \in V$. If we let T_w be the localization of $x \mapsto \partial_x f(x, w)$ on $\mathbb{B}(\bar{x}; \delta) \times V$, it follows from [14, Proposition 3.5] that the following inclusion also holds:

$$\partial g_w(v) = G(w, v) = P_\delta(w, v) \subseteq S_\delta(w, v) = \big(T_w\big)^{-1}(v) \tag{16}$$

for all $v \in V$.

Now we define h_w to be the convex conjugate of g_w so that

$$h_w(x) = \sup_{v\in\mathbb{R}^n}\{\langle x, v\rangle - g_w(v)\}.$$

It follows that h_w is a proper, lower semicontinuous, convex function whose maximal monotone subgradient mapping satisfies $\partial h_w = \big(\partial g_w\big)^{-1}$. From the inclusion (16), the subgradient mapping ∂h_w must also satisfy

$$\Big(\operatorname{gph}\partial h_w \cap \big(\mathbb{B}(\bar{x};\delta)\times V\big)\Big) \subseteq \operatorname{gph} T_w. \tag{17}$$

We claim that this inclusion is actually an identity. To show this claim, we notice that $T_w + \rho I$ is monotone for any ρ greater than the modulus r of prox-regularity. Since ∂h_w is maximal monotone, it follows that $\partial h_w + \rho I$ is also maximal monotone. If H is the localization of ∂h_w on $\mathbb{B}(\bar{x};\delta)\times V$, then the inclusion (17) implies that H satisfies $\operatorname{gph}\big(H+\rho I\big) \subseteq \operatorname{gph}\big(T_w+\rho I\big)$, which by the maximal monotonicity of $\partial h_w + \rho I$ translates into the identity $H + \rho I = T_w + \rho I$, which proves the claim. By its definition then, the stationary point mapping S_δ satisfies

$$S_\delta(w,v) = \big(\partial h_w\big)^{-1}(v) = \partial g_w(v) = G(w,v)$$

for all $v \in V$ and $w \in W$. Therefore, S_δ inherits the properties of $G = P_\delta$ on $W \times V$ which include monotonicity with respect to v.

The final claim about the equivalence between P_δ and S_δ on $W \times V$ clearly also follows from the above. □

Remark. In the special unparameterized case where $f(x,w) = f(x)$, the constraint qualification is trivially satisfied and the mapping P_δ depends only on the tilt parameter v. In this setting, our Proposition 3.2 provides a complementary condition (ii) in terms of outer graphical derivatives to the characterizations of tilt stability from [18, Theorem 1.3] (those authors studied the unparameterized case only and provided several other equivalent conditions based on a generalized coderivative).

Notice that according to our argument above, any restricted version of the strong partial outer graphical derivative can replace $\tilde{D}^2_{xx}f(\bar{x},0|0)$ in Proposition 3.2, as long as the final outer graphical limit used to construct it includes the sequences $x \to \bar{x}$, $w \to 0$, and $v \to 0$ for which $D^2_{xx}f(x,w|v)$ satisfies (11). Note also that such sequences are plentiful, because (cf. the proof of Proposition 3.2) there is a locally Lipschitzian mapping F_w whose inverse agrees with $x \mapsto \partial_x f(x,w) + \rho\, x$ near $(\bar{x}, \rho\bar{x})$ for some fixed $\rho > 0$. By its construction, the mapping F_w is differentiable precisely when $D^2_{xx}f(x,w|v)$ satisfies (11). According to Rademacher's theorem then, this occurs almost everywhere in the domain of F_w.

THEOREM 3.1. *Consider a function $f : \mathbb{R}^{n+d} \to \mathbb{R}\cup\{\infty\}$ that is continuously prox-regular at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ with compatible parameterization in w at 0. If the constraint qualification* (4) *holds then the following are equivalent:*

(i) *The point $\bar{x}$ gives both a tilt stable local minimum for f locally uniformly in w and a calm local minimum for f.*

(ii) *The strong partial outer graphical derivative $\tilde{D}^2_{xx}f(\bar{x},0|0)$ is positive-definite* (5) *and the partial outer graphical derivative $D^2_x f(\bar{x},0|0)$ satisfies* (6).

Proof. According to Proposition 3.2, tilt stability is equivalent to the positive-definiteness of $\tilde{D}^2_{xx}f(\bar{x},0|0)$. One consequence of tilt stability shown in Proposition 3.2 is that the optimal solution mapping P_δ agrees near $(0,0)$ with the stationary

point mapping S_δ. According to [11, Proposition 4.1], the (single-valued) stationary point mapping $S_\delta(w,v)$ satisfies

$$|S_\delta(w,v) - \bar{x}| \leq K|(w,v)| \tag{18}$$

for all pairs (w,v) near $(0,0)$ if and only if the outer graphical derivative of S_δ satisfies

$$DS_\delta(0,0|\bar{x})(0,0) = \{0\}. \tag{19}$$

However, by the definition of S_δ, the outer graphical derivative of S_δ also satisfies

$$DS_\delta(0,0|\bar{x})(w',v') = \left\{x' \,|\, v' \in D_x^2 f(\bar{x},0|0)(x',w')\right\},$$

so that the conditions (19) and (18) are equivalent to the kernel condition (6) on the partial outer graphical derivative $D_x^2 f(\bar{x},0|0)$. Thus, we see that condition (ii) above is equivalent to the paired conditions of tilt stability and joint calmness (18).

To finish the proof, we note that in the presence of tilt stability, the (generally stronger) joint calmness property (18) is equivalent to the calmness property (2). This follows since calmness in the presence of tilt stability implies the joint calmness property (18) via the following series of inequalities:

$$\begin{aligned}
|P_\delta(w,v) - P_\delta(0,0)|^2 &\leq |P_\delta(w,v) - P_\delta(w,0)|^2 + |P_\delta(w,0) - P_\delta(0,0)|^2 \\
&\leq K_1|v|^2 + K_2|w|^2 \\
&\leq \max\{K_1,K_2\}\left(|v|^2 + |w|^2\right) \\
&= \max\{K_1,K_2\}|(w,v)|^2. \qquad \square
\end{aligned}$$

Proof of Theorem 1.1. The first part of this result follows from the tilt stability guaranteed under (5) by Proposition 3.2, and the calmness under (6) follows from Theorem 3.1. $\square$

Remark. Notice that Theorem 3.1 exposes exactly the gap between the sufficient condition in Theorem 1.1 for calm local minima: Only points $\bar{x}$ which do not give a tilt stable local minimum for f locally uniformly in w have a chance of giving a calm local minimum for f while violating the sufficient condition.

An alternate sufficient condition for calmness can be given in terms of a single second-order condition on a different strong partial outer graphical derivative of f, $\tilde{D}_x^2 f(\bar{x},0|0) : \mathbb{R}^{n+d} \rightrightarrows \mathbb{R}^n$ which is outer graphical limit as $(x,w,v) \to (\bar{x},0,0)$ of the outer graphical derivatives at (x,w) for $v \in \partial_x f(x,w)$ of the mappings $(x,w) \mapsto \partial_x f(x,w)$. The drawback of this sufficient condition is that the gap between it and the calmness property is in general larger than the gap in Theorem 1.1.

THEOREM 3.2. *Consider a function $f : \mathbb{R}^{n+d} \to \mathbb{R} \cup \{\infty\}$ that is continuously prox-regular at $\bar{x}$ for $0 \in \partial_x f(\bar{x},0)$ with compatible parameterization in w at 0. If the constraint qualification* (4) *holds and the strong partial outer graphical derivative $\tilde{D}_x^2 f(\bar{x},0|0)$ satisfies*

$$v' \in \tilde{D}_x^2 f(\bar{x},0|0)(x',0) \Rightarrow \langle x', v'\rangle > 0 \text{ unless } x' = 0,$$

then $\bar{x}$ gives a calm local minimum for f.

Proof. This follows from Theorem 1.1 since by its definition, $\tilde{D}^2_x f(\bar{x}, 0|0)(x', 0)$ satisfies the following two inclusions:

$$\tilde{D}^2_{xx} f(\bar{x}, 0|0)(x') \subseteq \tilde{D}^2_x f(\bar{x}, 0|0)(x', 0) \text{ and}$$

$$D^2_x f(\bar{x}, 0|0)(x', 0) \subseteq \tilde{D}^2_x f(\bar{x}, 0|0)(x', 0).$$

The first inclusion follows since $v' \in \tilde{D}^2_{xx} f(\bar{x}, 0|0)(x')$ implies that there exist sequences $v'_n \to v'$, $x'_n \to x'$, $x \to \bar{x}$, $w \to 0$, and $v \to 0$ such that $v'_n \in D^2_{xx} f(x, w|v)(x'_n)$. This means that there are sequences $\tau \downarrow 0$, $v^\tau_n \to v'_n$, and $x^\tau_n \to x'_n$ such that $v + \tau v^\tau_n \in \partial_x f(x + \tau x^\tau_n, w)$. By defining a sequence $w^\tau = 0$, we see that $v'_n \in D^2_x f(x, w|v)(x'_n, 0)$, from which it follows that $v' \in \tilde{D}^2_x f(\bar{x}, 0|0)(x', 0)$.

The second inclusion follows because the constant sequence $(\bar{x}, 0, 0)$ is one of the candidate sequences for the final outer limit used to construct the strong partial outer graphical derivative $\tilde{D}^2_x f(\bar{x}, 0|0)(x', 0)$. ∎

Notice that just as in Theorem 1.1, the conditions in Theorem 3.2 are actually enough to ensure tilt stability as well as calmness.

4. Tilt stability in nonlinear programming. An important example of a continuously prox-regular function is the essential objective function associated with the parameterized nonlinear program (3)

$$f(x, w) = g_0(x, w) + \delta_{C(w)}(x), \tag{20}$$

where $\delta_{C(w)}$ represents the indicator function associated with the set $C(w)$:

$$\delta_{C(w)}(x) = \begin{cases} 0 & \text{if } x \in C(w), \\ \infty & \text{otherwise.} \end{cases}$$

To apply Theorem 1.1 in this situation, we need to compute the two generalized second-order derivatives given there. To do this, it will be useful to consider the Lagrangian

$$L(x, w, y) := g_0(x, w) + y_1\, g_1(x, w) + \cdots + y_m\, g_m(x, w),$$

defined in terms of Lagrange multiplier vectors $y \in \mathbb{R}^m$. A vector $y \in \mathbb{R}^s_+ \times \mathbb{R}^{m-s}$ (here $\mathbb{R}^s_+$ denotes the nonnegative orthant in $\mathbb{R}^s$) is a *Lagrange multiplier* for the pair of parameters (w, v) at the point $x \in C(w)$ if y satisfies $\nabla_x L(x, w, y) = v$, and either y_i or $g_i(x, w)$ is equal to 0 for $i = 1, \ldots, s$. The set of Lagrange multipliers can be defined compactly in terms of the mapping $G(x, w) := \big(g_1(x, w), \ldots, g_m(x, w)\big)$ and the normal cone

$$N_K(q) := \{y \in \mathbb{R}^s_+ \times \mathbb{R}^{m-s} : y_i = 0 \text{ or } q_i = 0 \text{ for } i = 1, \ldots, s\}$$

to $K = \mathbb{R}^s_- \times \{0\}^{m-s}$ at a point $q \in K$. In terms of these, the set of Lagrange multipliers $Y(x, w, v)$ for parameters (w, v) at the point $x \in C(w)$ is

$$Y(x, w, v) := \{y \in N_K(G(x, w)) : v = \nabla_x L(x, w, y)\},$$

and it is well known that under the Mangasarian–Fromovitz constraint qualification, this set is bounded for any pair of parameters (w, v) near 0 and any point x near $\bar{x}$.

We can also use the mapping G and the normal cone N_K to state one form of the *Mangasarian–Fromovitz constraint qualification at* $\bar{x} \in C(0)$:

$$y \in N_K(G(\bar{x},0)) \text{ and } \sum_1^m y_i \nabla_x g_i(\bar{x},0) = 0 \quad \Rightarrow \quad y = 0.$$

We now modify formulas for partial outer graphical derivatives from [12, Theorem 3.2] (these formulas were developed in [12] for partial protoderivatives (cf. [13]), but when they exist these are always the same as the partial outer graphical derivatives). According to [12], the image set $D^2_{xx} f(x,w|v)(x')$ is nonempty only if $Y(x,w,v)$ is nonempty and x' is in the critical cone $Q(x,w) \cap (v - \nabla_x g_0(x,w))^\perp$, where $Q(x,w)$ denotes the tangent cone

$$Q(x,w) := \left\{ x' \in \mathbb{R}^n : \begin{array}{l} \langle \nabla_x g_i(x,w), x' \rangle \le 0 \text{ for active } i = 1,\dots,s \\ \langle \nabla_x g_i(x,w), x' \rangle = 0 \text{ for } i = s+1,\dots,m \end{array} \right\}.$$

According to [12, Theorem 3.2], the value of the partial outer graphical derivative can be expressed in terms of the mapping $G(x,w) := \big(g_1(x,w),\dots,g_m(x,w)\big)$, the subset of Lagrange multipliers defined by

$$Y_{\max}(x,w,v,x') := \operatorname*{argmax}_{y \in Y(x,w,v)} \big\langle x', \nabla^2_{xx} L(x,w,y)\, x' \big\rangle,$$

and the polyhedral cone

$$Y'(x,w,x') := \big\{ y' \in N_K\big(G(x,w)\big) : y'_i \langle \nabla_x g_i(x,w), x' \rangle = 0 \big\}.$$

At any vector x' in the critical cone $Q(x,w) \cap (v - \nabla_x g_0(x,w))^\perp$, the partial outer graphical derivative is defined by

$$D^2_{xx} f(x,w|v)(x') := \left\{ v' = \nabla^2_{xx} L(x,w,y)\, x' + \sum_{i=1}^m y'_i \nabla_x g_i(x,w) - y'_0 \left(v - \nabla_x g_0(x,w)\right) \right\},$$

where this set is constructed from arbitrary choices of $y \in Y_{\max}(x,w,v,x')$, $y' \in Y'(x,w,x')$, and $y'_0 \in \mathbb{R}$. Since the elements $(\bar{x}', \bar{v}')$ of the graph of the strong partial outer graphical derivative are obtained as limits of pairs (x', v') from the graphs of $D^2_{xx} f(x,w|v)$, and since according to the preceding discussion the inner product of such pairs satisfies

$$\langle x', v' \rangle = \big\langle x', \nabla^2_{xx} L(x,w,y)\, x' \big\rangle$$

for some Lagrange multiplier $y \in Y_{\max}(x,w,v,x')$, we can translate the positive-definiteness condition in Theorem 3.1 to obtain the following characterization of tilt stability.

THEOREM 4.1. *For the essential objective* f (20) *associated with the parameterized nonlinear program* (3)*, if the Mangasarian–Fromovitz constraint qualification holds at* $\bar{x} \in C(0)$ *and the set* $Y(\bar{x},0,0)$ *of Lagrange multipliers is nonempty, then the following are equivalent:*

(i) $\bar{x}$ *gives a tilt stable local minimum for* f *locally uniformly in* w.

(SOC1) *For any sequences* $x \to \bar{x}$, $w \to 0$, *and* $v \to 0$ *satisfying* $x \in C(w)$, *and any convergent sequence of points* $x' \in Q(x,w) \cap (v - \nabla_x g_0(x,w))^\perp$ *with nonzero*

limit $\bar{x}'$, together with any corresponding convergent sequence of multipliers $y \in Y_{\max}(x, w, v, x')$ with limit $\bar{y}$, the Hessian satisfies $\langle \bar{x}', \nabla^2_{xx} L(\bar{x}, 0, \bar{y})\, \bar{x}' \rangle > 0$.

Proof. According to our discussion prior to this theorem, the only thing to show is that a sequence of Lagrange multipliers $y \in Y_{\max}(x, w, v, x')$ for sequences $x \to \bar{x}$, $w \to 0$, and $v \to 0$ has at least a subsequence that converges to a vector $\bar{y}$ in $Y(\bar{x}, 0, 0)$. This follows from the upper semicontinuity of the Lagrange multiplier mapping proved in [21, Theorem 2.3]. □

The second-order condition (SOC1) in Theorem 4.1 is unprecedented, and it relates in the following way to a simpler second-order condition.

LEMMA 4.1. *The second-order condition* (SOC1) *in Theorem* 4.1 *implies the second-order condition*

$$\max_{y \in Y(\bar{x},0,0)} \left\langle \bar{x}', \nabla^2_{xx} L(\bar{x}, 0, y)\, \bar{x}' \right\rangle > 0 \text{ for all nonzero } \bar{x}' \in Q(\bar{x}, 0) \cap \nabla_x g_0(\bar{x}, 0)^{\perp}. \tag{21}$$

Proof. The second-order condition here is just the special case of (SOC1) where the sequences $\{x\}$, $\{w\}$, and $\{v\}$ are taken to be the constant sequences $\bar{x}$, 0, and 0, respectively. □

The following example shows that the implication in Lemma 4.1 cannot be reversed.

Example A. Consider the minimization problem

$$\min\{g_0(x_1, x_2) := (x_1)^2 - (x_2)^2\} \text{ over all } x \in C,$$

where the constraint set is defined as follows:

$$C := \{x \in \mathbb{R}^2 : g_1(x_1, x_2) := -x_1 + 2\,x_2 \le 0 \text{ and } g_2(x_1, x_2) := -x_1 - 2\,x_2 \le 0\}.$$

At the point $\bar{x} = (0, 0)$, both constraints are active and their gradients $\nabla g_1(0, 0) = (-1, 2)$ and $\nabla g_2(0, 0) = (-1, -2)$ are linearly independent (so the Mangasarian–Fromovitz constraint qualification is certainly satisfied). Moreover, the set of Lagrange multipliers $Y(\bar{x}, 0, 0)$ consists of the singleton $y = (0, 0)$, the gradient $\nabla_x g_0(0, 0) = (0, 0)$, and the tangent cone $Q(\bar{x}, 0)$ is the same as the feasible region $\{x' \in \mathbb{R}^2 : x_1 \ge 0,\ x_2 \in [-x_1/2, x_1/2]\}$. It follows that the second-order condition (21) is satisfied. However, if we consider the tilted minimization problem for the vector $v = (\epsilon, 0)$ for any $\epsilon > 0$, there are two optimal solutions $(2\epsilon/3, \epsilon/3)$ and $(2\epsilon/3, -\epsilon/3)$, so $\bar{x} = (0, 0)$ certainly cannot give a tilt stable minimum (it does not even give a unique local minimum for these tilt parameters). It follows from Theorem 4.1 that the second-order condition (SOC1) is violated in this case.

According to the remark following Proposition 3.2, we can further refine our characterization of tilt stability by focusing our attention on sequences (x, w, v) converging to $(\bar{x}, 0, 0)$ for which the graph of the $D^2_{xx} f(x, w|v)$ is an n-dimensional subspace. From the formula for the partial outer graphical derivative, it follows that if its graph is an n-dimensional subspace then its domain is a subspace too. Moreover, its domain $Q(x, w) \cap (v - \nabla_x g_0(x, w))^{\perp}$ can be rewritten in terms of any Lagrange multiplier $y \in Y(x, w, v)$ as the set

$$D(x, w, y) := \left\{ x' \in \mathbb{R}^n : \begin{array}{l} \langle \nabla_x g_i(x, w), x' \rangle \le 0 \text{ for active } i = 1, \ldots, s \\ \langle \nabla_x g_i(x, w), x' \rangle = 0 \text{ for } i = s+1, \ldots, m \\ \sum_{i=1}^m y_i \langle \nabla_x g_i(x, w), x' \rangle = 0 \end{array} \right\},$$

which, since any multiplier $y \in Y(x,w,v)$ is an element of $N_K(G(x,w))$, can be further refined to

$$D(x,w,y) = \left\{ \bigcap_{i \in I_1(x,w,y)} \nabla_x g_i(x,w)^\perp \right\} \cap \{x' : \langle \nabla_x g_i(x,w), x' \rangle \leq 0 \text{ for } i \in I_0(x,w,y)\} \tag{22}$$

in terms of the index sets

$$I_1(x,w,y) := \{i = 1, \ldots, s : g_i(x,w) = 0 \text{ and } y_i > 0\} \cup \{i = s+1, \ldots, m\},$$

and

$$I_0(x,w,y) := \{i = 1, \ldots, s : g_i(x,w) = 0 \text{ and } y_i = 0\}$$

(where the first intersection in the formula (22) is interpreted to be all of $\mathbb{R}^n$ if the set of indices $I_1(x,w,y)$ is empty). From the expression (22), it is clear that the domain $D(x,w,y)$ of $D^2_{xx} f(x,w|v)$ is a subspace if and only if it satisfies

$$D(x,w,y) = \left\{ \bigcap_{i \in I_1(x,w,y) \cup I_0(x,w,y)} \nabla_x g_i(x,w)^\perp \right\} = \left\{ \bigcap_{i \in [1,m] \text{ with } g_i(x,w)=0} \nabla_x g_i(x,w)^\perp \right\}. \tag{23}$$

This observation leads to a different characterization of tilt stability in parameterized nonlinear programming.

THEOREM 4.2. *For the essential objective f* (20) *associated with the parameterized nonlinear program* (3)*, if the Mangasarian–Fromovitz constraint qualification holds at $\bar{x} \in C(0)$ and the set $Y(\bar{x},0,0)$ of Lagrange multipliers is nonempty, then the following are equivalent:*

(i) *$\bar{x}$ gives a tilt stable local minimum for f locally uniformly in w.*

(SOC2) *For any sequences $x \to \bar{x}$, $w \to 0$, and $v \to 0$ satisfying $x \in C(w)$, and any convergent sequence of points $x' \in \bigcap_{i \in [1,m] \text{ with } g_i(x,w)=0} \nabla_x g_i(x,w)^\perp$ with nonzero limit $\bar{x}'$, together with any corresponding convergent sequence of multipliers $y \in Y_{\max}(x,w,v,x')$, the Hessian at the limit multiplier $\bar{y}$ satisfies $\langle \bar{x}', \nabla^2_{xx} L(\bar{x},0,\bar{y})\, \bar{x}' \rangle > 0$.*

Proof. This follows from the remark following Proposition 3.2, since according to the discussion above, the class of partial outer graphical derivatives considered here includes all of those whose graphs are n-dimensional subspaces of $\mathbb{R}^{2n}$. □

The second-order condition (SOC2) in Theorem 4.2 is also unprecedented, but relates in the following way to a standard strong second-order condition.

LEMMA 4.2. *The (equivalent) second-order conditions* (SOC2) *and* (SOC1) *are implied by the* general strong second-order sufficient condition [4]*: For every Lagrange multiplier $y \in Y(\bar{x},0,0)$, the Hessian $\nabla^2_{xx} L(\bar{x},0,y)$ is positive-definite on the set $\bigcap_{i \in I_1(\bar{x},0,\bar{y})} \nabla_x g_i(\bar{x},0)^\perp$.*

Proof. This follows since any vector $\bar{y}$ obtained as the limit of multipliers $y \in Y(x,w,v)$ is an element of $Y(\bar{x},0,0)$, and its positive components must come from positive components of the sequence of multipliers (so $I_1(x,w,y) \supseteq I_1(\bar{x},0,\bar{y})$). Thus, any $\bar{x}'$ obtained as a limit of x' perpendicular to the gradients $\nabla_x g_i(x,w)$ for indices $i \in I_1(x,w,y)$, must have $\bar{x}'$ perpendicular to $\nabla_x g_i(\bar{x},0)$ for any index $i \in I_1(\bar{x},0,\bar{y})$. □

The following example shows that the implication in Lemma 4.2 cannot be reversed, so that the second-order conditions (SOC1) and (SOC2) are weaker than the general strong second-order sufficient condition.

Example B. Consider the minimization problem

$$\min\{g_0(x_1, x_2) := (x_1)^2 - x_1 + (x_2)^2\} \text{ over all } x \in C,$$

where the constraint set is defined as

$$C := \{x \in \mathbb{R}^2 : g_1(x_1, x_2) := x_1 - x_2^2 \leq 0 \text{ and } g_2(x_1, x_2) := x_1 \leq 0\}$$

(note that the first constraint function is superfluous). The objective tilted by $\langle x, (v_1, v_2)\rangle$ is

$$x_1^2 - (1 + v_1)\, x_1 + x_2^2 - v_2\, x_2$$

which has a unique (global) constrained minimum at the point $(0, v_2/2)$ as long as v_1 is sufficiently small. It follows that $\bar{x} = (0, 0)$ gives a tilt stable local minimum. Moreover, at $\bar{x}$ both constraints are active and their gradients are both equal to $(1, 0)$ so the Mangasarian–Fromovitz constraint qualification is satisfied. Theorems 4.1 and 4.2 then ensure that the second-order conditions (SOC1) and (SOC2) hold in this case.

However, the set of Lagrange multipliers $Y(\bar{x}, 0, 0)$ consists of the $y \in \mathbb{R}^2_+$ satisfying $y_1 + y_2 = 1$ and the Hessian of the Lagrangian is the matrix

$$\begin{bmatrix} 2 & 0 \\ 0 & 2(1 - y_1) \end{bmatrix}.$$

For the choice of multiplier $y = (1, 0)$, this matrix is only positive semidefinite on $\{0\} \times \mathbb{R}$ (which is the perpendicular subspace to both active constraint gradients). Therefore, the general strong second-order sufficient condition is not satisfied in this case.

Thus we see through Lemmas 4.1 and 4.2 that the (equivalent) second-order conditions (SOC1) and (SOC2) characterizing tilt stability are sandwiched properly between the second-order condition (21) and the general strong second-order sufficient condition. It would be nice if the conditions (SOC1) and (SOC2) could be refined in such a way as to depend only on information at the base point like more traditional second-order conditions and not on the limits of nearby parameter values. In particular, such a refinement would likely make (SOC1) and (SOC2) less daunting to verify in many situations. However, such refinements are not generally possible, as can be understood by considering two different parameterizations of an unperturbed problem that exhibits tilt stability: One parameterization is trivial, with each perturbed problem being the same as the unperturbed model, and the other parameterization is anything that creates a lack of tilt stability.

For instance, consider the objective function $g_0(x_1, x_2) := x_1^2/2$ with parameterized constraint set

$$C(w) := \{(x_1, x_2) \in \mathbb{R}^2 : g_1(x, w) := x_2 - w^{2k} \leq 0 \text{ and } g_2(x, w) := -x_2 - w^{2k} \leq 0\}$$

for any positive integer k. For the unperturbed problem (with $w = 0$), the constraint set reduces to $C(0) = \{(x_1, x_2) \in \mathbb{R}^2 : x_2 = 0\}$, and the tilted minimization has a unique solution at $x(v_1, v_2) = (v_1, 0)$. It follows that the point $\bar{x} = (0, 0)$ gives a

tilt stable local minimum for the trivially parameterized essential objective function $g_0 + \delta_{C(0)}$ locally uniformly in w. Notice, moreover, that the general strong second-order condition is satisfied in this case. However, the tilted minimization of g_0 over $C(w)$ for $w \neq 0$ has multiple solutions when $v_2 = 0$, so tilt stability does not hold. In this case, only the derivatives of order $2k$ with respect to w of g_1 and g_2 evaluated at the base point $(\bar{x}, 0)$ distinguish the two different parameterizations. Since k is any positive integer, no test using only derivatives evaluated at the base point will be able to distinguish the stability discrepancies illustrated in this example.

Our study of tilt stability here complements a long line of results concerning the Lipschitz stability of solutions to nonlinear programs (as opposed to calmness where a base point is fixed). The previous work has focused primarily on the more general parameters (denoted in this paper by w), but of course tilt stability can be viewed as a special case of such a parameterization. Robinson showed in [20] that the general strong second-order sufficient condition can be combined with the linear independence of the gradients of all the binding constraints to give the local Lipschitz continuity with respect to w of the optimal solutions to the nonlinear program (3). This result was complemented by Liu [15] and Ralph and Dempe [19] where a constant rank condition was paired with the Mangasarian–Fromovitz constraint to replace the linear independence condition in Robinson's result. Moreover, this sufficient condition for Lipschitz stable optimal solutions was completed in [3, Corollary 3.5] where under the constant rank condition, Lipschitz stability of optimal solutions was shown to be equivalent to the Mangasarian–Fromovitz constraint qualification together with the general strong second-order sufficient condition. Robinson [21] also gave an example to show that Mangasarian–Fromovitz and the general strong second-order sufficient condition (without the constant rank condition) are not enough to give Lipschitz stability.

5. Calm local minima in nonlinear programming. According to [12] and [11, Theorem 5.1],[1] the partial outer graphical derivative $D_x^2 f(\bar{x}, 0|0)$ satisfies the kernel condition in Theorem 1.1 if and only if

(KER) $\bar{x}' = 0$ is the only point in $Q(\bar{x}, 0) \cap (-\nabla_x g_0(\bar{x}, 0))^\perp$ satisfying

$$\nabla^2_{xx} L(\bar{x}, 0, \bar{y}) \cdot \bar{x}' + \nabla_x L(\bar{x}, 0, \bar{y}') = \bar{y}_0 \nabla_x g_0(\bar{x}, 0)$$

for some $\bar{y}' \in Y'(\bar{x}, 0, \bar{x}')$, some $\bar{y}_0 \in \mathbb{R}$, and some $\bar{y}$ achieving the maximum of the inner product $\langle \bar{x}', \nabla^2_{xx} L(\bar{x}, 0, y)\, \bar{x}' \rangle$ taken over all multipliers $y \in Y(\bar{x}, 0, 0)$ having the same gradient $\nabla_w L(\bar{x}, 0, \bar{y})$ with respect to the parameter w.

This kernel condition follows from the *general second-order sufficient condition* [4]: for every Lagrange multiplier $y \in Y(\bar{x}, 0, 0)$, the Hessian $\nabla^2_{xx} L(\bar{x}, 0, y)$ is positive-definite on the set $D(\bar{x}, 0, y)$ (22). Clearly the general second-order sufficient condition is weaker than the general strong second-order sufficient condition and stronger than the second-order condition (21). Now we show that it is stronger than our kernel condition (KER): Consider any $\bar{x}'$ in $Q(\bar{x}, 0) \cap (-\nabla_x g_0(\bar{x}, 0))^\perp$ that satisfies

$$\nabla^2_{xx} L(\bar{x}, 0, \bar{y}) \cdot \bar{x}' + \nabla_x L(\bar{x}, 0, \bar{y}') = \bar{y}_0 \nabla_x g_0(\bar{x}, 0) \tag{24}$$

for some $\bar{y}' \in Y'(\bar{x}, 0, \bar{x}')$, some $\bar{y}_0 \in \mathbb{R}$, and some $\bar{y}$ achieving the maximum of the inner product $\langle \bar{x}', \nabla^2_{xx} L(\bar{x}, 0, y)\, \bar{x}' \rangle$ taken over all multipliers $y \in Y(\bar{x}, 0, 0)$ having

[1] See [10] for a corrected version of [11, Theorem 5.1]

the same gradient $\nabla_w L(\bar{x}, 0, \bar{y})$. After multiplying (24) by the vector $\bar{x}'$, we conclude that $\langle \bar{x}', \nabla^2_{xx} L(\bar{x}, 0, \bar{y})\, \bar{x}' \rangle = 0$. It follows that our kernel condition (KER) is no stronger than the general second-order sufficient condition, but it is actually weaker since it characterizes a calmness property weaker (cf. [11, Theorem 5.1]) than the one Robinson established in [21] using the general second-order sufficient condition.

It is interesting to note that the general second-order sufficient condition is neither weaker nor stronger in general than our equivalent second-order conditions (SOC1) and (SOC2), as can be seen from Examples A and B in the previous section. The general second-order sufficient condition holds in Example A where both (SOC1) and (SOC2) fail, while the opposite is true in Example B.

The results of [12] and [11] thus lead to the following direct corollary to Theorems 3.1, 4.1, and 4.2.

COROLLARY 5.1. *Consider the essential objective f* (20) *associated with the parameterized nonlinear program* (3)*, and consider a point $\bar{x}$ for which there exists a Lagrange multiplier. If the Mangasarian–Fromovitz constraint qualification holds, then the following are equivalent:*

(i) *The point $\bar{x}$ gives both a tilt stable local minimum for f locally uniformly in w and a calm local minimum for f.*

(ii) *One of the (equivalent) second-order conditions* (SOC1) *and* (SOC2) *holds, and the kernel condition (KER) holds.*

Remark. Some results involving stability in nonlinear programming which are particularly closely related to Corollary 5.1 include those in [21] where Robinson used the general second-order sufficient condition together with the Mangasarian–Fromovitz constraint qualification to prove that there exist locally optimal solutions to the perturbed optimization problems (3) for any parameter w near 0, and that any single-valued selection of such local solutions satisfied the bound (2). Shapiro [23] proved a similar result but giving only the bound (2) (and not necessarily the existence) under a different second-order condition. Robinson's result in [21] is complemented by a result of Kojima [9] which states that if the general strong second-order sufficient condition holds in tandem with the Mangasarian–Fromovitz constraint qualification, then there exist neighborhoods $X \subseteq \mathbb{R}^n$ of $\bar{x}$ and $W \subseteq \mathbb{R}^d$ of 0 for which there are unique optimal solutions to (3) when the minimization is restricted to X and that these solutions behave continuously with respect to the parameter w on W. Robinson's and Kojima's results together yield the calmness of the local minimum for f as in our Corollary 5.1, but only under the general strong second-order sufficient condition which we have shown to be stronger than any of our conditions (SOC1), (SOC2), or (KER). In this way, our Corollary 5.1 is an improvement on the combination of Robinson's and Kojima's results. Moreover, our Corollary 5.1 actually characterizes the calmness of the local minimum for f when it is combined with tilt stability, so the gap between condition (ii) of Corollary 5.1 as a sufficient condition for calmness of local minima is precisely identified here.

Acknowledgments. I thank René Poliquin for his comments on this paper, particularly with regard to the proof of Proposition 3.2, and I thank the referees whose comments were very helpful.

REFERENCES

[1] J.F. BONNANS AND A. SHAPIRO, *Optimization problems with perturbations, a guided tour*, SIAM Rev., 40 (1998), pp. 228–264.

[2] 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.
[3] A.L. 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, Dekker, New York, 1998, pp. 65–82.
[4] A.V. Fiacco and J. Kyparisis, *Sensitivity analysis in nonlinear programming under second order assumptions*, in Systems and Optimization (Enschede, 1984), Lecture Notes in Control and Inform. Sci. 66, Springer, New York, 1985, pp. 74–97.
[5] H. Gfrerer, *Hölder continuity of solutions of perturbed optimization problems under Mangasarian–Fromovitz constraint qualification*, in Parametric Optimization and Related Topics, J. Guddat et al., eds., Akademie-Verlag, Berlin, 1987, pp. 113–125.
[6] D. Klatte, *Nonlinear optimization problems under data perturbations*, in Modern Methods of Optimization, W. Krabs and J. Zowe, eds., Lecture Notes in Econom. and Math. Systems 378, Springer, Berlin, 1992, pp. 204–235.
[7] D. Klatte, *On quantitative stability for non-isolated minima*, Control Cybernet., 23 (1994), pp. 183–200.
[8] D. Klatte and B. Kummer, *Generalized Kojima-functions and Lipschitz stability of critical points*, Comput. Optim. Appl., 13 (1999), pp. 61–85.
[9] M. Kojima, *Strongly stable stationary solutions in nonlinear programs*, in Analysis and Computation of Fixed Points, S. M. Robinson, ed., Academic Press, New York, 1980, pp. 93–138.
[10] A.B. Levy, *Errata in: "Implicit multifunction theorems for the sensitivity analysis of variational conditions,"* Math. Programming, 86 (1999), pp. 439–441.
[11] A.B. Levy, *Implicit multifunction theorems for the sensitivity analysis of variational conditions*, Math. Programming, 74 (1996), pp. 333–350.
[12] A.B. Levy and R.T. Rockafellar, *Sensitivity of solutions in nonlinear programming problems with nonunique multipliers*, in Recent Advances in Nonsmooth Optimization, D.-Z. Du, L. Qi, and R.S. Wommersly, eds., World Scientific, River Edge, NJ, 1995, pp. 215–223.
[13] A.B. Levy and R.T. Rockafellar, *Variational conditions and the proto-differentiation of partial subgradient mappings*, Nonlinear Anal., 26 (1995), pp. 1951–1964.
[14] A.B. Levy, R.A. Poliquin, and R.T. Rockafellar, *Stability of locally optimal solutions*, SIAM J. Optim., 10 (2000), pp. 580–604.
[15] J. Liu, *Sensitivity analysis in nonlinear programs and variational inequalities via continuous selections*, SIAM J. Control Optim., 33 (1995), pp. 1040–1060.
[16] R.A. Poliquin, *Integration of subdifferentials of nonconvex functions*, Nonlinear Anal., 17 (1991), pp. 385–398.
[17] R.A. Poliquin and R.T. Rockafellar, *Prox-regular functions in variational analysis*, Trans. Amer. Math. Soc., 348 (1995), pp. 1805–1838.
[18] R.A. Poliquin and R.T. Rockafellar, *Tilt stability of a local minimum*, SIAM J. Optim., 8 (1998), pp. 287–299.
[19] D. Ralph and S. Dempe, *Directional derivatives of the solutions of a parametric nonlinear program*, Math. Programming, 70 (1995), pp. 159–172.
[20] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.
[21] S. M. Robinson, *Generalized equations and their solutions, part* II*: Applications to nonlinear programming*, Math. Programming Study, 19 (1982), pp. 200–221.
[22] R.T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer-Verlag, Berlin, New York, Heidelburg, 1998.
[23] A. Shapiro, *Sensitivity analysis of nonlinear programs and differentiability properties of metric projections*, SIAM J. Control Optim., 26 (1988), pp. 628–645.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 179–188

GLOBAL OPTIMALITY CONDITIONS FOR QUADRATIC OPTIMIZATION PROBLEMS WITH BINARY CONSTRAINTS*

AMIR BECK† AND MARC TEBOULLE†

Abstract. We consider nonconvex quadratic optimization problems with binary constraints. Our main result identifies a class of quadratic problems for which a given feasible point is global optimal. We also establish a necessary global optimality condition. These conditions are expressed in a simple way in terms of the problem's data. We also study the relations between optimal solutions of the nonconvex binary quadratic problem versus the associated relaxed and convex problem defined over the l_∞ norm. Our approach uses elementary arguments based on convex duality.

Key words. quadratic programming, optimality conditions, nonconvex optimization, integer programming, convex duality, max-cut problem

AMS subject classifications. 90C20, 90C26, 90C09

PII. S1052623498336930

1. Introduction.

This work is concerned with quadratic optimization problems with binary constraints of the form

$$\text{(D)} \quad \min\{q(x) : \ x \in D := \{-1, 1\}^n\}, \tag{1.1}$$

where q is the quadratic function $q(x) = \frac{1}{2}x^tQx + b^tx$, where Q is an $n \times n$ symmetric matrix, and where $b \in \mathbb{R}^n$ are the given data. Problems of the above type arise naturally in several important combinatorial optimization problems, such as the max-cut problem. These problems are known to be NP hard; see, e.g., Garey and Johnson [2]. One typical approach to solve these problems is to construct lower bounds for approximating the optimal value. The classical technique to obtain bounds is either via a continuous relaxation or via the dual problem, which is usually followed by branch and bound type algorithms for refining it. This kind of approach was used, e.g., by Shor [5], and several variants of this technique, including various relaxations of the constraint set can be found in several works; see, e.g., the recent survey paper of [1] and references therein. More recently, semidefinite programming relaxations of (D) have been studied and proven to be quite powerful for finding approximate optimal solutions; see, e.g., [3] and references therein.

1.1. Motivation.

This paper is not concerned with computation of bounds for problem (D). Our main goal here is to exploit the peculiar structure of problem (D) in order to characterize global optimal solutions of problem (D), as well as to study the relations between the optimal solutions of (D) and the optimal solutions of its continuous relaxation (C) defined by

$$\text{(C)} \quad \min\{q(x) : \ x \in C := \{x : \ -1 \le x_i \le 1, \ i = 1, \dots, n\}\}.$$

We derive a sufficient optimality condition which guarantees that a given feasible point in D is a global optimal for problem (D) as well as a necessary global optimality

*Received by the editors April 16, 1998; accepted for publication (in revised form) February 13, 2000; published electronically July 25, 2000.
http://www.siam.org/journals/siopt/11-1/33693.html
†School of Mathematical Sciences, Tel-Aviv University, Ramat-Aviv 69978, Israel (amirb@math.tau.ac.il, teboulle@math.tau.ac.il). The second author was partially supported by the Israel Ministry of Science under grant 9636-1-96.

condition. An interesting fact about these conditions is that they are simply expressed in terms of the problem's data $[Q,b]$ involving only primal variables and do not involve any dual variables. To motivate the kind of conditions we are looking at, consider the following trivial example. Let Q be the diagonal matrix $Q = \mathrm{diag}(\lambda_j)_{j=1}^n$, where $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_n > 0$, and let $b \in \mathbb{R}^n$ be the given data. We then ask under which conditions on the data $[Q, b]$ we can write

$$\min\{q(x) : x \in D\} = \min\{q(x) : x \in C\}.$$

In this example, the function q is separable and can be written as

$$q(x) = \frac{1}{2}x^tQx + b^tx = \sum_{j=1}^{n} \frac{1}{2}\lambda_j x_j^2 + b_j x_j.$$

It is easy to verify that for any $a, b \in \mathbb{R}$ we have

$$\min\left\{\frac{1}{2}ax^2 + bx : -1 \leq x \leq 1\right\} = \begin{cases} -b^2(2a)^{-1} & \text{if } |ba^{-1}| < 1, \\ (2a)^{-1} + b & \text{if } |ba^{-1}| \geq 1, b \leq 0, \\ (2a)^{-1} - b & \text{if } |ba^{-1}| \geq 1, b \geq 0. \end{cases}$$

From the above computation, we thus have that a sufficient (and in this case necessary) condition to have the optimal value of the continuous minimum (C) equal to the optimal value of the discrete one (D) is simply

$$\lambda_j \leq |b_j| \quad \forall j = 1, \ldots, n. \tag{1.2}$$

This condition shows that for this particular example, we need to ask that the matrix Q is in the sense of inequality (1.2) *smaller* than the vector b. Another way to look at (1.2) is that when Q is in some sense smaller than b, then we can disregard the quadratic term and solve the trivial problem $\min_{x\in D} b^tx$.

In the next section, using simple convex duality arguments, we derive the sufficient global optimality condition for the general problem (D). This condition, like the condition derived for the trivial example above, also requires that Q is in some sense "smaller" than b. We also derive a necessary global optimality condition which is similar in form to the sufficient condition. Both conditions are simply expressed in terms of the problem's data $[Q, b]$ and do not involve any dual variables. In section 3 we treat the special case of (D), when the matrix Q is positive semidefinite. In that case, problem (D) remains nonconvex due to the constraints set; however, its continuous relaxation (C) becomes a *convex problem*. Applying the results of section 2, we then establish relations between the optimal solutions of (C) and (D). In particular, we find necessary and sufficient conditions for a vector $x \in D$ to be the solution of both (C) and (D). Furthermore, we characterize a global optimal solution of (D), whenever it is close enough to an optimal solution of the corresponding relaxed convex problem (C). We conclude the paper in section 4 with a simple application.

1.2. Notations and definitions. Throughout this paper we will use the following notations and definitions. The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$, and $\mathbb{R}^n_+, \mathbb{R}^n_{++}$ stand for the nonnegative and positive orthant, respectively. For a vector $x \in \mathbb{R}^n$, the Euclidean norm (l_2-norm) and l_∞- norm are denoted, respectively, by $||x|| := (\sum_{i=1}^n x_i^2)^{1/2}$ and $||x||_\infty := \max_{1\leq i\leq n} |x_i|$. Let $\{e_j\}_{j=1}^n$ be the canonical basis of $\mathbb{R}^n$, and let the vector of all 1's be denoted by e, i.e., $e = (1, \ldots, 1)^T$.

Given an $n \times n$ matrix Q, Diag(Q) denotes the $n \times n$ diagonal matrix with entries q_{ii}. For $x \in \mathbb{R}^n$, the corresponding capital letter will define the *diagonal* $n \times n$ matrix $X := \text{diag}(x)$ with ith diagonal element $x_i,\ i = 1, \ldots, n$, and thus we will also write $x = Xe$.

The feasible set $\{-1, 1\}^n$ of problem (D) can be written in a continuous form equivalently as:

$$D := \{x \in \mathbb{R}^n :\ x_i^2 = 1,\ i = 1, \ldots, n\}.$$

The following three equivalent formulations of the convex relaxation of D will be useful to us:

$$\begin{aligned} C &= \{x \in \mathbb{R}^n :\ ||x||_\infty \leq 1\} \\ &= \{x \in \mathbb{R}^n :\ -1 \leq x_i \leq 1,\ i = 1, \ldots, n\} \\ &= \{x \in \mathbb{R}^n :\ x_i^2 \leq 1,\ i = 1, \ldots, n\}. \end{aligned}$$

Clearly, the following relation holds: $D \subset C$.

We will denote the optimization problem of minimizing the quadratic function $q(x)$ over the set D by (D) and its global optimal value by $q_D(x)$. A similar notation is used when optimizing $q(x)$ over the set C.

For a symmetric $n \times n$ real matrix Q with elements $q_{ij} = q_{ji},\ i, j = 1, \ldots, n$, we denote by $\lambda_i(Q) \equiv \lambda_i, i = 1, \ldots, n$ its eigenvalues ordered as

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n.$$

We also use $\lambda_n \equiv \lambda_{\min}(Q) = \min\{x^T Q x,\ ||x|| = 1\}$. The matrix Q is positive semidefinite, denoted by $Q \succeq 0$ (positive definite, denoted by $Q \succ 0$) if and only if $\lambda_n \geq 0$ ($\lambda_n > 0$). The trace of Q is defined by $\text{tr}(Q) = \sum_{i=1}^n q_{ii} = \sum_{i=1}^n \lambda_i$ and it holds that $n\lambda_{\min}(Q) \leq \text{tr}(Q)$.

2. Global optimality conditions. Consider the nonconvex quadratic problem

$$\text{(D)} \quad \min\{q(x) :\ x_i^2 = 1,\ i = 1, \ldots, n\}.$$

This section is divided in two parts in which we first derive the sufficient globally optimality conditions and then the necessary one.

Sufficient conditions. Let $y \in \mathbb{R}^n$ be the multiplier associated with the constraints of (D) and form the Lagrangian

$$L(x, y) = q(x) + \sum_{i=1}^n y_i(x_i^2 - 1).$$

Defining the diagonal matrix $Y = \text{diag}(y)$, L can be written as

$$L(x, y) = \frac{1}{2} x^T (Q + Y) x + b^T x - \frac{e^T y}{2}. \tag{2.1}$$

The dual problem corresponding to (D) is then defined by the *concave* maximization problem

$$\text{(DD)} \quad \sup\{h(y) : y \in \mathbb{R}^n \cap \text{dom} h\},$$

where here h is the dual functional

$$h(y) := \inf\{L(x,y) : \ x \in \mathbb{R}^n\}, \tag{2.2}$$

and $\mathrm{dom}h = \{y \in \mathbb{R}^n : \ h(y) > -\infty\}$.

From standard duality we always have the *weak* duality relation

$$q(x) \geq h(y) \quad \forall x \in D, \ \forall y \in \mathbb{R}^n \cap \mathrm{dom}h.$$

Strong duality here of course does not hold since problem (D) is nonconvex. However, we recall the following useful result, which follows from basic duality theory [4].

LEMMA 2.1. *If there exists $\bar{x} \in D$ and $\bar{y} \in \mathbb{R}^n \cap \mathrm{dom}h$ such that $q(\bar{x}) = h(\bar{y}) = \inf_x L(x,\bar{y})$, then $\bar{x}$ is a global optimal solution of* (D).

Thus, if we are lucky enough to guess such a pair $(\bar{x},\bar{y})$ satisfying the conditions of Lemma 2.1, we can conclude that $\bar{x}$ globally solves (D). The special structure of problem (D) precisely allows us to identify such a pair. First we need to recall an elementary result on quadratic functions which will be helpful to make explicit the feasible set of the dual problem (DD).

LEMMA 2.2. *Let A be an $n \times n$ symmetric matrix, and let $f : \mathbb{R}^n \to \mathbb{R}$ be the quadratic function $f(x) = \frac{1}{2}x^T Ax + b^T x$, where $b \in \mathbb{R}^n$. Then,* $\inf\{f(x) : \ x \in \mathbb{R}^n\} > -\infty$ *if and only if the following two conditions hold:*

(i) $\exists x \in \mathbb{R}^n : \ Ax + b = 0$.

(ii) *The matrix A is positive semidefinite.*

We can now establish the following sufficient global optimality condition.

THEOREM 2.3. *Consider problem* (D) *with the data $[Q,b]$, with Q a real symmetric matrix. Let $x = Xe \in D$. If*

$$[\mathrm{SC}] \qquad \lambda_n(Q)e \geq XQXe + Xb,$$

then x is a global optimal solution for (D).

Proof. Applying Lemma 2.2 on the dual objective h defined via (2.1)–(2.2), we have $\inf\{L(x,y) : \ x \in \mathbb{R}^n\} > -\infty$ if and only if the following conditions hold:

$$\exists x \in \mathbb{R}^n : \ (Q+Y)x + b = 0, \tag{2.3}$$

$$Q + Y \succeq 0. \tag{2.4}$$

Let x be any feasible point of (D). Then, since $x = Xe$ with $X = \mathrm{diag}(x)$, from $x_i^2 = 1, \ i = 1,\ldots,n$, we also have $X^2 = I$. Now, let

$$y := -(Xb + XQXe). \tag{2.5}$$

We first show that the pair (x,y) just defined above satisfies (2.3). Indeed with $x = Xe$ and y defined in (2.5),

$$\begin{aligned}(Q+Y)x + b &= QXe + YXe + b \\ &= QXe + Xy + b \\ &= QXe - X^2 b - X^2 QXe + b \\ &= 0 \qquad (\text{since } X^2 = I).\end{aligned}$$

Now using (2.3) we can rewrite the dual objective h as

$$\begin{aligned}h(y) &= \inf_{x \in \mathbb{R}^n} \left\{ \frac{1}{2}x^T(Q+Y)x + b^T x - e^T y \right\} \\ &= -\frac{1}{2}x^T(Q+Y)x - \frac{e^T y}{2},\end{aligned}$$

with x satisfying (2.3) and such that $Q + Y \succeq 0$. Using the above expression for h, we now compute for the pair $(x = Xe, y = -XQXe - Xb)$:

$$\begin{aligned} h(y) &= -\frac{1}{2}e^T X(Q+Y)Xe - \frac{1}{2}e^T y \\ &= -\frac{1}{2}e^T XQXe - e^T y \\ &= \frac{1}{2}e^T XQXe + b^T Xe = q(Xe) = q(x). \end{aligned}$$

To complete the proof it thus remains to show that y defined in (2.5) is feasible for (DD), i.e., that $Q + Y \succeq 0$, and the result will follow from Lemma 2.1. For that, note that we always have

$$\lambda_n(Q+Y) \geq \lambda_n(Q) + \lambda_n(Y),$$

and hence $Q+Y$ is positive semidefinite if $\lambda_n(Q) \geq -\lambda_n(Y)$. But since Y is diagonal, from (2.5) we have $-\lambda_n(Y) = \max_i(Xb + XQXe)_i$ and the later inequality can thus be written as $\lambda_n(Q)e \geq XQXe + Xb$, and the proof is completed. □

Necessary conditions. We now derive global necessary optimality conditions which resemble the sufficient conditions derived in Theorem 2.3.

THEOREM 2.4. *Consider problem* (D) *with the data* $[Q, b]$, *where* Q *is a real symmetric matrix. If* $x \in D$ *is a global minimum for* (D), *then*

$$\text{[NC]} \qquad XQXe + Xb \leq \text{Diag}(Q)e.$$

Proof. If $x \in D$ is a global minimum for (D), then

$$q(x) \leq q(z) \quad \forall z \in D.$$

In particular, for $z = z_1 := -2x_1e_1 + x = (-x_1, x_2, \ldots, x_n)^T \in D$, where $e_1 = (1, 0, \ldots 0)^T$, we obtain

$$\begin{aligned} \frac{1}{2}x^T Qx + b^T x &\leq \frac{1}{2}(x - 2x_1e_1)^T Q(x - 2x_1e_1) + b^T(x - 2x_1e_1) \\ &= \frac{1}{2}x^T Qx + 2x_1^2 e_1^T Qe_1 - 2x_1 e_1^T Qx - 2x_1 b^T e_1 + b^T x. \end{aligned}$$

Since $x_i^2 = 1, e_1^t Qe_1 = q_{11}$, the later inequality reduces to

$$x_1 e_1^T Qx + x_1 b^T e_1 \leq q_{11}.$$

In a similar way we can show that for any $j = 1, \ldots, n$

$$x_j e_j^T Qx + x_j b^T e_j \leq q_{jj},$$

which proves the relation [NC]. □

It is interesting to note that both the necessary and optimality conditions are expressed only in terms of the primal variables and do not involve any dual variables. Moreover, rewriting the optimality conditions in the form

$$\begin{aligned} \text{[SC]} \quad & -Xb \geq X(Q - \lambda_{\min}(Q)I)Xe, \\ \text{[NC]} \quad & -Xb \geq X(Q - \text{Diag}(Q)I)Xe, \end{aligned}$$

we can interpret these as mentioned in the introduction by saying that a global optimal solution of (D) can be identified when the matrix Q is "smaller" than the vector b in the sense of the inequalities above. Several remarks are now in order regarding the derived optimality conditions.

Remark 2.5. In the case of pure quadratic optimization problems, i.e., when $b \equiv 0$, then the sufficient condition [SC] becomes $\lambda_{\min}(Q)e \geq XQXe$, which forces Xe to be the minimum eigenvector of Q. Thus, in the case of pure quadratic optimization problems, the sufficient condition becomes less informative. However, this difficulty can be handled by converting the pure quadratic problem into an equivalent one with a nonzero linear term in the objective. A standard and simple way to do this is just to observe that when in problem (D) $q(x) := 1/2x^TQx$, then since $q(x) = q(-x)$, we can just fix the value of an arbitrary component of x, say $x_k = 1$, and immediately get a nonhomogeneous quadratic objective, which has the same objective function value (see section 4 for an application).

Remark 2.6. Recall that $q_{jj} \geq \lambda_{\min}(Q)\ \forall j = 1,\ldots,n$, i.e., $\mathrm{Diag}(Q)e \geq \lambda_{\min}(Q)e$. Thus, using the sufficient optimality condition [SC] derived in Theorem 2.3, we have the natural implication

$$\lambda_{\min}(Q)e \geq XQXe + Xb \Longrightarrow \mathrm{Diag}(Q)e \geq XQXe + Xb.$$

Remark 2.7. Let $\bar{x} := Xe \in D$. Then [NC] implies

$$\mathrm{tr}(Q) \geq \bar{x}^TQ\bar{x} + b^T\bar{x},$$

where $\mathrm{tr}(Q) = \sum_{i=1}^n q_{ii}$. On the other hand, [SC] implies

$$n\lambda_{\min}(Q) \geq \bar{x}^TQ\bar{x} + b^T\bar{x}.$$

Since $\mathrm{tr}(Q) \geq n\lambda_{\min}(Q)$, one could be tempted to conjecture that $n\lambda_{\min}(Q) \geq \bar{x}^TQ\bar{x} + b^T\bar{x}$ could be considered as a potentially "better" sufficient condition for $\bar{x} \in D$ to be a global minimum. This is, however, not true as illustrated by the following simple example.

EXAMPLE 2.8. *Consider problem* (D) *in* $\mathbb{R}^2$ *with* $q(x) := x_1^2 - \frac{1}{2}x_2^2 + 6x_1 + 2x_2$. *The optimal solution is obtained at* $x^* = (-1,-1)^T$. *Now, let* $\bar{x} = (-1,1)^T$. *Since here* $\lambda_{\min}(Q) = -1$ *and* $n = 2$, *one can easily verify that* $n\lambda_{\min}(Q) = -2 \geq \bar{x}^TQ\bar{x} + b^T\bar{x} = -3$, *yet* $\bar{x}$ *is not global optimal.*

Remark 2.9. Let $x = (\sigma(b_i))_{i=1}^n$, where $\sigma(b_i) = 1$ if $b_i \geq 0$ and -1 otherwise. Then [SC] reduces to $XQXe \leq |b| + \lambda_{\min}(Q)e$. Thus, if the later inequality holds with $X = \mathrm{diag}(\sigma(b))$, the optimal solution of problem (D) is given by $x = (\sigma(b_i))_{i=1}^n$, namely, as the solution of the trivial problem $\min\{b^Tx :\ x \in D\}$; i.e., problem (D) can be solved by removing the quadratic term from the objective function.

We end this section by mentioning that we can state global optimality conditions for more general quadratic problems (and in particular for $\{0,1\}$ quadratic programs) of the form

$$\min\{q(x) :\ x \in \{a,c\}^n\}, \tag{2.6}$$

where $a < c$ are given real numbers. Using the linear transformation

$$x = \frac{c-a}{2}y + \frac{c+a}{2}e,$$

the above problem is transformed to $\min\{q'(y) : y \in D\}$, where $q'(y)$ can be explicitly written in terms of Q, b, a, c. A straightforward computation shows that [SC] and [NC] become, respectively,

$$\frac{c-a}{2}\lambda_{\min}(Q)e \geq \frac{a+c}{2}YQYe + Yb + \frac{a+c}{2}YQe, \tag{2.7}$$

$$\frac{c-a}{2}\mathrm{Diag}(Q)e \geq \frac{a+c}{2}YQYe + Yb + \frac{a+c}{2}YQe, \tag{2.8}$$

and the optimal solution x of problem (2.6) can be recovered from the optimal solution y via the linear transformation given above.

3. The positive semidefinite case. Let Q be a positive semidefinite matrix. Then (D) is still nonconvex because of the constraints $x \in D = \{-1, 1\}^n = \{x \in \mathbb{R}^n : x_i^2 = 1,\ i = 1, \ldots, n\}$. However, the corresponding relaxed problem (C) becomes the *convex* problem:

$$\text{(C)} \quad \min\{q(x) :\ x_i^2 \leq 1,\ i = 1, \ldots, m\}.$$

Then the question of the relations between the solution of the "easy" convex problem (C) versus the "hard" nonconvex problem (D) arises. Our first result shows that there is a simple necessary and sufficient condition for a point in D to be the solution of both the convex problem (C) and the nonconvex problem (D).

THEOREM 3.1. *Consider the nonconvex problem* (D) *with data [Q,b], with Q a real symmetric positive semidefinite matrix. Let $x = Xe \in D$. Then x is a solution of both* (C) *and* (D) *if and only if*

$$XQXe + Xb \leq 0.$$

Proof. First, suppose that $XQXe + Xb \leq 0$. Since (C) is convex and satisfies Slater's condition, strong duality applies and we have $\min\{q(x) : x \in C\} = \max\{h(y) : y \geq 0\}$, where h is the dual objective function of (C), which is the same as the one given in (2.2), except that here $y \in \mathbb{R}^n_+$. As in the proof of Theorem 2.3 with $y = -(XQXe + Xb)$, which is nonnegative by our assumption, we obtain $h(y) = q_C(Xe) = q_C(x)$, showing that x is a solution of (C) and hence of (D). To prove the converse, suppose $x = Xe$ solves (C) and (D). From the KKT optimality conditions for (C) we have $(Q + Y)x + b = 0,\ Y \succeq 0$, where $y \in \mathbb{R}^n_+$ are the multipliers for the constraints of (C). Therefore,

$$\begin{aligned} XQXe + Xb &= X(Qx + b) \\ &= -XYx \\ &= -Y, \ \text{ since } \ x \in D, \end{aligned}$$

and hence the result follows since $y \in \mathbb{R}^n_+$. ∎

Our next result characterizes an optimal solution of (D) whenever it is "close enough" to an optimal solution of the relaxed convex problem (C).

THEOREM 3.2. *Consider the problem* (D) *with data [Q,b], with Q a real symmetric positive semidefinite matrix. Let x be an optimal solution of the convex problem* (C). *If $y \in D$ satisfies the conditions*

(i) $y_i = x_i$ *when* $x_i^2 = 1$,

(ii) $YQ(y - x) \leq \lambda_{min}(Q)e$,

then y is a global optimal solution for (D).

Proof. Since (C) is a convex problem and Slater's condition holds, then x solves (C) if and only if the KKT conditions hold, i.e., there exists $\lambda \geq 0$ such that

$$(Q + \Lambda)x + b = 0, \tag{3.1}$$

$$\lambda_i(x_i^2 - 1) = 0,\ i = 1, \ldots, n, \tag{3.2}$$

where $\Lambda := \mathrm{diag}(\lambda)$. Set $\delta := y - x$, and $\Delta := \mathrm{diag}(\delta)$. Then

$$\begin{aligned} YQYe + Yb &= Y(Qy + b) \\ &= Y(Q(x + \delta) + b) \\ &= Y(-\Lambda x + Q\delta) \ \text{(using (3.1))} \\ &= (X + \Delta)(-\Lambda x + Q\delta) \\ &= -X\Lambda x + (X + \Delta)Q\delta - \Delta\Lambda x \\ &= -\lambda + YQ\delta - \Delta\Lambda x, \end{aligned}$$

where in the last equality we use (3.2). Now, we claim that $\Delta\Lambda x = 0$. Indeed, if $\delta_i = 0$, then $\lambda_i\delta_i = 0$, and if $\delta_i \neq 0$, then from the assumption of the theorem, this means $x_i^2 \neq 1$, and hence from (3.2) this implies $\lambda_i = 0$. Therefore, $\lambda_i\delta_i = 0\ \forall i$, and from the above computations, together with the fact that $\lambda \geq 0$, we have obtained

$$YQYe + Yb = -\lambda + YQ\delta \leq YQ\delta = YQ(y - x).$$

Invoking Theorem 2.3 then completes the proof. ∎

Note that when $x_i^2 \neq 1$ for some i, then the corresponding binary value y_i in the theorem above can be chosen as $y_i = \sigma(x_i)$, where $\sigma(x_i) = 1$ if $x_i \geq 0$ and -1 otherwise.

EXAMPLE 3.3. *Consider the problem* (D) *with data [Q,b], where*

$$Q = \begin{pmatrix} 4 & 2 & 0 & 2 \\ 2 & 4 & 0 & 2 \\ 0 & 0 & 4 & 2 \\ 2 & 2 & 2 & 4 \end{pmatrix},\ b = \begin{pmatrix} 4 \\ 4 \\ 3 \\ 3 \end{pmatrix}.$$

Here, we have $\lambda_{\min}(Q) = 1.036$ so that Q is positive definite. The solution of the relaxed convex problem (C) is obtained at the point $x = (-0.875, -0.875, -1, 0.625)^T$ and thus we can take (by rounding as explained above) as a "closest" point $y \in D$ to x the vector $y = (-1, -1, -1, 1)^T$. Now we compute $YQ(y - x) = (0, 0, -0.75, 1)$ so that the inequality $YQ(y - x) \leq \lambda_{\min}(Q)e$ is satisfied, and therefore from Theorem 3.2, y is the minimizing vector of (D).

4. An application. We consider a simple application of our results to problems with pure quadratic objectives, originally motivated from the max-cut problem. Given an undirected weighted graph $G = (V, E), V = \{1, 2, \ldots, n\}$, with weights $w_{ij} = w_{ji} \geq 0$ on the edges $(i, j) \in E$ and with $w_{ij} = 0$ if $(i, j) \notin E$, the max-cut problem is to find the set of vertices $S \subset V$ that maximizes the weight of the edges with one end point in S and the other in its complement $\bar{S}$, i.e., to maximize the total weight across the cut $(S, \bar{S})$. The cut can be defined by the integer variables $x_i \in \{-1, 1\}$ assigned to each vertex i. Then, with $x_i = 1$ if $i \in S$ and -1 otherwise, the weight of the

cut is $\sum_{i<j} w_{ij}(1 - x_i x_j)/2$, and the max-cut problem is equivalent to the quadratic optimization problem (see, e.g., [3]):

$$\text{(MC)} \quad \max \left\{ \sum_{i<j} w_{ij} \frac{1 - x_i x_j}{2} : \; x_i^2 = 1, \; i = 1, \ldots, n \right\}.$$

Problem (MC) can be reformulated equivalently as

$$\text{(MC)} \quad \min \left\{ \sum_{i=j} w_{ij} x_i x_j : \; x_i^2 = 1, \; i = 1, \ldots, n \right\},$$

with $w_{ii} = 0$. Defining the matrix $W = 2(w_{ij}), i, j = 1, \ldots, n$, we then obtain the formulation of (MC) as a pure quadratic problem fitting our generic formulation (D) with data $[W, 0]$, namely,

$$\text{(MC)} \quad \min \left\{ q(x) = \frac{1}{2} x^T W x : \; x \in D \right\}.$$

By elementary arguments we can obtain the following sufficient condition for a vertex to define a max-cut.

LEMMA 4.1. *Let $G = (V, E)$ be an undirected graph with $V = \{1, \ldots, n\}$ and with weight matrix W. Let l be a vertex that satisfies the following condition:*

$$\forall k \in V \setminus l : \quad w_{kl} \geq \sum_{i \neq l} w_{ik}. \tag{4.1}$$

Then l defines a max-cut; i.e., the max-cut is $S = \{l\}$ and $\bar{S}$ is the complementary set with the remaining vertices.

In other words, Lemma 4.1 says that under a particular condition as given in (4.1) on the matrix W, the vector $(-1, \ldots, -1, \underbrace{1}_{k}, -1, \ldots, -1)^T$ (meaning $x_k = 1, x_i = -1 \, \forall i \neq k$) is the minimizing vector of the problem (MC). This result relies on the fact that the matrix W in the max-cut problems satisfies the very special conditions $\text{Diag}(W) = 0$ and $w_{ij} \geq 0$. This motivates us to ask if a similar type of result can be established for an *arbitrary* pure quadratic problem, namely, when W is an $n \times n$ arbitrary symmetric matrix. An application of Theorem 2.3 leads us to establish a similar result for a class of matrices satisfying a sort of "eigenvalue-row-dominance" condition akin to the concept of diagonally dominant matrices.

PROPOSITION 4.2. *Let W be an $n \times n$ symmetric matrix that satisfies the following condition:*

$$\forall k \neq l : \quad w_{kl} \geq \sum_{i \neq l} w_{ik} - \lambda_{\min}(W(k)),$$

where $W(k)$ is the $(n-1) \times (n-1)$ matrix obtained from W by removing the kth row and column. Then the vector $(-1, \ldots, -1, \underbrace{1}_{k}, -1, \ldots, -1)^T$ is the minimizing point of problem (D) *with data [W,0].*

Proof. Without loss of generality we prove the result only for $k = 1$. By Remark 2.5, we can substitute $x_1 = 1$ and obtain a nonhomogeneous equivalent problem with data $[W', b']$ defined by

$$w_{ij}^{'} = w_{ij} \text{ if } i \neq 1, j \neq 1;\ w_{ij}^{'} = 0 \text{ if } i = 1 \text{ or } j = 1,$$
$$b' = (w_{j1})_{j=1}^{n}.$$

The above transformation obviously reduces the dimension of the original problem with data $[W, 0]$ posed in $\mathbb{R}^n$ to a nonhomogeneous problem which can now be defined in $\mathbb{R}^{n-1}$, with data $[W(1), b(1)]$, where $W(1)$ is obtained by removing the first row and column of W and $b(1)$ the first row of b'. Then, letting $X := -I_{n-1\times n-1}$, in Theorem 2.3 it follows that if

$$\lambda_{\min}(W(1))e \geq W'e - b(1),$$

then $(-1, \ldots, -1)^T \in \mathbb{R}^{n-1}$ is the solution of problem (D) with data $[W(1), b(1)]$ and thus $(\underbrace{1}_{1}, -1, \ldots, -1)^T \in \mathbb{R}^n$ is the solution of (D) with data $[W, 0]$. Similarly, the above argument can be repeated for each k, and the proof is completed. □

Acknowledgments. The authors thank the referees for the careful review and constructive comments and suggestions.

REFERENCES

[1] C. A. Floudas and V. Visweswaran, *Quadratic optimization*, in Handbook of Global Optimization, R. Horst and P. M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995, pp. 217–269.

[2] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, San Francisco, CA, 1979.

[3] M. X. Goemans and D. P. Williamson, *Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming*, J. ACM, 42 (1995), pp. 1115–1145.

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

[5] N. Z. Shor, *On a bounding method for quadratic extremal problems with* 0–1 *variables*, Kibernetika, 2 (1985), pp. 48–50.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 189–198

ON A GENERIC EXISTENCE RESULT IN OPTIMIZATION*

ALEXANDER J. ZASLAVSKI†

Dedicated to Alexander Rubinov on his 60th birthday

Abstract. In this work we consider the minimization problem $f(x) + h(x) \to \min$, $x \in X$, with $f \in C_l(X)$ and $h \in M$. Here X is a complete metric space, $C_l(X)$ is the space of all lower semicontinuous bounded from below functions on X endowed with an appropriate uniform structure, and M is a countable union of compact subsets of $C_l(X)$. We will establish the existence of a set $\mathcal{F} \subset C_l(X)$ which is a countable intersection of open everywhere dense sets in $C_l(X)$ such that for each $f \in \mathcal{F}$ and each $h \in M$ the minimization problem has a solution and, moreover, the set of its solutions is compact. In particular this result is valid when M is a countable union of a convex hull of finite sets in $C_l(X)$ satisfying certain assumptions.

Key words. existence of solutions, generic property, metric space, uniformity

AMS subject classifications. 49J99, 90C30

PII. S1052623498349280

Introduction. Let (X, ρ) be a complete metric space. Denote by $C_l(X)$ the set of all lower semicontinuous bounded from below functions $f : X \to R^1 \cup \{\infty\}$ which are not identically ∞. The set $C_l(X)$ will be endowed with an appropriate complete uniformity (see section 1). In this paper we consider the minimization problem

$$\text{(P)} \qquad f(x) + h(x) \to \min, \quad x \in X,$$

with $f \in \mathcal{L}$ and $h \in M$, where M is a countable union of compact sets in $C_l(X)$ and $\mathcal{L}$ is a subset of $C_l(X)$ with an appropriate complete uniformity. (In particular, $\mathcal{L} = C_l(X)$ and M is a countable union of a convex hull of finite sets in $C_l(X)$ satisfying certain assumptions.) We will establish the existence of a set $\mathcal{F} \subset C_l(X)$ which is a countable intersection of open everywhere dense sets in $\mathcal{L}$ such that for each $f \in \mathcal{F}$ and each $h \in M$ the minimization problem (P) has a solution and, moreover, the set of its solutions is compact.

A result of this kind when M is a singleton, X is a Banach space, and $\mathcal{L}$ is a subspace of $C_l(X)$ satisfying certain assumptions has been obtained by Deville, Godefroy, and Zizler [2]. Their result implies that given $h \in \mathcal{L}$ we can find a small perturbation $f \in \mathcal{L}$ such that the problem (P) has a solution. In this paper we will show that we can find a small perturbation $f \in \mathcal{L}$ such that the problem (P) has a solution for all functions h belonging to a countable union of compact sets in $C_l(X)$.

Since a countable intersection of everywhere dense G_δ subsets of a complete metric space is also an everywhere dense G_δ set, the result of Deville, Godefroy, and Zizler [2] can easily be extended to a countable set M. For an uncountable set M the situation is more difficult and less understood.

In [2] and in this paper, instead of studying the existence of solutions for a single cost function f we consider it for the space $\mathcal{L}$ of all such functions and show that the existence result holds for most of them. This approach has already been successfully applied in optimization theory and the calculus of variations [1, 3, 5, 7, 8, 9, 10]. This

*Received by the editors December 14, 1998; accepted for publication (in revised form) December 30, 1999; published electronically August 3, 2000.
http://www.siam.org/journals/siopt/11-1/34928.html
†Department of Mathematics, The Technion, Haifa 32000, Israel (ajzasl@techunix.technion.ac.il).

allows us to establish the existence of solutions of minimization problems without restrictive assumptions on the space X.

1. The main results. Let (K_1, d_1) and (K_2, d_2) be metric spaces. We say that a mapping $a : K_1 \to K_2$ is uniformly continuous if for each $\epsilon > 0$ there is a $\delta > 0$ such that $d_2(a(x), a(y)) \le \epsilon$ for each $x, y \in K_1$ satisfying $d_1(x, y) \le \delta$.

Assume that a set K is endowed with two metrics d_1 and d_2. We say that the metric d_2 is stronger than the metric d_1 if the identity operator $I : (K, d_2) \to (K, d_1)$ ($Ix = x$ for all $x \in X$) is uniformly continuous.

Let (X, ρ) be a complete metric space. For each $f : X \to R^1 \cup \{\infty\}$ set

$$\inf(f) = \inf\{f(x) : x \in X\} \text{ and } \mathrm{dom}(f) = \{x \in X : f(x) < \infty\}.$$

For each $f : X \to R^1 \cup \{\infty\}$ and each number $\lambda \ge 0$ we define the function $\lambda f : X \to R^1 \cup \{\infty\}$ by $(\lambda f)(x) = \lambda f(x)$, $x \in X$ (we define $0 \cdot \infty = 0$).

For each $x \in X$ and each $A \subset X$ set

$$\rho(x, A) = \inf\{\rho(x, y) : y \in A\}.$$

For the set $C_l(X)$ we consider the uniformity determined by the following base:

$$U_1(\epsilon) = \{(f, g) \in C_l(X) \times C_l(X) : \tag{1.1}$$

$$f(x) \le g(x) + \epsilon \text{ and } g(x) \le f(x) + \epsilon \text{ for all } x \in X\},$$

where $\epsilon > 0$. Clearly the space $C_l(X)$ with this uniformity is metrizable (by a metric $d_1(\cdot, \cdot)$) and complete [6]. Fix $\theta \in X$. The set $C_l(X)$ is also equipped with the uniformity determined by the following base:

$$U_0(n) = \{(f, g) \in C_l(X) \times C_l(X) : \tag{1.2}$$

$$f(x) \le g(x) + n^{-1} \text{ and } g(x) \le f(x) + n^{-1} \text{ for all } x \in X \text{ such that}$$

$$\rho(x, \theta) \le n \text{ or } \min\{f(x), g(x)\} \le n\},$$

where $n = 1, 2, \ldots$. Clearly the space $C_l(X)$ with this uniformity is metrizable (by a metric $d_0(\cdot, \cdot)$) and complete. This uniformity does not depend on our choice of θ. Evidently the metric d_1 is stronger than the metric d_0.

Suppose that $M, \mathcal{L} \subset C_l(X)$. We endow the space $\mathcal{L}$ with the metrics d_0 and d_1. Assume that $\mathcal{L}$ is also equipped with a metric $d_2(\cdot, \cdot) : \mathcal{L} \times \mathcal{L} \to [0, \infty)$, which is stronger than d_1, and the metric space $(\mathcal{L}, d_2)$ is complete. For the space $\mathcal{L}$ we consider the strong and weak topologies induced by the metrics d_2 and d_0, respectively.

We will use the following assumption for the pair $(M, \mathcal{L})$.

(A) For each $f \in \mathcal{L}$ and each $\gamma, \epsilon > 0$ there exist $\bar{f} \in \mathcal{L}$, $\eta > 0$, and a finite set $\{\bar{x}_i : i = 1, \ldots q\}$ such that

$$d_2(f, \bar{f}) \le \epsilon, \qquad \sup\{\inf(h + \bar{f}) : h \in M\} < \infty,$$

and for each $h \in M$ and each $y \in X$ satisfying

$$h(y) + \bar{f}(y) \le \inf(h + \bar{f}) + \eta$$

the following relation holds:

$$\inf\{\rho(y, \bar{x}_i) : \ i = 1, \dots, q\} \le \gamma.$$

We will establish the following result.

THEOREM 1.1. *Assume that assumption* (A) *holds and there is a constant* c *such that* $h(x) \ge c$ *for all* $x \in X$ *and all* $h \in M$. *Then there exists a set* $\mathcal{F} \subset \mathcal{L}$ *which is a countable intersection of open (in the weak topology) everywhere dense (in the strong topology) sets in* $\mathcal{L}$ *such that for each* $f \in \mathcal{F}$ *and each* $h \in M$ *the set* $\{x \in X : \ f(x) + h(x) = \inf(f + h)\}$ *is nonempty and compact.*

Theorem 1.1 implies the following result.

THEOREM 1.2. *Let* $\mathcal{M} = \cup_{i=1}^{\infty} M_i$ *where each* $M_i \subset C_l(X)$, *and for all integers* $i \ge 1$, *let assumption* (A) *hold with* $M = M_i$ *and*

$$\inf\{h(x) : \ h \in M_i \text{ and } x \in X\} > -\infty.$$

Then there exists a set $\mathcal{F} \subset \mathcal{L}$ *which is a countable intersection of open (in the weak topology) everywhere dense (in the strong topology) sets in* $\mathcal{L}$ *such that for each* $f \in \mathcal{F}$ *and each* $h \in \mathcal{M}$ *the set* $\{x \in X : \ f(x) + h(x) = \inf(f + h)\}$ *is nonempty and compact.*

In this paper we will also prove the following result.

THEOREM 1.3. *Suppose that* $M \subset C_l(X)$ *is compact in the weak topology and the following property holds:*

For each $f \in \mathcal{L}$, *each finite set* $A \subset X$, *and each* $\gamma \in (0, 1)$ *the function*

$$f_\gamma^A(x) = f(x) + \gamma \min\{\rho(x, A), 1\}, \qquad x \in X,$$

belongs to $\mathcal{L}$ *and* $d_2(f_\gamma^A, f) \to 0$ *as* $\gamma \to 0$ *uniformly on*

$$\{(f, A) : \ f \in \mathcal{L} \text{ and } A \text{ is a finite subset of } X\}.$$

Then assumption (A) *holds.*

For the set $C_l(X)$ we consider the uniformity determined by the following base:

$$U_2(n) = \{(f, g) \in C_l(X) \times C_l(X) : \tag{1.3}$$

$$f(x) \le g(x) + n^{-1} \text{ and } g(x) \le f(x) + n^{-1} \text{ for each } x \in X \text{ and}$$

$$f(x) + g(y) \le f(y) + g(x) + n^{-1}\rho(x, y) \text{ for each } x, y \in X\},$$

where $n = 1, 2, \dots$. Clearly the space $C_l(X)$ with this uniformity is metrizable.

Theorem 1.3 implies the following result in which the metric d_2 is compatible with the uniformity determined by (1.3) and the space $\mathcal{L}$ is either $C_l(X)$ or

$$\{f \in C_l(X) : \ f \text{ is finite-valued and continuous}\}$$

or the set of all finite-valued continuous functions $f \in C_l(X)$ such that

$$\sup\{|f(y) - f(x)|/\rho(x, y) : \ x, y \in X \text{ and } x \ne y\} < \infty.$$

(Note that in all these cases the metric space $(\mathcal{L}, d_2)$ is complete.)

THEOREM 1.4. *Assume that M is a countable union of compact sets in $C_l(X)$ with the weak topology. There exists a set $\mathcal{F} \subset \mathcal{L}$ which is a countable intersection of open (in the weak topology) everywhere dense (in the strong topology) sets in $\mathcal{L}$ such that for each $f \in \mathcal{F}$ and each $h \in M$ the set $\{x \in X : \ f(x) + h(x) = \inf(f + h)\}$ is nonempty and compact.*

Remark. Theorem 1.4 is true with $M = \cup_{i=1}^{\infty} M_i$, where for each integer $i \geq 1$

$$M_i = \left\{ \sum_{j=1}^{n_i} \alpha_j h_{ij} : \ \alpha_j \geq 0, \ j = 1, \dots, n_i \right\},$$

$h_{ij} \in C_l(X)$ is bounded on bounded subsets of $\mathrm{dom}(h_{ij})$ for $j = 1, \dots, n_i$, and

$$\cap_{j=1}^{n_i} \mathrm{dom}(h_{ij}) \neq \emptyset.$$

To show this it is sufficient to verify that for each integer $i \geq 1$, each $\epsilon \in (0, n_i^{-1})$, and each number $r > 1$ the set

$$\left\{ \sum_{j=1}^{n_i} \alpha_j h_{ij} : \ \alpha_j \geq \epsilon \text{ for } j = 1, \dots, n_i \text{ and } \sum_{j=1}^{n_i} \alpha_j \leq r \right\}$$

is compact in the weak topology. This verification can be done in a straightforward manner.

2. Proof of Theorem 1.1. By assumption (A) for each $f \in \mathcal{L}$ and each natural number n there exist $g(f,n) \in \mathcal{L}$, $\eta(f,n) \in (0,1)$, $c(f,n) > 0$, a natural number $q(f,n)$, and a finite set $\{x_i(f,n) : \ i = 1, \dots, q(f,n)\}$ such that

$$d_2(f, g(f,n)) \leq n^{-1}, \tag{2.1}$$

$$\inf(h + g(f,n)) \leq c(f,n) \quad \text{for all } h \in M, \tag{2.2}$$

and for each $h \in M$ and each $y \in X$ satisfying

$$h(y) + g(f,n)(y) \leq \inf(h + g(f,n)) + 4\eta(f,n) \tag{2.3}$$

the following relation holds:

$$\inf\{\rho(y, x_i(f,n)) : \ i = 1, \dots, q(f,n)\} \leq n^{-1}. \tag{2.4}$$

Let $f \in \mathcal{L}$ and let n be a natural number. Fix a natural number

$$s(f,n) > \max\{\rho(\theta, x_i(f,n)) : \ i = 1, \dots, q(f,n)\} + 4 + c(f,n) + |c| + \eta(f,n)^{-1} \tag{2.5}$$

and set

$$A(f,n) = \{x_i(f,n) : \ i = 1, \dots, q(f,n)\}.$$

By the definition of the metric d_0 (see (1.2)) there exists an open neighborhood $U(f,n)$ of $g(f,n)$ in $\mathcal{L}$ with the weak topology such that

$$U(f,n) \subset \{h \in \mathcal{L} : \ (h, g(f,n)) \in U_0(s(f,n))\}. \tag{2.6}$$

We will show that the following property holds:

(i) For each $h \in M$, each $\xi \in U(f,n)$, and each $y \in X$ satisfying

$$h(y) + \xi(y) \leq \inf(h + \xi) + \eta(f,n) \tag{2.7}$$

the relation (2.4) is valid.

Let $h \in M$ and let $\xi \in U(f,n)$. It follows from the definition of $g(f,n)$ and $\eta(f,n)$ (see (2.3), (2.4)) that

$$\inf(h + g(f,n)) = \inf\{h(y) + g(f,n)(y) : \ y \in X \text{ and } \tag{2.8}$$

$$\inf\{\rho(y, x_i(f,n)) : \ i = 1 \ldots, q(f,n)\} \leq n^{-1}\}.$$

By (2.8), (2.5), and (1.2)

$$\begin{aligned} \inf(h + \xi) &\leq \inf\{h(y) + \xi(y) : \ y \in X \text{ and } \rho(y, A(f,n)) \leq n^{-1}\} \\ &\leq \inf(h + g(f,n)) + s(f,n)^{-1}. \end{aligned} \tag{2.9}$$

Assume that $y \in X$ and (2.7) is valid. Then by (2.7), (2.9), (2.2), and (2.5)

$$h(y) + \xi(y) \leq \inf(h + g(f,n)) + s(f,n)^{-1} + \eta(f,n) \tag{2.10}$$

and

$$\xi(y) \leq c(f,n) + 2 + |c|. \tag{2.11}$$

It follows from (2.11), (2.6), (1.2), (2.5), and (2.10) that

$$g(f,n)(y) \leq \xi(y) + s(f,n)^{-1}$$

and

$$\begin{aligned} h(y) + g(f,n)(y) &\leq \inf(h + g(f,n)) + 2s(f,n)^{-1} + \eta(f,n) \\ &\leq \inf(h + g(f,n)) + 3\eta(f,n). \end{aligned} \tag{2.12}$$

By (2.12) and the definition of $g(f,n)$ and $\eta(f,n)$ the inequality (2.4) is valid. Therefore we have shown that property (i) holds. Define

$$\mathcal{F} = \cap_{m=1}^{\infty} \cup \{U(f,n) : \ f \in \mathcal{L} \text{ and } n \geq m\}.$$

Clearly $\mathcal{F}$ is a countable intersection of open (in the weak topology) everywhere dense (in the strong topology) sets in $\mathcal{L}$.

Assume that $h \in M$ and $\xi \in \mathcal{F}$. Consider a sequence $\{z_i\}_{i=1}^{\infty} \subset X$ such that

$$\lim_{i \to \infty} (h(z_i) + \xi(z_i)) = \inf(h + \xi). \tag{2.13}$$

Let $\epsilon \in (0,1)$. Choose a natural number k for which

$$k > 64\epsilon^{-1}. \tag{2.14}$$

There exist $f \in \mathcal{L}$ and an integer $n \geq k$ such that

$$\xi \in U(f,n). \tag{2.15}$$

It follows from (2.15), (2.13), (2.14), and property (i) that for all large enough integers i the relation

$$z_i \in \cup_{j=1}^{q(f,n)}\{z \in X:\ \rho(z, x_j(f,n)) \le 4^{-1}\epsilon\}$$

is true. Since ϵ is an arbitrary positive number in $(0,1)$ this relation implies that for each integer $p \ge 1$ there exists a subsequence $\{z_{i_k}^{(p)}\}_{k=1}^{\infty}$ of the sequence $\{z_i\}_{i=1}^{\infty}$ such that the following properties hold:

For each $p \ge 1$ the sequence $\{z_{i_k}^{(p+1)}\}_{k=1}^{\infty}$ is a subsequence of $\{z_{i_k}^{(p)}\}_{k=1}^{\infty}$.

For each integer $p \ge 1$ and each pair of integers $j, s \ge 1$,

$$\rho\left(z_{i_j}^{(p)}, z_{i_s}^{(p)}\right) \le p^{-1}.$$

These properties imply that there exists a subsequence $\{z_{i_k}^{*}\}_{k=1}^{\infty}$ of the sequence $\{z_i\}_{i=1}^{\infty}$ that is a Cauchy sequence. There exists $x_* = \lim_{k\to\infty} z_{i_k}^{*}$. By (2.13) and the lower semicontinuity of ξ and h

$$h(x_*) + \xi(x_*) = \inf(h + \xi). \tag{2.16}$$

Therefore we have shown that for each sequence $\{z_i\}_{i=1}^{\infty} \subset X$ satisfying (2.13) there exists a subsequence $\{z_{i_k}^{*}\}_{k=1}^{\infty}$ that converges to $x_* \in X$ and that x_* satisfies (2.16). This completes the proof of the theorem.

Remark. Actually we proved that every minimizing sequence (for the function $\xi + h$ with $\xi \in \mathcal{F}$ and $h \in M$) has a convergent subsequence. That is, the optimization problem is well posed in the generalized sense (see [4]).

3. Proof of Theorem 1.3. For the proof of Theorem 1.3 we need the following auxiliary lemmas.

LEMMA 3.1. *Assume that a set $M \subset C_l(X)$ is compact in the weak topology. Then there is $c \in R^1$ such that $h(x) \ge c$ for all $x \in X$ and all $h \in M$.*

Proof. Let $f \in M$. Choose a number $c(f)$ and a natural number $n(f)$ such that

$$f(x) \ge c(f) \quad \text{for each } x \in X \text{ and } n(f) \ge 2 + |c(f)|. \tag{3.1}$$

There exists an open neighborhood $U(f)$ of f in $C_l(X)$ in the weak topology such that

$$U(f) \subset \{g \in \mathcal{L}:\ (f,g) \in U_0(n(f))\}. \tag{3.2}$$

We will show that for each $h \in U(f)$ and each $x \in X$ the relation $h(x) \ge c(f) - 2$ is true.

Assume the contrary. Then there exist $h \in U(f)$ and $x \in X$ such that $h(x) < c(f) - 2$. By the definition of $U(f)$ (see (3.2)), (3.1), and (1.2)

$$f(x) \le h(x) + n(f)^{-1} \le c(f) - 1.$$

This relation is contradictory to (3.1). The obtained contradiction proves that $h(x) \ge c(f) - 2$ for each $h \in U(f)$ and each $x \in X$. Since $M \subset \cup\{U(f):\ f \in M\}$ there exists a finite set $\{f_1, \dots, f_k\} \subset M$ such that $M \subset \cup_{i=1}^{k} U(f_i)$. This implies that for each $h \in M$ and each $x \in X$

$$h(x) \ge \min\{c(f_i):\ i = 1, \dots, k\} - 2.$$

The lemma is proved. □

LEMMA 3.2. *Assume that a set $M \subset C_l(X)$ is compact in the weak topology and $g \in C_l(X)$. Then*

$$\sup\{\inf(h+g) : \ h \in M\} < \infty. \tag{3.3}$$

Proof. By Lemma 3.1 there exists a number c such that

$$\inf(h) \geq c \quad \text{for all } h \in M \text{ and } \inf(g) \geq c. \tag{3.4}$$

Let $f \in M$. Choose a natural number

$$n(f) > 2 + |c| + |\inf(f+g)|. \tag{3.5}$$

There exists an open neighborhood $U(f)$ of f in $C_l(X)$ with the weak topology such that

$$U(f) \subset \{h \in C_l(X) : \ (f,h) \in U_0(n(f))\}. \tag{3.6}$$

Let $h \in U(f)$. There is $x \in X$ such that

$$f(x) + g(x) < \inf(f+g) + 1. \tag{3.7}$$

The inequalities (3.7), (3.4), and (3.5) imply that

$$f(x) < \inf(f+g) + 1 - c < n(f). \tag{3.8}$$

By (3.8), (3.6), (1.2), and (3.7)

$$h(x) \leq f(x) + 1$$

and

$$\inf(h+g) \leq g(x) + h(x) \leq g(x) + f(x) + 1 < \inf(f+g) + 2. \tag{3.9}$$

Clearly there exists a finite set $\{f_1, \ldots, f_n\} \subset M$ such that $M \subset \cup_{i=1}^n U(f_i)$. Together with (3.9) this implies that for each $h \in M$

$$\inf(h+g) \leq \max\{\inf(f_i+g) : \ i = 1, \ldots, n\} + 2.$$

The lemma is proved. □

LEMMA 3.3. *Let $q \geq 1$ be an integer, $\epsilon \in (0,1)$, $f_1, \ldots, f_q \in C_l(X)$ and let $\{x_i\}_{i=1}^q \subset X$,*

$$f_i(x_i) \leq \inf(f_i) + 8^{-1}\epsilon^2, \qquad i = 1, \ldots, q. \tag{3.10}$$

For $i = 1, \ldots, q$ define $\bar{f}_i : \ X \to R^1 \cup \{\infty\}$ by

$$\bar{f}_i(z) = f_i(z) + \epsilon \min\{1, \rho(z, \{x_j : \ j = 1, \ldots, q\})\}, \qquad z \in X. \tag{3.11}$$

Then for each $p \in \{1, \ldots, q\}$ and each $y \in X$ satisfying

$$\bar{f}_p(y) \leq \inf \bar{f}_p + 4^{-1}\epsilon^2 \tag{3.12}$$

the following relation holds:

$$y \in \cup_{j=1}^{q}\{z \in X:\ \rho(z, x_j) \le 2^{-1}\epsilon\}. \tag{3.13}$$

Proof. Clearly $\bar{f}_p \in C_l(X)$, $p = 1, \dots, q$. Assume that $p \in \{1, \dots, q\}$, $y \in X$, and (3.12) holds. By (3.11), (3.12), and (3.10)

$$f_p(y) + \epsilon \min\{1, \rho(y, \{x_j:\ j = 1, \dots, q\})\} = \bar{f}_p(y) \le \inf(\bar{f}_p) + 4^{-1}\epsilon^2$$

$$\le 4^{-1}\epsilon^2 + \bar{f}_p(x_p) \le 4^{-1}\epsilon^2 + f_p(x_p) \le 4^{-1}\epsilon^2 + 8^{-1}\epsilon^2 + \inf(f_p)$$

$$\le 4^{-1}\epsilon^2 + 8^{-1}\epsilon^2 + f_p(y).$$

Therefore

$$\inf\{1, \rho(y, \{x_j:\ j = 1, \dots, q\})\} \le 4^{-1}\epsilon + 8^{-1}\epsilon.$$

This implies (3.13). The lemma is proved. □

For each set $A \subset X$ define $\phi_A:\ X \to R^1$ by

$$\phi_A(x) = \min\{1, \rho(x, A)\}, \qquad x \in X. \tag{3.14}$$

LEMMA 3.4. *Assume that a set $M \subset C_l(X)$ is compact in the weak topology, $g \in C_l(X)$, and $\epsilon \in (0, 1)$. Then there exists a finite set $A = \{x_i:\ i = 1, \dots, q\} \subset X$, where q is a natural number such that for each $f \in M$ and each $y \in X$ satisfying*

$$f(y) + g(y) + \epsilon\phi_A(y) \le \inf(f + g + \epsilon\phi_A) + 32^{-1}\epsilon^2 \tag{3.15}$$

the following relation holds:

$$\inf\{\rho(y, x_j):\ j = 1, \dots, q\} \le 2^{-1}\epsilon. \tag{3.16}$$

Proof. By Lemmas 3.1 and 3.2 there are numbers c_0, c_1 such that

$$\inf(g) \ge c_0, \quad \inf(h) \ge c_0 \quad \text{for all } h \in M \text{ and } \sup_{h \in M} |\inf(h + g)| < c_1. \tag{3.17}$$

Choose a natural number

$$n_0 > 64\epsilon^{-2} + 4 + |c_0| + |c_1|. \tag{3.18}$$

There exists a finite set $\{f_i:\ i = 1, \dots, q\} \subset M$ such that

$$M \subset \cup_{i=1}^{q}\{h \in C_l(X):\ (h, f_i) \in U_0(n_0)\}. \tag{3.19}$$

For each $i = 1, \dots, q$ choose $x_i \in X$ such that

$$f_i(x_i) + g(x_i) \le \inf(f_i + g) + 64^{-1}\epsilon^2. \tag{3.20}$$

Set

$$A = \{x_1, \dots, x_q\}. \tag{3.21}$$

Assume that $f \in M$, $y \in X$, and (3.15) holds. We will show that (3.16) is valid. There is $p \in \{1, \dots, q\}$ such that

$$(f, f_p) \in U_0(n_0). \tag{3.22}$$

We will show that

$$f_p(y) + g(y) + \epsilon\phi_A(y) \le \inf(f_p + g + \epsilon\phi_A) + 16^{-1}\epsilon^2. \tag{3.23}$$

By (3.17) for any $h \in M$

$$\inf(h + g + \epsilon\phi_A) = \inf\{h(z) + g(z) + \epsilon\phi_A(z) : \ z \in X \text{ and}$$

$$h(z) + g(z) + \epsilon\phi_A(z) \le \inf(h + g + \epsilon\phi_A) + 1\}$$

$$= \inf\{h(z) + g(z) + \epsilon\phi_A(z) : \ z \in X \text{ and } h(z) \le \inf(h + g + \epsilon\phi_A) + 1 + |c_0|\}$$

$$= \inf\{h(z) + g(z) + \epsilon\phi_A(z) : \ z \in X \text{ and } h(z) \le 2 + |c_0| + |c_1|\}.$$

It follows from this relation, (3.18), (3.22) and (1.2) that

$$\inf(f + g + \epsilon\phi_A) = \inf\{f(z) + g(z) + \epsilon\phi_A(z) : \tag{3.24}$$

$$z \in X \text{ and } f(z) \le 3 + |c_0| + |c_1|\} \le \inf\{f_p(z) + g(z) + \epsilon\phi_A(z) :$$

$$z \in X \text{ and } f(z) \le 3 + |c_0| + |c_1|\} + n_0^{-1}$$

$$\le \inf\{f_p(z) + g(z) + \epsilon\phi_A(z) : \ z \in X \text{ and } f_p(z) \le 2 + |c_0| + |c_1|\} + n_0^{-1}$$

$$= \inf(f_p + g + \epsilon\phi_A) + n_0^{-1}.$$

The relations (3.14), (3.15), (3.17), and (3.18) imply that

$$f(y) + g(y) + \epsilon\phi_A(y) \le c_1 + 2 \quad \text{and} \quad f(y) \le |c_1| + 2 + |c_0| < n_0.$$

By this relation, (3.22), (1.2), (3.15), (3.24), and (3.18)

$$f_p(y) \le f(y) + n_0^{-1}$$

and

$$f_p(y) + g(y) + \epsilon\phi_A(y) \le f(y) + g(y) + \epsilon\phi_A(y) + n_0^{-1}$$

$$\le n_0^{-1} + \inf(f + g + \epsilon\phi_A) + 32^{-1}\epsilon^2$$

$$\le n_0^{-1} + 32^{-1}\epsilon^2 + \inf(f_p + g + \epsilon\phi_A) + n_0^{-1} \le 16^{-1}\epsilon^2 + \inf(f_p + g + \epsilon\phi_A).$$

Therefore (3.23) is valid. By Lemma 3.3 the inequality (3.16) holds. The lemma is proved. □

Lemmas 3.4 and 3.2 implies Theorem 1.3.

REFERENCES

[1] J. Beer and R. Lucchetti, *Convex optimization and the epi-distance topology*, Trans. Amer. Math. Soc., 327 (1991), pp. 795–813.
[2] R. Deville, R. Godefroy, and V. Zizler, *Smoothness and Renormings in Banach Spaces*, Longman, London, 1993.
[3] A.L. Dontchev and I. Zolezzi, *Well-Posed Optimization Problems*, Lecture Notes in Math. 1543, Springer-Verlag, Berlin, Heidelberg, New York, 1993.
[4] M. Furi and V. Vignoli, *About well-posed minimization problems for functionals in metric spaces*, J. Optim. Theory Appl., 5 (1970), pp. 225–229.
[5] A.D. Ioffe and A.J. Zaslavski, *Variational principles and well-posedness in optimization and calculus of variations*, SIAM J. Control Optim., 38 (2000), pp. 566–581.
[6] J.L. Kelley, *General Topology*, Van Nostrand, New York, 1955.
[7] S. Reich and A.J. Zaslavski, *On the minimization of convex functionals*, in Calculus of Variations and Differential Equations, Chapman and Hall/CRC Res. Notes Math. 410, Chapman and Hall/CRC Press, Boca Raton, FL, 1999, pp. 200–209.
[8] J.P. Revalski, *Generic properties concerning well-posed optimization problems*, C. R. Acad. Bulgare Sci., 38 (1985), pp. 1431–1434.
[9] A.J. Zaslavski, *Optimal programs on infinite horizon.* I, II, SIAM J. Control Optim., 33 (1995), pp. 1643–1686.
[10] A.J. Zaslavski, *Existence of solutions of optimal control problems without convexity assumptions*, Nonlinear Anal., in press.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 199–222

A FEASIBLE BFGS INTERIOR POINT ALGORITHM FOR SOLVING CONVEX MINIMIZATION PROBLEMS*

PAUL ARMAND†, JEAN CHARLES GILBERT‡, AND SOPHIE JAN-JÉGOU§

Abstract. We propose a BFGS primal-dual interior point method for minimizing a convex function on a convex set defined by equality and inequality constraints. The algorithm generates feasible iterates and consists in computing approximate solutions of the optimality conditions perturbed by a sequence of positive parameters μ converging to zero. We prove that it converges q-superlinearly for each fixed μ. We also show that it is globally convergent to the analytic center of the primal-dual optimal set when μ tends to 0 and strict complementarity holds.

Key words. analytic center, BFGS quasi-Newton approximations, constrained optimization, convex programming, interior point algorithm, line-search, primal-dual method, superlinear convergence

AMS subject classifications. 90Cxx, 90C25, 90C51, 90C53

PII. S1052623498344720

1. Introduction.

We consider the problem of minimizing a smooth convex function on a convex set defined by inequality constraints. The problem is written as

$$\left\{ \begin{array}{l} \min\ f(x), \\ c(x) \geq 0, \end{array} \right. \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is the function to minimize and $c(x) \geq 0$ means that each component $c_{(i)} : \mathbb{R}^n \to \mathbb{R}$ $(1 \leq i \leq m)$ of c must be nonnegative at the solution. To simplify the presentation and to avoid complicated notation, the case when linear equality constraints are present is discussed at the end of the paper. Since we assume that the components of c are *concave*, the feasible set of this problem is convex.

The algorithm proposed in this paper and the convergence analysis require that f and c are differentiable and that at least one of the functions $f, -c_{(1)}, \ldots, -c_{(m)}$ is *strongly* convex. The reason for this latter hypothesis will be clarified below. Since the algorithm belongs to the class of interior point (IP) methods, it may be well suited for problems with many inequality constraints. It is also more efficient when the number of variables remains small or medium, say, fewer than 500, because it updates $n \times n$ matrices by a quasi-Newton (qN) formula. For problems with more variables, limited memory BFGS updates [39] can be used, but we will not consider this issue in this paper.

Our motivation is based on practical considerations. During the last 15 years much progress has been realized on IP methods for solving linear or convex minimization problems (see the monographs [29, 10, 38, 44, 23, 42, 47, 49]). For nonlinear convex problems, these algorithms assume that the second derivatives of the functions used to define the problem are available (see [43, 35, 36, 12, 38, 26]). In practice, how-

*Received by the editors September 15, 1998; accepted for publication (in revised form) January 26, 2000; published electronically August 3, 2000.
http://www.siam.org/journals/siopt/11-1/34472.html
†LACO, Faculté des Sciences, 123 av. A. Thomas, 87060 Limoges Cedex, France (Paul.Armand@unilim.fr).
‡INRIA Rocquencourt, BP 105, 78153 Le Chesnay Cedex, France (Jean-Charles.Gilbert@inria.fr).
§MIP, UFR MIG, Université Paul Sabatier, 118 route de Narbonne, 31062 Toulouse Cedex 4, France (jan@mip.ups-tlse.fr).

ever, it is not uncommon to find situations where this requirement cannot be satisfied, in particular for large scale engineering problems (see [27] for an example, which partly motivates this study and deals with the estimation of parameters in a three phase flow in a porous medium). Despite the possible use of computational differentiation techniques [8, 19, 3, 28], the computing time needed to evaluate Hessians or Hessian-vector products may be so large that IP algorithms using second derivatives may be unattractive.

This situation is familiar in unconstrained optimization. In that case, qN techniques, which use first derivatives only, have proved to be efficient, even when there are millions of variables (see [32, 20] and [9] for an example in meteorology). This fact motivates the present paper, in which we explore the possibility of combining the IP approach and qN techniques. Our ambition remains modest, however, since we confine ourselves to the question of whether the elegant BFGS theory for unconstrained convex optimization [41, 6] is still valid when inequality constraints are present. For the applications, it would be desirable to have a qN-IP algorithm in the case when f and $-c$ are nonlinear and not necessarily convex. We postpone this more difficult subject for future research (see [21, 48] for possible approaches).

Provided the constraints satisfy some qualification assumptions, the Karush–Kuhn–Tucker (KKT) optimality conditions of problem (1.1) can be written (see [17], for example) as follows: there exists a vector of multipliers $\lambda \in \mathbb{R}^m$ such that

$$\left\{ \begin{array}{l} \nabla f(x) - \nabla c(x)\lambda = 0, \\ C(x)\lambda = 0, \\ (c(x), \lambda) \geq 0, \end{array} \right.$$

where $\nabla f(x)$ is the gradient of f at x (for the Euclidean scalar product), $\nabla c(x)$ is a matrix whose columns are the gradients $\nabla c_{(i)}(x)$, and $C = \mathrm{diag}(c_{(1)}, \ldots, c_{(m)})$ is the diagonal matrix, whose diagonal elements are the components of c. The Lagrangian function associated with problem (1.1) is defined on $\mathbb{R}^n \times \mathbb{R}^m$ by

$$\ell(x, \lambda) = f(x) - \lambda^\top c(x).$$

Since f is convex and each component $c_{(i)}$ is concave, for any fixed $\lambda \geq 0$, $\ell(\cdot, \lambda)$ is a convex function from $\mathbb{R}^n$ to $\mathbb{R}$. When f and c are twice differentiable, the gradient and Hessian of ℓ with respect to x are given by

$$\nabla_x \ell(x, \lambda) = \nabla f(x) - \nabla c(x)\lambda \quad \text{and} \quad \nabla^2_{xx} \ell(x, \lambda) = \nabla^2 f(x) - \sum_{i=1}^{m} \lambda_{(i)} \nabla^2 c_{(i)}(x).$$

Our primal-dual IP approach is rather standard (see [24, 36, 35, 11, 12, 1, 26, 25, 15, 7, 5]). It computes iteratively approximate solutions of the perturbed optimality system

$$\left\{ \begin{array}{l} \nabla f(x) - \nabla c(x)\lambda = 0, \\ C(x)\lambda = \mu e, \\ (c(x), \lambda) > 0 \end{array} \right. \tag{1.2}$$

for a sequence of parameters $\mu > 0$ converging to zero. In (1.2), $e = (1 \ \cdots \ 1)^\top$ is the vector of all ones whose dimension will be clear from the context. The last inequality means that all the components of both $c(x)$ and λ must be positive. By perturbing the complementarity equation of the KKT conditions with the parameter

μ, the combinatorial aspect of the problem, inherent in the determination of the active constraints or the zero multipliers, is avoided. We use the word *inner* to qualify those iterations that are used to find an approximate solution of (1.2) for fixed μ, while an *outer iteration* is the collection of inner iterations corresponding to the same value of μ.

The Newton step for solving the first two equations in (1.2) with fixed μ is the solution $d = (d^x, d^\lambda) \in \mathbb{R}^n \times \mathbb{R}^m$ of the linear system

$$\begin{pmatrix} M & -\nabla c(x) \\ \Lambda \nabla c(x)^\top & C(x) \end{pmatrix} \begin{pmatrix} d^x \\ d^\lambda \end{pmatrix} = \begin{pmatrix} -\nabla f(x) + \nabla c(x)\lambda \\ \mu e - C(x)\lambda \end{pmatrix}, \tag{1.3}$$

in which $M = \nabla^2_{xx}\ell(x,\lambda)$ and $\Lambda = \operatorname{diag}(\lambda_{(1)}, \ldots, \lambda_{(m)})$. This direction is sometimes called the primal-dual step, since it is obtained by linearizing the primal-dual system (1.2), while the primal step is the Newton direction for minimizing in the primal variable x the *barrier function*

$$\varphi_\mu(x) := f(x) - \mu \sum_{i=1}^{m} \log c_{(i)}(x)$$

associated with (1.1) (the algorithms in [16, 33, 4] are in this spirit). The two problems are related since, after elimination of λ, (1.2) represents the optimality conditions of the unconstrained *barrier problem*

$$\begin{cases} \min \varphi_\mu(x), \\ c(x) > 0. \end{cases} \tag{1.4}$$

As a result, an approximate solution of (1.2) is also an approximate minimizer of the barrier problem (1.4). However, algorithms using the primal-dual direction have been shown to present a better numerical efficiency (see, for example, [46]).

In our algorithm for solving (1.2) or (1.4) approximately, a search direction d is computed as a solution of (1.3) in which M is now a positive definite symmetric matrix approximating $\nabla^2_{xx}\ell(x,\lambda)$ and updated by the BFGS formula (see [14, 17] for material on qN techniques). By eliminating d^λ from (1.3) we obtain

$$(M + \nabla c(x) C(x)^{-1} \Lambda \nabla c(x)^\top) d^x = -\nabla f(x) + \mu \nabla c(x) C(x)^{-1} e = -\nabla \varphi_\mu(x). \tag{1.5}$$

Since the iterates will be forced to remain strictly feasible, i.e., $(c(x), \lambda) > 0$, the positive definiteness of M implies that d^x is a descent direction of φ_μ at x. Therefore, to force convergence of the inner iterates, a possibility could be to force the decrease of φ_μ at each iteration. However, since the algorithm also generates dual variables λ, we prefer to add to φ_μ the function (see [45, 1, 18])

$$\mathcal{V}(x,\lambda) := \lambda^\top c(x) - \mu \sum_{i=1}^{m} \log\left(\lambda_{(i)} c_{(i)}(x)\right)$$

to control the change in λ. This function is also used in [30, 31] as a potential function for nonlinear complementarity problems. Even though the map $(x,\lambda) \mapsto \varphi_\mu(x) + \mathcal{V}(x,\lambda)$ is not necessarily convex, we will show that it has a unique minimizer, which is the solution of (1.2), and that it decreases along the direction $d = (d^x, d^\lambda)$. Therefore, this primal-dual merit function can be used to force the convergence of the pairs (x,λ) to the solution of (1.2), using line-searches. It will be shown that the additional

function $\mathcal{V}$ does not prevent unit step-sizes from being accepted asymptotically, which is an important point for the efficiency of the algorithm.

Let us stress the fact that our algorithm is not a standard BFGS algorithm for solving the barrier problem (1.4), since it is the Hessian of the Lagrangian that is approximated by the updated matrix M, not the Hessian of φ_μ. This is motivated by the following arguments. First, the difference between $\nabla^2_{xx}\ell(x, \mu C(x)^{-1}e)$ and

$$\nabla^2\varphi_\mu(x) = \nabla^2 f(x) + \mu \sum_{i=1}^{m} \left(\frac{1}{c_{(i)}(x)^2} \nabla c_{(i)}(x) \nabla c_{(i)}(x)^\top - \frac{1}{c_{(i)}(x)} \nabla^2 c_{(i)}(x) \right) \tag{1.6}$$

involves first derivatives only. Since these derivatives are considered to be available, they need not be approximated. Second, the Hessian $\nabla^2_{xx}\ell$, which is approximated by M, is independent of μ and does not become ill-conditioned as μ goes to zero. Third, the approximation of $\nabla^2_{xx}\ell$ obtained at the end of an outer iteration can be used as the starting matrix for the next outer iteration. If this looks attractive, it has also the inconvenience of restricting the approach to (strongly) convex functions, as we now explain.

After the computation of the new iterates $x_+ = x + \alpha d^x$ and $\lambda_+ = \lambda + \alpha d^\lambda$ (α is the step-size given by the line-search), the matrix M is updated by the BFGS formula using two vectors δ and γ. Since we want the new matrix M_+ to be an approximation of $\nabla^2_{xx}\ell(x_+, \lambda_+)$ and because it satisfies the qN equation $M_+\delta = \gamma$ (a property of the BFGS formula), it makes sense to define δ and γ by

$$\delta := x_+ - x \quad \text{and} \quad \gamma := \nabla_x \ell(x_+, \lambda_+) - \nabla_x \ell(x, \lambda_+).$$

The formula is well defined and generates stable positive definite matrices provided these vectors satisfy $\gamma^\top\delta > 0$. This inequality, known as the curvature condition, expresses the strict monotonicity of the gradient of the Lagrangian between two successive iterates. In unconstrained optimization, it can always be satisfied by using the Wolfe line-search, provided the function to minimize is bounded below. If this is a reasonable assumption in unconstrained optimization, it is no longer the case when constraints are present, since the optimization problem may be perfectly well defined even when ℓ is unbounded below. Now, assuming this hypothesis on the boundedness of ℓ would have been less restrictive than assuming its strong convexity, but it is not satisfactory. Indeed, with a bounded below Lagrangian, the curvature condition can be satisfied by the Wolfe line-search as in unconstrained optimization, but near the solution the information on $\nabla^2_{xx}\ell$ collected in the matrix M could come from a region far from the optimal point, which would prevent q-superlinear convergence of the iterates. Because of this observation, we assume that f or one of the functions $-c_{(i)}$ is strongly convex, so that the Lagrangian becomes a strongly convex function of x for any fixed $\lambda > 0$. With this assumption, the curvature condition will be satisfied independently of the kind of line-search techniques actually used in the algorithm. The question whether the present theory can be adapted to convex problems, hence including linear programming, is puzzling. We will come back to this issue in the discussion section.

A large part of the paper is devoted to the analysis of the qN algorithm for solving the perturbed KKT conditions (1.2) with fixed μ. The algorithm is detailed in the next section, while its convergence speed is analyzed in sections 3 and 4. In particular, it is shown that, for fixed $\mu > 0$, the primal-dual pairs (x, λ) converge q-superlinearly toward a solution of (1.2). The tools used to prove convergence are essentially those of

the BFGS theory [6, 13, 40]. In section 5, the overall algorithm is presented and it is shown that the sequence of outer iterates is globally convergent, in the sense that it is bounded and that its accumulation points are primal-dual solutions of problem (1.1). If, in addition, strict complementarity holds, the whole sequence of outer iterates converges to the analytic center of the primal-dual optimal set.

2. The algorithm for solving the barrier problem. The Euclidean or ℓ_2 norm is denoted by $\|\cdot\|$. We recall that a function $\xi : \mathbb{R}^n \to \mathbb{R}$ is said to be *strongly convex* with modulus $\kappa > 0$, if for all $(x, y) \in \mathbb{R}^n \times \mathbb{R}^n$ one has $\xi(y) \geq \xi(x) + \nabla\xi(x)^\top(y - x) + \kappa\|y - x\|^2$ (for other equivalent definitions, see, for example, [22, Chapter IV]). Our minimal assumptions are the following.

Assumption 2.1. (i) The functions f and $-c_{(i)}$ $(1 \leq i \leq m)$ are convex and differentiable from $\mathbb{R}^n$ to $\mathbb{R}$ and at least one of the functions $f, -c_{(1)}, \ldots, -c_{(m)}$ is strongly convex. (ii) The set of strictly feasible points for problem (1.1) is nonempty, i.e., there exists $x \in \mathbb{R}^n$ such that $c(x) > 0$.

Assumption 2.1(i) was motivated in section 1. Assumption 2.1(ii), also called the (strong) Slater condition, is necessary for the well-posedness of a *feasible* interior point method. With the convexity assumption, it is equivalent to the fact that the set of multipliers associated with a given solution is nonempty and compact (see [22, Theorem VII.2.3.2], for example). These assumptions have the following clear consequence.

LEMMA 2.2. *Suppose that Assumption* 2.1 *holds. Then, the solution set of problem* (1.1) *is nonempty and bounded.*

By Lemma 2.2, the level sets of the logarithmic barrier function φ_μ are compact, a fact that will be used frequently. It is a consequence of [16, Lemma 12], which we recall for completeness.

LEMMA 2.3. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex continuous function and $c : \mathbb{R}^n \to \mathbb{R}^m$ be a continuous function having concave components. Suppose that the set $\{x \in \mathbb{R}^n : c(x) > 0\}$ is nonempty and that the solution set of problem* (1.1) *is nonempty and bounded. Then, for any $\alpha \in \mathbb{R}$ and $\mu > 0$, the set*

$$\left\{ x \in \mathbb{R}^n : c(x) > 0,\ f(x) - \mu \sum_{i=1}^{m} \log c_{(i)}(x) \leq \alpha \right\}$$

is compact (*and possibly empty*).

Let x_1 be the first iterate of our *feasible* IP algorithm, hence satisfying $c(x_1) > 0$, and define the level set

$$\mathcal{L}_1^{\mathrm{p}} := \{x \in \mathbb{R}^n : c(x) > 0 \text{ and } \varphi_\mu(x) \leq \varphi_\mu(x_1)\}.$$

LEMMA 2.4. *Suppose that Assumption* 2.1 *holds. Then, the barrier problem* (1.4) *has a unique solution, which is denoted by $\hat{x}_\mu$.*

Proof. By Assumption 2.1, Lemma 2.2, and Lemma 2.3, $\mathcal{L}_1^{\mathrm{p}}$ is nonempty and compact, so that the barrier problem (1.4) has at least one solution. This solution is also unique, since φ_μ is strictly convex on $\{x \in \mathbb{R}^n : c(x) > 0\}$. Indeed, by Assumption 2.1(i), $\nabla^2\varphi_\mu(x)$ given by (1.6) is positive definite. ∎

To simplify the notation we denote by

$$z := (x, \lambda)$$

a typical pair of primal-dual variables and by $\mathcal{Z}$ the set of strictly feasible z's:

$$\mathcal{Z} := \{z = (x, \lambda) \in \mathbb{R}^n \times \mathbb{R}^m : (c(x), \lambda) > 0\}.$$

The algorithm generates a sequence of pairs (z, M), where $z \in \mathcal{Z}$ and M is a positive definite symmetric matrix. Given a pair (z, M), the next one (z_+, M_+) is obtained as follows. First

$$z_+ := z + \alpha d,$$

where $\alpha > 0$ is a step-size and $d = (d^x, d^\lambda)$ is the unique solution of (1.3). The uniqueness comes from the positivity of $c(x)$ and from the positive definiteness of M (for the unicity of d^x, use (1.5)). Next, the matrix M is updated into M_+ by the BFGS formula

$$M_+ := M - \frac{M\delta\delta^\top M}{\delta^\top M\delta} + \frac{\gamma\gamma^\top}{\gamma^\top\delta}, \tag{2.1}$$

where γ and δ are given by

$$\delta := x_+ - x \quad \text{and} \quad \gamma := \nabla_x \ell(x_+, \lambda_+) - \nabla_x \ell(x, \lambda_+). \tag{2.2}$$

This formula gives a symmetric positive definite matrix M_+, provided M is symmetric positive definite and $\gamma^\top\delta > 0$ (see [14, 17]). This latter condition is satisfied because of the strong convexity assumption. Indeed, since at least one of the functions f or $-c_{(i)}$ is strongly convex, for any fixed $\lambda > 0$, the function $x \mapsto \ell(x, \lambda)$ is strongly convex, that is, there exists a constant $\kappa > 0$ such that

$$2\kappa\|x - x'\|^2 \leq \left(\nabla_x \ell(x, \lambda) - \nabla_x \ell(x', \lambda)\right)^\top (x - x') \quad \text{for all } x \text{ and } x'.$$

Since α sizes the displacement in x and λ, the merit function used to estimate the progress to the solution must depend on both x and λ. We follow an idea of Anstreicher and Vial [1] and add to φ_μ a function forcing λ to take the value $\mu C(x)^{-1}e$. The merit function is defined for $z = (x, \lambda) \in \mathcal{Z}$ by

$$\psi_\mu(z) := \varphi_\mu(x) + \mathcal{V}(z),$$

where

$$\mathcal{V}(z) = \lambda^\top c(x) - \mu \sum_{i=1}^{m} \log\left(\lambda_{(i)} c_{(i)}(x)\right).$$

Note that

$$\nabla\psi_\mu(z) = \begin{pmatrix} \nabla f(x) - 2\mu\nabla c(x) C(x)^{-1}e + \nabla c(x)\lambda \\ c(x) - \mu\Lambda^{-1}e \end{pmatrix}. \tag{2.3}$$

Using ψ_μ as a merit function is reasonable provided the problem

$$\begin{cases} \min \psi_\mu(z), \\ z \in \mathcal{Z} \end{cases} \tag{2.4}$$

has for unique solution the solution of (1.2) and the direction $d = (d^x, d^\lambda)$ is a descent direction of ψ_μ. This is what we check in Lemmas 2.5 and 2.6 below.

LEMMA 2.5. *Suppose that Assumption* 2.1 *holds. Then, problem* (2.4) *has a unique solution* $\hat{z}_\mu := (\hat{x}_\mu, \hat{\lambda}_\mu)$, *where* $\hat{x}_\mu$ *is the unique solution of the barrier problem* (1.4) *and* $\hat{\lambda}_\mu$ *has its* i*th component defined by* $(\hat{\lambda}_\mu)_{(i)} := \mu / c_{(i)}(\hat{x}_\mu)$. *Furthermore,* ψ_μ *has no other stationary point than* $\hat{z}_\mu$.

Proof. By optimality of the unique solution $\hat{x}_\mu$ of the barrier problem (1.4)

$$\varphi_\mu(\hat{x}_\mu) \leq \varphi_\mu(x) \quad \text{for any } x \text{ such that } c(x) > 0.$$

On the other hand, since $t \to t - \mu \log t$ is minimized at $t = \mu$ and since $\mu = c_{(i)}(\hat{x}_\mu)(\hat{\lambda}_\mu)_{(i)}$ for all index i, we have

$$\mathcal{V}(\hat{z}_\mu) \leq \mathcal{V}(z) \quad \text{for any } z \in \mathcal{Z}.$$

Adding up the preceding two inequalities gives $\psi_\mu(\hat{z}_\mu) \leq \psi_\mu(z)$ for all $z \in \mathcal{Z}$. Hence $\hat{z}_\mu$ is a solution of (2.4).

It remains to show that $\hat{z}_\mu$ is the unique stationary point of ψ_μ. If z is stationary, it satisfies

$$\left\{ \begin{array}{rcl} \nabla f(x) - 2\mu \nabla c(x)\, C(x)^{-1} e + \nabla c(x)\, \lambda & = & 0, \\ c(x) - \mu \Lambda^{-1} e & = & 0. \end{array} \right.$$

Canceling λ from the first equality gives $\nabla f(x) - \mu \nabla c(x) C(x)^{-1} e = 0$, and thus $x = \hat{x}_\mu$ is the unique minimizer of the convex function φ_μ. Now, $\lambda = \hat{\lambda}_\mu$ by the second equation of the system above. □

LEMMA 2.6. *Suppose that* $z \in \mathcal{Z}$ *and that* M *is symmetric positive definite. Let* $d = (d^x, d^\lambda)$ *be the solution of* (1.3). *Then*

$$\nabla \psi_\mu(z)^\top d = -(d^x)^\top (M + \nabla c(x) \Lambda C(x)^{-1} \nabla c(x)^\top) d^x - \|C(x)^{-1/2} \Lambda^{-1/2} (C(x)\lambda - \mu e)\|^2,$$

so that d *is a descent direction of* ψ_μ *at a point* $z \neq \hat{z}_\mu$, *meaning that* $\nabla \psi_\mu(z)^\top d < 0$.

Proof. We have $\nabla \psi_\mu(z)^\top d = \nabla \varphi_\mu(x)^\top d^x + \nabla \mathcal{V}(z)^\top d$. Using (1.5),

$$\nabla \varphi_\mu(x)^\top d^x = -(d^x)^\top (M + \nabla c(x) C(x)^{-1} \Lambda \nabla c(x)^\top) d^x,$$

which is nonpositive. On the other hand, when d satisfies the second equation of (1.3), one has (see [1])

$$\begin{aligned} \nabla \mathcal{V}(z)^\top d &= (\nabla c(x)\lambda - \mu \nabla c(x)\, C(x)^{-1} e)^\top d^x + (c(x) - \mu \Lambda^{-1} e)^\top d^\lambda \\ &= (e - \mu C(x)^{-1} \Lambda^{-1} e)^\top (\Lambda \nabla c(x)^\top d^x + C(x) d^\lambda) \\ &= -(\mu e - C(x)\lambda)^\top C(x)^{-1} \Lambda^{-1} (\mu e - C(x)\lambda) \\ &= -\|C(x)^{-1/2} \Lambda^{-1/2} (C(x)\lambda - \mu e)\|^2, \end{aligned}$$

which is also nonpositive. The formula for $\nabla \psi_\mu(z)^\top d$ given in the statement of the lemma follows from this calculation. Furthermore, $\nabla \psi_\mu(z)^\top d < 0$, if $z \neq \hat{z}_\mu$. □

We can now state precisely one iteration of the algorithm used to solve the perturbed KKT system (1.2). The constants $\omega \in]0, 1[$ and $0 < \tau < \tau' < 1$ are given independently of the iteration index.

ALGORITHM A_μ (for solving (1.2); one iteration).

0. At the beginning of the iteration, the current iterate $z = (x, \lambda) \in \mathcal{Z}$ is supposed available, as well as a positive definite matrix M approximating the Hessian of the Lagrangian $\nabla^2_{xx}\ell(x, \lambda)$.
1. Compute $d := (d^x, d^\lambda)$, the solution of the linear system (1.3).
2. Compute a step-size α by means of a backtracking line search.
   2.0. Set $\alpha = 1$.
   2.1. Test the *sufficient decrease condition*:

$$\psi_\mu(z + \alpha d) \le \psi_\mu(z) + \omega\alpha\nabla\psi_\mu(z)^\top d. \tag{2.5}$$

   2.2. If (2.5) is not satisfied, choose a new trial step-size α in $[\tau\alpha, \tau'\alpha]$ and go to Step 2.1. If (2.5) is satisfied, set $z_+ := z + \alpha d$.
3. Update M by the BFGS formula (2.1) where γ and δ are given by (2.2).

By Lemma 2.6, d is a descent direction of ψ_μ at z, so that a step-size $\alpha > 0$ satisfying (2.5) can be found. In the line-search, it is implicitly assumed that (2.5) is not satisfied if $z + \alpha d \notin \mathcal{Z}$, so that $(c(x_+), \lambda_+) > 0$ holds for the new iterate z_+.

We conclude this section with a result that gives the contribution of the line-search to the convergence of the sequence generated by Algorithm A_μ. It is in the spirit of a similar result given by Zoutendijk [50] (for a proof, see [6]). We say that a function is $C^{1,1}$ if it has Lipschitz continuous first derivatives. We denote the level set of ψ_μ determined by the first iterate $z_1 = (x_1, \lambda_1) \in \mathcal{Z}$ by

$$\mathcal{L}_1^{\text{PD}} := \{z \in \mathcal{Z} : \psi_\mu(z) \le \psi_\mu(z_1)\}.$$

LEMMA 2.7. *If ψ_μ is $C^{1,1}$ on an open convex neighborhood of the level set $\mathcal{L}_1^{\text{PD}}$, there is a positive constant K such that for any $z \in \mathcal{L}_1^{\text{PD}}$, if α is determined by the line-search in Step 2 of Algorithm A_μ, one of the following two inequalities holds:*

$$\psi_\mu(z + \alpha d) \le \psi_\mu(z) - K|\nabla\psi_\mu(z)^\top d|,$$

$$\psi_\mu(z + \alpha d) \le \psi_\mu(z) - K\frac{|\nabla\psi_\mu(z)^\top d|^2}{\|d\|^2}.$$

It is important to mention here that this result holds even though ψ_μ may not be defined for all positive step-sizes along d, so that the line-search may have to reduce the step-size in a first stage to enforce feasibility.

3. The global and *r*-linear convergence of Algorithm A_μ. In the convergence analysis of BFGS, the path to q-superlinear convergence traditionally leads through r-linear convergence (see [41, 6]). In this section, we show that the iterates generated by Algorithm A_μ converge to $\hat{z}_\mu = (\hat{x}_\mu, \hat{\lambda}_\mu)$, the solution of (1.2), with that convergence speed. We use the notation

$$\hat{C}_\mu := \text{diag}(c_{(1)}(\hat{x}_\mu), \ldots, c_{(m)}(\hat{x}_\mu)) \quad \text{and} \quad \hat{\Lambda}_\mu := \text{diag}((\hat{\lambda}_\mu)_{(1)}, \ldots, (\hat{\lambda}_\mu)_{(m)}).$$

Our first result shows that, because the iterates (x, λ) remain in the level set $\mathcal{L}_1^{\text{PD}}$, the sequence $\{(c(x), \lambda)\}$ is bounded and bounded away from zero.

LEMMA 3.1. *Suppose that Assumption* 2.1 *holds. Then, the level set* $\mathcal{L}_1^{\rm PD}$ *is compact and there exist positive constants* K_1 *and* K_2 *such that*

$$K_1 \le (c(x), \lambda) \le K_2 \quad \textit{for all } z \in \mathcal{L}_1^{\rm PD}.$$

Proof. Since $\lambda^\top c(x) - \mu \sum_i \log(\lambda_{(i)} c_{(i)}(x))$ is bounded below by $m\mu(1 - \log \mu)$, there is a constant $K_1' > 0$ such that $\varphi_\mu(x) \le K_1'$ for all $z = (x, \lambda) \in \mathcal{L}_1^{\rm PD}$. By Assumption 2.1 and Lemma 2.3, the level set $\mathcal{L}' := \{x : c(x) > 0,\ \varphi_\mu(x) \le K_1'\}$ is compact. By continuity, $c(\mathcal{L}')$ is also compact, so that $c(x)$ is bounded and bounded away from zero for all $z \in \mathcal{L}_1^{\rm PD}$.

What we have just proven implies that $\{\varphi_\mu(x) : z = (x, \lambda) \in \mathcal{L}_1^{\rm PD}\}$ is bounded below, so that there is a constant $K_2' > 0$ such that $\lambda^\top c(x) - \mu \sum_i \log(\lambda_{(i)} c_{(i)}(x)) \le K_2'$ for all $z = (x, \lambda) \in \mathcal{L}_1^{\rm PD}$. Hence the λ-components of the z's in $\mathcal{L}_1^{\rm PD}$ are bounded and bounded away from zero.

We have shown that $\mathcal{L}_1^{\rm PD}$ is included in a compact set. Now, it is itself compact by continuity of ψ_μ. ∎

The next proposition is crucial for the technique we use to prove global convergence (see [6]). It claims that the proximity of a point z to the unique solution of (2.4) can be measured by the value of $\psi_\mu(z)$ or the norm of its gradient $\nabla\psi_\mu(z)$. In unconstrained optimization, the corresponding result is a direct consequence of strong convexity. Here, ψ_μ is not necessarily convex, but the result can still be established by using Lemma 2.5 and Lemma 3.1. The function ψ_μ is nonconvex, for example, when $f(x) = x^2$ is minimized on the half-line of nonnegative real numbers.

PROPOSITION 3.2. *Suppose that Assumption* 2.1 *holds. Then, there is a constant* $a > 0$ *such that for any* $z \in \mathcal{L}_1^{\rm PD}$

$$a\|z - \hat{z}_\mu\|^2 \le \psi_\mu(z) - \psi_\mu(\hat{z}_\mu) \le \frac{1}{a}\|\nabla\psi_\mu(z)\|^2. \tag{3.1}$$

Proof. Let us show that ψ_μ is strongly convex in a neighborhood of $\hat{z}_\mu$. Using (2.3) and the fact that $\hat{C}_\mu \hat{\lambda}_\mu = \mu e$, the Hessian of ψ_μ at $\hat{z}_\mu$ can be written as

$$\nabla^2\psi_\mu(\hat{z}_\mu) = \begin{pmatrix} \nabla^2_{xx}\ell(\hat{x}_\mu, \hat{\lambda}_\mu) + 2\mu\nabla c(\hat{x}_\mu)\, \hat{C}_\mu^{-2} \nabla c(\hat{x}_\mu)^\top & \nabla c(\hat{x}_\mu) \\ \nabla c(\hat{x}_\mu)^\top & \frac{1}{\mu}\hat{C}_\mu^2 \end{pmatrix}.$$

From Assumption 2.1, for fixed $\lambda > 0$, the Lagrangian is a strongly convex function in the variable x. It follows that its Hessian with respect to x is positive definite at $(\hat{x}_\mu, \hat{\lambda}_\mu)$. Let us show that the above matrix is also positive definite. Multiplying the matrix on both sides by a vector $(u, v) \in \mathbb{R}^n \times \mathbb{R}^m$ gives

$$\begin{aligned} &u^\top \nabla^2_{xx}\ell(\hat{x}_\mu, \hat{\lambda}_\mu)u + 2\mu u^\top \nabla c(\hat{x}_\mu)\, \hat{C}_\mu^{-2} \nabla c(\hat{x}_\mu)^\top u + 2u^\top \nabla c(\hat{x}_\mu) v + \frac{1}{\mu} v^\top \hat{C}_\mu^2 v \\ &= u^\top \nabla^2_{xx}\ell(\hat{x}_\mu, \hat{\lambda}_\mu)u + \mu u^\top \nabla c(\hat{x}_\mu) \hat{C}_\mu^{-2} \nabla c(\hat{x}_\mu)^\top u + \|\mu^{1/2} \hat{C}_\mu^{-1} \nabla c(\hat{x}_\mu)^\top u + \mu^{-1/2} \hat{C}_\mu v\|^2. \end{aligned}$$

Since $\nabla^2_{xx}\ell(\hat{x}_\mu, \hat{\lambda}_\mu)$ is positive definite and $c(\hat{x}_\mu) > 0$, this quantity is nonnegative. If it vanishes, one deduces that $u = 0$ and next that $v = 0$. Hence $\nabla^2\psi_\mu(\hat{z}_\mu)$ is positive definite.

Let us now prove a local version of the proposition: there exist a constant $a' > 0$ and an *open* neighborhood $\mathcal{N} \subset \mathcal{Z}$ of $\hat{z}_\mu$ such that

$$a'\|z - \hat{z}_\mu\|^2 \le \psi_\mu(z) - \psi_\mu(\hat{z}_\mu) \le \frac{1}{a'}\|\nabla\psi_\mu(z)\|^2 \quad \text{for all } z \in \mathcal{N}. \tag{3.2}$$

The inequality on the left comes from the fact that $\nabla\psi_\mu(\hat{z}_\mu) = 0$ and the strong convexity of ψ_μ near $\hat{z}_\mu$. For the inequality on the right, we first use the local convexity of ψ_μ: for an arbitrary z near $\hat{z}_\mu$, $\psi_\mu(\hat{z}_\mu) \geq \psi_\mu(z) + \nabla\psi_\mu(z)^\top(\hat{z}_\mu - z)$. With the Cauchy–Schwarz inequality and the inequality on the left of (3.2), one gets

$$\psi_\mu(z) - \psi_\mu(\hat{z}_\mu) \leq \|\nabla\psi_\mu(z)\| \left(\frac{\psi_\mu(z) - \psi_\mu(\hat{z}_\mu)}{a'}\right)^{\frac{1}{2}}.$$

Simplifying and squaring give the inequality on the right of (3.2).

To extend the validity of (3.2) for all $z \in \mathcal{L}_1^{\text{PD}}$, it suffices to note that, by virtue of Lemma 2.5, the ratios

$$\frac{\psi_\mu(z) - \psi_\mu(\hat{z}_\mu)}{\|z - \hat{z}_\mu\|^2} \quad \text{and} \quad \frac{\psi_\mu(z) - \psi_\mu(\hat{z}_\mu)}{\|\nabla\psi_\mu(z)\|^2}$$

are well defined and continuous on the compact set $\mathcal{L}_1^{\text{PD}} \setminus \mathcal{N}$. Since $\hat{z}_\mu$ is the unique minimizer of ψ_μ on $\mathcal{L}_1^{\text{PD}}$ (Lemma 2.5), the ratios are respectively bounded away from zero and bounded above on $\mathcal{L}_1^{\text{PD}} \setminus \mathcal{N}$, by some positive constants K_1' and K_2'. The conclusion of the proposition now follows by taking $a = \min(a', K_1', 1/K_2')$. ∎

The proof of the r-linear convergence rests on the following lemma, which is part of the theory of BFGS updates. It can be stated independently of the present context (see Byrd and Nocedal [6]). We denote by θ_k the angle between $M_k\delta_k$ and δ_k:

$$\cos\theta_k := \frac{\delta_k^\top M_k \delta_k}{\|M_k\delta_k\|\,\|\delta_k\|}$$

and by $\lceil\cdot\rceil$ the roundup operator: $\lceil x\rceil = i$ when $i - 1 < x \leq i$ and $i \in \mathbb{N}$.

LEMMA 3.3. *Let $\{M_k\}$ be positive definite matrices generated by the BFGS formula using pairs of vectors $\{(\gamma_k, \delta_k)\}_{k\geq 1}$, satisfying for all $k \geq 1$*

$$\gamma_k^\top\delta_k \geq a_1\|\delta_k\|^2 \quad \textit{and} \quad \gamma_k^\top\delta_k \geq a_2\|\gamma_k\|^2, \tag{3.3}$$

where $a_1 > 0$ and $a_2 > 0$ are independent of k. Then, for any $r \in]0,1[$, there exist positive constants b_1, b_2, and b_3, such that for any index $k \geq 1$,

$$b_1 \leq \cos\theta_j \quad \textit{and} \quad b_2 \leq \frac{\|M_j\delta_j\|}{\|\delta_j\|} \leq b_3 \tag{3.4}$$

for at least $\lceil rk\rceil$ indices j in $\{1,\ldots,k\}$.

The assumptions (3.3) made on γ_k and δ_k in the above lemma are satisfied in our context. The first one is due to the strong convexity of one of the functions f, $-c_{(1)}$, …, $-c_{(m)}$, and to the fact that λ is bounded away from zero (Lemma 3.1). When f and c are $C^{1,1}$, the second one can be deduced from the Lipschitz inequality, the boundedness of λ (Lemma 3.1), and the first inequality in (3.3).

THEOREM 3.4. *Suppose that Assumption 2.1 holds and that f and c are $C^{1,1}$ functions. Then, Algorithm A_μ generates a sequence $\{z_k\}$ converging to $\hat{z}_\mu$ r-linearly, meaning that $\limsup_{k\to\infty}\|z_k - \hat{z}_\mu\|^{1/k} < 1$. In particular,*

$$\sum_{k\geq 1}\|z_k - \hat{z}_\mu\| < \infty.$$

Proof. We denote by K_1', K_2', … positive constants (independent of the iteration index). We also use the notation

$$c_j := c(x_j) \quad \text{and} \quad C_j := \operatorname{diag}(c_{(1)}(x_j),\ldots,c_{(m)}(x_j)).$$

The bounds on $(c(x), \lambda)$ given by Lemma 3.1 and the fact that f and c are $C^{1,1}$ imply that ψ_μ is $C^{1,1}$ on some open convex neighborhood of the level set $\mathcal{L}_1^{\text{PD}}$, for example, on

$$\left(c^{-1}\left(\left]\frac{K_1}{2}, +\infty\right[^m\right) \times \left]\frac{K_1}{2}, 2K_2\right[^m\right) \cap \mathcal{O},$$

where $\mathcal{O}$ is an open bounded convex set containing $\mathcal{L}_1^{\text{PD}}$ (this set $\mathcal{O}$ is used to have ∇c bounded on the given neighborhood).

Therefore, by the line-search and Lemma 2.7, there is a positive constant K_1' such that either

$$\psi_\mu(z_{k+1}) \leq \psi_\mu(z_k) - K_1' |\nabla \psi_\mu(z_k)^\top d_k| \tag{3.5}$$

or

$$\psi_\mu(z_{k+1}) \leq \psi_\mu(z_k) - K_1' \frac{|\nabla \psi_\mu(z_k)^\top d_k|^2}{\|d_k\|^2}. \tag{3.6}$$

Let us now apply Lemma 3.3: fix $r \in]0,1[$ and denote by J the set of indices j for which (3.4) holds. Using Lemma 2.6 and the bounds from Lemma 3.1, one has for $j \in J$

$$\begin{aligned}
|\nabla \psi_\mu(z_j)^\top d_j| &= (d_j^x)^\top (M_j + \nabla c_j \Lambda_j C_j^{-1} \nabla c_j^\top) d_j^x + \|C_j^{-1/2} \Lambda_j^{-1/2} (C_j \lambda_j - \mu e)\|^2 \\
&\geq (d_j^x)^\top M_j d_j^x + K_2^{-2} \|C_j \lambda_j - \mu e\|^2 \\
&\geq \frac{b_1}{b_3} \|M_j d_j^x\|^2 + K_2^{-2} \|C_j \lambda_j - \mu e\|^2 \\
&\geq K_2' \left(\|M_j d_j^x\|^2 + \|C_j \lambda_j - \mu e\|^2\right).
\end{aligned}$$

Let us denote by K_4' a positive constant such that $\|\nabla c(x)\| \leq K_4'$ for all $x \in \mathcal{L}_1^{\text{PD}}$. By using (2.3), (1.5), and the inequality $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$, we obtain

$$\begin{aligned}
&\|\nabla \psi_\mu(z_j)\|^2 \\
&= \|\nabla_x \psi_\mu(z_j)\|^2 + \|\nabla_\lambda \psi_\mu(z_j)\|^2 \\
&= \| - (M_j + \nabla c_j C_j^{-1} \Lambda_j \nabla c_j^\top) d_j^x \; + \; \nabla c_j (\lambda_j - \mu C_j^{-1} e)\|^2 + \|c_j - \mu \Lambda_j^{-1} e\|^2 \\
&\leq \left(\|M_j d_j^x\| + K_1^{-1} K_2 {K_4'}^2 \|d_j^x\| + K_1^{-1} K_4' \|C_j \lambda_j - \mu e\|\right)^2 + K_1^{-2} \|C_j \lambda_j - \mu e\|^2 \\
&\leq 3 \left(1 + \frac{K_1^{-2} K_2^2 {K_4'}^4}{b_2^2}\right) \|M_j d_j^x\|^2 + K_1^{-2} (3 {K_4'}^2 + 1) \|C_j \lambda_j - \mu e\|^2 \\
&\leq K_3' \left(\|M_j d_j^x\|^2 + \|C_j \lambda_j - \mu e\|^2\right)
\end{aligned}$$

and also, by (1.3),

$$\begin{aligned}
\|d_j\|^2 &= \|d_j^x\|^2 + \|d_j^\lambda\|^2 \\
&= \|d_j^x\|^2 + \|\mu C_j^{-1} e - \lambda_j - C_j^{-1} \Lambda_j \nabla c_j^\top d_j^x\|^2 \\
&\leq \|d_j^x\|^2 + 2\|C_j^{-1} \Lambda_j \nabla c_j^\top d_j^x\|^2 + 2\|C_j^{-1} (C_j \lambda_j - \mu e)\|^2 \\
&\leq \frac{1 + 2K_1^{-2} K_2^2 {K_4'}^2}{b_2^2} \|M_j d_j^x\|^2 + 2K_1^{-2} \|C_j \lambda_j - \mu e\|^2 \\
&\leq K_5' \left(\|M_j d_j^x\|^2 + \|C_j \lambda_j - \mu e\|^2\right).
\end{aligned}$$

Combining these inequalities with (3.5) or (3.6) gives for some positive constant K_6' and for any $j \in J$

$$\psi_\mu(z_{j+1}) \leq \psi_\mu(z_j) - K_6' \|\nabla \psi_\mu(z_j)\|^2.$$

The end of the proof is standard (see [41, 6]). Using Proposition 3.2, for $j \in J$,

$$\begin{aligned}\psi_\mu(z_{j+1}) - \psi_\mu(\hat{z}_\mu) &\leq \psi_\mu(z_j) - \psi_\mu(\hat{z}_\mu) - K_6' \|\nabla \psi_\mu(z_j)\|^2 \\ &\leq \tau^{\frac{1}{r}} (\psi_\mu(z_j) - \psi_\mu(\hat{z}_\mu)),\end{aligned}$$

where $\tau := (1 - K_6' a)^r \in [0, 1[$. On the other hand, by the line-search, $\psi_\mu(z_{k+1}) - \psi_\mu(\hat{z}_\mu) \leq \psi_\mu(z_k) - \psi_\mu(\hat{z}_\mu)$ for any $k \geq 1$. By Lemma 3.3, $|[1, k] \cap J| \geq \lceil rk \rceil \geq rk$, so that the last inequality gives for any $k \geq 1$

$$\psi_\mu(z_{k+1}) - \psi_\mu(\hat{z}_\mu) \leq K_7' \tau^k,$$

where K_7' is the positive constant $(\psi_\mu(z_1) - \psi_\mu(\hat{z}_\mu))$. Now, using the inequality on the left in (3.1), one has for all $k \geq 1$

$$\|z_{k+1} - \hat{z}_\mu\| \leq \frac{1}{\sqrt{a}} (\psi_\mu(z_{k+1}) - \psi_\mu(\hat{z}_\mu))^{\frac{1}{2}} \leq \left(\frac{K_7'}{a}\right)^{\frac{1}{2}} \tau^{\frac{k}{2}},$$

from which the r-linear convergence of $\{z_k\}$ follows. □

4. The q-superlinear convergence of Algorithm $\mathbf{A}_\mu$. With the r-linear convergence result of the previous section, we are now ready to establish the q-superlinear convergence of the sequence $\{z_k\}$ generated by Algorithm A_μ. By definition, $\{z_k\}$ converges q-superlinearly to $\hat{z}_\mu$ if the following estimate holds:

$$z_{k+1} - \hat{z}_\mu = o(\|z_k - \hat{z}_\mu\|),$$

which means that $\|z_{k+1} - \hat{z}_\mu\| / \|z_k - \hat{z}_\mu\| \to 0$ (assuming $z_k \neq \hat{z}_\mu$). To get this result, f and c have to be a little bit smoother, namely twice continuously differentiable near $\hat{x}_\mu$. We use the notation

$$\hat{M}_\mu := \nabla^2_{xx} \ell(\hat{x}_\mu, \hat{\lambda}_\mu).$$

We start by showing that the unit step-size is accepted asymptotically by the line-search condition (2.5), provided the updated matrix M_k becomes good (or sufficiently large) in a sense specified by inequality (4.1) below and provided the iterate z_k is sufficiently close to the solution $\hat{z}_\mu$.

Given two sequences of vectors $\{u_k\}$ and $\{v_k\}$ in some normed spaces and a positive number β, we write $u_k \geq o(\|v_k\|^\beta)$, if there exists a sequence of $\{\epsilon_k\} \subset \mathbb{R}$ such that $\epsilon_k \to 0$ and $u_k \geq \epsilon_k \|v_k\|^\beta$ for all k.

PROPOSITION 4.1. *Suppose that Assumption* 2.1 *holds and that f and c are twice continuously differentiable near $\hat{x}_\mu$. Suppose also that the sequence $\{z_k\}$ generated by Algorithm A_μ converges to $\hat{z}_\mu$ and that the positive definite matrices M_k satisfy the estimate*

$$(d_k^x)^\top \left(M_k - \hat{M}_\mu\right) d_k^x \geq o(\|d_k^x\|^2) \tag{4.1}$$

when $k \to \infty$. Then the sufficient decrease condition (2.5) *is satisfied with $\alpha_k = 1$ for k sufficiently large provided that $\omega < \frac{1}{2}$.*

Proof. Observe first that the positive definiteness of $\hat{M}_\mu$ with (4.1) implies that

$$(d_k^x)^\top M_k d_k^x \geq K' \|d_k^x\|^2 \tag{4.2}$$

for some positive constant K' and sufficiently large k. Observe also that $d_k \to 0$ (for $d_k^x \to 0$, use (1.5), (4.2), and $\nabla\varphi_\mu(x_k) \to 0$). Therefore, for k large enough, z_k and $z_k + d_k$ are near $\hat{z}_\mu$ and one can expand $\psi_\mu(z_k + d_k)$ about z_k. A second order expansion gives for the left-hand side of (2.5)

$$\begin{aligned}
&\psi_\mu(z_k + d_k) - \psi_\mu(z_k) - \omega\nabla\psi_\mu(z_k)^\top d_k \\
&= (1-\omega)\nabla\psi_\mu(z_k)^\top d_k + \frac{1}{2} d_k^\top \nabla^2\psi_\mu(z_k) d_k + o(\|d_k\|^2) \\
&= \left(\frac{1}{2} - \omega\right) \nabla\psi_\mu(z_k)^\top d_k \\
&\quad + \frac{1}{2}\left(\nabla\psi_\mu(z_k)^\top d_k + d_k^\top \nabla^2\psi_\mu(z_k) d_k\right) + o(\|d_k\|^2).
\end{aligned} \tag{4.3}$$

We want to show that this quantity is negative for k large.

Our first aim is to show that $\left(\nabla\psi_\mu(z_k)^\top d_k + d_k^\top \nabla^2\psi_\mu(z_k) d_k\right)$ is smaller than a term of order $o(\|d_k\|^2)$. For this purpose, one computes

$$\begin{aligned}
&d_k^\top \nabla^2\psi_\mu(z_k) d_k \\
&= (d_k^x)^\top \nabla^2_{xx}\ell(x_k, \tilde{\lambda}_k) d_k^x + 2\mu (d_k^x)^\top \nabla c_k C_k^{-2} \nabla c_k^\top d_k^x \\
&\quad + 2(d_k^x)^\top \nabla c_k d_k^\lambda + \mu (d_k^\lambda)^\top \Lambda_k^{-2} d_k^\lambda,
\end{aligned}$$

where $\tilde{\lambda}_k = 2\mu C_k^{-1} e - \lambda_k$. On the other hand, using

$$C_k^{-1/2}\Lambda_k^{-1/2}(C_k\lambda_k - \mu e) = -C_k^{-1/2}\Lambda_k^{1/2}\nabla c_k^\top d_k^x - C_k^{1/2}\Lambda_k^{-1/2} d_k^\lambda,$$

one gets from Lemma 2.6

$$\begin{aligned}
&\nabla\psi_\mu(z_k)^\top d_k \\
&= -(d_k^x)^\top M_k d_k^x - (d_k^x)^\top \nabla c_k C_k^{-1}\Lambda_k \nabla c_k^\top d_k^x - \|C_k^{-1/2}\Lambda_k^{-1/2}(C_k\lambda_k - \mu e)\|^2 \\
&= -(d_k^x)^\top M_k d_k^x - 2(d_k^x)^\top \nabla c_k C_k^{-1}\Lambda_k \nabla c_k^\top d_k^x - 2(d_k^x)^\top \nabla c_k d_k^\lambda - (d_k^\lambda)^\top C_k \Lambda_k^{-1} d_k^\lambda.
\end{aligned}$$

With these estimates, (4.1), and the fact that $\nabla^2_{xx}\ell(x_k, \tilde{\lambda}_k) \to \hat{M}_\mu$ and $C_k\lambda_k \to \mu e$, with Lemma 3.1 and the boundedness of $\{\nabla c_k\}$, (4.3) becomes

$$\begin{aligned}
&\psi_\mu(z_k + d_k) - \psi_\mu(z_k) - \omega\nabla\psi_\mu(z_k)^\top d_k \\
&= \left(\frac{1}{2} - \omega\right) \nabla\psi_\mu(z_k)^\top d_k \\
&\quad - \frac{1}{2}(d_k^x)^\top \left(M_k - \nabla^2_{xx}\ell(x_k, \tilde{\lambda}_k)\right) d_k^x + (d_k^x)^\top \nabla c_k \left(\mu C_k^{-2} - C_k^{-1}\Lambda_k\right) \nabla c_k^\top d_k^x \\
&\quad + \frac{1}{2}(d_k^\lambda)^\top \left(\mu\Lambda_k^{-2} - C_k\Lambda_k^{-1}\right) d_k^\lambda + o(\|d_k\|^2) \\
&\leq \left(\frac{1}{2} - \omega\right) \nabla\psi_\mu(z_k)^\top d_k + o(\|d_k\|^2).
\end{aligned} \tag{4.4}$$

Since $\omega < \frac{1}{2}$, it is clear that the result will be proven if we show that, for some positive constant K and k large, $\nabla\psi_\mu(z_k)^\top d_k \leq -K\|d_k\|^2$. To show this, we use the

last expression of $\nabla\psi_\mu(z_k)^\top d_k$ and an upper bound of $|(d_k^x)^\top \nabla c_k d_k^\lambda|$, obtained by the Cauchy–Schwartz inequality:

$$\begin{aligned} 2\left|(d_k^x)^\top \nabla c_k d_k^\lambda\right| &= 2\left|\left(C_k^{-1/2}\Lambda_k^{1/2}\nabla c_k^\top d_k^x\right)^\top \left(C_k^{1/2}\Lambda_k^{-1/2} d_k^\lambda\right)\right| \\ &\le 2\left\|C_k^{-1/2}\Lambda_k^{1/2}\nabla c_k^\top d_k^x\right\| \left\|C_k^{1/2}\Lambda_k^{-1/2} d_k^\lambda\right\| \\ &\le \frac{3}{2}(d_k^x)^\top \nabla c_k C_k^{-1}\Lambda_k \nabla c_k^\top d_k^x + \frac{2}{3}(d_k^\lambda)^\top C_k \Lambda_k^{-1} d_k^\lambda. \end{aligned}$$

It follows that

$$\nabla\psi_\mu(z_k)^\top d_k \le -(d_k^x)^\top M_k d_k^x - \frac{1}{2}(d_k^x)^\top \nabla c_k C_k^{-1}\Lambda_k \nabla c_k^\top d_k^x - \frac{1}{3}(d_k^\lambda)^\top C_k \Lambda_k^{-1} d_k^\lambda.$$

Therefore, using (4.2) and Lemma 3.1, one gets

$$\nabla\psi_\mu(z_k)^\top d_k \le -K\|d_k\|^2$$

for some positive constant K and k large. □

Proposition 4.1 shows in particular that the function $\mathcal{V}$, which was added to φ_μ to get the merit function ψ_μ, has the right curvature around $\hat{z}_\mu$, so that the unit step-size in both x and λ is accepted by the line-search.

In the following proposition, we establish a necessary and sufficient condition of q-superlinear convergence of the Dennis and Moré [13] type. The analysis assumes that the unit step-size is taken and that the updated matrix M_k is sufficiently good asymptotically in a manner given by the estimate (4.5), which is slightly different from (4.1).

PROPOSITION 4.2. *Suppose that Assumption* 2.1 *holds and that f and c are twice differentiable at $\hat{x}_\mu$. Suppose that the sequence $\{z_k\}$ generated by Algorithm A_μ converges to $\hat{z}_\mu$ and that, for k sufficiently large, the unit step-size $\alpha_k = 1$ is accepted by the line-search. Then $\{z_k\}$ converges q-superlinearly towards $\hat{z}_\mu$ if and only if*

$$(M_k - \hat{M}_\mu)d_k^x = o(\|d_k\|). \tag{4.5}$$

Proof. Let us denote by $\mathcal{M}$ the nonsingular Jacobian matrix of the perturbed KKT conditions (1.2) at the solution $\hat{z}_\mu = (\hat{x}_\mu, \hat{\lambda}_\mu)$:

$$\mathcal{M} = \begin{pmatrix} \hat{M}_\mu & -\nabla c(\hat{x}_\mu) \\ \hat{\Lambda}_\mu \nabla c(\hat{x}_\mu)^\top & \hat{C}_\mu \end{pmatrix}.$$

A first order expansion of the right-hand side of (1.3) about $\hat{z}_\mu$ and the identities $\nabla f(\hat{x}_\mu) = \nabla c(\hat{x}_\mu)\hat{\lambda}_\mu$ and $\hat{C}_\mu\hat{\lambda}_\mu = \mu e$ give

$$\begin{pmatrix} M_k & -\nabla c_k \\ \Lambda_k \nabla c_k^\top & C_k \end{pmatrix} \begin{pmatrix} d_k^x \\ d_k^\lambda \end{pmatrix} = -\mathcal{M}(z_k - \hat{z}_\mu) + o(\|z_k - \hat{z}_\mu\|).$$

Subtracting $\mathcal{M}d_k$ from both sides and assuming a unit step-size, we obtain

$$\begin{aligned} &\begin{pmatrix} M_k - \hat{M}_\mu & -(\nabla c_k - \nabla c(\hat{x}_\mu)) \\ \Lambda_k \nabla c_k^\top - \hat{\Lambda}_\mu \nabla c(\hat{x}_\mu)^\top & C_k - \hat{C}_\mu \end{pmatrix} \begin{pmatrix} d_k^x \\ d_k^\lambda \end{pmatrix} \\ &\qquad = -\mathcal{M}(z_{k+1} - \hat{z}_\mu) + o(\|z_k - \hat{z}_\mu\|). \end{aligned} \tag{4.6}$$

Suppose now that $\{z_k\}$ converges q-superlinearly. Then, the right-hand side of (4.6) is of order $o(\|z_k - \hat{z}_\mu\|)$, so that

$$(M_k - \hat{M}_\mu)d_k^x + o(\|d_k^\lambda\|) = o(\|z_k - \hat{z}_\mu\|).$$

Then (4.5) follows from the fact that, by the q-superlinear convergence of $\{z_k\}$, $z_k - \hat{z}_\mu = O(\|d_k\|)$.

Let us now prove the converse. By (4.5), the left-hand side of (4.6) is an $o(\|d_k\|)$ and due to the nonsingularity of $\mathcal{M}$, (4.6) gives $z_{k+1} - \hat{z}_\mu = o(\|z_k - \hat{z}_\mu\|) + o(\|d_k\|)$. With a unit step-size, $d_k = (z_{k+1} - \hat{z}_\mu) - (z_k - \hat{z}_\mu)$, so that we finally get $z_{k+1} - \hat{z}_\mu = o(\|z_k - \hat{z}_\mu\|)$. □

For proving the q-superlinear convergence of the sequence $\{z_k\}$, we need the following result from the BFGS theory (see [40, Theorem 3] and [6]).

LEMMA 4.3. *Let $\{M_k\}$ be a sequence of matrices generated by the BFGS formula from a given symmetric positive definite matrix M_1 and pairs (γ_k, δ_k) of vectors verifying*

$$\gamma_k^\top \delta_k > 0 \quad \textit{for all } k \geq 1 \qquad \textit{and} \qquad \sum_{k\geq 1} \frac{\|\gamma_k - M\delta_k\|}{\|\delta_k\|} < \infty, \tag{4.7}$$

where M is a symmetric positive definite matrix. Then, the sequences $\{M_k\}$ and $\{M_k^{-1}\}$ are bounded and

$$(M_k - M)\delta_k = o(\|\delta_k\|). \tag{4.8}$$

By using this lemma, we will see that the BFGS formula gives the estimate

$$(M_k - \hat{M}_\mu)d_k^x = o(\|d_k^x\|).$$

Note that the above estimate implies (4.5), from which the q-superlinear convergence of $\{z_k\}$ will follow.

A function ϕ, twice differentiable in a neighborhood of a point $x \in \mathbb{R}^n$, is said to have a *locally radially Lipschitzian* Hessian at x, if there exists a positive constant L such that for x' near x, one has

$$\|\nabla^2\phi(x) - \nabla^2\phi(x')\| \leq L\|x - x'\|.$$

THEOREM 4.4. *Suppose that Assumption 2.1 holds and that f and c are $C^{1,1}$ functions, twice continuously differentiable near $\hat{x}_\mu$ with locally radially Lipschitzian Hessians at $\hat{x}_\mu$. Suppose that the line-search in Algorithm A_μ uses the constant $\omega < \frac{1}{2}$. Then the sequence $\{z_k\} = \{(x_k, \lambda_k)\}$ generated by this algorithm converges to $\hat{z}_\mu = (\hat{x}_\mu, \hat{\lambda}_\mu)$ q-superlinearly and, for k sufficiently large, the unit step-size $\alpha_k = 1$ is accepted by the line-search.*

Proof. Let us start by showing that Lemma 4.3 with $M = \hat{M}_\mu$ can be applied. First, $\gamma_k^\top \delta_k > 0$, as this was already discussed after Lemma 3.3. For the convergence of the series in (4.7), we use a Taylor expansion, assuming that k is large enough (f and c are C^2 near $\hat{x}_\mu$):

$$\begin{aligned}\gamma_k - \hat{M}_\mu\delta_k = &\int_0^1 \left(\nabla^2_{xx}\ell(x_k + t\delta_k, \lambda_{k+1}) - \nabla^2_{xx}\ell(\hat{x}_\mu, \lambda_{k+1})\right)\delta_k \,\mathrm{d}t \\ &+ \left(\nabla^2_{xx}\ell(\hat{x}_\mu, \lambda_{k+1}) - \hat{M}_\mu\right)\delta_k.\end{aligned}$$

With the local radial Lipschitz continuity of $\nabla^2 f$ and $\nabla^2 c$ at $\hat{x}_\mu$ and the boundedness of $\{\lambda_{k+1}\}$, there exist positive constants K' and K'' such that

$$\begin{aligned}\|\gamma_k - \hat{M}_\mu\,\delta_k\| &\le K'\|\delta_k\|\left(\int_0^1 \|x_k + t\,\delta_k - \hat{x}_\mu\|\mathrm{d}t + \|\lambda_{k+1} - \hat{\lambda}_\mu\|\right)\\ &\le K'\|\delta_k\|\left(\int_0^1 \Big((1-t)\|x_k - \hat{x}_\mu\| + t\|x_{k+1} - \hat{x}_\mu\|\Big)\,\mathrm{d}t\right.\\ &\qquad\left. + \|\lambda_{k+1} - \hat{\lambda}_\mu\|\right)\\ &\le K''\|\delta_k\|\Big(\|x_k - \hat{x}_\mu\| + \|z_{k+1} - \hat{z}_\mu\|\Big).\end{aligned}$$

Hence the series in (4.7) converges by Theorem 3.4. Therefore, by (4.8) with $M = \hat{M}_\mu$ and the fact that δ_k is parallel to d_k^x,

$$(M_k - \hat{M}_\mu)d_k^x = o(\|d_k^x\|). \tag{4.9}$$

By the estimate (4.9) and Proposition 4.1, the unit step-size is accepted when k is large enough. The q-superlinear convergence of $\{z_k\}$ follows from Proposition 4.2. ∎

5. The overall primal-dual algorithm. In this section, we consider an overall algorithm for solving problem (1.1). Recall from Lemma 2.2 that the set of primal solutions of this problem is nonempty and bounded. By the Slater condition (Assumption 2.1(ii)), the set of dual solutions is also nonempty and bounded. Let us denote by $\hat{z} = (\hat{x}, \hat{\lambda})$ a primal-dual solution of problem (1.1), which is also a solution of the necessary and sufficient conditions of optimality

$$\begin{cases} \nabla f(\hat{x}) - \nabla c(\hat{x})\hat{\lambda} = 0,\\ C(\hat{x})\hat{\lambda} = 0,\\ (c(\hat{x}), \hat{\lambda}) \ge 0. \end{cases} \tag{5.1}$$

Our overall algorithm for solving (1.1) or (5.1), called Algorithm A, consists in computing approximate solutions of the perturbed optimality conditions (1.2), for a sequence of μ's converging to zero. For each μ, the primal-dual Algorithm A_μ is used to find an approximate solution of (1.2). This is done by so-called *inner* iterations. Next μ is decreased and the process of solving (1.2) for the new value of μ is repeated. We call an *outer* iteration the collection of inner iterations for solving (1.2) for a fixed value of μ. We index the outer iterations by superscripts $j \in \mathbb{N}\backslash\{0\}$.

ALGORITHM A (for solving problem (1.1); one outer iteration).

0. At the beginning of the jth outer iteration, an approximation $z_1^j := (x_1^j, \lambda_1^j) \in \mathcal{Z}$ of the solution $\hat{z}$ of (5.1) is supposed available, as well as a positive definite matrix M_1^j approximating the Hessian of the Lagrangian. A value $\mu^j > 0$ is given, as well as a precision threshold $\epsilon^j > 0$.
1. Starting from z_1^j, use Algorithm A_μ until $z^j := (x^j, \lambda^j)$ satisfies

$$\|\nabla f(x^j) - \nabla c(x^j)\lambda^j\| \le \epsilon^j \quad \text{and} \quad \|C(x^j)\lambda^j - \mu^j e\| \le \epsilon^j. \tag{5.2}$$

2. Choose a new starting iterate $z_1^{j+1} \in \mathcal{Z}$ for the next outer iteration, as well as a positive definite matrix M_1^{j+1}. Set the new parameters $\mu^{j+1} > 0$ and $\epsilon^{j+1} > 0$, such that $\{\mu^j\}$ and $\{\epsilon^j\}$ converge to zero when $j \to \infty$.

To start the $(j+1)$th outer iteration, a possibility is to take $z_1^{j+1} = z^j$ and $M_1^{j+1} = M^j$, the updated matrix obtained at the end of the jth outer iteration.

As far as the global convergence is concerned, how z^j, M^j, and μ^j are determined is not important. Therefore, on that point, Algorithm A leaves the user much freedom to maneuver, while Theorem 5.1 gives us a global convergence result for such a general algorithm.

THEOREM 5.1. *Suppose that Assumption* 2.1 *holds and that f and c are $C^{1,1}$ functions. Then Algorithm A generates a bounded sequence $\{z^j\}$ and any limit point of $\{z^j\}$ is a primal-dual solution of problem* (1.1).

Proof. By Theorem 3.4, any outer iteration of Algorithm A terminates with an iterate z^j satisfying the stopping criteria in Step 1. Therefore Algorithm A generates a sequence $\{z^j\}$. Since the sequences $\{\mu^j\}$ and $\{\epsilon^j\}$ converge to zero, any limit point of $\{z^j\}$ is a solution of problem (1.1). It remains to show that $\{z^j\}$ is bounded.

Let us first prove the boundedness of $\{x^j\}$. The convexity of the Lagrangian implies that

$$\ell(x^j, \lambda^j) + \nabla_x \ell(x^j, \lambda^j)^\top (x^1 - x^j) \leq \ell(x^1, \lambda^j).$$

Using the positivity of λ^j and $c(x^1)$ and next the stopping criteria of Algorithm A, it follows that

$$\begin{aligned} f(x^j) &\leq f(x^1) + (\lambda^j)^\top c(x^j) + \nabla_x \ell(x^j, \lambda^j)^\top (x^j - x^1) \\ &\leq f(x^1) + o(1) + o(\|x^j - x^1\|). \end{aligned}$$

If $\{x^j\}$ is unbounded, setting $t^j := \|x^j - x^1\|$ and $y^j := \frac{x^j - x^1}{t^j}$, one can choose a subsequence J such that

$$\lim_{\substack{j \to +\infty \\ j \in J}} t^j = +\infty \quad \text{and} \quad \lim_{\substack{j \to +\infty \\ j \in J}} y^j = y \neq 0.$$

From the last inequality we deduce that

$$f'_\infty(y) := \lim_{\substack{j \to +\infty \\ j \in J}} \frac{f(x^1 + t^j y^j) - f(x^1)}{t^j} \leq 0.$$

Moreover, since $c(x^j) > 0$, we have $(-c_{(i)})'_\infty(y) \leq 0$ for $i = 1, \ldots, m$. It follows that $\hat{x} + \mathbb{R}_+ y \subset \{x : c(x) \geq 0,\ f(x) \leq f(\hat{x})\}$ (see, for example, [22, Proposition IV.3.2.5] or [2, Formula (1)]). Therefore, the solution set of problem (1.1) would be unbounded, which is in contradiction with what is claimed in Lemma 2.2.

To prove the boundedness of the multipliers, suppose that the algorithm generates an unbounded sequence of positive vectors $\{\lambda^j\}_{j \in J'}$ for some subsequence J'. The sequence $\{(x^j, \lambda^j / \|\lambda^j\|)\}_{j \in J'}$ is bounded and thus has at least one limit point, say, (x^*, ν^*). Dividing the two inequalities in (5.2) by $\|\lambda^j\|$ and taking limits when $j \to \infty$, $j \in J'$, we deduce that $\nu^* \geq 0$, $\nabla c(x^*)\nu^* = 0$, and $(\nu^*)^\top c(x^*) = 0$. Using the concavity of the components $c_{(i)}$, one has

$$c(x^*) + \nabla c(x^*)^\top (x^1 - x^*) \geq c(x^1) > 0,$$

where the inequality on the right follows from the strict feasibility of the first iterate. Multiplying by ν^*, we deduce that $(\nu^*)^\top c(x^1) = 0$, and thus $\nu^* = 0$, a contradiction with $\|\nu^*\| = 1$. ∎

In the rest of this section, we give conditions under which the whole sequence $\{z^j\}$ converges to a particular point called the analytic center of the primal-dual optimal set. This actually occurs when the following two conditions hold: strict complementarity and a proper choice of the forcing sequence ϵ^j in Algorithm A, which has to satisfy the estimate

$$\epsilon^j = o(\mu^j),$$

meaning that $\epsilon^j/\mu^j \to 0$ when $j \to \infty$.

Let us first recall the notion of analytic center of the optimal sets, which under Assumption 2.1 is uniquely defined (see Monteiro and Zhou [37], for related results). We denote by $\mathrm{opt}(P)$ and $\mathrm{opt}(D)$ the sets of primal and dual solutions of problem (1.1). The analytic center of $\mathrm{opt}(P)$ is defined as follows. If $\mathrm{opt}(P)$ is reduced to a single point, its analytic center is precisely that point. Otherwise, $\mathrm{opt}(P)$ is a convex set with more than one point. In that case, f is not strongly convex and, by Assumption 2.1(i), at least one of the constraint functions, $-c_{(i_0)}$ say, is strongly convex. It follows that the index set

$$B := \{i : \text{there exists } \hat{x} \in \mathrm{opt}(P) \text{ such that } c_{(i)}(\hat{x}) > 0\}$$

is nonempty (it contains i_0). The analytic center of $\mathrm{opt}(P)$ is then defined as the unique solution of the following problem:

$$\max_{\substack{\hat{x}\in\mathrm{opt}(P)\\ c_B(\hat{x})>0}} \left(\sum_{i\in B} \log c_{(i)}(\hat{x})\right). \tag{5.3}$$

The fact that this problem is well defined and has a unique solution is the matter of Lemma 5.2 below. Similarly, if $\mathrm{opt}(D)$ is reduced to a single point, its analytic center is that point. In case of multiple dual solutions, the index set

$$N := \{i : \text{there exists } \hat{\lambda} \in \mathrm{opt}(D) \text{ such that } \hat{\lambda}_{(i)} > 0\}$$

is nonempty (otherwise $\mathrm{opt}(D)$ would be reduced to $\{0\}$). The analytic center of $\mathrm{opt}(D)$ is then defined as the unique solution of the following problem:

$$\max_{\substack{\hat{\lambda}\in\mathrm{opt}(D)\\ \hat{\lambda}_N>0}} \left(\sum_{i\in N} \log \hat{\lambda}_{(i)}\right). \tag{5.4}$$

LEMMA 5.2. *Suppose that Assumption* 2.1 *holds. If* $\mathrm{opt}(P)$ (*resp.,* $\mathrm{opt}(D)$) *is not reduced to a singleton, then problem* (5.3) (*resp.,* (5.4)) *has a unique solution.*

Proof. Consider first problem (5.3) and suppose that $\mathrm{opt}(P)$ is not a singleton. We have seen that B is nonempty. By the convexity of the set $\mathrm{opt}(P)$ and the concavity of the functions $c_{(i)}$, there exists $\hat{x} \in \mathrm{opt}(P)$ such that $c_B(\hat{x}) > 0$. Therefore the feasible set in (5.3) is nonempty. On the other hand, let $\hat{x}_0$ be a point satisfying the constraints in (5.3). Then the set

$$\left\{\hat{x} : \hat{x} \in \mathrm{opt}(P),\ c_B(\hat{x}) > 0,\ \text{and } \sum_{i\in B} \log c_i(\hat{x}) \geq \sum_{i\in B} \log c_i(\hat{x}_0)\right\}$$

is nonempty, bounded (Lemma 2.2), and closed. Therefore, problem (5.3) has a solution. Finally, by Assumption 2.1(i), we know that there is an index $i_0 \in B$ such

that $-c_{(i_0)}$ is strongly convex. It follows that the objective in (5.3) is strongly concave and that problem (5.3) has a unique solution.

By similar arguments and the fact that the objective function in (5.4) is strictly concave, it follows that problem (5.4) has a unique solution. □

By complementarity (i.e., $C(\hat{x})\hat{\lambda} = 0$) and convexity of problem (1.1), the index sets B and N do not intersect, but there may be indices that are neither in B nor in N. It is said that problem (1.1) has the *strict complementarity* property if $B \cup N = \{1, \ldots, n\}$. This is equivalent to the existence of a primal-dual solution satisfying strict complementarity.

THEOREM 5.3. *Suppose that Assumption* 2.1 *holds and that f and c are $C^{1,1}$ functions. Suppose also that problem* (1.1) *has the strict complementarity property and that the sequence $\{\epsilon^j\}$ in Algorithm A satisfies the estimate $\epsilon^j = o(\mu^j)$. Then the sequence $\{z^j\}$ generated by Algorithm A converges to the point $\hat{z}_0 := (\hat{x}_0, \hat{\lambda}_0)$, where $\hat{x}_0$ is the analytic center of the primal optimal set and $\hat{\lambda}_0$ is the analytic center of the dual optimal set.*

Proof. Let $(\hat{x}, \hat{\lambda})$ be an arbitrary primal-dual solution of (1.1). Then $\hat{x}$ minimizes $\ell(\cdot, \hat{\lambda})$ and $\hat{\lambda}^\top c(\hat{x}) = 0$, so that

$$f(\hat{x}) = \ell(\hat{x}, \hat{\lambda}) \leq \ell(x^j, \hat{\lambda}) = f(x^j) - \hat{\lambda}^\top c(x^j).$$

Using the convexity of $\ell(\cdot, \lambda^j)$ and the stopping criterion (5.2) of the inner iterations in Algorithm A, one has

$$\begin{aligned} f(\hat{x}) - (\lambda^j)^\top c(\hat{x}) &= \ell(\hat{x}, \lambda^j) \\ &\geq \ell(x^j, \lambda^j) + \nabla_x \ell(x^j, \lambda^j)^\top (\hat{x} - x^j) \\ &= f(x^j) - (\lambda^j)^\top c(x^j) - \epsilon^j \|x^j - \hat{x}\| \\ &\geq f(x^j) - m\mu^j - m^{\frac{1}{2}} \epsilon^j - \epsilon^j \|x^j - \hat{x}\|, \end{aligned}$$

because $(\lambda^j)^\top c(x^j) = m\mu^j + e^\top (C(x^j)\lambda^j - \mu^j e) \leq m\mu^j + m^{\frac{1}{2}} \epsilon^j$. By Theorem 5.1, there is a constant C_1 such that $m^{\frac{1}{2}} + \|x^j - \hat{x}\| \leq C_1$. Then, adding the corresponding sides of the two inequalities above leads to

$$\hat{\lambda}_N^\top c_N(x^j) + (\lambda_B^j)^\top c_B(\hat{x}) = \hat{\lambda}^\top c(x^j) + (\lambda^j)^\top c(\hat{x}) \leq m\mu^j + C_1 \epsilon^j. \tag{5.5}$$

We pursue this by adapting an idea used by McLinden [34] to give properties of the limit points of the path $\mu \mapsto (\hat{x}_\mu, \hat{\lambda}_\mu)$. Let us define $\Gamma^j := C(x^j)\lambda^j - \mu^j e$. One has for all indices i

$$c_{(i)}(x^j) = \frac{\mu^j + \Gamma_{(i)}^j}{\lambda_{(i)}^j} \qquad \text{and } \lambda_{(i)}^j = \frac{\mu^j + \Gamma_{(i)}^j}{c_{(i)}(x^j)}.$$

Substituting this in (5.5) and dividing by μ^j give

$$\sum_{i \in N} \frac{\hat{\lambda}_{(i)}}{\lambda_{(i)}^j} \frac{\mu^j + \Gamma_{(i)}^j}{\mu^j} + \sum_{i \in B} \frac{c_{(i)}(\hat{x})}{c_{(i)}(x^j)} \frac{\mu^j + \Gamma_{(i)}^j}{\mu^j} \leq m + C_1 \frac{\epsilon^j}{\mu^j}.$$

By assumptions, $\epsilon^j = o(\mu^j)$, so that $\Gamma_{(i)}^j = o(\mu^j)$. Now supposing that $(\hat{x}_0, \hat{\lambda}_0)$ is a limit point of $\{(x^j, \lambda^j)\}$ and taking the limit in the preceding estimate provide

$$\sum_{i \in N} \frac{\hat{\lambda}_{(i)}}{(\hat{\lambda}_0)_{(i)}} + \sum_{i \in B} \frac{c_{(i)}(\hat{x})}{c_{(i)}(\hat{x}_0)} \leq m.$$

Necessarily, $c_B(\hat{x}_0) > 0$ and $(\hat{\lambda}_0)_N > 0$. Observe now that, by strict complementarity, there are exactly m terms on the left-hand side of the preceding inequality. Hence, by the arithmetic-geometric mean inequality

$$\left(\prod_{i\in N} \frac{\hat{\lambda}_{(i)}}{(\hat{\lambda}_0)_{(i)}}\right)\left(\prod_{i\in B} \frac{c_{(i)}(\hat{x})}{c_{(i)}(\hat{x}_0)}\right) \leq 1$$

or

$$\left(\prod_{i\in N} \hat{\lambda}_{(i)}\right)\left(\prod_{i\in B} c_{(i)}(\hat{x})\right) \leq \left(\prod_{i\in N} (\hat{\lambda}_0)_{(i)}\right)\left(\prod_{i\in B} c_{(i)}(\hat{x}_0)\right).$$

One can take $\hat{\lambda}_N = (\hat{\lambda}_0)_N > 0$ or $c_B(\hat{x}) = c_B(\hat{x}_0) > 0$ in this inequality, so that

$$\prod_{i\in B} c_{(i)}(\hat{x}) \leq \prod_{i\in B} c_{(i)}(\hat{x}_0) \qquad \text{and} \quad \prod_{i\in N} \hat{\lambda}_{(i)} \leq \prod_{i\in N} (\hat{\lambda}_0)_{(i)}.$$

This shows that $\hat{x}_0$ is a solution of (5.3) and that $\hat{\lambda}_0$ is a solution of (5.4). Since the problems in (5.3) and (5.4) have unique solutions, all the sequence $\{x^j\}$ converges to $\hat{x}_0$ and all the sequence $\{\lambda^j\}$ converges to $\hat{\lambda}_0$. ∎

6. Discussion. By way of conclusion, we discuss the results obtained in this paper, give some remarks, and raise some open questions.

Problems with linear constraints. The algorithm is presented with convex inequality constraints only, but it can also be used when linear constraints are present. Consider the problem

$$\left\{\begin{array}{l} \min\ f(x), \\ Ax = b, \\ c(x) \geq 0, \end{array}\right. \tag{6.1}$$

obtained by adding linear constraints to problem (1.1). In (6.1), A is a $p \times n$ matrix with $p < n$ and $b \in \mathbb{R}^p$ is given in the range space of A.

Problem (6.1) can be reduced to problem (1.1) by using a basis of the null space of the matrix A. Indeed, let x_1 be the first iterate, which is supposed to be strictly feasible in the sense that

$$Ax_1 = b \quad \text{and} \quad c(x_1) > 0.$$

Let us denote by Z an $n \times q$ matrix whose columns form a basis of the null space of A. Then, any point satisfying the linear constraints of (6.1) can be written

$$x = x_1 + Zu \quad \text{with } u \in \mathbb{R}^q.$$

With this notation, problem (6.1) can be rewritten as the problem in $u \in \mathbb{R}^q$:

$$\left\{\begin{array}{l} \min\ f(x_1 + Zu), \\ c(x_1 + Zu) \geq 0, \end{array}\right. \tag{6.2}$$

which has the form (1.1).

Thanks to this transformation, we can deduce from Assumption 2.1 what are the minimal assumptions under which our algorithm for solving problem (6.2) or, equivalently, problem (6.1) will converge.

Assumption 6.1. (i) The real-valued functions f and $-c_{(i)}$ $(1 \le i \le m)$ are convex and differentiable on the affine subspace $X := \{x : Ax = b\}$ and at least one of the functions f, $-c_{(1)}$, ..., $-c_{(m)}$ is strongly convex on X. (ii) There exists an $x \in \mathbb{R}^n$ such that $Ax = b$ and $c(x) > 0$.

With these assumptions, all the previous results apply. In particular, Algorithm A_μ converges r-linearly (if f and c are also $C^{1,1}$) and q-superlinearly (if f and c are also $C^{1,1}$, twice continuously differentiable near $\hat{x}_\mu$ with locally radially Lipschitzian Hessian at $\hat{x}_\mu$). Similarly, the conclusions of Theorem 5.1 apply if f and c are also $C^{1,1}$.

Feasible algorithms and qN techniques. In the framework of qN methods, the property of having to generate feasible iterates should not be only viewed as a restriction limiting the applicability of a feasible algorithm. Indeed, in the case of problem (6.2), if it is sometimes difficult to find a strictly feasible initial iterate, the matrix to update for solving this problem is of order q only, instead of order n for an infeasible algorithm solving problem (6.1) directly. When $q \ll n$, the qN updates will approach the reduced Hessian of the Lagrangian $Z^\top(\nabla^2\ell)Z$ more rapidly than the full Hessian $\nabla^2\ell$, so that a feasible algorithm is likely to converge more rapidly.

About the strong convexity hypothesis. Another issue concerns the extension of the present theory to convex problems, without the strong convexity assumption (Assumption 2.1(i)).

Without this hypothesis, the class of problems to consider encompasses linear programming (f and c are affine). It is clear that for dealing properly with linear programs, our algorithm needs modifications, since then $\gamma_k = 0$ and the BFGS formula is no longer defined. Of course, it would be very ineffective to solve linear programs with the qN techniques proposed in this paper ($M_k = 0$ is the desired matrix), but problems that are almost linear near the solution may be encountered, so that a technique for dealing with a situation where $\|\gamma_k\| \ll \|\delta_k\|$ can be of interest.

To accept $\gamma_k = 0$, one can look at the limit of the BFGS formula (2.1) when $\gamma_k \to 0$. A possible update formula could be

$$M_{k+1} := M_k - \frac{M_k \delta_k \delta_k^\top M_k}{\delta_k^\top M_k \delta_k}.$$

The updated matrix satisfies $M_{k+1}\delta_k = 0$ and is positive semidefinite, provided M_k is already positive semidefinite. The fact that M_{k+1} may be singular raises some difficulties, however. For example, the search direction d^x may no longer be defined (see formula (1.5), in which the matrix $M + \nabla c(x) C(x)^{-1} \Lambda \nabla c(x)^\top$ can be singular). Therefore, the present theory cannot be extended in a straightforward manner.

On the other hand, the strong convexity assumption may not be viewed as an important restriction, because a fictive strongly convex constraint can always be added. An obvious example of fictive constraint is "$x^\top x \le K$." If the constant K is large enough, the constraint is inactive at the solution, so that the solution of the original problem is not altered by this new constraint and the present theory applies.

Better control of the outer iterations. Last but not least, the global convergence result of section 5 is independent of the update rule of the parameters ϵ^j and μ^j. In practice, however, the choice of the decreasing values ϵ^j and μ^j is essential for the efficiency of the algorithm and would deserve a detailed numerical study.

From a theoretical viewpoint, it would be highly desirable to have an update rule that would allow the outer iterates of Algorithm A to converge q-superlinearly. Along

the same lines, an interesting problem is to design an algorithm in which the barrier parameter would be updated at every step, while having q-superlinear convergence of the iterates. Such extensions would involve more difficult issues.

The global convergence result proved in this paper gives us some reasons to believe that it is not unreasonable to tackle these open questions.

Acknowledgments. We would like to thank the referees for their valuable comments. One of them has shown us a direct argument for the last part of the proof of Proposition 3.2, which is the one we have finally chosen to give in the paper. The other referee has brought McLinden's paper to our attention, which led us to Theorem 5.3.

REFERENCES

[1] K.M. Anstreicher and J.-P. Vial, *On the convergence of an infeasible primal-dual interior-point method for convex programming*, Optim. Methods Softw., 3 (1994), pp. 273–283.

[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] M. Berz, C. Bischof, G. Corliss, and A. Griewank, eds., *Computational Differentiation: Techniques, Applications, and Tools*, SIAM, Philadelphia, 1996.

[4] J.F. Bonnans and C. Pola, *A trust region interior point algorithm for linearly constrained optimization*, SIAM J. Optim., 7 (1997), pp. 717–731.

[5] R.H. Byrd, J.Ch. Gilbert, and J. Nocedal, *A Trust Region Method Based on Interior Point Techniques for Nonlinear Programming*, Technical report 2896, INRIA, Le Chesnay, France, 1996; Math. Programming, submitted.

[6] R.H. Byrd and J. Nocedal, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 727–739.

[7] A.R. Conn, N.I.M. Gould, and Ph.L. Toint, *A Primal-Dual Algorithm for Minimizing a Non-convex Function Subject to Bound and Linear Equality Constraints*, Technical Report, Facultés Universitaires de Namur, Belgique, 1996.

[8] A. Griewank, *Evaluating Derivatives. Principles and Techniques of Algorithmic Differentiation,* Frontiers in Applied Mathematics 19, SIAM, Philadelphia, 2000.

[9] P. Courtier, J.N. Thépaut, and A. Hollingsworth, *A strategy for operational implementation of 4D-Var, using an incremental approach*, Quart. J. Royal Meteorological Society, 120 (1994), pp. 1367–1387.

[10] D. den Hertog, *Interior Point Approach to Linear, Quadratic and Convex Programming*, Math. Appl. 277, Kluwer, Dordrecht, The Netherlands, 1992.

[11] D. den Hertog, C. Roos, and T. Terlaky, *A Potential Reduction Method for a Class of Smooth Convex Programming Problems*, Report 90-01, Faculty of Technical Mathematics and Informatics, Technische Universiteit Delft, Holland, 1990.

[12] D. den Hertog, C. Roos, and T. Terlaky, *On the classical logarithmic barrier function method for a class of smooth convex programming problems*, J. Optim. Theory Appl., 73 (1992), pp. 1–25.

[13] 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.

[14] J.E. Dennis and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Englewood Cliffs, NJ, 1983.

[15] A.S. El-Bakry, R.A. Tapia, T. Tsuchiya, and Y. Zhang, *On the formulation and theory of the Newton interior-point method for nonlinear programming*, J. Optim. Theory Appl., 89 (1996), pp. 507–541.

[16] A.V. Fiacco and G.P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley, New York, 1968.

[17] R. Fletcher, *Practical Methods of Optimization,* 2nd ed., John Wiley, Chichester, UK, 1987.

[18] A. Forsgren and P.E. Gill, *Primal-dual interior methods for nonconvex nonlinear programming*, SIAM J. Optim., 8 (1998), pp. 1132–1152.

[19] J.Ch. Gilbert, G. Le Vey, and J. Masse, *La différentiation automatique de fonctions représentées par des programmes*, Rapport de Recherche 1557, INRIA, Le Chesnay, France, 1991; also available online from http://www.inria.fr/RRRT/RR-1557.html; available via ftp at ftp://ftp.inria.fr/INRIA/publication/RR, RR-1557.ps.gz.

[20] J.Ch. Gilbert and C. Lemaréchal, *Some numerical experiments with variable-storage quasi-Newton algorithms*, Math. Programming, 45 (1989), pp. 407–435.

[21] J. Herskovits, *Feasible direction interior-point technique for nonlinear optimization*, J. Optim. Theory Appl., 99 (1998), pp. 121–146.
[22] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Grundlehren Math. Wiss. 305–306, Springer-Verlag, Berlin, New York, 1993.
[23] B. Jansen, *Interior Point Techniques in Optimization—Complementarity, Sensitivity and Algorithms*, Appl. Optim. 6, Kluwer, Dordrecht, The Netherlands, 1997.
[24] F. Jarre, *On the method of analytic centers for solving smooth convex problems*, Lecture Notes in Math. 1405, Springer-Verlag, Berlin, 1989, pp. 69–85.
[25] F. Jarre, *Interior-point methods for classes of convex programs*, in Interior Point Methods of Mathematical Programming, T. Terlaky, ed., Kluwer, Dordrecht, The Netherlands, 1996, chapter 7.
[26] F. Jarre and M.A. Saunders, *A practical interior-point method for convex programming*, SIAM J. Optim., 5 (1995), pp. 149–171.
[27] S. Jégou, *Estimation des perméabilités relatives dans des expériences de déplacements triphasiques en milieu poreux*, Ph.D. thesis, Université de Paris IX (Dauphine), Paris, 1997.
[28] S. Jégou, *Using Maple for symbolic differentiation to solve inverse problems*, MapleTech, 4 (1997), pp. 32–40.
[29] 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.
[30] M. Kojima, S. Mizuno, and T. Noma, *A new continuation method for complementarity problems with uniform P-functions*, Math. Programming, 43 (1989), pp. 107–113.
[31] M. Kojima, S. Mizuno, and T. Noma, *Limiting behavior of trajectories generated by a continuation method for monotone complementarity problems*, Math. Oper. Res., 15 (1990), pp. 662–675.
[32] D.C. Liu and J. Nocedal, *On the limited memory BFGS method for large scale optimization*, Math. Programming, 45 (1989), pp. 503–520.
[33] G.P. McCormick, *The projective SUMT method for convex programming problems*, Math. Oper. Res., 14 (1989), pp. 203–223.
[34] L. McLinden, *An analogue of Moreau's proximation theorem, with application to the nonlinear complementarity problem*, Pacific J. Math., 88 (1980), pp. 101–161.
[35] S. Mehrotra and J. Sun, *An interior point algorithm for solving smooth convex programs based on Newton's method*, in Mathematical Developments Arising from Linear Programming, J.C. Lagarias and M.J. Todd, eds., Contemp. Math. 114, AMS, Providence, RI, 1990, pp. 265–284.
[36] R. Monteiro and I. Adler, *An extension of Karmarkar-type algorithms to a class of convex separable programming problems with global linear rate of convergence*, Math. Oper. Res., 15 (1990), pp. 408–422.
[37] R.D.C. Monteiro and F. Zhou, *On the existence and convergence of the central path for convex programming and some duality results*, Comput. Optim. Appl., 10 (1998), pp. 51–77.
[38] Y. Nesterov and A.S. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM Stud. Appl. Math. 13, SIAM, Philadelphia, 1994.
[39] J. Nocedal, *Updating quasi-Newton matrices with limited storage*, Math. Comput., 35 (1980), pp. 773–782.
[40] M.J.D. Powell, *On the convergence of the variable metric algorithm*, J. Inst. Math. Appl., 7 (1971), pp. 21–36.
[41] M.J.D. Powell, *Some global convergence properties of a variable metric algorithm for minimization without exact line searches*, in Nonlinear Programming, R.W. Cottle and C.E. Lemke, eds., SIAM-AMS Proceedings 9, AMS, Providence, RI, 1976.
[42] C. Roos, T. Terlaky, and J.-Ph. Vial, *Theory and Algorithms for Linear Optimization—An Interior Point Approach*, John Wiley, Chichester, UK, 1997.
[43] G. Sonnevend, *An "analytical center" for polyhedrons and new classes of global algorithms for linear (smooth, convex) programming*, in Lecture Notes in Control and Inform. Sci. 84, Springer-Verlag, New York, 1986, pp. 866–876.
[44] T. Terlaky ed., *Interior Point Methods of Mathematical Programming*, Kluwer, Dordrecht, The Netherlands, 1996.
[45] J.-P. Vial, *Computational experience with a primal-dual interior-point method for smooth convex programming*, Optim. Methods Softw., 3 (1994), pp. 285–310.

[46] M.H. Wright, *Why a pure primal Newton barrier step may be infeasible*, SIAM J. Optim., 5 (1995), pp. 1–12.
[47] S.J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.
[48] H. Yamashita and H. Yabe, *Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization*, Math. Programming, 75 (1996), pp. 377–397.
[49] Y. Ye, *Interior Point Algorithms—Theory and Analysis*, Wiley-Intersci. Ser. Discrete Math. Optim., John Wiley, New York, 1997.
[50] G. Zoutendijk, *Nonlinear programming, computational methods*, in Integer Nonlinear Programming, J. Abadie, ed., North-Holland, Amsterdam, 1970, pp. 37–86.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 223–242

SOME NEW SEARCH DIRECTIONS FOR PRIMAL-DUAL INTERIOR POINT METHODS IN SEMIDEFINITE PROGRAMMING*

KIM-CHUAN TOH†

Abstract. Search directions for primal-dual path-following methods for semidefinite programming (SDP) are proposed. These directions have the properties that (1) under certain nondegeneracy and strict complementarity assumptions, the Jacobian matrix of the associated symmetrized Newton equation has a bounded condition number along the central path in the limit as the barrier parameter μ tends to zero; and (2) the Schur complement matrix of the symmetrized Newton equation is symmetric and the cost for computing this matrix is $2mn^3 + 0.5m^2n^2$ flops, where n and m are the dimension of the matrix and vector variables of the semidefinite program, respectively. These two properties imply that a path-following method using the proposed directions can achieve the high accuracy typically attained by methods employing the direction proposed by Alizadeh, Haeberly, and Overton (currently the best search direction in terms of accuracy), but each iteration requires at most half the amount of flops (to leading order).

Key words. semidefinite programming, interior point methods, search directions

AMS subject classification. 90C05

PII. S1052623498335067

1. Introduction.

Let $\mathcal{S}^n$ be the vector space of $n \times n$ real symmetric matrices endowed with the inner product $A \bullet B = \mathrm{Tr}(AB)$. Let **svec** be an isometry identifying $\mathcal{S}^n$ with $\mathbb{R}^{n(n+1)/2}$ so that $K \bullet L = \mathbf{svec}(K)^t \mathbf{svec}(L)$ and let **smat** be the inverse of **svec**. Given any two $n \times n$ matrices G and K, we define the linear map $G \circledast K : \mathbb{R}^{n(n+1)/2} \longrightarrow \mathbb{R}^{n(n+1)/2}$ by

$$(G \circledast K)\,\mathbf{svec}(H) \;=\; \mathbf{svec}((GHK^t \,+\, KHG^t)/2).$$

Consider the semidefinite program

$$\begin{array}{lrcl} \min_X & C \bullet X, & & \\ & A_k \bullet X & = & b_k, \quad k = 1, \ldots, m, \\ & X & \succeq & 0, \end{array} \tag{1}$$

where $b \in \mathbb{R}^m$, $A_k, C \in \mathcal{S}^n$, and $X \succeq 0$ means that X is positive semidefinite. The dual of (1) is

$$\begin{aligned} &\max_{y,S} \; b^t y, \\ &\sum_{k=1}^{m} y_k A_k + S \;=\; C, \\ &\qquad\qquad S \succeq 0. \end{aligned} \tag{2}$$

We consider primal-dual path-following methods for SDP in which the general framework in each iteration is as follows (see [15], [17]): Given a current iterate (X, y, S) and a barrier parameter μ, where X, S are symmetric positive definite, the

*Received by the editors March 4, 1998; accepted for publication (in revised form) January 5, 2000; published electronically August 3, 2000.
http://www.siam.org/journals/siopt/11-1/33506.html
†Department of Mathematics, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119620 (mattohkc@math.nus.edu.sg).

methods find a search direction $(\Delta X, \Delta y, \Delta S)$, so as to generate the next iterate, by solving the following symmetrized Newton equation (with respect to a given nonsingular matrix P):

$$\mathcal{J}\,\Delta\mathcal{X} \;=\; \mathcal{R}, \tag{3}$$

where

$$\mathcal{J} \;=\; \begin{pmatrix} 0 & \mathcal{A}^t & \mathcal{I} \\ \mathcal{A} & 0 & 0 \\ \mathcal{E} & 0 & \mathcal{F} \end{pmatrix}, \quad \Delta\mathcal{X} \;=\; \begin{pmatrix} \mathbf{svec}(\Delta X) \\ \Delta y \\ \mathbf{svec}(\Delta S) \end{pmatrix}, \quad \mathcal{R} \;=\; \begin{pmatrix} r_p \\ \mathbf{svec}(R_d) \\ \mathbf{svec}(R_c) \end{pmatrix}, \tag{4}$$

$\mathcal{I}$ is the identity matrix of order $n(n+1)/2$, and

$$\mathcal{A}^t \;=\; [\mathbf{svec}(A_1)\;\cdots\;\mathbf{svec}(A_m)], \quad r_p \;=\; b - \mathcal{A}\,\mathbf{svec}(X),$$

$$R_d \;=\; C - S - \sum_{k=1}^{m} y_k A_k, \qquad\qquad R_c \;=\; \mu I \;-\; H_P(XS).$$

Here

$$\mathcal{E} \;=\; P \circledast P^{-T} S, \qquad \mathcal{F} \;=\; PX \circledast P^{-T}, \tag{5}$$

and $H_P : \mathbb{R}^{n\times n} \longrightarrow \mathcal{S}^n$ is the symmetrization operator with respect to P, defined by $H_P(M) = (PMP^{-1} + P^{-t}M^tP^t)/2$.

The matrix P used in the symmetrization process H_P is generally chosen to be a function of X and S. For different choices of P (up to left-multiplication by orthogonal matrices), the search direction generated will be different. Currently, the most commonly considered search directions in practice are

1. Alizadeh–Haeberly–Overton (AHO) direction, corresponding to $P = I$ [3];

2. Helmberg–Rendel–Vanderbei–Wolkowicz/Kojima–Shindoh–Hara/Monteiro (HKM) direction, corresponding to $P = S^{1/2}$ [8], [9], [10];

3. Nesterov–Todd (NT) direction, corresponding to $P = W^{-1/2}$, where W is the unique symmetric positive definite matrix satisfying $WSW = X$ [15].

It is observed that a path-following method using the AHO direction (henceforth called the AHO method) can usually produce a much more accurate solution than methods using the HKM and NT directions (henceforth referred to as the HKM and NT methods, respectively). Let $\kappa(\mathcal{J})$ be the 2-norm condition number of the matrix $\mathcal{J}$ in (4) and ϵ be the machine epsilon. Typically, the AHO method can reduce the duality gap to a level below $\mathcal{O}(\epsilon\kappa(\mathcal{J}))$, while the HKM and NT methods can usually only achieve a level of $\max[\mathcal{O}(\sqrt{\epsilon}), \mathcal{O}(\epsilon\kappa(\mathcal{J}))]$. A heuristic explanation for this difference is that the Jacobian matrix $\mathcal{J}$ for the AHO method typically has a bounded condition in the limit as μ tends to zero along the central path, but not for the other two methods. The AHO method is also observed to be the most efficient (in terms of number of iterations) among the three. However, the AHO method has the drawback that each of its iterations is at least two times as expensive as those for the HKM and NT methods when the directions are computed via the Schur complement equation. The reason that each iteration of the AHO method is more expensive is partly because the Schur complement matrix $\mathcal{M} := \mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^t$ involved in the Schur complement equation is nonsymmetric for the AHO method but symmetric for the other two methods.

Given the strengths and weaknesses of the AHO, HKM, and NT directions, it is desirable for us to look for search directions that combine the strengths of all these directions. In this paper, we propose search directions (equivalently, matrices P) for which the Jacobian matrices $\mathcal{J}$ are shown to have bounded condition numbers along the central path in the limit as μ tends to zero, under the assumptions of primal and dual nondegeneracy, and strict complementarity as defined in [2], [3], but the Schur complement matrices for our directions are symmetric. The computational complexity of our directions is $2mn^3 + 0.5m^2n^2$ flops if $\mathcal{A}$ is dense. This is comparable to those for the HKM and NT directions. In contrast, the complexity for the AHO direction is $4mn^3 + m^2n^2$ flops.

We conducted numerical experiments to compare the performance of a path-following method using one of our proposed directions (for ease of discussion, this direction will be called the GT direction) to path-following methods with the same algorithmic framework but using the AHO, HKM, and NT directions. The numerical results show that, just like the AHO method, the path-following method using the GT direction can typically solve a semidefinite program to an accuracy better than $\mathcal{O}(\epsilon\kappa(\mathcal{J}))$. However, just like the HKM and NT methods, each iteration of our method requires less CPU time than the AHO method. Our method is almost as efficient (in terms of the number of iterations required to gain a desired accuracy) as the AHO method. Our method is also robust in that all the SDP problems considered in our experiments, a total of 70 random instances from seven different classes of semidefinite programs with $n, m \approx 100$, are successfully solved.

An issue that is closely related to the implementation of a path-following algorithm for SDP is the exploitation of possible sparsity of the data $\mathcal{A}$. In this paper, we also discuss this issue for algorithms employing the AHO, HKM, and NT directions and the directions we proposed. An analogous discussion, but different from ours, for the HKM and NT directions can also be found in [5].

Our paper is motivated by the work of Gu in [7]. In that paper, the author carried out stability analysis of the Schur complement approach for computing the AHO and a subfamily of the Monteiro–Zhang (MZ) directions [13] which he called the Todd–Toh–Tütüncü (TTT) directions. (This subfamily is also called the commuting class in [13], where the authors also give an explicit parameterization of these directions.) Based on results from the stability analysis, Gu proposed a direction (henceforth called Gu's direction) in the TTT family that is able to achieve high accuracy solutions through limited computational testing. However, there was no discussion of the conditioning of the Jacobian matrix associated with this direction along the central path.

Our proposed directions also belong to the TTT family, and they include Gu's direction as a special case. To avoid possible confusion, we should mention that the objective of this paper is different from that of [7]. Here, our objective is the derivation of TTT directions that can achieve high accuracy solutions through the analysis of the conditioning of the associated Jacobian matrices and the implementation and performance of practical path-following methods employing these directions. In contrast, the main objective of [7] is the stability analysis of the Schur complement approach for computing the AHO and TTT directions. As a byproduct of the stability analysis, Gu derived his direction based on the desire to minimize the condition number of the matrices P in the TTT family.

Recently, Monteiro and Zanjácomo [12] also reported computational results on the performance of a search direction (called the S-MT direction) that can achieve solutions of high accuracy, but each iteration has roughly the same computational

complexity as the AHO direction. By employing a hybrid strategy between the S-MT direction and another direction known as the S-Ch-MT direction, they were able to reduce the complexity to no less than $\frac{5}{3}mn^3 + m^2n^2$. In their implementations, they observed that the hybrid direction is comparable to the AHO, HKM, and NT directions in terms of efficiency, and the number of flops used per iteration falls between that for the AHO direction and the HKM and NT directions. The CPU time spent per iteration is not reported in their paper. However, it is not obvious that the reduction in the number of flops for the hybrid direction will indeed translate into savings in CPU time since the overhead involved in their implementation may be significant. We add that in [12] there is no discussion of the conditioning of the associated Jacobian matrices.

Theoretical properties such as primal-dual symmetry and scale invariance of search directions for interior point methods in SDP are investigated by Todd [14]. Some 20 directions are considered, including the AHO, HKM, and NT directions, Gu's direction proposed in [7], the GT direction proposed in this paper, and the S-MT and S-Ch-MT directions in [12]. Todd showed that the first five directions just mentioned satisfy all but a small number of desirable theoretical properties. Based on limited computational testing, he observed that path-following methods using the first five directions also perform the best in terms of robustness and accuracy among the 20 directions considered. Among these five directions, the computational performance of the AHO direction is the best, followed by the GT direction, Gu's direction, and the HKM and NT directions. His computational results confirmed our finding from the numerical experiments presented in this paper.

Throughout, we shall assume X and S to be symmetric positive definite, unless stated otherwise. The 2-norm for vectors and matrices will be used throughout.

2. Search directions in the TTT family. The class of search directions generated by matrices P for which the matrices $\mathcal{E}^{-1}\mathcal{F}$ are symmetric is called the TTT family in [7]. The set of nonsingular matrices P such that the associated matrix $\mathcal{E}^{-1}\mathcal{F}$ is symmetric can be characterized as follows.

LEMMA 2.1. *The matrix $\mathcal{E}^{-1}\mathcal{F}$ is symmetric if and only if the matrix P (up to left-multiplication by an orthogonal matrix) satisfies the condition that the matrix $PXSP^{-1}$ is symmetric.*

Proof. For the proof, see [15]. □

Note that the TTT directions always exist if X and S are symmetric positive definite. In contrast, the AHO direction does not always exist without imposing further assumptions on X and S; see [15].

Now we shall derive our search directions. Note that our interest here is not to give an explicit parameterization of the matrices P in the TTT family but merely on specific search directions in this family. Readers who are interested in the explicit parameterization of the TTT family can refer to [13] for the details.

Consider the SVD of $X^{1/2}S^{1/2}$

$$X^{1/2}S^{1/2} \;=\; U\Sigma V^t. \tag{6}$$

Suppose

$$U^tX^{1/2} \;=\; \Psi\bar{G}, \qquad V^tS^{1/2} \;=\; \Phi\bar{H}, \tag{7}$$

where Ψ, Φ are positive diagonal scaling matrices chosen so that the rows of $\bar{G}$ and $\bar{H}$ all have unit norms. Let D be the diagonal matrix defined by

$$D \;=\; \Sigma(\Phi\Psi)^{-1}. \tag{8}$$

Then

$$\bar{G}\bar{H}^t \;=\; D. \tag{9}$$

In the rest of the paper, we shall consider matrices P of the form

$$P \;=\; \Gamma^{-1}\bar{H} \;=\; \Gamma^{-1}D\bar{G}^{-t}, \tag{10}$$

where $\Gamma = \mathrm{Diag}(\gamma_1, \ldots, \gamma_n)$ is a positive diagonal matrix whose diagonal elements are given real-valued functions of X and S. It is easy to check that $PXSP^{-1} = \Sigma^2$ for matrices P of the form given in (10). Thus by Lemma 2.1, our search directions are in the TTT family. For convenience, we shall call the class of search directions generated by matrices P of the form given in (10) the TTT* subfamily.

It is clear that different choices of Γ would lead to different search directions. For example, the choices $\Gamma = \Phi^{-1}$ and $\Gamma = \Phi^{-1}\Sigma^{1/2}$ lead to the HKM and NT directions, respectively. For $\Gamma = I$, we obtain the search direction proposed by Gu in [7]. In our numerical experiments in section 6, we will consider the direction corresponding to $\Gamma = D^{1/2}$. For easy reference, we will refer to this direction as the GT direction.

For the choice of P in (10), the matrices $\mathcal{E}$ and $\mathcal{F}$ are given as follows.

LEMMA 2.2. *Suppose P is given by* (10). *Then the matrices $\mathcal{E}$ and $\mathcal{F}$ in* (5) *become*

$$\mathcal{E} \;=\; (\Gamma\Phi^2 \circledast \Gamma^{-1})(\bar{H} \circledast \bar{H}) \;=\; \mathcal{D}_{\mathcal{E}}\,(\bar{G}^{-t} \circledast \bar{G}^{-t}), \tag{11}$$

$$\mathcal{F} \;=\; \mathcal{D}_{\mathcal{F}}\,(\bar{G} \circledast \bar{G}), \tag{12}$$

where

$$\mathcal{D}_{\mathcal{E}} \;=\; \Gamma D\Phi^2 \circledast \Gamma^{-1}D, \qquad \mathcal{D}_{\mathcal{F}} \;=\; \Gamma^{-1}D\Psi^2 \circledast \Gamma D^{-1}.$$

Proof. Let $B = \Gamma^{-1}\Phi^{-1}$. Then $P = B(V^tS^{1/2})$, and we have from (5) that

$$\mathcal{E} \;=\; B(V^tS^{1/2}) \circledast B^{-1}(V^tS^{1/2}), \qquad \mathcal{F} \;=\; B\Sigma(U^tX^{1/2}) \circledast B^{-1}\Sigma^{-1}(U^tX^{1/2}).$$

By substituting $B = \Gamma^{-1}\Phi^{-1}$ and (7), (8), (9) into the above equations, we get (11) and (12). ∎

PROPOSITION 2.3. *The direction corresponding to $\Gamma = D^{1/2}$ is primal-dual symmetric in the sense defined in* [14].

Proof. For the proof see [14]. ∎

3. Conditioning of the Jacobian along the central path. In this section, we shall consider $X = X_\mu$ and $S = S_\mu$ on the central path, that is, $X_\mu S_\mu = \mu I$ with $\mu > 0$. We will denote the matrices D and Γ by D_μ and Γ_μ, respectively, to show their dependence on μ explicitly.

Let Q_μ be an orthogonal matrix that simultaneously diagonalizes X_μ and S_μ so that

$$X_\mu \;=\; Q_\mu\,\Lambda_\mu^X\,Q_\mu^t, \qquad S_\mu \;=\; Q_\mu\,\Lambda_\mu^S\,Q_\mu^t, \tag{13}$$

where the eigenvalue matrices

$$\Lambda_\mu^X \;=\; \mathrm{Diag}(\lambda_1^\mu, \ldots, \lambda_n^\mu), \qquad \Lambda_\mu^S \;=\; \mathrm{Diag}(\omega_1^\mu, \ldots, \omega_n^\mu)$$

satisfy $\lambda_i^\mu \omega_i^\mu = \mu$ and the eigenvalues are ordered such that

$$\lambda_1^\mu \geq \lambda_2^\mu \geq \cdots \geq \lambda_n^\mu, \qquad \omega_1^\mu \leq \omega_2^\mu \leq \cdots \leq \omega_n^\mu.$$

Let the Jacobian matrix in (4) for the point (X_μ, S_μ) be denoted by $\mathcal{J}_\mu$ and let $\mathcal{Q}_\mu = Q_\mu \circledast Q_\mu$. On the central path, $\mathcal{J}_\mu$ can be expressed in terms of the eigenvector matrix Q_μ and the eigenvalue matrices Λ_X and Λ_S as follows.

THEOREM 3.1. *Suppose $X_\mu S_\mu = \mu I$ with $\mu > 0$. Then $D_\mu = I$ and*

$$\mathcal{J}_\mu \;=\; \mathrm{Diag}(\mathcal{Q}_\mu, I, \mathcal{I}) \;\; \widetilde{\mathcal{J}}_\mu \;\; \mathrm{Diag}(\mathcal{Q}_\mu^t, I, \mathcal{Q}_\mu^t), \tag{14}$$

where $\mathrm{Diag}(G, H, K)$ *denotes the block diagonal matrix with blocks G, H, K, and*

$$\widetilde{\mathcal{J}}_\mu \;=\; \begin{pmatrix} 0 & (\mathcal{A}\mathcal{Q}_\mu)^t & \mathcal{I} \\ \mathcal{A}\mathcal{Q}_\mu & 0 & 0 \\ \Gamma_\mu \Lambda_\mu^S \circledast \Gamma_\mu^{-1} & 0 & \Gamma_\mu^{-1}\Lambda_\mu^X \circledast \Gamma_\mu \end{pmatrix}. \tag{15}$$

Proof. Given the decompositions of X_μ and S_μ in (13), we have

$$X_\mu^{1/2} \;=\; Q_\mu (\Lambda_\mu^X)^{1/2} Q_\mu^t, \qquad S_\mu^{1/2} \;=\; Q_\mu (\Lambda_\mu^S)^{1/2} Q_\mu^t,$$

and the resulting SVD of $X_\mu^{1/2} S_\mu^{1/2}$ is

$$X_\mu^{1/2} S_\mu^{1/2} \;=\; Q_\mu (\Lambda_\mu^X \Lambda_\mu^S)^{1/2} Q_\mu^t \;=\; Q_\mu \Sigma Q_\mu^t.$$

By our definitions, we have $\Psi = (\Lambda_\mu^X)^{1/2}$ and $\bar{G} = Q_\mu^t$. Similarly, we have $\Phi = (\Lambda_\mu^S)^{1/2}$ and $\bar{H} = Q_\mu^t$. Thus $D_\mu = \Sigma(\Phi\Psi)^{-1} = I$, and hence the matrices $\mathcal{E}$ and $\mathcal{F}$ in (11) and (12) become

$$\mathcal{E} \;=\; (\Gamma_\mu \Lambda_\mu^S \circledast \Gamma_\mu^{-1}) \mathcal{Q}_\mu^t, \qquad \mathcal{F} \;=\; (\Gamma_\mu^{-1} \Lambda_\mu^X \circledast \Gamma_\mu) \mathcal{Q}_\mu^t.$$

With the above equations, (14) is readily verified. □

Under the assumptions that there exist a primal feasible point $X \succ 0$ for (1) and a dual feasible point (y, S) with $S \succ 0$ for (2), and that the set $\{A_1, \ldots, A_m\}$ is linearly independent, it is known that the following limit exists [9]:

$$\lim_{\mu \to 0} (X_\mu, y_\mu, S_\mu) \;=\; (X_*, y_*, S_*), \tag{16}$$

and (X_*, y_*, S_*) is a solution of the primal and dual semidefinite programs. Let Q_* be a limit point (not necessarily unique) of the set $\{Q_\mu \,:\, \mu > 0\}$. Then Q_* is an orthogonal matrix that simultaneously diagonalizes X_* and S_*, with

$$X_* \;=\; Q_* \, \Lambda_*^X \, Q_*^t, \qquad S_* \;=\; Q_* \, \Lambda_*^S \, Q_*^t, \tag{17}$$

where

$$\Lambda_*^X \;=\; \mathrm{Diag}(\lambda_1^*, \ldots, \lambda_n^*), \qquad \Lambda_*^S \;=\; \mathrm{Diag}(\omega_1^*, \ldots, \omega_n^*)$$

satisfy $\lambda_i^* \omega_i^* = 0$ and

$$\lambda_1^* \geq \lambda_2^* \geq \cdots \geq \lambda_n^*, \qquad \omega_1^* \leq \omega_2^* \leq \cdots \leq \omega_n^*.$$

We are now in the position to prove the main theorem of our paper.

THEOREM 3.2. *Suppose the solution* (X_*, y_*, S_*) *of* (1) *and* (2) *satisfies the primal and dual nondegeneracy and the strict complementarity conditions defined in* [3]. *Assume that* $\lim_{\mu\to 0} \Gamma_\mu = \Gamma_*$ *and the* 2*-norm condition number* $\kappa(\Gamma_*) < \infty$. *Let* $\mathcal{Q}_* = Q_* \circledast Q_*$ *and* $\widetilde{\mathcal{J}}_*$ *be the matrix in* (15) *with* μ *replaced by* $*$*; then* $\widetilde{\mathcal{J}}_*$ *is nonsingular and* $\kappa(\mathcal{J}_\mu)$ *satisfies*

$$\lim_{\mu\to 0} \kappa(\mathcal{J}_\mu) \;=\; \kappa(\widetilde{\mathcal{J}}_*) \;<\; \infty. \tag{18}$$

Hence there exists a $\delta > 0$ *such that* $\kappa(\mathcal{J}_\mu) \le 2\kappa(\widetilde{\mathcal{J}}_*)$ *for all* $\mu \le \delta$.

Proof. For readability, we divide the proof into three parts.

(a) Under the nondegeneracy and strict complementarity assumptions, and $\kappa(\Gamma_*) < \infty$, the matrix $\widetilde{\mathcal{J}}_*$ is nonsingular; a proof of this can be found in Theorem 1 of [3], with slight modifications.

(b) It is easily shown that $\widetilde{\mathcal{J}}_\mu \to \widetilde{\mathcal{J}}_*$ as $\mu \to 0$. Since $\widetilde{\mathcal{J}}_*$ is nonsingular, the matrix $\widetilde{\mathcal{J}}_\mu$ is nonsingular and

$$\|\widetilde{\mathcal{J}}_\mu^{-1} - \widetilde{\mathcal{J}}_*^{-1}\| \;\le\; \frac{\|\widetilde{\mathcal{J}}_*^{-1}\|^2 \, \|\widetilde{\mathcal{J}}_\mu - \widetilde{\mathcal{J}}_*\|}{1 - \|\widetilde{\mathcal{J}}_*^{-1}\| \, \|\widetilde{\mathcal{J}}_\mu - \widetilde{\mathcal{J}}_*\|}$$

for μ sufficiently small. (See [6, p. 58] for a proof of the above inequality.) This implies that $\widetilde{\mathcal{J}}_\mu^{-1} \to \widetilde{\mathcal{J}}_*^{-1}$ as $\mu \to 0$.

(c) Noting that $\mathcal{Q}_\mu$ is orthogonal, we have

$$\kappa(\mathcal{J}_\mu) \;=\; \|\mathcal{J}_\mu\| \, \|\mathcal{J}_\mu^{-1}\| \;=\; \|\widetilde{\mathcal{J}}_\mu\| \, \|\widetilde{\mathcal{J}}_\mu^{-1}\|.$$

Since part (b) implies $\|\widetilde{\mathcal{J}}_\mu\| \, \|\widetilde{\mathcal{J}}_\mu^{-1}\| \to \|\widetilde{\mathcal{J}}_*\| \, \|\widetilde{\mathcal{J}}_*^{-1}\|$ as $\mu \to 0$, (18) is established.

To be precise in the above proof, we should actually present the proof as follows. Given any sequence $\{\mu_k\}$ where $\mu_k \to 0$ as $k \to \infty$, extract a subsequence $\{\mu_{k_j}\}$ such that $Q_{\mu_{k_j}} \to Q_*$ as $j \to \infty$. Then prove (a)–(c) for that subsequence. Note that since Q_* is unique up to the signs of its columns, different Q_*'s still lead to the same limit $\kappa(\widetilde{\mathcal{J}}_*)$. □

For the choice $\Gamma = D^\alpha$ where $\alpha \in \mathbb{R}$, in particular, $\Gamma = I$, $\Gamma = D^{1/2}$, and $\Gamma = D$, we have $\Gamma_\mu = I$ for (X, S) on the central path. Thus, the condition of Theorem 3.2 on Γ_μ is satisfied, and hence the associated Jacobian matrices $\mathcal{J}_\mu$ have bounded condition numbers along the central path in the limit as $\mu \to 0$.

4. Computation of the TTT* directions. The matrix P given in (10) depends on the matrices D, $\bar{G}$, and $\bar{H}$. They in turn depend on the symmetric square roots of X and S. But in practice, we can avoid the costly step of explicitly forming these symmetric square roots. The mechanics are as follows. Suppose we have the following Cholesky factorizations and SVD:

$$X \;=\; G^t G, \qquad S \;=\; H^t H, \qquad GH^t \;=\; U_1 \Sigma V_1. \tag{19}$$

Then $X^{1/2} = Q_X G$ and $S^{1/2} = Q_S H$ for some orthogonal matrices Q_X and Q_S. In addition, the SVD of $X^{1/2} S^{1/2}$ is

$$X^{1/2} S^{1/2} \;=\; Q_X G H^t Q_S \;=\; \underbrace{(Q_X U_1)}_{U} \, \Sigma \, \underbrace{(Q_S V_1)^t}_{V^t},$$

and

$$\Psi\bar{G} \;:=\; U^tX^{1/2} \;=\; U_1^tG, \qquad \Phi\bar{H} \;:=\; V^tS^{1/2} \;=\; V_1^tH.$$

Thus the matrices D, $\bar{G}$, and $\bar{H}$ can be computed by using the Cholesky factorizations of X and S instead of their symmetric square roots.

Next we shall discuss the computation of the search direction $(\Delta X, \Delta y, \Delta S)$ in a path-following method via the Schur complement approach. In this approach, block Gaussian elimination is applied to (3) to eliminate the unknowns ΔX and ΔS to obtain an equation (known as the Schur complement equation) with unknown Δy:

$$\mathcal{M}\,\Delta y \;=\; h,$$

where

$$\mathcal{M} \;=\; \mathcal{A}\,(\mathcal{E}^{-1}\mathcal{F}\mathcal{A}^t), \tag{20a}$$

$$h \;=\; r_p + \mathcal{A}\mathcal{E}^{-1}\mathcal{F}\mathbf{svec}(R_d) - \mathcal{A}\mathcal{E}^{-1}\mathbf{svec}(R_c). \tag{20b}$$

To compute the search direction $(\Delta X, \Delta y, \Delta S)$, the unknown Δy is first computed from the above Schur complement equation, and then ΔX and ΔS are computed from the equations

$$\left.\begin{array}{l} \Delta S \;=\; R_d - \mathbf{smat}(\mathcal{A}^t\Delta y), \\ \Delta X \;=\; \mathbf{smat}(\mathcal{E}^{-1}\mathbf{svec}(R_c) - \mathcal{E}^{-1}\mathcal{F}\mathbf{svec}(\Delta S)). \end{array}\right\} \tag{20c}$$

For the TTT* directions, with the matrices $\mathcal{E}$ and $\mathcal{F}$ given as in Lemma 2.2, substitution into (20a) and (20c) gives

$$\mathcal{M} \;=\; \mathcal{A}\left[(\bar{G}^t \circledast \bar{G}^t)\,\mathcal{D}_{\mathcal{M}}\,(\bar{G} \circledast \bar{G})\,\mathcal{A}^t\right], \tag{21a}$$

where $\mathcal{D}_{\mathcal{M}}$ is the diagonal matrix

$$\mathcal{D}_{\mathcal{M}} \;=\; \mathcal{D}_{\mathcal{E}}^{-1}\,\mathcal{D}_{\mathcal{F}} \;=\; \mathrm{Diag}\left(\mathbf{svec}\left(\frac{d_i^2\,\psi_i^2\gamma_j^2 + d_j^2\,\psi_j^2\gamma_i^2}{d_i^2d_j^2\,[\phi_i^2\gamma_i^2 + \phi_j^2\gamma_j^2]}\right)\right),$$

and

$$h \;=\; r_p + \mathcal{A}(\bar{G}^t \circledast \bar{G}^t)\,\mathcal{D}_{\mathcal{M}}\,\mathbf{svec}(\bar{G}R_d\bar{G}^t) - \mathcal{A}(\bar{G}^t \circledast \bar{G}^t)\,\mathcal{D}_{\mathcal{E}}^{-1}\,\mathbf{svec}(R_c), \tag{21b}$$

$$\left.\begin{array}{l} \Delta S \;=\; R_d - \mathbf{smat}(\mathcal{A}^t\Delta y), \\ \Delta X \;=\; \bar{G}^t\,\mathbf{smat}\left(\mathcal{D}_{\mathcal{E}}^{-1}\mathbf{svec}(R_c) \;-\; \mathcal{D}_{\mathcal{M}}\,\mathbf{svec}(\bar{G}\,\Delta S\,\bar{G}^t)\right)\,\bar{G}. \end{array}\right\} \tag{21c}$$

If we let $\widetilde{\mathcal{A}}$ be the matrix defined by

$$\widetilde{\mathcal{A}} \;=\; \mathcal{A}\,(\bar{G}^t \circledast \bar{G}^t) \;=\; \left[\mathbf{svec}(\bar{G}A_1\bar{G}^t) \;\cdots\; \mathbf{svec}(\bar{G}A_m\bar{G}^t)\right]^t, \tag{22}$$

then (21a)–(21b) can, respectively, be rewritten as

$$\mathcal{M} \;=\; \widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}\,\widetilde{\mathcal{A}}^t, \tag{23a}$$

$$h \;=\; r_p \;+\; \widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{M}}\,\mathbf{svec}(\bar{G}\,R_d\,\bar{G}^t) \;-\; \widetilde{\mathcal{A}}\,\mathcal{D}_{\mathcal{E}}^{-1}\mathbf{svec}(R_c). \tag{23b}$$

Thus the Schur complement matrix $\mathcal{M}$ can be computed via either (21a) or (23a). Depending on the formula used to obtain $\mathcal{M}$, a different computational complexity

for the search direction $(\Delta X, \Delta y, \Delta S)$ will result. This will be the subject of the next few paragraphs.

In each iteration of a path-following method applied to a problem with dense data, the computation of the Schur complement matrix $\mathcal{M}$ is by far the most expensive step, with complexity that is $\mathcal{O}(\min(n, m))$ times more expensive than the rest. Thus, we will concentrate our discussion just on the computational complexity of the Schur complement matrix $\mathcal{M}$.

For the TTT* directions, we will show in Appendix A that if $\mathcal{M}$ is computed via (23a) by first computing and storing $\widetilde{\mathcal{A}}$, then the complexity of each iteration is $2mn^3 + 0.5m^2n^2$ (assuming that $\widetilde{\mathcal{A}}$ is dense and ignoring lower order terms in m and n), where the first term comes from computing $\widetilde{\mathcal{A}}$ and the second from computing $\mathcal{M}$ once $\widetilde{\mathcal{A}}$ is obtained, with the symmetry of $\mathcal{M}$ taken into account.

However, we should mention that by computing $\mathcal{M}$ via $\widetilde{\mathcal{A}}$ from (23a), one cannot take advantage of possible sparsity of $\mathcal{A}$ since $\widetilde{\mathcal{A}}$ is usually dense even if $\mathcal{A}$ is sparse. If one were to take advantage of possible sparsity of $\mathcal{A}$, then (21a) should be used instead, and this will give a complexity of $4\frac{1}{3}mn^3 + 0.5m^2n^2$ if the sparsity of $\mathcal{A}$ is ignored. This complexity is also derived in Appendix A. Of course, when the sparsity of $\mathcal{A}$ is taken into account, the complexity just given would be an overestimate, and it is not immediately obvious which formula is cheaper for computing $\mathcal{M}$. In practice, one may carry out an a priori analysis similar to the analyses applied to the HKM and NT directions in [5] to decide which way is best to compute $\mathcal{M}$. However, as the main purpose of this paper is not to investigate the issue of exploiting sparsity of $\mathcal{A}$ in the computation of $\mathcal{M}$, we shall not go into such an analysis in detail. The reader is referred to Appendix B for a brief discussion of the issue of exploiting possible sparsity of $\mathcal{A}$ when computing the Schur complement matrix $\mathcal{M}$ for the various search directions.

Note that the HKM and NT directions, being members of the TTT* family, also share the same complexities given in the previous paragraphs if they are computed via (23a) or (21a). But for these two directions, (23a) and (21a) can be simplified, and typically these two directions are computed via the simplified formulas where different computational complexities are obtained; see Appendix A for details.

For the sake of comparison, Table 1 summarizes the computational complexities of the AHO direction, the TTT* directions proposed in section 2, and the HKM and NT directions. These complexities are derived in Appendix A.

TABLE 1

Computational complexity of various search directions. The second and third columns correspond to the complexities obtained when $\mathcal{M}$ is computed via (23a) *and* (20a)*, respectively. We count one addition and one multiplication each as one flop. Note that all the search directions, except the HKM direction, require an eigenvalue decomposition of a symmetric matrix in their computations.*

Directions	Complexity by using (23a)	Complexity by using (20a)
AHO	$4mn^3 + m^2n^2$	$6\frac{1}{3}mn^3 + m^2n^2$
TTT*	$2mn^3 + 0.5m^2n^2$	$4\frac{1}{3}mn^3 + 0.5m^2n^2$
HKM	$2mn^3 + m^2n^2$	$4mn^3 + 0.5m^2n^2$
NT	$mn^3 + 0.5m^2n^2$	$2mn^3 + 0.5m^2n^2$

So far, we have concentrated our discussion only on the computational complexities of various directions when computed via (23a) and (21a). However, the storage requirement for each approach should also be taken into consideration. Although computing the various directions using (23a) can lead to lower complexity compared to using (21a), we should note that the former requires additional storage space to store the matrix $\widetilde{\mathcal{A}}$, which is generally dense even if $\mathcal{A}$ is sparse. The storage space required by (23a) is at least twice the amount required by (21a). In cases where m and n are large, $\widetilde{\mathcal{A}}$ is generally a large dense matrix; this implies that storing $\widetilde{\mathcal{A}}$ would require a huge amount of additional storage space. Therefore, if storage space is limited, one should not use (23a).

5. Numerical stability of the Schur complement approach for computing the TTT* directions. In this section, we will briefly discuss the numerical stability of the Schur complement approach for computing the TTT* directions. Our intention here is not to repeat the detailed analysis already given in [7] on these topics but to strengthen some of the results obtained in [7] for our search directions. In particular, we shall show that the achievable accuracy in the duality gap for our methods satisfies $X \bullet S = \mathcal{O}(\epsilon\kappa(\widetilde{\mathcal{J}}_*))$ when $\kappa(\widetilde{\mathcal{J}}_*)$ is finite.

Assume that the factorizations of X and S are computed by a backward stable algorithm, and Δy is computed from the Schur complement equation via Cholesky factorization. With $\mathcal{M}$, h, and ΔS, ΔX computed via the expressions presented in section 4, it can be shown by adapting the error analysis for the TTT methods in [7] that the computed search direction $\widehat{\Delta\mathcal{X}}$ satisfies

$$(\mathcal{J}_\mathcal{S} + \delta\mathcal{J}_\mathcal{S})\widehat{\Delta\mathcal{X}} \;=\; \mathcal{R}_\mathcal{S} + \delta\mathcal{R}_\mathcal{S},$$

where $\mathcal{S}$ is a positive diagonal scaling matrix chosen such that the rows of $\mathcal{J}_\mathcal{S} := \mathcal{S}\mathcal{J}$ have unit norms. The perturbations $\delta\mathcal{J}_\mathcal{S}$ and $\delta\mathcal{R}_\mathcal{S}$ satisfy

$$\|\delta\mathcal{R}_\mathcal{S}\| \;=\; \mathcal{O}(\epsilon\kappa(\Sigma)\,\|\mathcal{X}\|), \qquad \|\delta\mathcal{J}_\mathcal{S}\| \;=\; \mathcal{O}(\epsilon\kappa(\Sigma)[\kappa(X)+\kappa(S)]). \tag{24}$$

Note that for (X, S) close to the central path, we have $\kappa(\Sigma) = \mathcal{O}(1)$.

Now, from standard perturbation analysis of solutions of linear systems of equations, the computed search direction $\widehat{\Delta\mathcal{X}}$ must satisfy the inequality

$$\frac{\|\widehat{\Delta\mathcal{X}} - \Delta\mathcal{X}\|}{\|\Delta\mathcal{X}\|} \;\le\; \frac{\kappa(\mathcal{J}_\mathcal{S})}{1-\kappa(\mathcal{J}_\mathcal{S})\,\|\delta\mathcal{J}_\mathcal{S}\|/\|\mathcal{J}_\mathcal{S}\|}\left(\frac{\|\delta\mathcal{J}_\mathcal{S}\|}{\|\mathcal{J}_\mathcal{S}\|}+\frac{\|\delta\mathcal{R}_\mathcal{S}\|}{\|\mathcal{R}_\mathcal{S}\|}\right). \tag{25}$$

If $\|\delta\mathcal{J}_\mathcal{S}\|$ is so large that

$$\kappa(\mathcal{J}_\mathcal{S})\,\|\delta\mathcal{J}_\mathcal{S}\|/\|\mathcal{J}_\mathcal{S}\| \;=\; \Omega(1), \tag{26}$$

then the right-hand side of (25) becomes $\Omega(1)$ or even undefined, implying that the computed search direction $\widehat{\Delta\mathcal{X}}$ could be completely different from the actual direction $\Delta\mathcal{X}$, and consequently the algorithm will not make any progress. From (24) and (26), we would expect an algorithm to stop making progress when

$$\min(\lambda_{\min}(X), \lambda_{\min}(S)) \;=\; \mathcal{O}(\epsilon)\,\kappa(\mathcal{J}_\mathcal{S})\,\frac{\max(\|X\|,\|S\|)}{\|\mathcal{J}_\mathcal{S}\|} \;=\; \mathcal{O}(\epsilon)\,\kappa(\mathcal{J}_\mathcal{S}). \tag{27}$$

Using a result of [4], namely,

$$\kappa(\mathcal{J}_\mathcal{S}) \;\le\; \sqrt{m+n(n+1)}\;\min\{\kappa(\mathcal{D}\mathcal{J}) : \mathcal{D} \text{ is a diagonal matrix}\} \;\le\; \sqrt{m+n(n+1)}\;\kappa(\mathcal{J}),$$

(27) becomes

$$\min(\lambda_{\min}(X), \lambda_{\min}(S)) \;=\; \mathcal{O}(\epsilon)\,\kappa(\mathcal{J}). \tag{28}$$

Let $\mu = X \bullet S/n$. Under the assumption of nondegeneracy and strict complementarity, we have $\lambda_{\min}(X) = \Omega(\mu) = \lambda_{\min}(S)$ if (X, S) is close to the central path. By (28), we would expect the achievable accuracy of an algorithm that computes its search direction via the Schur complement equation to be

$$\mu \;=\; \mathcal{O}(\epsilon)\,\kappa(\mathcal{J}).$$

For the search directions corresponding to $\Gamma = D^\alpha$ where $\alpha \in \mathbb{R}$, Theorem 3.2 implies that $\kappa(\mathcal{J}) = \mathcal{O}(\kappa(\widetilde{\mathcal{J}}_*))$ for (X, S) close to the central path and when μ is small. In this case, the achievable accuracy of an algorithm using these search directions is $\mu = \mathcal{O}(\epsilon)\kappa(\widetilde{\mathcal{J}}_*)$.

We end this section with two numerical examples showing $\kappa(\mathcal{J})$ and $\kappa(\mathcal{J}_\mathcal{S})$ for various search directions with (X, S) that lie approximately on the central path. The numerical values of $\kappa(\mathcal{J})$ and $\kappa(\mathcal{J}_\mathcal{S})$ are given in Table 2. In the first example, (X, S) come from a random semidefinite program with $n = 20$, $m = 20$ (generated from the routine `randsdp.m` provided in [16]), and $\|XS - \mu I\|_F = 2.2 \times 10^{-13}$ with $\mu = 10^{-10}$. In the second example, (X, S) again come from a random semidefinite program, but $n = 20$, $m = 40$, and $\|XS - \mu I\|_F = 4.8 \times 10^{-13}$ with $\mu = 10^{-10}$. Based on the numerical results, we can see that the GT direction is distinctively different from the HKM and NT directions in that the condition number of the Jacobian (scaled Jacobian) associated with the former stays bounded when μ is very small while those associated with the latter blow up.

TABLE 2
Condition number of the Jacobian and scaled Jacobian for various search directions.

	Example 1		Example 2	
Directions	$\kappa(\mathcal{J})$	$\kappa(\mathcal{J}_\mathcal{S})$	$\kappa(\mathcal{J})$	$\kappa(\mathcal{J}_\mathcal{S})$
AHO	1.18×10^4	1.45×10^3	3.67×10^3	1.29×10^3
GT	2.49×10^4	2.08×10^3	1.06×10^4	1.84×10^3
HKM	1.32×10^8	9.94×10^7	5.14×10^7	5.51×10^7
NT	3.31×10^{10}	7.56×10^8	1.02×10^{10}	2.12×10^9

6. Numerical experiments. In this section, we present numerical results showing the performance of a path-following method using the search direction corresponding to $\Gamma = D^{1/2}$ (or $P = D^{-1/2}\bar{H}$), where the Cholesky factorizations $X = G^tG$ and $S = H^tH$ are used to compute D, $\bar{G}$, and $\bar{G}$. We prefer this direction to those corresponding to $\Gamma = I$ (proposed by Gu in [7]) and $\Gamma = D$ because the former is primal-dual symmetric but the latter are not.

We compare our method to the AHO, HKM, and NT methods on seven classes of SDP problems. For each class, we considered 10 random instances. All the computations were done in MATLAB. Here are the semidefinite programs we considered (the reader is referred to [16] for a description of these problems):

1. Random semidefinite program ($n = 100$, $m = 100$).
2. Norm minimization problem ($n = 200$, $m = 26$).

3. Chebyshev approximation of a matrix ($n = 200$, $m = 26$).
4. Maxcut problem ($n = 100$, $m = 100$).
5. Educational testing problem ($n = 200$, $m = 100$).
6. Lovász θ function for a graph ($n = 50$, $m \approx 610$).
7. Logarithmic Chebyshev approximation problem ($n = 900$, $m = 201$).

All the methods used the algorithmic framework (with Mehrotra-corrector) of Algorithm IPC in [16]. For the sake of completeness, we describe Algorithm IPC in Appendix A. The reader is referred to [15] for implementation details on the computation of the AHO, HKM, and NT directions. The computation of the Schur complement matrix for the various search directions is based on the formulas given in Appendix A. We have created MATLAB C-MEX routines to compute the various specialized matrix products given in Lemmas A.1 and A.2 in order to realize the computational complexities given there. We used the default infeasible starting point described in [16] for all the numerical experiments.

As discussed in section 4, there are two mathematically equivalent ways for computing the Schur complement matrix for the search directions we consider in this section. In our experiments, the Schur complement matrix for each search direction is computed via the formula that gives a lower flops count. The reader is referred to Appendix B for details.

Our numerical results are presented in Table 2. The results show the following:

1. The method employing the GT direction (henceforth called the GT method) is generally slightly more efficient than the HKM and NT methods. It is less efficient than the AHO method, but on the average, it takes about one iteration more, except for the educational testing problem (ETP).

2. As shown in the last four columns of Table 3, the achievable accuracy in the duality gap for the GT method is almost the same as that for the AHO method, and the accuracies for both are much better than those for the HKM and NT methods.

3. The CPU time taken for the GT method to reduce the duality gap by a factor of 10^{10} is smaller than the time taken by the AHO method. For problems where $\mathcal{A}$ is dense and $m \approx n$, namely, the random SDP problem, the norm minimization problem and the Chebyshev approximation problem of a matrix, the savings in CPU time range from about 10% to 30%. For the problems where $\mathcal{A}$ is sparse and $m \approx n$, namely, the maxcut and ETP problems, the savings in CPU time is about 40%. For problems where $\mathcal{A}$ is sparse and $m \gg n$, namely, the Lovász theta function problem, the savings in CPU time is about 10%. For problems with block diagonal structure consisting of a large number of small blocks, namely, the logarithmic Chebyshev problem, the savings in CPU time is about 20%. But compared to the HKM and NT methods, the GT method is more expensive on all the problems tested.

7. Conclusion. We proposed search directions for path-following algorithms in SDP that can achieve high accuracy solutions. The computational complexity per iteration for these directions is half of that for the AHO direction. The condition numbers of the Jacobian matrices associated with our directions are shown to be bounded in the limit as the barrier parameter tends to zero, under suitable nondegeneracy and strict complementarity conditions. The practical performance of one of our directions (the GT direction) is demonstrated through extensive numerical experiments. The efficiency of this direction is comparable to the AHO direction, currently the most efficient. The CPU time required to reduce the duality gap by a fixed factor for the GT direction is either comparable to or smaller than that for the AHO direction, with savings ranging from about 10% to 40% on the SDP problems we tested.

TABLE 3

Computational results on different classes of SDP for Algorithm IPC. Ten random instances are considered for each class. The computations were done in MATLAB *on a DEC AlphaStation/*500 (333*MHz*)*. The duality gap* $X \bullet S$ *is the gap delivered by the algorithm when no further progress can be made due to the numerical stability problem. Note that the primal and dual infeasibilities were reduced to a level smaller than* 10^{-13} *for all the problems.*

	Average no. of iterations to reduce the duality gap by 10^{10}				Average CPU time (mins.) to reduce the duality gap by 10^{10}				Mean($\lvert \log_{10} X \bullet S \rvert$)			
	AHO	GT	HKM	NT	AHO	GT	HKM	NT	AHO	GT	HKM	NT
Random SDP	10.2	11.0	11.4	10.9	2.5	1.7	1.6	1.2	8.0	8.0	7.3	7.0
Norm min. problem	9.3	10.9	11.1	12.1	4.2	3.7	2.4	2.9	10.2	12.0	9.5	8.8
Cheby. approx. of a matrix	10.8	11.0	11.5	12.4	4.9	3.8	2.5	3.0	13.2	13.7	10.8	10.6
Maxcut	10.0	11.0	11.6	11.8	1.4	0.8	0.4	0.4	10.3	10.2	8.5	8.0
ETP	19.6	24.6	25.1	25.0	3.0	1.9	1.0	1.0	7.8	7.4	6.5	6.3
Lovász θ function	11.6	12.1	12.3	12.2	3.5	3.1	2.3	2.0	11.2	11.2	10.2	10.0
Log. Cheby. problem	15.0	16.7	16.7	16.6	4.0	3.1	2.3	2.3	8.8	8.7	8.9	8.8

Appendix A. The complexities (without taking possible sparsity of $\mathcal{A}$ into account) in computing the Schur complement matrix $\mathcal{M}$ for various search directions are presented in Table 1. Assuming that m and n are sufficiently large so that terms with order lower than mn^3 or n^2m^2 can be ignored, these complexities will now be derived. We begin with two lemmas which will be useful for deriving the complexities.

LEMMA A.1. *Suppose A and B are real $n \times n$ matrices. Let $C = AB$ and $\omega(C)$ be the number of flops (to leading order) required to compute C. Then*

$$\omega(C) \;=\; \begin{cases} n^3 & \text{if } C \text{ is known a priori to be symmetric,} \\ n^3 & \text{if either } A \text{ or } B \text{ is a triangular matrix,} \\ n^3/3 & \text{if both } A,B \text{ are upper triangular or both are lower triangular,} \\ 2n^3/3 & \text{if } A \text{ is upper triangular and } B \text{ is lower triangular, or vice versa.} \end{cases}$$

Proof. The proof is omitted. ∎

LEMMA A.2. *Let A be a real $n \times n$ symmetric matrix. Suppose the Aasen decomposition (which costs $n^3/3$ flops to compute) of A is given, that is, $A = P^t L\, D\, L^t P$, where L is a unit lower triangular matrix, D is a symmetric tridiagonal matrix, and P is a permutation matrix* [1]. *Then for any real $n \times n$ matrix M, the matrix product MAM^t can be computed in $2n^3$ flops to leading order.*

Proof. We have $MAM^t = (MP^tL)\, D\, (MP^tL)^t = N\, D\, N^t$, where $N = MP^tL$. The cost for computing N is n^3 flops since MP^t can be formed by permuting the columns of M and L is lower triangular. With the matrix N, the matrix $N\, D\, N^t$ can be computed with just n^3 flops since D is tridiagonal and $N\, D\, N^t$ is symmetric. This completes the proof. ∎

For the search directions we consider in this paper, the Schur complement matrix $\mathcal{M}$ can be computed via two mathematically equivalent expressions. The first is

$$\mathcal{M} \;=\; \widetilde{\mathcal{A}}\, \widetilde{\mathcal{B}}^t, \tag{29}$$

$$\text{where} \quad \widetilde{\mathcal{A}} \;=\; \Big[\mathbf{svec}(\tilde{A}_1) \;\; \cdots \;\; \mathbf{svec}(\tilde{A}_m)\Big]^t, \quad \widetilde{\mathcal{B}} \;=\; \Big[\mathbf{svec}(\widetilde{B}_1) \;\; \cdots \;\; \mathbf{svec}(\widetilde{B}_m)\Big]^t,$$

and the second is

$$\mathcal{M} \;=\; \mathcal{A}\, \mathcal{B}^t, \tag{30}$$

$$\text{where} \qquad \mathcal{B} \;=\; [\mathbf{svec}(B_1) \;\; \cdots \;\; \mathbf{svec}(B_m)]^t.$$

To compute $\mathcal{M}$ via the first expression (29), the matrix $\widetilde{\mathcal{A}}$ is first computed and stored (thus incurring additional storage); then the kth column of $\mathcal{M}$ is computed from $\widetilde{\mathcal{A}}\,\mathbf{svec}(\widetilde{B}_k)$. The computation of $\mathcal{M}$ via the second expression (30) is straightforward and does not incur additional storage. In this case, the kth column of $\mathcal{M}$ is computed simply from $\mathcal{A}\,\mathbf{svec}(B_k)$.

The matrices $\tilde{A}_k$, $\widetilde{B}_k$, and B_k for the various directions and the complexity in computing $\mathcal{M}$ from (29) and (30) are given as follows. Throughout, we assume that the Aasen decomposition of A_k is computed a priori.

TTT* directions. Referring to (7), (8), and (9) for the meaning of various variables, we have

$$\tilde{A}_k \;=\; \bar{G}\, A_k\, \bar{G}^t, \qquad \widetilde{B}_k = \boldsymbol{D} \bigodot \tilde{A}_k,$$

$$B_k \;=\; \bar{G}^t\, \widetilde{B}_k\, \bar{G},$$

where $\odot$ denotes the Hadamard product (elementwise product) and

$$\boldsymbol{D} \;=\; \left(\frac{d_i^2\,\psi_i^2\gamma_j^2 + d_j^2\,\psi_j^2\gamma_i^2}{d_i^2 d_j^2\,[\phi_i^2\gamma_i^2 + \phi_j^2\gamma_j^2]} \right).$$

By Lemma A.2, $\tilde{A}_k$ can be computed with $2n^3$ flops given the Aasen decomposition of A_k. The matrix $\widetilde{B}_k$ is obtained simultaneously since it is just a scaling of $\tilde{A}_k$. Now, given $\widetilde{B}_k$, the matrix B_k can be computed with $2\frac{1}{3}n^3$ flops via the Aasen decomposition of $\widetilde{B}_k$. Thus forming $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ takes a total of $2mn^3$ flops, and forming $\mathcal{B}$ takes $4\frac{1}{3}mn^3$ flops. With the matrices $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$, or $\mathcal{B}$ computed, forming $\mathcal{M}$ from either (29) or (30) would take another $0.5m^2n^2$ flops, with the symmetry of $\mathcal{M}$ taken into account. This completes the derivation of computational complexity presented in Table 1 for the TTT* directions.

AHO direction. Let $S = QDQ^t$ be an eigenvalue decomposition of S, where Q is orthogonal and D is diagonal. Let $\tilde{X} = Q^t X Q$. Then

$$\tilde{A}_k \;=\; Q^t\,A_k\,Q, \qquad \widetilde{B}_k \;=\; \frac{1}{2}\boldsymbol{D} \odot \left[\tilde{X}\tilde{A}_k + \tilde{A}_k\tilde{X}\right],$$

$$B_k \;=\; Q\,\widetilde{B}_k\,Q^t,$$

where

$$\boldsymbol{D} \;=\; \left(\frac{2}{d_i + d_j} \right).$$

As before, Lemma A.2 implies that $\tilde{A}_k$ can be computed with $2n^3$ flops. Given $\tilde{A}_k$, the matrix $\widetilde{B}_k$ can be computed with another $2n^3$ flops and, in turn, B_k can be formed with an additional $2\frac{1}{3}n^3$ flops from $\widetilde{B}_k$. Thus the computation of $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ takes a total of $4mn^3$ flops and the computation of $\mathcal{B}$ takes $6\frac{1}{3}mn^3$ flops. Finally, forming $\mathcal{M}$ from either $\mathcal{B}$ or $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ would take another m^2n^2 flops. This explains the complexity for the AHO direction presented in Table 1.

HKM direction. Suppose $X = G^tG$ and $S = H^tH$ are the Cholesky factorizations of X and S, respectively. Then

$$\tilde{A}_k \;=\; G\,A_k\,H^{-1}, \qquad \widetilde{B}_k \;=\; \tilde{A}_k,$$

$$B_k \;=\; \frac{1}{2}\left[S^{-1}A_kX + (S^{-1}A_kX)^t\right].$$

By Lemma A.1, $\tilde{A}_k$ can be obtained with $2n^3$ flops since G and H^{-1} are triangular matrices. The matrix $\widetilde{B}_k$ is obtained at the same time for free. Thus computing $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ takes a total of $2mn^3$ flops. But note that $\tilde{A}_k$ is not a symmetric matrix. This implies that forming each entry of $\mathcal{M}$ from $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ takes $2n^2$ flops rather than n^2 flops. Hence given $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$, computing $\mathcal{M}$ from (29) would take m^2n^2 flops. To be mathematically precise, the notation **svec** in (29) should be replaced by an isometry between $n \times n$ matrices and column vectors of length n^2.

The computation of B_k involves two ordinary matrix products, thus taking $4n^3$ flops, and forming $\mathcal{B}$ would take $4mn^3$ flops. Note that B_k is symmetric in this case. Hence, given $\mathcal{B}$, forming $\mathcal{M}$ from (30) would take $0.5m^2n^2$ flops.

Notice that to compute $\mathcal{M}$ for the HKM direction, no SVD or eigenvalue decomposition is necessary.

NT direction. Let $A_k = R_k + R_k^t$, where R_k is upper triangular. Let W be the symmetric positive definite matrix that satisfies the equation $WSW = X$. Suppose $W = U^tU$ is the Cholesky factorization of W. Then

$$\tilde{A}_k \;=\; UA_kU^t \;=\; UR_kU^t \,+\, (UR_kU^t)^t, \qquad \widetilde{B}_k \;=\; \tilde{A}_k,$$
$$B_k \;=\; WA_kW.$$

By Lemma A.1, $\tilde{A}_k$ can be obtained with n^3 flops since UR_k can be computed with $n^3/3$ flops and $(UR_k)U^t$ with $2n^3/3$ flops. Of course, there is no extra cost in getting $\widetilde{B}_k$. Thus $\widetilde{\mathcal{A}}$ and $\widetilde{\mathcal{B}}$ can be computed with just mn^3 flops, and $\mathcal{M}$ is formed with another $0.5m^2n^2$ flops from (29).

On the other hand, the matrix B_k can be computed with at most $2n^3$ flops. Therefore, forming $\mathcal{B}$ takes at most $2mn^3$ flops. Again, forming $\mathcal{M}$ would take an additional $0.5m^2n^2$ flops.

So far as we are aware, the computational complexities given here for the TTT* directions and the AHO direction are new. But some of those for the HKM and NT directions are adapted from [12]. After this paper was submitted for publication, Monteiro and Zanjácomo revised [12] and showed that the HKM and NT directions can be computed, respectively, in $4mn^3/3 + m^2n^2$ and $2mn^3/3 + m^2n^2/2$ flops if (23a) is used and, respectively, in $10mn^3/3 + m^2n^2/2$ and $5mn^3/3 + m^2n^2/2$ flops if (20a) is used. Thus, there is a reduction of $mn^3/3$ or $2mn^3/3$ flops compared to the complexities derived in this appendix. However, this reduction in the number of flops may not translate into savings in CPU time because of the extra overhead incurred in obtaining better complexities. As such, we prefer to use the procedures described in this appendix to compute the HKM and NT directions.

Appendix B. In this appendix, we compare the computational complexities of the Schur complement matrix $\mathcal{M}$ for the AHO, HKM, NT, and TTT* directions when computed via (29) and (30), with possible sparsity of $\mathcal{A}$ taken into account. Throughout, we assume that X and S are dense matrices. We shall also assume that storage space is abundant so that $\widetilde{\mathcal{A}}$ can be stored without difficulty if $\mathcal{M}$ is computed via (29).

Let ρ_k be the fraction of nonzero elements of A_k. We will first derive a condition to decide whether (29) or (30) should be used to compute the TTT* directions.

TTT* directions. Let $\omega(\tilde{A}_k)$ be the number of flops required to form $\tilde{A}_k$. Noting that each $\tilde{A}_k$ is generally dense even if A_k is sparse when X and S are dense matrices, the number of flops $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ required to form $\mathcal{M} = \widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t$ is then given by

$$\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t) \;=\; \sum_{k=1}^{m} \omega(\tilde{A}_k) \,+\, \frac{1}{2}n^2m^2, \tag{31}$$

where the first term on the right-hand side comes from computing $\tilde{A}_k$ for each k and the second comes from the dense matrix-matrix multiplication $\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t$. On the other hand, the number of flops required to form $\mathcal{M} = \mathcal{A}\mathcal{B}^t$ is given by

$$\omega(\mathcal{A}\mathcal{B}^t) \;=\; \sum_{k=1}^{m} \omega(\tilde{A}_k) \,+\, \frac{7}{3}mn^3 \,+\, \left(\sum_{k=1}^{m}\sum_{i=1}^{k} \rho_i\right) n^2, \tag{32}$$

where the first and second terms on the right-hand side come from computing $\tilde{A}_k$ and B_k, respectively, for each k, and the third term comes from the matrix-matrix

multiplication $\mathcal{A}\mathcal{B}^t$ where $\mathcal{A}$ is possibly sparse. Comparing (31) and (32), it is readily seen that $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{7}{3}\frac{n}{m} + \frac{1}{m^2}\sum_{k=1}^{m}\sum_{i=1}^{k}\rho_i \lesssim \frac{1}{2}. \tag{33}$$

Thus, for the special case where each $\rho_k = \rho$, $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{n}{m} \lesssim \frac{3}{14}(1-\rho).$$

AHO direction. Let $\omega(\tilde{A}_k)$ be the number of flops required to form $\tilde{A}_k$. An analysis similar to what we had just given for the TTT* directions will give

$$\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t) = \sum_{k=1}^{m}\omega(\tilde{A}_k) + 2mn^3 + n^2m^2,$$

and

$$\omega(\mathcal{A}\mathcal{B}^t) = \sum_{k=1}^{m}\omega(\tilde{A}_k) + \frac{13}{3}mn^3 + mn^2\sum_{k=1}^{m}\rho_k.$$

Hence $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{7}{3}\frac{n}{m} + \frac{1}{m}\sum_{k=1}^{m}\rho_k \lesssim 1, \tag{34}$$

and for the special case where each $\rho_k = \rho$, $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{n}{m} \lesssim \frac{3}{7}(1-\rho).$$

HKM direction. With reference to the notation used in Appendix A for the HKM direction, let $\omega(S^{-1}A_kX)$ and $\omega(GA_kH^{-1})$ be the number of flops required to compute $S^{-1}A_kX$ and GA_kH^{-1}, respectively. Note that since G and H are triangular matrices, we have $\omega(GA_kH^{-1}) \approx \omega(S^{-1}A_kX)/2$. It is readily shown that the number of flops $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ and $\omega(\mathcal{A}\mathcal{B}^t)$ required to form $\mathcal{M}$ via (29) and (30) are, respectively, given by

$$\begin{aligned}\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t) &= \sum_{k=1}^{m}\omega(GA_kH^{-1}) + m^2n^2 \\ &\approx \frac{1}{2}\sum_{k=1}^{m}\omega(S^{-1}A_kX) + m^2n^2\end{aligned}$$

and

$$\omega(\mathcal{A}\mathcal{B}^t) = \sum_{k=1}^{m}\omega(S^{-1}A_kX) + \left(\sum_{k=1}^{m}\sum_{i=1}^{k}\rho_i\right)n^2.$$

Thus $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{n}{m}\sum_{k=1}^{m}\frac{\omega(S^{-1}A_kX)}{m\,n^3} + \frac{2}{m^2}\sum_{k=1}^{m}\sum_{i=1}^{k}\rho_i \lesssim 2.$$

Notice that to decide whether (29) or (30) should be used, we need to know

$\omega(S^{-1}A_kX)$ for each k. But these quantities can always be estimated a priori by replacing S^{-1} and X with any random dense matrices.

NT direction. Again, with reference to the notation used in Appendix A for the NT direction, let $\omega(WA_kW)$ and $\omega(UR_kU^t)$ be the number of flops required to compute WA_kW and UA_kU^t, respectively. By taking into account the structures of the matrix products, it can be shown that $\omega(UA_kU^t) \approx \omega(WA_kW)/2$. Just like the case of the HKM direction, it is easily shown that for the NT direction, we have

$$\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t) \;\approx\; \frac{1}{2}\sum_{k=1}^{m}\omega(WA_kW) \;+\; \frac{1}{2}\,m^2n^2$$

and

$$\omega(\mathcal{A}\mathcal{B}^t) \;=\; \sum_{k=1}^{m}\omega(WA_kW) \;+\; \left(\sum_{k=1}^{m}\sum_{i=1}^{k}\rho_i\right)\,n^2.$$

Therefore, $\omega(\widetilde{\mathcal{A}}\widetilde{\mathcal{B}}^t)$ is larger than $\omega(\mathcal{A}\mathcal{B}^t)$ when

$$\frac{n}{m}\sum_{k=1}^{m}\frac{\omega(WA_kW)}{m\,n^3} \;+\; \frac{2}{m^2}\sum_{k=1}^{m}\sum_{i=1}^{k}\rho_i \;\lesssim\; 1.$$

We believe that our analysis given here for the AHO and TTT* directions is new. As for the HKM and NT directions, a detailed analysis of how to exploit possible sparsity of $\mathcal{A}$ for computing $\mathcal{M}$ via (30) is given in [5]. For these two directions, our analysis here is somewhat different from that given in [5] since we are comparing the computational complexities produced by (29) and (30) instead of focusing our attention solely on (30).

Appendix C. In this appendix, we give the algorithmic framework of the Mehrotra predictor-corrector variant of a path-following algorithm.

ALGORITHM IPC. Suppose we are given an initial iterate (X^0, y^0, S^0) with X^0, S^0 positive definite. Decide the type of symmetrization operator $H_P(\cdot)$ to use. Set $\alpha = \beta = 0$. Choose the exponent *expon*.
For $k = 0, 1, \ldots$

(Let the current and the next iterate be (X, y, S) and (X^+, y^+, S^+), respectively.)

- Let $\mu = X \bullet S/n$.
- (Predictor step)

  Solve the linear system (3) via the Schur complement approach with $\sigma = 0$, i.e., with $R_c = -H_P(XS)$ in (4). Denote the solution by $(\delta X, \delta y, \delta S)$.
- Let α_p and β_p be γ_p times the maximum step-lengths $\hat{\alpha}_p$ and $\hat{\beta}_p$ that can be taken so that $X + \hat{\alpha}_p\delta X$ and $S + \hat{\beta}_p\delta S$ remain positive definite, where

$$\gamma_p \;=\; 0.9 + 0.09\min(\alpha, \beta).$$

  Take σ to be

$$\sigma \;=\; \left[\frac{(X + \alpha_p\,\delta X) \bullet (S + \beta_p\,\delta S)}{X \bullet S}\right]^{expon}.$$

- (Corrector step)

  Compute the search direction $(\Delta X, \Delta y, \Delta S)$ from the same linear system (3), but with R_c in (4) replaced with

$$R_q = \sigma\mu I - H_P(XS) - H_P(\delta X \delta S).$$

- Let α and β be γ times the maximum step-lengths $\hat{\alpha}$ and $\hat{\beta}$ that can be taken so that $X + \hat{\alpha}\Delta X$ and $S + \hat{\beta}\Delta S$ remain positive definite, where

$$\gamma = 0.9 + 0.09 \min(\alpha_p, \beta_p).$$

  Update (X, y, S) to (X^+, y^+, S^+) by

$$X^+ = X + \alpha\,\Delta X, \quad y^+ = y + \beta\,\Delta y, \quad S^+ = S + \beta\,\Delta S.$$

In our experiments, the exponent *expon* used in updating the parameter σ for the corrector step is set to $expon = 3$ for the AHO direction, whereas for the others, *expon* is chosen adaptively based on the step-lengths taken in the predictor step, specifically,

$$expon = \begin{cases} \max[1,\, 3\min(\alpha_p, \beta_p)^2] & \text{if } \mu > 10^{-6}, \\ 1 & \text{if } \mu \le 10^{-6}. \end{cases}$$

Acknowledgments. The author thanks M. J. Todd and R. H. Tütüncü for helpful discussions. He also thanks one of the anonymous referees for constructive comments that helped to improve the presentation of this paper.

REFERENCES

[1] J. O. Aasen, *On the reduction of a symmetric matrix to tridiagonal form*, BIT, 11 (1971), pp. 233–242.

[2] F. Alizadeh, J. A. Haeberly, and M. Overton, *Complementarity and nondegeneracy in semidefinite programming*, Math. Programming, 77 (1997), pp. 111–128.

[3] 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.

[4] J. W. Demmel, *The condition number of equivalence transformations that block diagonalize matrix pencils*, SIAM J. Numer. Anal., 20 (1983), pp. 599–610.

[5] K. Fujisawa, M. Kojima, and K. Nakata, *Exploiting sparsity in primal-dual interior-point methods for semidefinite programming*, Math. Programming, 79 (1997), pp. 235–253.

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins University Press, Baltimore, MD, 1989.

[7] M. Gu, *Primal-dual interior-point methods for semidefinite programming in finite precision*, SIAM J. Optim., 10 (2000), pp. 462–502.

[8] C. Helmberg, F. Rendl, R. Vanderbei, and H. Wolkowicz, *An interior-point method for semidefinite programming*, SIAM J. Optim., 6 (1996), pp. 342–361.

[9] 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.

[10] R. D. C. Monteiro, *Primal–dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.

[11] 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.

[12] R. D. C. Monteiro and P. R. Zanjácomo, *Implementation of primal-dual methods for semidefinite programming based on Monteiro and Tsuchiya Newton directions and their variants*, Optim. Methods Software, 11 (1999), pp. 91–140.

[13] 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.
[14] M. J. Todd, *On search directions in interior-point methods for semidefinite programming*, Optim. Methods Software, 11 (1999), pp. 1–46.
[15] 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.
[16] K. C. Toh, M. J. Todd, and R. H. Tütüncü, SDPT3—a *Matlab software package for semidefinite programming, version* 1.3, Optim. Methods Software, 11 (1999), pp. 545–581.
[17] 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. 11, No. 1, pp. 243–253

A NOTE ON THE AUGMENTED HESSIAN WHEN THE REDUCED HESSIAN IS SEMIDEFINITE*

KURT M. ANSTREICHER† AND MARGARET H. WRIGHT‡

Abstract. Certain matrix relationships play an important role in optimality conditions and algorithms for nonlinear and semidefinite programming. Let H be an $n \times n$ symmetric matrix, A an $m \times n$ matrix, and Z a basis for the null space of A. (In a typical optimization context, H is the Hessian of a smooth function and A is the Jacobian of a set of constraints.) When the reduced Hessian Z^THZ is positive definite, augmented Lagrangian methods rely on the known existence of a finite $\bar{\rho} \geq 0$ such that, for all $\rho > \bar{\rho}$, the augmented Hessian $H + \rho A^TA$ is positive definite. In this note we analyze the case when Z^THZ is positive semidefinite, i.e., singularity is allowed, and show that the situation is more complicated. In particular, we give a simple necessary and sufficient condition for the existence of a finite $\bar{\rho}$ so that $H + \rho A^TA$ is positive semidefinite for $\rho \geq \bar{\rho}$. A corollary of our result is that if H is nonsingular and indefinite while Z^THZ is positive semidefinite and singular, no such $\bar{\rho}$ exists.

Key words. augmented Hessian, reduced Hessian, inertia, augmented Lagrangian methods

AMS subject classifications. 49D30, 65K05

PII. S1052623499351791

1. Introduction.

Augmented Lagrangian methods, proposed independently in the late 1960s by Hestenes [12] and Powell [18], convert a constrained optimization problem into an unconstrained problem by adding a quadratic penalty term to the Lagrangian function. In contrast to classical quadratic penalty methods, the penalty parameter need not become infinite if the solution of the constrained problem satisfies standard sufficient optimality conditions. This crucial property is a consequence of the following well-known theorem, first quoted by Finsler in 1937 [8].

THEOREM 1.1. *Let H be an $n \times n$ symmetric matrix and A an $m \times n$ matrix of rank m, where $m < n$. Let Z denote a basis for the null space of A. Then Z^THZ is positive definite if and only if there exists a finite $\bar{\rho} \geq 0$ such that, for all $\rho > \bar{\rho}$, $H + \rho A^TA$ is positive definite.*

Proofs can be found in many textbooks; see, for example, [9, 10, 17]. (We also prove this result as part of Theorem 4.2, below.) In the context of constrained optimization, H is the Hessian of a smooth function and A is the Jacobian matrix of a set of constraints. The matrix Z^THZ is usually referred to as the *reduced Hessian*; we shall call $H + \rho A^TA$ the *augmented Hessian*.

In this note we consider the augmented Hessian when Z^THZ is positive semidefinite and thus is allowed to be singular. It is natural to conjecture that in this case there always exists a finite $\bar{\rho}$ such that for all $\rho \geq \bar{\rho}$, $H + \rho A^TA$ is positive semidefinite, but we show by example that this is not true. We also give a precise characterization of when such a finite $\bar{\rho}$ exists. A corollary of our result is that when H is nonsingular and indefinite but Z^THZ is positive semidefinite and singular, no such $\bar{\rho}$ exists.

The results in this paper are closely connected with the theory of augmented Lagrangian methods for constrained nonlinear programming, and are of interest in other

*Received by the editors February 8, 1999; accepted for publication (in revised form) April 7, 2000; published electronically August 24, 2000.
http://www.siam.org/journals/siopt/11-1/35179.html
†Department of Management Sciences, College of Business, The University of Iowa, Iowa City, IA 52242 (kurt-anstreicher@uiowa.edu).
‡Bell Laboratories, Murray Hill, NJ 07974 (mhw@research.bell-labs.com).

areas of optimization as well. In particular, recent work on semidefinite programming (SDP) has made it possible to directly solve optimization problems involving a semidefiniteness condition on a matrix $H(x)$ which is itself a linear function of a vector x of real variables. Such problems are relevant in systems theory, structural optimization, eigenvalue optimization, and combinatorial optimization; see, for example, [2] and [19]. The constraint that $H(x)$ is semidefinite on the null space of a given matrix A can arise in SDP formulations, and in such a case it might be tempting to "reformulate" the constraint as semidefiniteness of $H(x) + \rho A^T A$, where ρ is an additional variable. Our result shows that this reformulation is *not* in general valid. In fact, our interest in the topic arose from an SDP application of this type; see [3].

2. Related work and applications. In continuous optimization, necessary conditions for optimality typically involve semidefiniteness of certain symmetric matrices. In particular, positive semidefiniteness of $Z^T HZ$ is necessary for existence of a minimizer of the quadratic form $\frac{1}{2}x^T Hx + g^T x$ in the null space of A. Definiteness and semidefiniteness of $Z^T HZ$ are also important in studying generalized convexity of twice-differentiable functions on affine subspaces; see [7].

Conditions characterizing semidefiniteness have been studied in linear algebra and matrix theory for decades; see [5] for a selection of references. In analyzing the quadratic form $x^T Hx$ restricted to the null space of A, [6] shows how the inertia of $Z^T HZ$ is related to that of the bordered matrix

$$\begin{pmatrix} H & A^T \\ A & 0 \end{pmatrix}$$

(usually called the "KKT matrix" in optimization) and various Schur complements. In [5], positive semidefiniteness of $Z^T HZ$ is shown to be equivalent to four different criteria involving (i) the inertia of the bordered matrix, (ii) "augmentability" (that the number of negative eigenvalues of $H - \gamma A^T A$ is exactly equal to $\operatorname{rank}(A)$ for all sufficiently large positive γ), (iii) determinants, and (iv) roots of a polynomial equation.

A complete analysis of the properties of the quadratic form $x^T Hx$ restricted to a general subspace is given in the unifying paper of Maddocks [15], which presents a wide array of inertia theorems that specialize results originally proved in an infinite-dimensional setting [14]. In analyzing stability of KdV multisolitons, where the infinite-dimensional analogues of A and H have the special property that the null space of A contains the null space of H, Lemma 2.3 of [14] gives a sharp lower bound for a penalty parameter that ensures positive semidefiniteness of the augmented Hessian when the reduced Hessian is positive semidefinite.

3. Notation and background. We consider only real matrices throughout. For a symmetric matrix K we use $\lambda_{\min}(K)$ and $\lambda_{\max}(K)$ to denote the minimal and maximal eigenvalues of K, and $\|K\|$ to denote the spectral norm. For symmetric matrices J and K, $K \succeq J$ means that $K - J$ is positive semidefinite, and $K \succ J$ means that $K - J$ is positive definite. We use $\mathcal{N}(A)$ to denote the null space of a matrix A. When K is symmetric, $\operatorname{In}(K)$ denotes the inertia of K, a triple of nonnegative integers (k_+, k_-, k_0) representing the numbers of positive, negative, and zero eigenvalues of K.

We shall invoke several known properties of symmetric matrices K and J, where

K is sometimes symmetrically partitioned as

$$K = \begin{pmatrix} K_{11} & K_{12} \\ K_{12}^T & K_{22} \end{pmatrix}. \tag{3.1}$$

The following theorem, first proved in [1] (see also [4]), plays a central role in our analysis.

THEOREM 3.1. *The symmetric matrix K, partitioned as in* (3.1)*, is positive semidefinite if and only if the following three conditions hold:*

$$\text{(i) } K_{11} \succeq 0, \qquad \text{(ii) } \mathcal{N}(K_{11}) \subseteq \mathcal{N}(K_{12}^T), \quad \textit{and} \quad \text{(iii) } K_{22} - K_{12}^T K_{11}^\dagger K_{12} \succeq 0,$$

where $K_{11}^\dagger$ is the Moore–Penrose pseudoinverse of K_{11}.

In addition to Theorem 3.1, we use the following well-known results.

Result 1. $\lambda_{\max}(K)I \succeq K \succeq \lambda_{\min}(K)I$.

Result 2 [13, Observation 7.7.7]. $K \succeq J$ implies $B^T K B \succeq B^T J B$ for any matrix B.

Result 3 [13, Corollary 7.7.4]. If K and J are positive definite, then $K \succeq J$ if and only if $J^{-1} \succeq K^{-1}$.

Result 4 [13, Theorem 7.7.6]. When K is partitioned as in (3.1), K is positive definite if and only if both K_{11} and the Schur complement $K_{22} - K_{12}^T K_{11}^{-1} K_{12}$ are positive definite.

Following common practice in optimization, Z denotes a generic basis for the null space of A, i.e., an $n \times (n-m)$ matrix with full column rank such that $AZ = 0$. Any vector y satisfying $Ay = 0$ can be written as a linear combination of the columns of Z, and the columns of A^T and Z together span all of $\mathcal{R}^n$. If Z_1 and Z_2 are two bases for the null space of A, then $Z_1 = Z_2 F$ for some nonsingular matrix F; this implies, among other things, that the inertia of $Z^T H Z$ does not depend on the choice of Z. For any condition stated in terms of Z, there is an obvious equivalent condition involving A. For example, "$Z^T H Z$ is positive definite" is equivalent to "$x^T H x > 0$ for all x satisfying $Ax = 0$."

4. A general characterization. To begin, we show by example that the obvious generalization of Theorem 1.1, with "semidefinite" replacing "definite" throughout, is not valid. Consider

$$H = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} 1 & 1 \end{pmatrix}, \quad \text{with} \quad Z \text{ taken as } \begin{pmatrix} 1 \\ -1 \end{pmatrix}. \tag{4.1}$$

Observe that $Z^T H Z = 0$ and so is positive semidefinite; note also that $AHA^T = 0$. The augmented Hessian $H + \rho A^T A$ is

$$H + \rho A^T A = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} + \rho \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} = \begin{pmatrix} \rho + 1 & \rho \\ \rho & \rho - 1 \end{pmatrix},$$

with eigenvalues $\rho \pm \sqrt{\rho^2 + 1}$. Hence $H + \rho A^T A$ is indefinite for any finite value of ρ.

In the next two subsections, we develop a general approach that leads to a theorem covering both the positive semidefinite case and the (known) positive definite case.

4.1. Preliminaries. To simplify the analysis, we assume that A has full rank, but analogous results can be obtained without this restriction; see the end of section 4.2. By definition of Z and our assumption that A has full rank, the matrix

$(A^T,\ Z)$ is nonsingular. Let $\widetilde{H}_\rho$ be defined as

$$\widetilde{H}_\rho := \begin{pmatrix} Z^T \\ A \end{pmatrix} \left(H + \rho A^T A\right) \left(Z,\ A^T\right) = \begin{pmatrix} Z^T H Z & Z^T H A^T \\ AHZ & AHA^T + \rho (AA^T)^2 \end{pmatrix}. \tag{4.2}$$

We write the eigensystem of $Z^T H Z$ as

$$Z^T H Z = (V,\ U) \begin{pmatrix} \Phi & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V^T \\ U^T \end{pmatrix},$$

where $\Phi = V^T Z^T H Z V$ is a positive diagonal matrix whose dimension is the number of positive eigenvalues of $Z^T H Z$, $(V,\ U)$ is orthogonal, and $Z^T H Z U = 0$. Let n_V and n_U denote the numbers of columns of V and U. If $Z^T H Z$ is positive definite, U is empty and $n_U = 0$; if $Z^T H Z$ is positive semidefinite and singular, then $n_U \geq 1$; and if Φ is empty, then $Z^T H Z$ must be the zero matrix. The columns of U form a basis for the null space of $Z^T H Z$, so that every nonzero vector y satisfying $Z^T H Z y = 0$ can be written as $y = Uv$ for some nonzero v.

By analogy with (4.2), we define $\bar{H}_\rho$ as

$$\begin{aligned} \bar{H}_\rho &:= \begin{pmatrix} V^T & 0 \\ U^T & 0 \\ 0 & I \end{pmatrix} \widetilde{H}_\rho \begin{pmatrix} V & U & 0 \\ 0 & 0 & I \end{pmatrix} \\ &= \begin{pmatrix} \Phi & 0 & V^T Z^T H A^T \\ 0 & 0 & U^T Z^T H A^T \\ AHZV & AHZU & AHA^T + \rho (AA^T)^2 \end{pmatrix}. \end{aligned} \tag{4.3}$$

Using Sylvester's Law of Inertia (see, for example, [13, Theorem 4.5.8]) twice, observe that

$$\mathrm{In}(H + \rho A^T A) = \mathrm{In}(\widetilde{H}_\rho) = \mathrm{In}(\bar{H}_\rho). \tag{4.4}$$

We shall obtain conditions under which there is a finite ρ such that the augmented Hessian $H + \rho A^T A$ is positive semidefinite (or positive definite) by examining the structure of $\bar{H}_\rho$. It is clear from the form of the matrix in (4.3) that if $n_U > 0$, then $\bar{H}_\rho$ cannot be positive semidefinite if $AHZU \neq 0$. We show below that when $Z^T H Z$ is positive semidefinite and singular, the condition $AHZU = 0$ is in fact necessary and sufficient for the existence of $\bar{\rho}$ such that $H + \rho A^T A \succeq 0$ for all $\rho \geq \bar{\rho}$. First, however, we describe other conditions that are equivalent to the condition $AHZU = 0$.

LEMMA 4.1. *Let H be an $n \times n$ symmetric matrix and A an $m \times n$ matrix of rank m, where $m < n$. Let Z denote a basis for the null space of A. Assume that $Z^T H Z$ is positive semidefinite and singular, and let U be a matrix whose columns are a basis for $\mathcal{N}(Z^T H Z)$. Then the following three conditions are equivalent:*

(a) $AHZU = 0$
(b) $HZU = 0$
(c) $\mathcal{N}(Z^T H Z) = \mathcal{N}(Z^T H^2 Z)$.

Proof. We first show that (a) and (b) are equivalent. Obviously (b) implies (a). For the converse, assume that t_U is any vector for which $HZUt_U \neq 0$. By definition of U as a basis for the null space of $Z^T H Z$, it must be true that $Z^T H Z U t_U = 0$. Since Z is a basis for the null space of A, every nonzero vector ζ such that $Z^T \zeta = 0$ must have the form $\zeta = A^T \zeta_A$ for a nonzero ζ_A. Therefore there is a nonzero vector ξ such that

$HZUt_U = A^T\xi$. Since A has full row rank, AA^T is positive definite, and multiplication by A gives $AHZUt_U = AA^T\xi \neq 0$. It follows that $HZU \neq 0 \implies AHZU \neq 0$.

Next we show that (b) and (c) are equivalent. First, symmetry of H implies that $Z^THZ = Z^TH^THZ$, so that $Z^THZy = 0$ only if $HZy = 0$, i.e., $\mathcal{N}(Z^THZ) = \mathcal{N}(HZ)$. Thus any y in $\mathcal{N}(Z^THZ)$ also satisfies $Z^THZy = 0$, which means that

$$\mathcal{N}(Z^THZ) \subseteq \mathcal{N}(Z^THZ). \tag{4.5}$$

Suppose now that (b) holds. Then, since any nonzero y satisfying $Z^THZy = 0$ has the form $y = Uv$, it must hold that $HZy = 0$, which implies that $Z^THZy = 0$ and so $\mathcal{N}(Z^THZ) \subseteq \mathcal{N}(Z^THZ)$. Combining this result with (4.5), (b) implies (c).

On the other hand, assume that (b) does not hold, so that $HZU \neq 0$. Then there is a vector $y \in \mathcal{N}(Z^THZ)$ for which $HZy \neq 0$. It follows that $y^TZ^THZy \neq 0$, which means that $Z^THZy \neq 0$, so that $y \notin \mathcal{N}(Z^THZ)$. Thus $\mathcal{N}(Z^THZ) \neq \mathcal{N}(Z^THZ)$. ∎

4.2. The main theorem. We are now ready for our main result. Condition (c) from Lemma 4.1 appears in the theorem, but condition (a) or (b) from that lemma could be used instead. For completeness we also restate and prove Theorem 1.1.

THEOREM 4.2. *Let H be an $n \times n$ symmetric matrix and A an $m \times n$ matrix of rank m, where $m < n$. Let Z denote a basis for the null space of A.*

(a) *If Z^THZ is positive semidefinite and singular, then there exists a finite $\bar{\rho} \geq 0$ such that $H + \rho A^TA$ is positive semidefinite for all $\rho \geq \bar{\rho}$, if and only if $\mathcal{N}(Z^THZ) = \mathcal{N}(Z^THZ)$. In this case, $H + \rho A^TA$ is singular for all ρ.*

(b) *Z^THZ is positive definite if and only if there exists a finite $\bar{\rho} \geq 0$ such that $H + \rho A^TA$ is positive definite for all $\rho > \bar{\rho}$.*

Proof. According to (4.4), positive semidefiniteness of the augmented Hessian follows from that of $\bar{H}_\rho$ of (4.3). To determine whether $\bar{H}_\rho$ is positive semidefinite, we need to check the three necessary and sufficient conditions of Theorem 3.1, where

$$K_{11} = \begin{pmatrix} \Phi & 0 \\ 0 & 0 \end{pmatrix}, \quad K_{12}^T = (AHZV, \;\; AHZU), \quad \text{and} \quad K_{22} = AHA^T + \rho(AA^T)^2.$$

Condition (i), that $K_{11} \succeq 0$, is obviously satisfied because Φ is positive definite or empty.

To check condition (ii), we observe that the null space of K_{11} consists of all vectors u satisfying

$$\begin{pmatrix} \Phi & 0 \\ 0 & 0 \end{pmatrix} \quad \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \;\; \text{so that, since } \Phi \succ 0, \;\; \mathcal{N}(K_{11}) = \left\{ \begin{pmatrix} 0 \\ u_2 \end{pmatrix} \right\} \quad \text{for any } u_2,$$

where u_1 has dimension n_V and u_2 has dimension n_U. All vectors in the null space of K_{12}^T satisfy

$$(AHZV, \;\; AHZU) \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = AHZVu_1 + AHZUu_2 = 0. \tag{4.6}$$

Clearly, an arbitrary vector $(0, \; u_2)^T$ from the null space of K_{11} will not lie in the null space of K_{12}^T if $AHZU \neq 0$. The condition that $\mathcal{N}(K_{11}) \subseteq \mathcal{N}(K_{12}^T)$ thus holds only if $AHZU = 0$ or if U is empty. Since condition (ii) of Theorem 3.1 fails independently of ρ when $AHZU \neq 0$, we conclude that $H + \rho A^TA$ is not positive semidefinite in this case.

If, on the other hand, $AHZU = 0$, condition (ii) of Theorem 3.1 holds, since any vector $(0,\ u_2)^T$ from the null space of K_{11} satisfies (4.6).

To satisfy condition (iii) of Theorem 3.1, we need to show that the generalized Schur complement

$$\widetilde{K} = K_{22} - K_{12}^T K_{11}^\dagger K_{12}$$

is positive semidefinite. If Φ is nonempty, $\widetilde{K}$ is given by

$$\widetilde{K} = AHA^T + \rho(AA^T)^2 - (AHZV,\ 0)\begin{pmatrix}\Phi & 0\\ 0 & 0\end{pmatrix}^\dagger \begin{pmatrix}V^TZ^THA^T\\ 0\end{pmatrix} \tag{4.7}$$

$$= AHA^T + \rho(AA^T)^2 - (AHZV,\ 0)\begin{pmatrix}\Phi^{-1} & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}V^TZ^THA^T\\ 0\end{pmatrix} \tag{4.8}$$

$$= AHA^T + \rho(AA^T)^2 - AHZV\Phi^{-1}V^TZ^THA^T, \tag{4.9}$$

where (4.8) includes the Moore–Penrose pseudoinverse of K_{11}. Let

$$\chi = \lambda_{\min}(H), \quad \alpha = \lambda_{\min}(AA^T), \quad \text{and} \quad \phi = \lambda_{\min}(\Phi),$$

where $\alpha > 0$ because A has independent rows and $\phi > 0$ because Φ is positive definite. Applying Results 1, 2, and 3 to (4.9), we obtain

$$\widetilde{K} \ \succeq \ A\left[(\chi + \rho\alpha)I - \frac{1}{\phi}HZZ^TH\right]A^T, \tag{4.10}$$

where we use the fact that $VV^T \preceq I$ because V has orthonormal columns. It follows that $\widetilde{K} \ \succeq \ 0$ for all $\rho \geq \bar{\rho}$, where

$$\bar{\rho} = \max(0, \tilde{\rho}), \quad \text{with} \quad \tilde{\rho} = \frac{1}{\alpha}\left(\frac{\|HZZ^TH\|}{\phi} - \chi\right). \tag{4.11}$$

When Φ is empty, i.e. Z^THZ is the zero matrix, the last term in (4.7) is zero, and the generalized Schur complement in condition (iii) of Theorem 3.1 is simply $AHA^T + \rho(AA^T)^2$. Using Results 1 and 2, this matrix is positive semidefinite for $\rho \geq \bar{\rho}$, where $\bar{\rho} = \max(0, -\chi/\alpha)$.

Thus, when $AHZU = 0$ and $\rho \geq \bar{\rho}$, we conclude from Theorem 3.1 that $\bar{H}_\rho$ is positive semidefinite and, applying (4.4), that the augmented Hessian is positive semidefinite. Because we have previously shown that $H + \rho A^TA$ cannot be positive semidefinite when $AHZU \neq 0$, the first statement in (a) now follows from Lemma 4.1.

The second statement in (a) follows because, when Z^THZ is positive semidefinite and singular, there must be a nonzero vector y satisfying $Z^THZy = 0$. Since $\mathcal{N}(Z^THZ) = \mathcal{N}(Z^TH^2Z)$, it must also hold that $HZy = 0$, so that

$$(H + \rho A^TA)Zy = HZy = 0,$$

showing that the augmented Hessian is singular for any value of ρ.

To prove (b), note that positive-definiteness of Z^THZ means that V is square and U is empty, so that $K_{11} = \Phi \succ 0$. It follows directly from (4.10) that the Schur complement $K_{22} - K_{12}^TK_{11}^{-1}K_{12}$ satisfies

$$AHA^T + \rho(AA^T)^2 - AHZ\Phi^{-1}Z^THA^T \ \succ \ 0 \quad \text{when} \quad \rho > \bar{\rho},$$

where $\bar{\rho}$ is given in (4.11). Applying Result 4, we obtain that $H + \rho A^T A$ is positive definite for sufficiently large ρ when $Z^T H Z$ is positive definite. Finally, if $H + \rho A^T A$ is positive definite, then $y^T Z^T (H + \rho A^T A) Z y = y^T Z^T H Z y > 0$ for all $y \neq 0$, and therefore $Z^T H Z$ is positive definite. □

Although we use properties of $\bar{H}_\rho$ to obtain all the results in Theorem 4.2, there is a simple alternative proof for the "only if" part of (a). It always holds that $\mathcal{N}(Z^T H^2 Z) \subseteq \mathcal{N}(Z^T H Z)$ (see (4.5) in the proof of Lemma 4.1). Suppose that $\mathcal{N}(Z^T H Z)$ contains a vector that does not lie in $\mathcal{N}(Z^T H^2 Z)$, i.e., that there is a vector y such that $Z^T H Z y = 0$ and $H Z y \neq 0$. Let L denote the augmented Hessian, $L = H + \rho A^T A$, and let $x = Z y$. Then $L x = H Z y \neq 0$ and $x^T L x = y^T Z^T H Z y = 0$. But if the symmetric matrix L is positive semidefinite, $x^T L x$ can be zero only if $L x = 0$ [13, page 400], so that existence of such a y contradicts positive semidefiniteness of L, as desired.

Regarding the lower bound $\bar{\rho}$ of (4.11), note that if the columns of Z are orthonormal, then $\|H Z Z^T H\| \leq \|H\|^2$ because $Z Z^T \preceq I$. In this case it is also easy to see that the value of $\phi = \lambda_{\min}(\Phi)$ is independent of the choice of orthonormal basis Z.

Finally, it is unnecessary to assume that the rows of A are independent. Suppose that A has rank r, with $r < m$. Let $\hat{A}$ consist of r linearly independent rows of A, so that $A^T = (\hat{A}^T, \ R)$, where R is $n \times (m - r)$, and assume that $\rho \geq 0$. Then $H + \rho A^T A = H + \rho \hat{A}^T \hat{A} + \rho R R^T$, which shows that

$$H + \rho \hat{A}^T \hat{A} \succeq 0 \implies H + \rho A^T A \succeq 0 \quad \text{and} \quad H + \rho \hat{A}^T \hat{A} \succ 0 \implies H + \rho A^T A \succ 0.$$

Furthermore, there is an $m \times r$ matrix S of rank r such that $A = S \hat{A}$. Applying Result 1, we have

$$H + \rho A^T A \ = \ H + \rho \hat{A}^T S^T S \hat{A} \ \preceq \ H + \rho \, \|S^T S\| \, \hat{A}^T \hat{A},$$

from which it follows, letting $\hat{\rho} = \rho \, \|S^T S\|$, that

$$H + \rho A^T A \succeq 0 \implies H + \hat{\rho} \, \hat{A}^T \hat{A} \succeq 0 \quad \text{and} \quad H + \rho A^T A \succ 0 \implies H + \hat{\rho} \, \hat{A}^T \hat{A} \succ 0.$$

Since the null spaces of A and $\hat{A}$ coincide, it follows easily that Theorem 4.2 holds without assuming independence of the rows of A.

4.3. Special cases. It is interesting to consider the consequences of Theorem 4.2 in two special cases.

First, assume that H is nonsingular, in which case $Z^T H^2 Z$ is positive definite, with an empty null space. Consequently, if $Z^T H Z$ is positive semidefinite and singular, then $\mathcal{N}(Z^T H Z) \neq \mathcal{N}(Z^T H^2 Z)$, and Theorem 4.2(a) shows that the augmented Hessian is not positive semidefinite for any finite ρ. Thus the augmented Hessian can be positive semidefinite only if $Z^T H Z$ is positive definite, but in this case Theorem 4.2(b) implies that the augmented Hessian is strictly positive definite for all sufficiently large ρ. Unless H is positive definite, it follows from continuity of the eigenvalues with respect to ρ [13, p. 540] that there must be a positive value of ρ for which the augmented Hessian is positive semidefinite and singular. These observations are summarized in the following corollary.

COROLLARY 4.3. *Let H be a* nonsingular *symmetric $n \times n$ matrix with at least one negative eigenvalue, and let A be an $m \times n$ matrix. Let Z denote a basis for the null space of A. Then*

(a) *if $Z^T H Z$ is positive semidefinite and singular, $H + \rho A^T A$ is not positive semidefinite for any finite ρ;*

(b) *if Z^THZ is positive definite, then $H + \bar{\rho}A^TA$ is positive semidefinite and singular for some $\bar{\rho} > 0$, and $H + \rho A^TA$ is positive definite for all $\rho > \bar{\rho}$.*

A second result involves the case when H itself is positive semidefinite.

LEMMA 4.4. *Let H be a symmetric positive semidefinite $n \times n$ matrix and A an $m \times n$ matrix. Let Z denote a basis for the null space of A. If Z^THZ is singular, then $\mathcal{N}(Z^THZ) = \mathcal{N}(Z^TH^2Z)$.*

Proof. Since H is symmetric and positive semidefinite, it follows immediately that $H + \rho A^TA \succeq 0$ for $\rho \geq 0$ and $Z^THZ \succeq 0$, and that H has a symmetric square root $H^{1/2}$. Because Z^THZ is singular, there exists a nonzero vector y such that $Z^THZy = 0$. For any such y, we have

$$Z^TH^{1/2}H^{1/2}Zy = 0 \implies H^{1/2}Zy = 0 \implies HZy = 0 \implies Z^TH^2Zy = 0,$$

and therefore $\mathcal{N}(Z^THZ) \subseteq \mathcal{N}(Z^TH^2Z)$. As previously observed (see (4.5) in the proof of Lemma 4.1), $\mathcal{N}(Z^TH^2Z)$ is always a subset of $\mathcal{N}(Z^THZ)$, so we have $\mathcal{N}(Z^THZ) = \mathcal{N}(Z^TH^2Z)$, as required. ∎

4.4. Examples. Example (4.1) shows that the augmented Hessian is not positive semidefinite when H is nonsingular and Z^THZ is positive semidefinite and singular (case (a) of Corollary 4.3). The condition that $HZU \neq 0$ (which disallows a positive semidefinite augmented Hessian) can also occur when H is singular. Consider

$$H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \text{ and } A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & -1 \end{pmatrix}, \text{ with } Z \text{ taken as } \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}. \tag{4.12}$$

Then

$$A^TA = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \quad Z^THZ = 0, \quad \text{and} \quad U = 1, \quad \text{so} \quad HZU = \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}.$$

The augmented Hessian is given by

$$H + \rho A^TA = \begin{pmatrix} \rho + 1 & 0 & -\rho \\ 0 & \rho & 0 \\ -\rho & 0 & \rho - 1 \end{pmatrix}$$

and has a negative eigenvalue for any finite ρ.

An instance in which $HZU = 0$ but H is not positive semidefinite is

$$H = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix} \text{ and } A = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix}, \text{ with } Z \text{ taken as } \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}. \tag{4.13}$$

Then

$$A^TA = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad Z^THZ = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad \text{and} \quad U = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

The augmented Hessian is

$$H + \rho A^TA = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \rho - 1 \end{pmatrix} \tag{4.14}$$

and is obviously positive semidefinite for $\rho \geq 1$. Note that, as indicated in Theorem 4.2(a), the augmented Hessian (4.14) is always singular.

To illustrate situation (b) of Corollary 4.3, where H is nonsingular but not positive definite and Z^THZ is positive definite, consider H from (4.1) with a different A:

$$H = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and} \quad A = \begin{pmatrix} 0 & 1, \end{pmatrix} \quad \text{with} \quad Z \text{ taken as } \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Then $Z^THZ = 1$ and the augmented Hessian is

$$H + \rho A^TA = \begin{pmatrix} 1 & 0 \\ 0 & \rho - 1 \end{pmatrix},$$

so that the augmented Hessian is positive semidefinite and singular for $\rho = 1$ and positive definite for $\rho > 1$.

4.5. A limiting property. A final general property is suggested by examples (4.1) and (4.12), where the smallest eigenvalue of $H + \rho A^TA$ is negative for any finite ρ but converges to zero as $\rho \to \infty$. In the next lemma we show that this is always the case when $HZU \neq 0$ and Z^THZ is positive semidefinite and singular.

LEMMA 4.5. *Let H be an $n \times n$ symmetric matrix and A an $m \times n$ matrix. Let Z denote a basis for the null space of A. Assume that Z^THZ is positive semidefinite and singular and that $HZU \neq 0$, where U is a matrix whose columns are a basis for the null space of Z^THZ. Then $\lambda_{\min}(H + \rho A^TA) \to 0$ as $\rho \to \infty$.*

Proof. Let $\{\rho_k\}$ be a sequence of monotonically increasing positive scalars with $\rho_k \to \infty$. Since $HZU \neq 0$, we know from Theorem 4.2 that $H + \rho A^TA$ has a negative eigenvalue for any finite ρ, so that $\lambda_{\min}(H+\rho_k A^TA) < 0$ for all k. If $\lambda_{\min}(H+\rho_k A^TA)$ does not converge to zero as $\rho_k \to \infty$, then there must exist a scalar $\beta > 0$ and vectors x_k, with $\|x_k\| = 1$, such that for all k,

$$x_k^T(H + \rho_k A^TA)x_k = x_k^THx_k + \rho_k\|Ax_k\|^2 \leq -\beta. \tag{4.15}$$

Letting $\bar{x}$ denote any accumulation point of $\{x_k\}$, (4.15) cannot hold as $\rho_k \to \infty$ unless $A\bar{x} = 0$ and $\bar{x}^TH\bar{x} < 0$, which contradicts the assumption that Z^THZ is positive semidefinite. ∎

5. Relationship to optimality conditions. Consider the equality-constrained quadratic program (QP) of minimizing $\frac{1}{2}x^THx + x^Tg$ subject to $Ax = b$, where A has full rank. Let x_A satisfy $AA^Tx_A = b$. As discussed in [11], if Z^THZ is positive semidefinite and singular, then the QP has weak minimizers if and only if the linear system

$$Z^THZx_Z = -Z^Tg - Z^THA^Tx_A \tag{5.1}$$

is compatible. It is interesting that there is no direct correspondence between compatibility of (5.1) and positive semidefiniteness of the augmented Hessian.

When $AHZ = 0$, we know from Lemma 4.1 and Theorem 4.2 that $H + \rho A^TA$ is positive semidefinite for sufficiently large finite ρ. However, the system (5.1), which reduces in this case to $Z^THZx_Z = -Z^Tg$, need not be compatible. In example (4.13), the QP is

$$\begin{aligned} &\text{minimize } \tfrac{1}{2}(x_1^2 - x_3^2) + x_1g_1 + x_2g_2 + x_3g_3 \\ &\text{subject to } x_3 = b. \end{aligned}$$

Using Z from (4.13), we have

$$Z^T g = \begin{pmatrix} g_1 \\ g_2 \end{pmatrix} \quad \text{and} \quad Z^T H Z = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}.$$

Thus (5.1) is compatible if and only if $g_2 = 0$, which means that x_2 does not appear in the objective function. If $g_2 \neq 0$, the system (5.1) is not compatible and no weak minimizers exist, although $H + \rho A^T A$ is positive semidefinite for sufficiently large ρ.

When $AHZU \neq 0$, there is no finite ρ for which $H + \rho A^T A$ is positive semidefinite. Even so, there are associated QPs with weak minimizers. The QP problem associated with (4.1) is

$$\begin{aligned} &\text{minimize } \tfrac{1}{2}(x_1^2 - x_2^2) + g_1 x_1 + g_2 x_2 \\ &\text{subject to } x_1 + x_2 = b. \end{aligned}$$

Since $Z^T H Z = 0$, the system (5.1) is compatible when $Z^T g + Z^T H A^T x_A = 0$, for example, when

$$g = \begin{pmatrix} -1 \\ 0 \end{pmatrix} \quad \text{and} \quad b = 1,$$

and the QP has weak minimizers of the form $(\frac{1}{2} + \beta, \frac{1}{2} - \beta)^T$ for any scalar β.

Acknowledgments. We are very grateful to John Maddocks for his enlightening discussion of many aspects of the problem and for bringing [14], [15], and [16] to our attention. We also thank an anonymous referee for helpful suggestions and for making us aware of the Finsler reference [8].

REFERENCES

[1] A. Albert, *Conditions for positive and nonnegative definiteness in terms of pseudoinverses*, SIAM J. Appl. Math., 17 (1969), pp. 434–440.
[2] F. Alizadeh, *Interior point methods in semidefinite programming with applications to combinatorial optimization*, SIAM J. Optim., 5 (1995), pp. 13–51.
[3] K. M. Anstreicher, *Eigenvalue bounds versus semidefinite relaxations for the quadratic assignment problem*, SIAM J. Optim., 11 (2000), pp. 254–265.
[4] D. Carlson, E. Haynsworth, and T. Markham, *A generalization of the Schur complement by means of the Moore–Penrose inverse*, SIAM J. Appl. Math., 26 (1974), pp. 169–175.
[5] Y. Chabrillac and J.-P. Crouzeix, *Definiteness and semidefiniteness of quadratic forms revisited*, Linear Algebra Appl., 63 (1984), pp. 283–292.
[6] R. W. Cottle, *Manifestations of the Schur complement*, Linear Algebra Appl., 8 (1974), pp. 189–211.
[7] J.-P. Crouzeix, J. A. Ferland, and S. Schaible, *Generalized convexity on affine subspaces with an application to potential functions*, Math. Programming, 56 (1992), pp. 223–232.
[8] P. Finsler, *Über das Vorkommen definiter und semidefiniter Formen und Scharen quadratischer Formen*, Commentarii Mathematici Helvetica, 9 (1937), pp. 188–192.
[9] R. Fletcher, *Practical Methods of Optimization*, 2nd ed., John Wiley and Sons, Chichester, UK, 1987.
[10] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, London, New York, 1981.
[11] N. I. M. Gould, *On practical conditions for the existence and uniqueness of solutions to the general equality quadratic programming problem*, Math. Programming, 32 (1985), pp. 90–99.
[12] M. R. Hestenes, *Multiplier and gradient methods*, J. Optim. Theory Appl., 4 (1969), pp. 303–320.
[13] R. A. Horn and C. R. Johnson, *Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1985.

[14] J. H. Maddocks, *Restricted quadratic forms and their application to bifurcation and stability in constrained variational principles*, SIAM J. Math. Anal., 16 (1985), pp. 47–68. *Corrigendum*, SIAM J. Math. Anal, 19 (1988), pp. 1256–1257.
[15] J. H. Maddocks, *Restricted quadratic forms, inertia theorems, and the Schur complement*, Linear Algebra Appl., 108 (1988), pp. 1–36.
[16] J. H. Maddocks and R. L. Sachs, *On the stability of KdV multi-solitons*, Comm. Pure Appl. Math., 46 (1993), pp. 867–901.
[17] S. G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw-Hill, New York, 1996.
[18] M. J. D. Powell, *A method for nonlinear constraints in minimization problems*, in Optimization, R. Fletcher, ed., Academic Press, London, New York, 1969, pp. 283–298.
[19] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 254–265

EIGENVALUE BOUNDS VERSUS SEMIDEFINITE RELAXATIONS FOR THE QUADRATIC ASSIGNMENT PROBLEM*

K. M. ANSTREICHER†

Abstract. It was recently demonstrated that a well-known eigenvalue bound for the quadratic assignment problem (QAP) actually corresponds to a semidefinite programming (SDP) relaxation. However, for this bound to be computationally useful, the assignment constraints of the QAP must first be eliminated and the bound then applied to a lower-dimensional problem. The resulting "projected eigenvalue bound" is one of the best available bounds for the QAP, especially when considering the quality of bounds relative to the complexity of obtaining them. In this paper we show that the projected eigenvalue bound is also related to an SDP relaxation of the original QAP. This "implicit" SDP relaxation is similar to SDP relaxations of the QAP proposed by Lin and Saigal [*On Solving Large-scale Semidefinite Programming Problems—A Case Study of Quadratic Assignment Problem*, Department of Industrial Engineering and Operations Research, University of Michigan, 1997] and Zhao et al. [J. Combin. Optim., 2 (1998), pp. 71-109].

Key words. quadratic assignment problem, eigenvalue bounds, semidefinite programming

AMS subject classifications. 90C09, 90C20

PII. S1052623499354904

1. Introduction.

The quadratic assignment problem (QAP) is a well-studied problem in discrete optimization. For recent surveys see, for example, [6], [8], and [18]. In this paper we consider the "Koopmans–Beckmann" form of the problem, which can be written

$$\begin{aligned} \mathrm{QAP}(A,B,C): \quad & \min \ \ \mathrm{tr}(AXB+C)X^T \\ & \text{subject to (s.t.) } X \in \Pi, \end{aligned}$$

where A, B, and C are $n \times n$ matrices, tr denotes the trace of a matrix, and Π is the set of $n \times n$ permutation matrices. Throughout we assume that A and B are symmetric. We write $\mathrm{QAP}(A,B)$ for the "homogenous" problem with $C = 0$. $\mathrm{QAP}(A,B,C)$ arises naturally in facility planning and can also be used to model certain other well-known combinatorial optimization problems, such as graph partitioning and the traveling salesman problem. The problem is of interest both for its applicability and its difficulty; several problems of dimension $n = 30$ have remained open for many years [7].

Algorithms that attempt to solve the QAP to optimality must incorporate both primal heuristics that obtain good feasible solutions and lower-bounding methods, in a branch-and-bound structure. At present the greatest obstacle to obtaining provably optimal solutions for QAP problems is the lack of an efficient lower-bounding method that produces reasonably tight bounds. There are a number of different classes of lower-bounding methods for the QAP, including

1. the Gilmore–Lawler bound (GLB), and related bounds;
2. bounds based on linear programming (LP) relaxations;
3. eigenvalue-based bounds;
4. bounds based on semidefinite programming (SDP).

*Received by the editors April 22, 1999; accepted for publication (in revised form) January 19, 2000; published electronically August 24, 2000.
http://www.siam.org/journals/siopt/11-1/35490.html
†Department of Management Sciences, University of Iowa, Iowa City, IA 52242 (kurt-anstreicher@uiowa.edu).

Research is active in all four of these areas. In this paper we will concentrate on eigenvalue bounds and their relationship to SDP. See, for example, [12], [14], [17], [20] and references therein for recent work on variants of the GLB and LP relaxations.

SDP refers to optimization over matrices that are constrained to be positive semidefinite. Although the potential applicability of SDP has been known for some time, it is only recently that interior-point algorithms have provided a practical solution approach. See [1] and [21] for descriptions of different types of problems that can be formulated as SDPs. SDP-based approaches to the QAP have been considered by [15] and [22]. Bounds for the QAP developed in these two papers are highly competitive, but the solution times required on modest-sized problems exceed what could realistically be expended at each node in a branch-and-bound tree.

The basic eigenvalue bound for QAP was introduced in [9] and has been modified in a variety of ways; see, for example, [11] and [19]. It was recently demonstrated [4] that the simplest eigenvalue bound for QAP(A, B) actually corresponds to a semidefinite relaxation of the problem. This result is potentially interesting because the work to obtain the eigenvalue bound is far less than that required to solve a general SDP. Unfortunately, the basic eigenvalue bound for the QAP is known to be too weak to be computationally useful. One technique for strengthening the bound, from [11], is to implicitly enforce the assignment constraints of the QAP by first projecting out, or eliminating, these constraints before applying the eigenvalue bound. The resulting "projected eigenvalue bound" is a competitive bound for many problems, especially considering the quality of the bound versus the computational effort required to obtain it.

In the next section we review eigenvalue bounds for the QAP, including the projected eigenvalue bound PB(A, B, C). In section 3 we consider the SDP interpretation of the basic eigenvalue bound proved in [4] and use this interpretation to derive a new semidefinite programming problem, SDP$^+(A, B)$. Our main result, Theorem 3.5, shows that PB(A, B, C) corresponds to first applying a simple transformation to the QAP and then using SDP$^+(\cdot, \cdot)$ to bound the quadratic term. (The linear term is bounded separately by solving a linear assignment problem.) We also demonstrate that the "implicit" semidefinite program SDP$^+(A, B)$ is closely related to SDP relaxations for the QAP proposed in [15] and [22].

Notation. We use $\operatorname{tr} A$ to denote the trace of a square matrix A, and $A \bullet B = \operatorname{tr} AB^T$. For symmetric matrices A and B we use $B \succeq A$ to denote that $B - A$ is positive semidefinite and $B \succ A$ to denote that $B - A$ is positive definite. We use e to denote a vector of arbitrary dimension with each component equal to one, and $E = ee^T$. The Kronecker product of matrices A and B is denoted $A \otimes B$. See [10] or [13] for basic properties of Kronecker products. We sometimes abuse notation by writing, for example, $e = (e \otimes e)$, where e on the left side is a vector in $\Re^{n^2}$, and each e on the right is a vector in $\Re^n$. For an $n \times n$ symmetric matrix A, $\lambda(A) \in \Re^n$ denotes the vector of eigenvalues of A. For a vector x, $\operatorname{Diag}(x)$ is the diagonal matrix with diagonal entries equal to the components of x, and for a square matrix X, $\operatorname{diag}(X)$ is the vector whose components are the diagonal entries of X.

We use $\mathcal{O}$ to denote the set of orthogonal matrices ($XX^T = X^TX = I$), $\mathcal{E}$ to denote the set of matrices with row and column sums equal to one ($Xe = X^Te = e$), and Π to denote the set of permutation matrices. The "minimal product" of two vectors x and y in $\Re^n$ is denoted $\langle x, y \rangle_-$, and is defined by

$$\langle x, y \rangle_- = \min_{\pi} \prod_{i=1}^{n} x_i y_{\pi(i)},$$

where $\pi(\cdot)$ is a permutation of $1, 2, \ldots, n$. It is easy to show that if $x_1 \leq x_2 \leq \cdots \leq x_n$, and $y_1 \geq y_2 \geq \cdots \geq y_n$, then $\langle x, y \rangle_- = x^T y$. We sometimes use $r(A)$ to denote the vector of row sums of a matrix A, $r(A) = Ae$, and we use $s(A)$ to denote the sum of the entries of A, $s(A) = e^T A e = e^T r(A)$.

Throughout the paper we use the convention of letting the name of an optimization problem, such as $\mathrm{QAP}(A, B, C)$, also refer to the solution value of the problem.

2. Eigenvalue bounds for the QAP. One well-known relaxation for the QAP is based on relaxing $X \in \Pi$ to $X \in \mathcal{O}$ and further separating the linear and quadratic terms in the objective. The result is

$$\min_{X \in \mathcal{O}} \ \mathrm{tr}\, AXBX^T + \mathrm{LAP}(C), \tag{2.1}$$

where $\mathrm{LAP}(C)$ is a linear assignment problem with cost matrix C. The relaxation (2.1) is potentially useful because a closed-form expression exists for the quadratic term. Specifically, it is known [9], [19] that

$$\min_{X \in \mathcal{O}} \ \mathrm{tr}\, AXBX^T = \langle \lambda(A), \lambda(B) \rangle_-, \tag{2.2}$$

and therefore (2.1) can be computed by performing spectral decompositions of A and B and solving $\mathrm{LAP}(C)$. Unfortunately, however, the basic eigenvalue bound from (2.1) is generally a very weak bound for $\mathrm{QAP}(A, B, C)$, even when $C = 0$.

There are several ways to improve (2.1). One approach is based on first applying perturbations to A, B, and C that leave the objective invariant for $X \in \Pi$ and then evaluating the bound from (2.1) using the perturbed data. Specifically, let

$$A' = A + eg^T + ge^T + \mathrm{Diag}(r), \tag{2.3a}$$

$$B' = B + eh^T + he^T + \mathrm{Diag}(s), \tag{2.3b}$$

$$\begin{aligned} C' = C - 2[&Aeh^T + ge^T B + gs^T + rh^T + ngh^T + (e^T g)eh^T] \\ &-[as^T + rb^T + rs^T], \end{aligned} \tag{2.3c}$$

where $a = \mathrm{diag}(A)$, $b = \mathrm{diag}(B)$, and g, h, r, and s are all in $\Re^n$. It is then easy to show that $\mathrm{QAP}(A', B', C') = \mathrm{QAP}(A, B, C)$.[1] One choice for the perturbation vectors g, h, r, and s, described in [9], [19] is based on minimizing the *spectral variance* of A' and B'. Another possibility [19], which we will refer to as the *parametric eigenvalue bound*, is obtained by approximately maximizing the eigenvalue bound

$$\langle \lambda(A'), \lambda(B') \rangle_- + \mathrm{LAP}(C')$$

over the perturbation vectors g, h, r, and s. The result is one of the strongest known bounds for the QAP, but performing the approximate maximization is difficult due to the fact that as a function of the perturbations the bound is a nondifferentiable, nonconcave function.

A different approach to improving the basic eigenvalue bound (2.1) was introduced in [11]. The idea of this improvement is to continue to work with an orthogonal relaxation of the quadratic term, but to enforce the assignment constraints $X \in \mathcal{E}$ that are ignored in (2.1). The mechanism to do so is provided by the following result.

[1] Formulas similar to those in (2.3) appear with sign errors in a number of standard references on the QAP, including [11] and [18].

PROPOSITION 2.1 (see [11, Lemma 3.1]). *Let X be an $n \times n$ matrix with $X \in \mathcal{O} \cap \mathcal{E}$. Then there is an $(n-1) \times (n-1)$ orthogonal matrix $\hat{X}$ such that $X = V\hat{X}V^T + (1/n)E$, where V is an $n \times (n-1)$ matrix whose columns are an orthonormal basis for the nullspace of e^T. Conversely, if $\hat{X}$ is an $(n-1)\times(n-1)$ orthogonal matrix, then $X = V\hat{X}V^T + (1/n)E \in \mathcal{O} \cap \mathcal{E}$.*

In [11], Proposition 2.1 is used to obtain the *projected eigenvalue bound* $\mathrm{PB}(A, B, C)$ for the QAP described in the following theorem.

THEOREM 2.2. *Let V be an $n \times (n-1)$ matrix whose columns are an orthonormal basis for the nullspace of e^T, and define $\hat{A} = V^T AV$, $\hat{B} = V^T BV$, $D = C + (2/n)r(A)r(B)^T$. Let $\mathrm{PB}(A, B, C) = \langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_- + \mathrm{LAP}(D) - s(A)s(B)/n^2$. Then*

1. $\mathrm{QAP}(A, B, C) \geq \mathrm{PB}(A, B, C)$;
2. $\mathrm{PB}(A, B) = \langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_- + (2/n)\langle r(A), r(B) \rangle_- - s(A)s(B)/n^2$;
3. *if e is an eigenvector of A or B,* $\mathrm{PB}(A, B) = \langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_- + s(A)s(B)/n^2$.

Proof. Part 1 is [11, Theorem 4.1]. Parts 2 and 3 are proved in Corollaries 4.1 and 4.2 of [11], but we describe the arguments here also. In the case that $C = 0$ we have $D = (2/n)r(A)r(B)^T$, and it is then easy to show that $\mathrm{LAP}(D) = (2/n)\langle r(A), r(B) \rangle_-$, which gives part 2. To show part 3, assume that $Ae = r(A) = \mu e$, from which it follows that $\mu = s(A)/n$. Then $\langle r(A), r(B) \rangle_- = \mu e^T r(B) = s(A)s(B)/n$, and part 2 implies part 3. The argument when e is an eigenvector of B is similar. ∎

Computational results reported in [11] show that the projected eigenvalue bound is often close to the parametric eigenvalue bound but is much more practical to compute.

3. SDP relaxations. A new interpretation of the basic eigenvalue bound (2.1) in terms of SDP was recently given in [4]. For an $n^2 \times n^2$ matrix Y, let $Y_{[ij]}$ denote the $n \times n$ matrix which is the ij "block" of Y, $i, j = 1, \ldots, n$. In other words,

$$Y = \begin{pmatrix} Y_{[11]} & \cdots & Y_{[1n]} \\ \vdots & \ddots & \vdots \\ Y_{[n1]} & \cdots & Y_{[nn]} \end{pmatrix}.$$

Define [22] the linear operators from $\Re^{n^2 \times n^2}$ to $\Re^{n \times n}$:

$$\begin{aligned} \mathrm{bdiag}(Y) &= \sum_{i=1}^{n} Y_{[ii]} \\ (\mathrm{odiag}(Y))_{ij} &= \mathrm{tr}\, Y_{[ij]}, \quad i, j = 1, \ldots, n. \end{aligned}$$

It is then easy to show that $\mathrm{bdiag}(\cdot)$ and $\mathrm{odiag}(\cdot)$ are the adjoints of the operators $S \to I \otimes S$ and $T \to T \otimes I$, from $\Re^{n \times n}$ to $\Re^{n^2 \times n^2}$, respectively. Consider the following pair of SDP problems:

$$\begin{aligned} \mathrm{SDP}(A, B): \quad & \min \ (B \otimes A) \bullet Y \\ & \text{s.t.} \ \ \mathrm{bdiag}(Y) = I, \\ & \qquad \mathrm{odiag}(Y) = I, \\ & \qquad Y \succeq 0, \end{aligned}$$

$$\begin{aligned} \mathrm{SDD}(A, B): \quad & \max \ \mathrm{tr}\, S + \mathrm{tr}\, T \\ & \text{s.t.} \ \ (I \otimes S) + (T \otimes I) \preceq (B \otimes A), \\ & \qquad S = S^T, \ T = T^T. \end{aligned}$$

THEOREM 3.1. $\mathrm{SDP}(A,B) = \mathrm{SDD}(A,B) = \langle \lambda(A), \lambda(B) \rangle_-$.

Proof. $\mathrm{SDD}(A,B) = \langle \lambda(A), \lambda(B) \rangle_-$ is proved in [4, Theorem 3.2]. $\mathrm{SDP}(A,B) = \mathrm{SDD}(A,B)$ follows from the fact that these are dual semidefinite programming problems, both of which have strictly feasible solutions; see, for example, [16, Theorem 4.2.1]. ∎

The problem $\mathrm{SDP}(A,B)$ can be viewed as a semidefinite relaxation of $\mathrm{QAP}(A,B)$. Note that $\operatorname{tr} AXBX^T = \mathbf{vec}(X)^T (B \otimes A)\, \mathbf{vec}(X) = (B \otimes A) \bullet \mathbf{vec}(X)\, \mathbf{vec}(X)^T$, and clearly $\mathbf{vec}(X)\, \mathbf{vec}(X)^T \succeq 0$. The equality constraints of $\mathrm{SDP}(A,B)$ are relaxations of the orthogonality condition on the matrix X. Specifically, if $Y = \mathbf{vec}(X)\, \mathbf{vec}(X)^T$, then $Y_{[ij]} = X_i X_j^T$, where X_i is the ith column of X. It follows that for such a Y,

$$\operatorname{bdiag}(Y) = XX^T, \qquad \operatorname{odiag}(Y) = X^T X.$$

The fact that $\langle \lambda(A), \lambda(B) \rangle_- = \mathrm{SDD}(A,B)$ can be viewed as a rather surprising Lagrangian strong duality result for a nonconvex problem [4]. It is particularly interesting that this result holds only with *both* of the conditions $\operatorname{bdiag}(Y) = I$, $\operatorname{odiag}(Y) = I$ enforced in $\mathrm{SDP}(A,B)$, despite the fact that the original constraints $XX^T = I$ and $X^T X = I$ are completely equivalent. See [3] for an analogue of Theorem 3.1 for a relaxation of $\mathrm{QAP}(A,B)$ with the semidefinite inequality $XX^T \preceq I$ in place of the orthogonality condition $XX^T = I$.

Theorem 3.1 shows that the original eigenvalue bound from (2.1) corresponds to using the SDP relaxation $\mathrm{SDP}(A,B)$ in place of the quadratic term of the QAP. Unfortunately, as noted above, it is known that this bound is in general too weak to be of computational use. However, it is obvious that Theorem 3.1 can also be applied to obtain an SDP representation for the term $\langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_-$ in the projected eigenvalue bound of Theorem 2.2. Our goal now is to show that this SDP representation of the "projected" problem can be lifted back to the original problem to obtain a new, stronger SDP relaxation of the original QAP.

From Theorem 3.1, $\langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_- = \mathrm{SDD}(\hat{A}, \hat{B})$, which can be written as

$$\begin{aligned} \mathrm{SDD}(\hat{A}, \hat{B}): \quad & \max \ \operatorname{tr} \hat{S} + \operatorname{tr} \hat{T} \\ & \text{s.t.} \ \ (I \otimes \hat{S}) + (\hat{T} \otimes I) \preceq \hat{B} \otimes \hat{A}, \end{aligned}$$

where $\hat{A} = V^T A V$, $\hat{B} = V^T B V$. (Henceforth we consider the symmetry constraints on $\hat{S}$ and $\hat{T}$ to be implicit.) But any $\hat{S}$ and $\hat{T}$ can be written in the form $\hat{S} = V^T S V$, $\hat{T} = V^T T V$ for $n \times n$ symmetric matrices S and T. Since $V^T V = I$, $\mathrm{SDD}(\hat{A}, \hat{B})$ is then equivalent to the problem

$$\begin{aligned} \max \quad & \operatorname{tr} V^T S V + \operatorname{tr} V^T T V \\ \text{s.t.} \quad & (V^T \otimes V^T)\big[(B \otimes A) - (I \otimes S) - (T \otimes I)\big](V \otimes V) \succeq 0. \end{aligned} \tag{3.1}$$

The following proposition is well known from the theory of augmented Lagrangian methods; see, for example, [2, Corollary 12.9].

PROPOSITION 3.2. *Let H be a $k \times k$ symmetric matrix, and let F be an $m \times k$ matrix. Let Z be a matrix whose columns are a basis for the nullspace of F. Then the following three conditions are equivalent:*

1. *$x^T H x > 0$ for all $x \neq 0$ having $Fx = 0$.*
2. *$Z^T H Z \succ 0$.*
3. *$H + \rho F^T F \succ 0$ for all sufficiently large ρ.*

(In our application it would be convenient if a "semidefinite" version of Proposition 3.2, with "$\geq$" replacing "$>$" in part 1, and "$\succeq$" replacing "$\succ$" in parts 2 and 3, were true. Unfortunately this is not the case; see [5].) Let F be the $2n \times n^2$ matrix

$$F = \begin{pmatrix} e^T \otimes I \\ I \otimes e^T \end{pmatrix}.$$

The matrix F arises naturally in the representation of the assignment constraints of the QAP when the matrix X is written as a vector $\mathbf{vec}(X)$. Specifically, the constraints $Xe = e$, $X^T e = e$ are exactly equivalent to $F\,\mathbf{vec}(X) = e$.

LEMMA 3.3. *The columns of $V \otimes V$ are a basis for the nullspace of F.*

Proof. The columns of $V \otimes V$ are certainly in the nullspace of F, since $(e^T \otimes I)(V \otimes V) = e^T V \otimes V = 0$, and $(I \otimes e^T)(V \otimes V) = V \otimes e^T V = 0$. Moreover, it follows from the fact that $(V, e) \otimes (V, e)$ is a nonsingular matrix that the columns of $V \otimes V$ are independent. Finally, it is very well known that the rank of F is $2n - 1$, and therefore the dimension of the nullspace of F is $n^2 - (2n - 1) = (n - 1)^2$ which is exactly the number of columns of $V \otimes V$. ∎

Motivated by Proposition 3.2 and Lemma 3.3, we define the semidefinite program

$$\begin{aligned} \widehat{\mathrm{SDD}}(A, B): \qquad & \sup\ VV^T \bullet S + VV^T \bullet T \\ & \text{s.t. } (I \otimes S) + (T \otimes I) - \rho F^T F \preceq B \otimes A. \end{aligned}$$

In the next lemma we demonstrate that $\mathrm{SDD}(\hat{A}, \hat{B})$ and $\widehat{\mathrm{SDD}}(A, B)$ are equivalent.

LEMMA 3.4. *If S, T, ρ are feasible in $\widehat{\mathrm{SDD}}(A, B)$, then $\hat{S} = V^T S V$, $\hat{T} = V^T T V$ are feasible in $\mathrm{SDD}(\hat{A}, \hat{B})$, and $\operatorname{tr}\hat{S} + \operatorname{tr}\hat{T} = VV^T \bullet S + VV^T \bullet T$. Conversely, if $\hat{S}$ and $\hat{T}$ are feasible in $\mathrm{SDD}(\hat{A}, \hat{B})$, then for every $\epsilon > 0$ there are $S_\epsilon, T_\epsilon, \rho_\epsilon$ feasible in $\widehat{\mathrm{SDD}}(A, B)$ such that $VV^T \bullet S_\epsilon + VV^T \bullet T_\epsilon = \operatorname{tr}\hat{S} + \operatorname{tr}\hat{T} - \epsilon$.*

Proof. Assume that S, T, ρ are feasible in $\widehat{\mathrm{SDD}}(A, B)$, and let $H = B \otimes A - (I \otimes S) - (T \otimes I)$. If $x \in \Re^{n^2}$ is in the nullspace of F, then $x^T H x = x^T (H + \rho F^T F) x \geq 0$, since $H + \rho F^T F \succeq 0$. Using Lemma 3.3, it follows that $(V^T \otimes V^T) H (V \otimes V) \succeq 0$, so S and T are feasible in (3.1). Therefore $\hat{S}$ and $\hat{T}$ are feasible in $\mathrm{SDD}(\hat{A}, \hat{B})$, and $\operatorname{tr}\hat{S} = VV^T \bullet S$, $\operatorname{tr}\hat{T} = VV^T \bullet T$.

Next assume that $\hat{S}, \hat{T}$ are feasible in $\mathrm{SDD}(\hat{A}, \hat{B})$. Then $S = V\hat{S}V^T$, $T = V\hat{T}V^T$ are feasible in (3.1). For $\epsilon > 0$ let $S_\epsilon = S - [\epsilon/(n-1)]I$, and $H_\epsilon = B \otimes A - (I \otimes S_\epsilon) - (T \otimes I)$. Then $\operatorname{tr} V^T S_\epsilon V + \operatorname{tr} V^T T V = \operatorname{tr}\hat{S} + \operatorname{tr}\hat{T} - \epsilon$, and $(V^T \otimes V^T) H_\epsilon (V \otimes V) \succ 0$. Applying Proposition 3.2 and Lemma 3.3, there is a ρ_ϵ so that $H_\epsilon + \rho_\epsilon F^T F \succ 0$, and therefore $S_\epsilon, T, \rho_\epsilon$ is feasible in $\widehat{\mathrm{SDD}}(A, B)$. ∎

The dual of $\widehat{\mathrm{SDD}}(A, B)$ is the semidefinite program

$$\begin{aligned} \widehat{\mathrm{SDP}}(A, B): \qquad & \min\ (B \otimes A) \bullet \bar{Y} \\ & \text{s.t. } \operatorname{bdiag}(\bar{Y}) = VV^T, \\ & \quad \operatorname{odiag}(\bar{Y}) = VV^T, \\ & \quad \bar{Y} \bullet F^T F = 0, \\ & \quad \bar{Y} \succeq 0. \end{aligned}$$

Lemma 3.4 and Theorem 3.1 imply that $\widehat{\mathrm{SDD}}(A, B) = \mathrm{SDD}(\hat{A}, \hat{B}) = \langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_-$. Since $\widehat{\mathrm{SDD}}(A, B)$ is strictly feasible (for example, take $T = 0$, $\rho = 0$, and $S = -\alpha I$ for sufficiently large α) [16, Theorem 4.2.1] then implies that $\widehat{\mathrm{SDP}}(A, B) = \widehat{\mathrm{SDD}}(A, B)$ and also that the solution value in $\widehat{\mathrm{SDP}}(A, B)$ is attained.

The problem $\widehat{\mathrm{SDP}}(A,B)$ can be viewed as an SDP relaxation of $\mathrm{QAP}(A,B)$, but this can be seen more clearly by reformulating $\widehat{\mathrm{SDP}}(A,B)$ in terms of the matrix $Y = \bar{Y} + (1/n^2)E$. It is straightforward to compute that for an $n^2 \times n^2$ matrix E,

$$\mathrm{bdiag}(E) = nE, \tag{3.2a}$$

$$\mathrm{odiag}(E) = nE, \tag{3.2b}$$

$$E \bullet F^T F = 2n^3. \tag{3.2c}$$

In addition, note that $(V,v)(V,v)^T = I$, where $v = (1/\sqrt{n})e$, and therefore $VV^T = I - vv^T = I - (1/n)E$. It follows that $\bar{Y}$ being feasible in $\widehat{\mathrm{SDP}}(A,B)$ is equivalent to $Y = \bar{Y} + (1/n^2)E$ being feasible in the SDP problem

$$\begin{aligned} \mathrm{SDP}^+(A,B): \qquad \min \ & (B \otimes A) \bullet Y \\ \text{s.t.} \ & \mathrm{bdiag}(Y) = I, \\ & \mathrm{odiag}(Y) = I, \\ & Y \bullet F^T F = 2n, \\ & Y \succeq (1/n^2)E. \end{aligned}$$

Finally,

$$\begin{aligned} (B \otimes A) \bullet E &= \mathrm{tr}(B \otimes A)(ee^T \otimes ee^T) \\ &= \mathrm{tr}(B \otimes A)(e \otimes e)(e^T \otimes e^T) \\ &= \mathrm{tr}(e^T \otimes e^T)(B \otimes A)(e \otimes e) \\ &= s(A)s(B), \end{aligned} \tag{3.3}$$

so $Y = \bar{Y} + (1/n^2)E$ implies that

$$\mathrm{SDP}^+(A,B) = \widehat{\mathrm{SDP}}(A,B) + \frac{s(A)s(B)}{n^2} = \langle \lambda(\hat{A}), \lambda(\hat{B}) \rangle_- + \frac{s(A)s(B)}{n^2}. \tag{3.4}$$

The problem $\mathrm{SDP}^+(A,B)$ is a stronger semidefinite relaxation of $\mathrm{QAP}(A,B)$ than $\mathrm{SDP}(A,B)$. The additional equality constraint $Y \bullet F^T F = 2n$ corresponds to a relaxation of the assignment constraints $Xe = X^T e = e$. Note that $\mathrm{tr}\, YF^T F = \mathrm{tr}\, FYF^T$, so the constraint $\bar{Y} \bullet F^T F = 0$ of $\widehat{\mathrm{SDP}}(A,B)$ is equivalent to $\mathrm{tr}\, F\bar{Y}F^T = 0$. However, $F\bar{Y}F^T \succeq 0$, and therefore $\mathrm{tr}\, F\bar{Y}F^T = 0$ is equivalent to $F\bar{Y}F^T = 0$. For $Y = \bar{Y} + (1/n^2 E)$, the latter is equivalent to $FYF^T = E$. Finally, if $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$, then

$$\begin{aligned} FYF^T &= \begin{pmatrix} e^T \otimes I \\ I \otimes e^T \end{pmatrix} \mathbf{vec}(X)\,\mathbf{vec}(X)^T \begin{pmatrix} e^T \otimes I \\ I \otimes e^T \end{pmatrix}^T \\ &= \begin{pmatrix} \mathbf{vec}(Xe) \\ \mathbf{vec}(e^T X) \end{pmatrix} \begin{pmatrix} \mathbf{vec}(Xe) \\ \mathbf{vec}(e^T X) \end{pmatrix}^T \\ &= \begin{pmatrix} Xe \\ X^T e \end{pmatrix} \begin{pmatrix} Xe \\ X^T e \end{pmatrix}^T, \end{aligned}$$

so $Xe = X^T e = e$ implies that $FYF^T = E$.

Comparing (3.4) with part 3 of Theorem 2.2, it is clear that $\mathrm{SDP}^+(A,B) = \mathrm{PB}(A,B)$ when e is an eigenvector of either A or B. In the next theorem we show that

in all cases $\mathrm{PB}(A,B,C)$, as defined in Theorem 2.2, corresponds to applying $\mathrm{SDP}^+(\cdot,\cdot)$ to bound the quadratic term of $\mathrm{QAP}(A,B,C)$, after a preliminary transformation (2.3) that makes e an eigenvector of A.

THEOREM 3.5. *Let* $A' = A + eg^T + ge^T$, *where* $g = (-1/n)Ae$, *and let* $C' = C + (2/n)Aee^TB = C + (2/n)r(A)r(B)^T$. *Then* $\mathrm{PB}(A,B,C) = \mathrm{SDP}^+(A',B) + \mathrm{LAP}(C')$.

Proof. Note that

$$A'e = Ae + (g^Te)e + (e^Te)g = -\frac{s(A)}{n}e,$$

so e is an eigenvector of A', and $s(A') = -s(A)$. From (3.4), and the fact that $V^TA'V = V^TAV = \hat{A}$, we then have

$$\mathrm{SDP}^+(A',B) = \langle \lambda(\hat{A}), \lambda(\hat{B})\rangle_- + \frac{s(A')s(B)}{n^2} = \langle \lambda(\hat{A}), \lambda(\hat{B})\rangle_- - \frac{s(A)s(B)}{n^2}.$$

That $\mathrm{PB}(A,B,C) = \mathrm{SDP}^+(A',B) + \mathrm{LAP}(C')$ then follows from the definition of $\mathrm{PB}(A,B,C)$. ∎

Theorem 3.5 uses a transformation of A that makes e an eigenvector of A', but it is easy to see that a similar result holds if an analogous transformation is applied to B instead. As mentioned above, the problem $\mathrm{SDP}^+(A,B)$ can be interpreted as a strengthened semidefinite relaxation of $\mathrm{QAP}(A,B)$. In particular, note that if $X \in \mathcal{O} \cap \mathcal{E}$, then $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$ is feasible for all of the equality constraints in $\mathrm{SDP}^+(A,B)$ and also has $(B \otimes A) \bullet Y = \mathbf{vec}(X)^T(B \otimes A)\,\mathbf{vec}(X) = \operatorname{tr} AXBX^T$. However, such a Y will *not* in general be feasible for the constraint $Y \succeq (1/n^2)E$. In the next theorem we show that when e is an eigenvector of A or B, $\mathrm{SDP}^+(A,B)$ *is* in fact a valid relaxation of $\mathrm{QAP}(A,B)$.

THEOREM 3.6. *Assume that* $X \in \mathcal{O} \cap \mathcal{E}$, *and let* $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$, $Y' = Y - (1/n^2)[YE + EY] + (2/n^2)E$. *Then* Y' *is feasible in* $\mathrm{SDP}^+(A,B)$. *Moreover, if* e *is an eigenvector of* A *or* B, *then* $(B \otimes A) \bullet Y' = (B \otimes A) \bullet Y$.

Proof. To begin, we will show that

$$\mathrm{bdiag}(YE) = \mathrm{bdiag}(EY) \;=\; nE, \tag{3.5a}$$

$$\mathrm{odiag}(YE) = \mathrm{odiag}(EY) \;=\; nE, \tag{3.5b}$$

$$(YE) \bullet F^TF = (EY) \bullet F^TF \;=\; 2n^3. \tag{3.5c}$$

By inspection, using the fact that $Xe = X^Te = e$, we have

$$(YE)_{[ij]} = nX_ie^T, \qquad (EY)_{[ij]} = neX_j^T,$$

for $i,j = 1,\ldots,n$, where X_i is the ith column of X. Then

$$\mathrm{bdiag}(YE) = \sum_{i=1}^{n}(YE)_{[ii]} = n(Xe)e^T = nE.$$

The argument that $\mathrm{bdiag}(EY) = nE$ is similar, proving (3.5a). Next,

$$\operatorname{tr}(YE)_{[ij]} = n \operatorname{tr} X_ie^T = ne^TX_i = n$$

for all i,j, so $\mathrm{odiag}(YE) = nE$. The argument that $\mathrm{odiag}(EY) = nE$ is similar, proving (3.5b). Finally $YE = \mathbf{vec}(X)\,\mathbf{vec}(X)^Tee^T = n\,\mathbf{vec}(X)e^T = n\,\mathbf{vec}(X)(e^T \otimes e^T)$, and $F^TF = (E \otimes I) + (I \otimes E)$, so

$$(YE) \bullet F^TF = 2n^2 \operatorname{tr}\big(\mathbf{vec}(X)(e^T \otimes e^T)\big) = 2n^2e^T\,\mathbf{vec}(X) = 2n^3,$$

and $(EY) \bullet F^T F = (YE) \bullet F^T F$, proving (3.5c). Since Y is feasible for the equality constraints in $\mathrm{SDP}^+(A,B)$, (3.2) and (3.5) together imply that Y' is also feasible for the equality constraints in $\mathrm{SDP}^+(A,B)$.

Since $X \in \mathcal{O} \cap \mathcal{E}$, Proposition 2.1 implies that there is an $(n-1) \times (n-1)$ orthogonal matrix $\hat{X}$ such that $X = V\hat{X}V^T + (1/n)E$, and therefore

$$\mathbf{vec}(X) = (V \otimes V)\,\mathbf{vec}(\hat{X}) + \frac{1}{n}e. \tag{3.6}$$

It follows that $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$ can be represented in the form

$$\begin{aligned} Y &= (V \otimes V)\,\mathbf{vec}(\hat{X})\,\mathbf{vec}(\hat{X})^T(V^T \otimes V^T) + \frac{1}{n}e\,\mathbf{vec}(\hat{X})^T(V^T \otimes V^T) \\ &\quad + \frac{1}{n}(V \otimes V)\,\mathbf{vec}(\hat{X})e^T + \frac{1}{n^2}ee^T. \end{aligned} \tag{3.7}$$

Using (3.6), and writing $\hat{Y} = \mathbf{vec}(\hat{X})\,\mathbf{vec}(\hat{X})^T$, (3.7) implies that

$$\begin{aligned} Y &= (V \otimes V)\hat{Y}(V^T \otimes V^T) + \frac{1}{n}\left[\mathbf{vec}(X)e^T + e\,\mathbf{vec}(X)^T\right] - \frac{1}{n^2}ee^T \\ &= (V \otimes V)\hat{Y}(V^T \otimes V^T) + \frac{1}{n^2}\left[YE + EY\right] - \frac{1}{n^2}E. \end{aligned}$$

It follows that

$$Y' = Y - \frac{1}{n^2}[YE + EY] + \frac{2}{n^2}E = (V \otimes V)\hat{Y}(V^T \otimes V^T) + \frac{1}{n^2}E \succeq \frac{1}{n^2}E,$$

so Y' is feasible for the semidefinite inequality constraint of $\mathrm{SDP}^+(A,B)$ as well.

Assume now that e is an eigenvector of A. Then $Ae = (s(A)/n)e$, and

$$\begin{aligned} (B \otimes A)EY &= (B \otimes A)(e \otimes e)e^T Y \\ &= n(Be \otimes Ae)\,\mathbf{vec}(X)^T \\ &= s(A)(Be \otimes e)\,\mathbf{vec}(X)^T \\ &= s(A)(B \otimes I)(e \otimes e)\,\mathbf{vec}(X)^T. \end{aligned}$$

It follows that

$$\begin{aligned} \mathrm{tr}(B \otimes A)EY &= s(A)(e^T \otimes e^T)(B \otimes I)\,\mathbf{vec}(X) \\ &= s(A)(e^T \otimes e^T)\,\mathbf{vec}(XB) \\ &= s(A)(e^T XBe) \\ &= s(A)s(B). \end{aligned} \tag{3.8}$$

Combining (3.3), (3.8), and the fact that $(B \otimes A) \bullet EY = (B \otimes A) \bullet YE$, we obtain $(B \otimes A) \bullet Y = (B \otimes A) \bullet Y'$. The proof when e is an eigenvector of B is similar. ∎

When e is an eigenvector of A or B, the form of $\mathrm{SDP}^+(A,B)$ provides an easy proof that the projected eigenvalue bound $\mathrm{PB}(A,B,C)$ cannot be lower than the basic eigenvalue bound $\langle \lambda(A), \lambda(B) \rangle_- + \mathrm{LAP}(C)$. The following lemma generalizes [11, Theorem 4.2].

LEMMA 3.7. *Assume that e is an eigenvector of A or B. Then* $\mathrm{PB}(A,B,C) \geq \langle \lambda(A), \lambda(B) \rangle_- + \mathrm{LAP}(C)$.

Proof. When e is an eigenvector of A or B it is easy to show (using an argument similar to that used to prove part 3 of Theorem 2.2) that

$$\begin{aligned}\mathrm{PB}(A,B,C) &= \langle \lambda(\hat{A}), \lambda(\hat{B})\rangle_- + s(A)s(B)/n^2 + \mathrm{LAP}(C)\\ &= \mathrm{SDP}^+(A,B) + \mathrm{LAP}(C),\end{aligned}$$

where the second equality uses (3.4). But clearly $\mathrm{SDP}^+(A,B) \geq \mathrm{SDP}(A,B)$, and $\mathrm{SDP}(A,B) = \langle \lambda(A), \lambda(B)\rangle_-$, from Theorem 3.1. □

It is quite interesting to compare the semidefinite relaxation $\mathrm{SDP}^+(A,B)$ with SDP relaxations of the QAP devised in [15] and [22]. The basic relaxation of [15] is

$$\begin{array}{ll}\min & (B\otimes A)\bullet Y\\ \text{s.t.} & Y \bullet F^TF = 2n,\\ & Y\bullet E = n^2,\\ & Y - \mathrm{diag}(Y)\,\mathrm{diag}(Y)^T \succeq 0.\end{array} \tag{3.9}$$

It is well known that the nonlinear semidefinite inequality $Y - \mathrm{diag}(Y)\,\mathrm{diag}(Y)^T \succeq 0$ can be expressed as a linear semidefinite constraint, for example by writing $\mathcal{A}(Y) \succeq -e_0e_0^T$, where

$$\mathcal{A}(Y) = \begin{pmatrix} 0 & \mathrm{diag}(Y)^T\\ \mathrm{diag}(Y) & Y\end{pmatrix},$$

and e_0 is the unit vector in $\Re^{n^2+1}$ with a one in the zeroth position. Note that (3.9) does not impose the constraints $\mathrm{bdiag}(Y) = I$, $\mathrm{odiag}(Y) = I$ of $\mathrm{SDP}^+(A,B)$. In addition, it is easy to see that the constraint $\bar{Y}\bullet F^TF = 0$ of $\widehat{\mathrm{SDP}}(A,B)$ implies that $e^TF\bar{Y}F^Te = 0$ (see the discussion following (3.4)), which is equivalent to $\bar{Y}\bullet E = 0$. As a result the constraint $\bar{Y}\bullet E = n^2$ would be redundant in $\mathrm{SDP}^+(A,B)$. These observations suggest that the bound from (3.9) could be inferior to $\mathrm{PB}(A,B)$ in some cases, as can be verified by comparing the bounds in Table 1 of [15] with the projected eigenvalue bounds for the same problems (see, for example, Table 3 of [22]).

For a homogenous problem ($C = 0$) the basic SDP bound of [22] is very similar to $\mathrm{SDP}^+(A,B)$, the main difference being the representation of the assignment constraints. (The general construction of [22] also provides an SDP bound for problems with $C \neq 0$, but we omit the details here.) Since the original assignment constraints of the QAP can be written $F\,\mathbf{vec}(X) - e = 0$, where $e \in \Re^{2n}$, it is certainly true that

$$\mathbf{vec}(X)^TF^TF\,\mathbf{vec}(X) - 2e^TF\,\mathbf{vec}(X) + 2n = 0. \tag{3.10}$$

For $X \in \Pi$ and $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$ we have $\mathbf{vec}(X) = \mathrm{diag}(Y)$ and $e^TF = 2e^T$, so (3.10) can be written

$$Y\bullet F^TF - 4e^T\,\mathrm{diag}(Y) + 2n = 0.$$

The basic SDP relaxation for $\mathrm{QAP}(A,B)$ from [22] is

$$\begin{array}{ll}\min & (B\otimes A)\bullet Y\\ \text{s.t.} & \mathrm{bdiag}(Y) = I,\\ & \mathrm{odiag}(Y) = I,\\ & Y\bullet F^TF - 4e^T\,\mathrm{diag}(Y) = -2n,\\ & Y - \mathrm{diag}(Y)\,\mathrm{diag}(Y)^T \succeq 0.\end{array} \tag{3.11}$$

Note, however, that $\operatorname{bdiag}(Y) = I$ implies that $e^T \operatorname{diag}(Y) = n$, so the constraint $Y \bullet F^T F - 4e^T \operatorname{diag}(Y) = -2n$ of (3.11) is actually equivalent to $Y \bullet F^T F = 2n$. In the computational results reported in [22], the bound from (3.11) is never worse than $\mathrm{PB}(A, B)$. However, it is interesting to note that there are several problems for which these two bounds coincide (see Tables 1 and 3 of [22]).

Both [15] and [22] consider strengthenings of the basic SDP bounds in (3.9) and (3.11), respectively. These improved bounds are based on two classes of constraints that are valid for $Y = \mathbf{vec}(X)\,\mathbf{vec}(X)^T$, $X \in \Pi$:

1. all components of Y should be nonnegative;
2. certain components of Y should be zero.

By imposing additional constraints of one or both of the above types, [15] and [22] obtain substantial improvements over the basic SDP bounds from (3.9) and (3.11). Unfortunately, however, the computational cost of obtaining these improved bounds is considerable.

4. Conclusion. We have shown that the well-known projected eigenvalue bound for the QAP corresponds to first applying a simple transformation to the problem and then using a semidefinite relaxation to bound the quadratic term. The implicit semidefinite relaxation is closely related to SDP relaxations for the QAP proposed in [15] and [22]. In addition to its purely theoretical interest, there are several possible applications for this result. For example, because the projected eigenvalue bound corresponds to a particular $X \in \mathcal{O} \cap \mathcal{E}$, this "solution" might be useful for warm-starting a stronger SDP relaxation of the QAP. In addition, the knowledge that the bound corresponds to implicitly solving a convex optimization problem may make it possible to derive stronger bounds for the QAP that do not require explicit solution of an SDP.

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. Avriel, *Nonlinear Programming: Analysis and Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1976.

[3] K. M. Anstreicher, X. Chen, H. Wolkowicz, and Y. Yuan, *Strong duality for a trust-region type relaxation of the quadratic assignment problem*, Linear Algebra Appl., 301 (1999), pp. 121–136.

[4] K. M. Anstreicher and H. Wolkowicz, *On Lagrangian relaxation of quadratic matrix constraints*, SIAM J. Matrix Anal. Appl., 22 (2000), pp. 41–55.

[5] K. M. Anstreicher and M. H. Wright, *A note on the augmented Hessian when the reduced Hessian is semidefinite*, SIAM J. Optim., 11 (2000), pp. 243–253.

[6] R. E. Burkard, E. Çela, P. M. Pardalos, and L. S. Pitsoulis, *The quadratic assignment problem*, in Handbook of Combinatorial Optimization, Vol. 3, D.-Z. Zhu and P. M. Pardalos, eds., Kluwer, Dordrecht, The Netherlands, 1998, pp. 241–337.

[7] R. E. Burkard, S. E. Karisch, and F. Rendl, *QAPLIB—A quadratic assignment problem library*, J. Global Optim., 10 (1997), pp. 391-403.

[8] E. Çela, *The Quadratic Assignment Problem: Theory and Algorithms*, Kluwer, Dordrecht, The Netherlands, 1998.

[9] G. Finke, R. Burkard, and F. Rendl, *Quadratic assignment problems*, Ann. Discrete Math. 31, North-Holland, Amsterdam, 1987, pp. 61–82.

[10] A. Graham, *Kronecker Products and Matrix Calculus: With Applications*, Ellis Horwood, Chichester, UK, 1981.

[11] S. Hadley, F. Rendl, and H. Wolkowicz, *A new lower bound via projection for the quadratic assignment problem*, Math. Oper. Res., 17 (1992), pp. 727–739.

[12] P. Hahn and T. Grant, *Lower bounds for the quadratic assignment problem based upon a dual formulation*, Oper. Res., 46 (1998), pp. 912–922.

[13] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, UK, 1991.

[14] S. E. Karisch, E. Çela, J. Clausen, and T. Espersen, *A dual framework for lower bounds of the quadratic assignment problem based on linearization*, Computing, 63 (1999), pp. 351–403.

[15] C.-J. Lin and R. Saigal, *On Solving Large-scale Semidefinite Programming Problems—A Case Study of Quadratic Assignment Problem*, Working paper, Dept. of Industrial Engineering and Operations Research, University of Michigan, Ann Arbor, MI, 1997.

[16] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[17] P. M. Pardalos, K. G. Ramakrishnan, M. G. C. Resende, and Y. Li, *Implementation of a variance reduction-based lower bound in a branch-and-bound algorithm for the quadratic assignment problem*, SIAM J. Optim., 7 (1997), pp. 280–294.

[18] P. Pardalos, F. Rendl, and H. Wolkowicz, *The quadratic assignment problem: A survey and recent developments*, in Proceedings of the DIMACS Workshop on Quadratic Assignment Problems, DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 16, AMS, Providence, RI, 1994, pp. 1–41.

[19] F. Rendl and H. Wolkowicz, *Applications of parametric programming and eigenvalue maximization to the quadratic assignment problem*, Math. Programming, 53 (1992), pp. 63–79.

[20] M. G. C. Resende, K. G. Ramakrishnan, and Z. Drezner, *Computing lower bounds for the quadratic assignment problem with an interior point algorithm for linear programming*, Oper. Res., 43 (1995), pp. 781–791.

[21] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[22] Q. Zhao, S. Karisch, F. Rendl, and H. Wolkowicz, *Semidefinite programming relaxations for the quadratic assignment problem*, J. Combin. Optim., 2 (1998), pp. 71–109.

SIAM J. OPTIM.
Vol. 11, No. 1, pp. 266–288

MULTIPLE CUTS IN THE ANALYTIC CENTER CUTTING PLANE METHOD*

JEAN-LOUIS GOFFIN† AND JEAN-PHILIPPE VIAL‡

Abstract. We analyze the multiple cut generation scheme in the analytic center cutting plane method. We propose an optimal primal and dual updating direction when the cuts are central. The direction is optimal in the sense that it maximizes the product of the new dual slacks and of the new primal variables within the trust regions defined by Dikin's primal and dual ellipsoids. The new primal and dual directions use the variance-covariance matrix of the normals to the new cuts in the metric given by Dikin's ellipsoid.

We prove that the recovery of a new analytic center from the optimal restoration direction can be done in $O(p\log(p+1))$ damped Newton steps, where p is the number of new cuts added by the oracle, which may vary with the iteration. The results and the proofs are independent of the specific scaling matrix—primal, dual, or primal-dual—that is used in the computations.

The computation of the optimal direction uses Newton's method applied to a self-concordant function of p variables.

The convergence result of [Ye, *Math. Programming*, 78 (1997), pp. 85–104] holds here also: the algorithm stops after $O^*(\frac{\bar{p}^2 n^2}{\varepsilon^2})$ cutting planes have been generated, where $\bar{p}$ is the maximum number of cuts generated at any given iteration.

Key words. primal Newton algorithm, analytic center, cutting plane method, multiple cuts, interior-point methods, self-concordance

AMS subject classifications. 90C06, 90C25

PII. S1052623498340266

1. Introduction.

The analytic center cutting plane (ACCPM) algorithm [5, 18] is an efficient algorithm in practice [2, 4]. The complexity of related algorithms was given in [1, 13] and subsequently in [6]. Extensions to deep cuts were given in [7] and to very deep cuts in [8]. The method studied in [8] corresponds to the practical implementation of ACCPM [11] with a single cut.

In practice, it often occurs that the oracle in the cutting plane scheme generates multiple cuts. The paper by Ye [19] shows that it is possible to handle several cuts at a time provided they are central; the direction used is the primal direction suggested by Mitchell and Todd [12]. Although this analysis shows how one can recover feasibility after introducing multiple cuts, there is no clear argument as to the choice of a feasibility restoration direction. Intuitive, but well justified, arguments about how to introduce multiple cuts were given in [2] in the context of a primal projective algorithm and two cuts (one shallow, one deep) and in [10] with an infeasible primal-dual approach to the introduction of several cuts in general position.

The case of two central cuts was analyzed in [9]. It was shown that there exist explicit primal-dual directions which allow a best move towards primal-dual feasibility. An argument using the primal, dual, and primal-dual potentials at this new optimal

*Received by the editors June 4, 1998; accepted for publication (in revised form) October 4, 1999; published electronically August 24, 2000. This work was completed with support from the Fonds National Suisse de la Recherche Scientifique, grant 12-42503.94, the Natural Sciences and Engineering Research Council of Canada, grant OPG0004152, and the FCAR of Quebec.

http://www.siam.org/journals/siopt/11-1/34026.html

†GERAD/Faculty of Management, McGill University, 1001 Sherbrooke West, Montreal, Quebec, H3A 1G5, Canada (jlg@crt.umontreal.ca).

‡LOGILAB/Management Studies, University of Geneva, 102 Bd Carl-Vogt, CH-1211 Genève 4, Switzerland (jpvial@uni2a.unige.ch).

primal-dual point proves that $O(1)$ damped Newton steps are enough to recover centrality. The updating direction depends on the cosine in the metric of Dikin's ellipsoid of the normals to the cuts.

In this paper, we analyze the multiple central cut generation scheme in the analytic center cutting plane method. An approach based upon weighted potentials applied to the primal direction proposed in [12] and studied in [19] leads to a number of recentering steps that depends upon the total number of cuts and on the data.

We propose an optimal updating direction when the cuts are central. The direction is optimal in the sense that it maximizes the product of the new slacks within the trust region defined by Dikin's ellipsoid. The new primal-dual directions use the variance-covariance matrix of the normals to the new cuts in the metric given by Dikin's ellipsoid.

We prove that the recovery of a new analytic center from the optimal restoration point can be done in $O(p \log(p+1))$ damped Newton steps, where p is the number of new cuts added by the oracle; the number of cuts may vary at each iteration. The number of damped Newton steps does not depend upon the data, i.e., it is strongly polynomial.

The results and the proofs are independent of the specific scaling matrix—primal, dual, or primal-dual—that is used in the computations. The proof of a complexity that does not depend on the data of the problem relies on the use of the primal-dual potential function as a proximity measure.

The computation of the optimal direction uses Newton's method applied to a self-concordant function of p variables. The number of iterations needed is polynomial in the problem data; but this could be very advantageous in practice if the number of cuts p is small with respect to n, the dimension of the space, as Newton's method takes place in a p-dimensional space.

The argument made here is very classical in nonlinear optimization and involves computing an optimal direction within a trust region, here defined by Dikin's ellipsoid, and only then searching for the new center.

The convergence result of [19] holds here also: the algorithm stops after $O^*(\frac{\bar{p}^2 n^2}{\varepsilon^2})$ cutting planes have been generated, where $\bar{p}$ is the maximum number of cuts generated by the oracle at any iteration. No improvement on this result can be offered here, as the worst case answer from the oracle is p copies of the same cutting plane, in which case the optimal direction proposed here is the same as the one studied in [19]. The long-step analysis given in [19] shows an average number of Newton steps of $O(\bar{p})$.

2. Analytic center cutting plane method.

2.1. Cutting planes. The problem of interest is that of finding a point in a convex set $C \subset \mathbb{R}^n$. We make the following assumptions.

ASSUMPTION 2.1. *The set C is convex, contains a ball of radius $\varepsilon > 0$, and is contained in the cube $0 \leq y \leq e$.*

ASSUMPTION 2.2. *The set C is described by an oracle. That is, the oracle either confirms that $y \in C$, or answers at least one cutting plane that contains C and does not contain y in its interior.*

A cut at $\bar{y} \notin C$ takes the form

$$a^T y \leq a^T \bar{y} - \bar{\gamma}.$$

If $\bar{\gamma} > 0$, the cut is deep; if $\bar{\gamma} < 0$, the cut is shallow; if $\bar{\gamma} = 0$, the cut passes through $\bar{y}$, and we will refer to this as a central cut.

The algorithm may generate multiple cuts at a time. They take the form

$$a_j^T y \le a_j^T \bar{y} - \bar{\gamma}_j,\ j = 1, \ldots, p \quad \forall y \in C.$$

We define the matrix B by

$$B = (a_1, a_2, \ldots, a_p);$$

p may vary at each iteration and, when necessary, this shall be denoted as p_k.

ASSUMPTION 2.3. *All the cutting planes generated have been scaled so that* $\|a\| = 1$ *(wlog). We also assume that* $\bar{\gamma} = 0$*, and thus that all cuts go through* $\bar{y}$*.*

A cutting plane algorithm constructs a sequence of query points $\{y^k\}$. The answers of the oracle to the queries, together with the cube $0 \le y \le e$, define a polyhedral outer approximation

$$\mathcal{F}_D = \{y \,:\, A^T y \le c\}$$

of C. Since A contains the identity matrix associated with the cube, A has full row rank. Therefore there is a one-to-one correspondence between points $y \in \mathcal{F}_D$ and the slack $s = c - A^T y$, leading to the equivalent definition of $\mathcal{F}_D$

$$\mathcal{F}_D = \{s \ge 0 \,:\, A^T y + s = c\}.$$

The number of columns in A is denoted as m (or m_k) and is equal to $2n$ plus the number of cutting planes generated until the k^{th} iteration; i.e., $m_k = 2n + \sum_{j=0}^{k-1} p_j \le 2n + k * \bar{p}$.

The analytic center cutting plane method chooses as a query point an approximate analytic center of $\mathcal{F}_D$.

2.2. Analytic center. The analytic center of $\mathcal{F}_D$ is the unique point maximizing the dual potential

$$\varphi_D(s) = \sum_{i=1}^{m} \log s_i$$

with $s = c - A^T y > 0$. We formally introduce the optimization problem

$$\max \left\{ \varphi_D(s) \,:\, s = c - A^T y > 0 \right\} \tag{1}$$

and the associated first-order optimality conditions

$$\begin{aligned} xs &= e, \\ A^T y + s &= c, \quad s > 0, \\ Ax &= 0, \quad x > 0, \end{aligned}$$

where x is a vector in R^m. The notation xs indicates the Hadamard or componentwise product of the two vectors x and s.

The analytic center can alternatively be defined as the optimal solution of

$$\max \left\{ \varphi_P(x) \,:\, Ax = 0,\, x > 0 \right\}, \tag{2}$$

where

$$\varphi_P(x) = -c^T x + \sum_{i=1}^{m} \log x_i$$

denotes the primal potential. One easily checks that problem (2) shares with (1) the same first order optimality conditions.

At this stage, it is convenient to introduce the primal-dual potential

$$\varphi_{PD}(x,s) = \varphi_P(x) + \varphi_D(s)$$

and an associated duality relationship.

LEMMA 2.4. *Let $x \in \text{int}\mathcal{F}_P$ and $s \in \text{int}\mathcal{F}_D$. Then $\varphi_{PD}(x,s) \leq -m$, with equality if and only if $xs = e$.*

Proof. Consider the simple inequality

$$\log t \leq t - 1, \quad \forall t > 0, \tag{3}$$

with equality if and only if $t = 1$. Let $x \in \text{int}\mathcal{F}_P$ and $s \in \text{int}\mathcal{F}_D$. Apply (3) with $t = x_i s_i$. By summing the resulting inequalities, one gets

$$\sum_{i=1}^{m} \log x_i + \sum_{i=1}^{m} \log s_i \leq x^T s - m = c^T x - m,$$

with equality if and only if $xs = e$. Therefore,

$$\varphi_P(x) + \varphi_D(s) \leq -m, \tag{4}$$

with equality if and only if $xs = e$. ∎

Finally, we define approximate centers by relaxing the condition $xs = e$ in the first order optimality conditions. Formally, any solution (x,s) of

$$\|e - xs\| \leq \theta < 1, \tag{5}$$

$$A^T y + s = c, \quad s > 0, \tag{6}$$

$$Ax = 0, \quad x > 0, \tag{7}$$

defines a pair of θ-approximate centers, or θ-centers in short.

2.3. Analytic center cutting plane method. ACCPM can be briefly stated as follows.

Initialization. Let $\mathcal{F}_D^0 = \{y \geq 0 : y \leq e\}$ be the unit cube and $y^0 = \frac{1}{2}e$ be its center. The centering parameter is $0 < \theta < 1$.

Basic step. y^k is a θ-center of $\mathcal{F}_D^k$; $m_k = 2n + \sum_{j=0}^{k-1} p_j$ the total number of hyperplanes describing $\mathcal{F}_D^k$.

(1) The oracle returns the cuts a_{m_k+j}, $j = 1, \ldots, p_k$, at y^k.

(2) Update
$\mathcal{F}_D^{k+1} = \mathcal{F}_D^k \cap \{y : a_{m_k+j}^T (y - y^k) \leq 0,\ j = 1, \ldots, p_k\}$.

(3) Compute a θ-center of $\mathcal{F}_D^{k+1}$.

The computation of a new θ-center after adding new cuts will be discussed in a later section.

3. Some useful properties. The literature on interior point methods essentially proposes three approaches for computing analytic centers. All of them are based on Newton's method. The primal (resp., dual) Newton direction is initiated at an interior primal (resp., dual) feasible point; it involves the scaling matrix $D = X$ (resp., $D = S^{-1}$). (We recall the standard notation X which denotes the diagonal matrix $\text{diag}(x)$.) The primal-dual direction is initiated at an interior primal-dual feasible pair, i.e., $(x,s) \in \text{int}\mathcal{F}_P \times \text{int}\mathcal{F}_D$; it involves the scaling matrix $D = (XS^{-1})^{1/2}$.

Let us briefly recall the formulas. The primal direction is given by $\Delta x = xp(x)$ with $p(x) = e - xs(x)$, $s(x) = c - A^T(AD^2A^T)^{-1}AX^2c$, and $D = X$. The dual direction is given by $\Delta s = sq(s)$ with $q(s) = e - sx(s)$, $x(s) = (I - DA^T(AD^2A^T)^{-1}AD)e$, and $D = S^{-1}$. Finally the primal-dual direction is $\Delta s = A^T(AD^2A^T)^{-1}As^{-1}$, $\Delta x = -x + s^{-1} - D^2\Delta s$, and $D = (XS^{-1})^{1/2}$.

3.1. Properties of the Newton step. There are two basic properties, a local one in the vicinity of the analytic center, and a global one. Since the results are well known we state them without proofs. Missing proofs can be found in the books [17] or [20].

Let us start with the local properties. Proximity to analytic center is measured with the quantity $\|e - sx\|$. In this definition, either

(i) $x \in \text{int}\mathcal{F}_P$ and $s = s(x)$ (primal case),
(ii) $s \in \text{int}\mathcal{F}_D$ and $x = x(s)$ (dual case),
(iii) $x \in \text{int}\mathcal{F}_P$ and $s \in \text{int}\mathcal{F}_D$ (primal-dual case).

Note that if $\|e - sx\| \le \theta < 1$, then $s(x) > 0$ and thus $s(x) \in \mathcal{F}_D$ (primal case), and $x(s) > 0$ and thus $x(s) \in \mathcal{F}_P$ (dual case). The Newton step defines a pair (x^+, s^+) as follows:

(i) $x^+ = x + \Delta x$ and $s^+ = s(x^+)$ (primal case),
(ii) $s^+ = s + \Delta s$ and $x^+ = x(s^+)$ (dual case),
(iii) $x^+ = x + \Delta x$ and $s^+ = s + \Delta s$ (primal-dual case).

THEOREM 3.1. *Assume $\|e - sx\| \le \theta < \frac{2}{3}$. Let (x^+, s^+) be the point resulting from a Newton step (primal, dual, or primal-dual). Then, $(s^+, x^+) \in \text{int}\mathcal{F}_D \times \text{int}\mathcal{F}_P$.* In the primal and dual cases, the theorem holds with any $0 < \theta < 1$.

One can derive from the above theorem a useful corollary that yields lower bounds on the potentials near the analytic center. Let (x^c, y^c) be the pair of exact analytic centers. Denote $\varphi_P^c = \varphi_P(x^c)$ and $\varphi_D^c = \varphi_D(s^c)$.

COROLLARY 3.2. *Assume (5)–(7) at (x, s). Then*

(1) $\varphi_P^c \ge \varphi_P(x) \ge \varphi_P^c - \frac{\theta^2}{1-\theta^2}$;
(2) $\varphi_D^c \ge \varphi_D(s) \ge \varphi_D^c - \frac{\theta^2}{1-\theta^2}$;
(3) $-m \ge \varphi_{PD}(x, s) \ge -m - 2\frac{\theta^2}{1-\theta^2}$.

Now let us consider the global properties of a damped Newton step. The properties are consequences of the well-known inequality on the logarithm function [17, p. 439].

LEMMA 3.3. *Let h be any point in R^m such that $\|h\| < 1$. Then,*

$$\sum_{i=1}^{m} \log(1 + h_i) \ge e^T h + \|h\| + \log(1 - \|h\|).$$

The main result bounds the variation of the potentials after a damped Newton step.

THEOREM 3.4. *Assume $\|e - xs\| \ge \theta > 0$. Define $x(\alpha) = x + \alpha\Delta x$ and $s(\alpha) = s + \alpha\Delta s$. ($\Delta x$ and Δs may be the primal, dual, or primal-dual directions.) Then, there exists a step size $\alpha > 0$ and constants σ_P, σ_D, and σ_{PD} such that*

(1) $\varphi_P(x(\alpha)) \ge \varphi_P(x) + \sigma_P$;
(2) $\varphi_D(x(\alpha)) \ge \varphi_D(s) + \sigma_D$;
(3) $\varphi_{PD}(x(\alpha), s(\alpha)) \ge \varphi_{PD}(x, s) + \sigma_{PD}$.

In the primal and dual cases the constants are $\sigma_P = \sigma_D = \theta - \log(1 + \theta)$, while in the primal-dual case $\sigma_{PD} = \frac{\theta}{2(1+\theta)} - \log(1 + \frac{\theta}{2(1+\theta)})$. The above result allows for the design of a potential increase algorithm based on damped Newton steps. The convergence estimate is given by the following theorem.

THEOREM 3.5. *Let $x^0 \in \text{int}\mathcal{F}_P$ and $s^0 \in \text{int}\mathcal{F}_D$. Any potential increase algorithm (primal, dual, primal-dual) produces an interior feasible pair such that $\|e - xs\| \leq \theta < 1$ in a number of iterations not greater than*

$$\left\lceil \frac{\varphi_{PD}(x^0, s^0) + m}{\sigma} \right\rceil,$$

with $\sigma = \sigma_P, \sigma_D,$ or σ_{PD}, depending on which approach (primal, dual, or primal-dual) is taken.

3.2. Dikin's ellipsoids. Let $x \in \text{int}\mathcal{F}_P$. From the observation that $x + \Delta x > 0$, for all Δx such that $\left\|x^{-1}\Delta x\right\| < 1$, we can define an ellipsoidal neighborhood of x that is entirely contained in $\mathcal{F}_P$. Formally,

$$\mathcal{E}_P = \left\{\Delta x \,:\, A\Delta x = 0, \left\|X^{-1}\Delta x\right\| \leq 1\right\}.$$

We shall be particularly concerned with ellipsoids around a θ-center.

We can extend the definition of Dikin ellipsoid to include a different scaling.

LEMMA 3.6. *Let (x, s) be a pair of θ-centers.*

(1) *If $D = S^{-1}$ (dual scaling),*

$$(1-\theta)\mathcal{E}_P \subset \left\{\Delta x \,:\, A\Delta x = 0, \left\|D^{-1}\Delta x\right\| \leq 1\right\} \subset (1+\theta)\mathcal{E}_P.$$

(2) *If $D = X^{1/2}S^{-1/2}$ (primal-dual scaling),*

$$\sqrt{1-\theta}\mathcal{E}_P \subset \left\{\Delta x \,:\, A\Delta x = 0, \left\|D^{-1}\Delta x\right\| \leq 1\right\} \subset \sqrt{1+\theta}\mathcal{E}_P.$$

Proof. For the dual scaling the proof follows from

$$\left\|X^{-1}\Delta x\right\| = \left\|DX^{-1}D^{-1}\Delta x\right\| \leq \left\|(XS)^{-1}\right\|_\infty \left\|D^{-1}\Delta x\right\| \leq \frac{1}{1-\theta}$$

and

$$\left\|D^{-1}\Delta x\right\| = \left\|D^{-1}XX^{-1}\Delta x\right\| \leq \|XS\|_\infty \left\|X^{-1}\Delta x\right\| \leq 1 + \theta.$$

For the primal-dual scaling the proof follows from

$$\left\|X^{-1}\Delta x\right\| = \left\|DX^{-1}D^{-1}\Delta x\right\| \leq \left\|(XS)^{-1/2}\right\|_\infty \left\|D^{-1}\Delta x\right\| \leq \frac{1}{\sqrt{1-\theta}}$$

and

$$\left\|D^{-1}\Delta x\right\| = \left\|D^{-1}XX^{-1}\Delta x\right\| \leq \left\|(XS)^{1/2}\right\|_\infty \left\|X^{-1}\Delta x\right\| \leq \sqrt{1+\theta}. \qquad \square$$

We can similarly define Dikin's ellipsoids in the dual. Let $s \in \text{int}\mathcal{F}_D$. The dual ellipsoid is

$$\mathcal{E}_D = \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \left\|S^{-1}\Delta s\right\| \leq 1\right\}.$$

The extension of Dikin's ellipsoid to a different scaling at a θ-center is given by the following lemma.

LEMMA 3.7.
(1) *If $D = X$ (primal scaling),*

$$(1-\theta)\mathcal{E}_D \subset \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le 1\right\} \subset (1+\theta)\mathcal{E}_D.$$

(2) *If $D = X^{1/2}S^{-1/2}$ (primal-dual scaling),*

$$\sqrt{(1-\theta)}\mathcal{E}_D \subset \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le 1\right\} \subset \sqrt{(1+\theta)}\mathcal{E}_D.$$

The proof is the same as for Lemma 3.6.

It is well known that an homothety of Dikin's ellipsoid contains the feasible set. We shall use this property in the restricted context of the set $\mathcal{F}_D$.

LEMMA 3.8. *Let (x, s) be a θ-centered feasible pair. Then*

$$\mathcal{F}_D \subset \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le \frac{1+\theta}{1-\theta}(m+1)\right\}.$$

Proof. Let (x, s) be a θ-centered feasible pair and $\tilde{s} = c - A^T\tilde{y} > 0$ be any interior point of $\mathcal{F}_D$. Since $x(\tilde{s} - s)$ and e are orthogonal,

$$\begin{aligned}\|x(\tilde{s}-s)\|^2 + m &= \|x\tilde{s} + (e - xs)\|^2, \\ &\le (\|x\tilde{s}\| + \|e - xs\|)^2.\end{aligned}$$

Since $\tilde{s} > 0$, then $\|x\tilde{s}\| \le x^T\tilde{s} = x^Ts$. From $\|e - xs\| \le \theta$, one has $x^Ts \le (1+\theta)m$. We thus obtain the weak bound

$$\|x(\tilde{s}-s)\| \le (1+\theta)m + \theta \le (1+\theta)(m+1).$$

Finally, from $\|D(\tilde{s}-s)\| \le \left\|Dx^{-1}\right\|_\infty \|x(\tilde{s}-s)\|$, one gets

$$\|D(\tilde{s}-s)\| \le \frac{1+\theta}{1-\theta}(m+1).$$

Hence,

$$\mathcal{F}_D \subset \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le \frac{1+\theta}{1-\theta}(m+1)\right\}. \qquad \square$$

4. Multiple central cuts. We assume now that a θ-center $(x; s, y)$ has been computed, i.e.,

$$\|e - xs\| \le \theta < 1, \tag{8}$$

$$A^Ty + s = c, \quad s > 0, \tag{9}$$

$$Ax = 0, \quad x > 0. \tag{10}$$

The cuts are

$$a_{m+j}^T\tilde{y} \le a_{m+j}^Ty,\ j = 1, \ldots, p, \quad \forall y \in C.$$

We define

$$B = (a_{m+1}, a_{m+2}, \ldots, a_{m+p}).$$

The new cuts lead to two new sets:

$$\tilde{\mathcal{F}}_D = \left\{\tilde{y} \,:\, A^T\tilde{y} \le c, B^T\tilde{y} \le B^T y\right\}$$

or

$$\tilde{\mathcal{F}}_D = \left\{\tilde{s} = (\hat{s}, \gamma) \ge 0 \,:\, A^T\tilde{y} + \hat{s} = c, B^T\tilde{y} + \gamma = B^T y\right\},$$

and

$$\tilde{\mathcal{F}}_P = \left\{\tilde{x} = (\hat{x}, \beta) \ge 0 \,:\, A\hat{x} + B\beta = 0\right\}.$$

We shall use the notation

$$\Delta y = (\tilde{y} - y),$$

so

$$\gamma = -B^T \Delta y.$$

After adding the cuts, one has

$$\tilde{s} = \begin{pmatrix} s \\ \gamma = 0 \end{pmatrix} \in \tilde{\mathcal{F}}_D$$

and

$$\tilde{x} = \begin{pmatrix} x \\ \beta = 0 \end{pmatrix} \in \tilde{\mathcal{F}}_P.$$

Let us introduce the notation

$$\tilde{c} = \begin{pmatrix} c \\ B^T y \end{pmatrix}.$$

The primal-dual potentials at the new points $(\hat{x}, \beta)$ and $(\hat{s}, \gamma)$ are

$$\tilde{\varphi}_D(\tilde{s}) = \sum_{i=1}^{m} \log \hat{s}_i + \sum_{i=1}^{p} \log \gamma_i = \varphi_D(\hat{s}) + \sum_{i=1}^{p} \log \gamma_i$$

and

$$\begin{aligned} \tilde{\varphi}_P(\tilde{x}) &= -c^T\hat{x} + \sum_{i=1}^{m} \log \hat{x}_i - y^T B\beta + \sum_{i=1}^{p} \log \beta_i, \\ &= \varphi_P(\hat{x}) - y^T B\beta + \sum_{i=1}^{p} \log \beta_i. \end{aligned}$$

The points $\tilde{x}$ and $\tilde{s}$ (or $\tilde{y}$) lie on the boundary of the new primal and dual sets, respectively. To recover the new analytic center, one has to increase the components β and γ. Since the terms $\sum_{i=1}^{p} \log \beta_i$ and $\sum_{i=1}^{p} \log \gamma_i$ are dominant near $\beta = 0$ and $\gamma = 0$, maximizing those terms while limiting the variation on φ_P and φ_D is likely to produce a good step towards the solution.

This approach requires the knowledge of the level sets of the potential, something that we don't have, but that can be approximated by Dikin's ellipsoids. Therefore, we are interested in solving the following problems:

$$\max \left\{ \sum_{i=1}^{p} \log \beta_i \ : \ \beta \geq 0,\ A\Delta x + B\beta = 0, \left\| D^{-1}\Delta x \right\| \leq 1 \right\} \tag{11}$$

and

$$\max \left\{ \sum_{i=1}^{p} \log \gamma_i \ : \ \gamma \geq 0,\ B^T\Delta y + \gamma = 0, \left\| DA^T\Delta y \right\| \leq 1 \right\}. \tag{12}$$

Here D is one of the scaling matrices X, S^{-1}, or $(XS^{-1})^{\frac{1}{2}}$, depending on whether the computations are done with the primal, the dual, or the primal-dual algorithm.

Let us show here that the above problems are well defined and have a finite optimum.

LEMMA 4.1. *Under Assumptions* 2.1 *and* 2.2, *Problems* (11) *and* (12) *are well defined and have a finite optimum that is uniquely defined by the first-order optimality conditions.*

Proof. Both problems have a strictly concave objective. Their optimum, if it exists, is unique in β (resp., γ).

By Assumptions 2.1 and 2.2, there exists a $\bar{\gamma} > 0$ and a $\bar{\Delta}y$ such that $B^T\bar{\Delta}y + \bar{\gamma} = 0$. Problem (12) is well defined. Since Δy is bounded, γ is bounded and the feasible set is compact. Since the objective tends to $-\infty$ close to the boundary, the problem has a finite solution that is uniquely defined by the set of first order optimality conditions.

To show that Problem (11) is also well defined, we note that the equation $A\Delta x + B\beta = 0$ has a solution for any $\beta > 0$ since A has full row rank. Let us show that the feasible set is bounded. Indeed, let $\beta \geq 0$ and $B\beta = 0$; then

$$0 = \beta^T(B^T\bar{\Delta}y + \bar{\gamma}) = \beta^T\bar{\gamma}.$$

Since $\beta \geq 0$ and $\bar{\gamma} > 0$, then $\beta = 0$. Recalling that A has full row rank, we conclude from $A\Delta x + B\beta = 0$ that $\Delta x \neq 0$ whenever $\beta \geq 0$, $\beta \neq 0$; thus β is bounded, since Δx is bounded by $\left\| D^{-1}\Delta x \right\| \leq 1$. Problem (11) is thus well defined and has a finite optimum. ∎

The solutions of problems (11) and (12) define the primal-dual pair of rays

$$\tilde{x}(\alpha) = \begin{pmatrix} x + \alpha\Delta x \\ \alpha\beta \end{pmatrix}$$

and

$$\tilde{s}(\alpha) = \begin{pmatrix} s + \alpha\Delta s \\ \alpha\gamma \end{pmatrix} = \begin{pmatrix} s - \alpha A^T\Delta y \\ \alpha\gamma \end{pmatrix}$$

for $\alpha > 0$.

If $\|e - xs\| \leq \theta < 1$, then for $\alpha < 1 - \theta$

$$\tilde{x}(\alpha) \in \text{int}\tilde{\mathcal{F}}_P \quad \text{and} \quad \tilde{s}(\alpha) \in \text{int}\tilde{\mathcal{F}}_D.$$

The following positive semidefinite matrix

$$V = B^T(AD^2A^T)^{-1}B$$

plays a fundamental role in the analysis. V can be interpreted as the variance-covariance matrix between the vectors (a_{m+j}), $j = 1, \ldots, p$, in the metric induced by the matrix $(AD^2A^T)^{-1}$, i.e., Dikin's metric.

THEOREM 4.2. *The solution of problems* (11) *and* (12) *is given by*

$$\Delta x = -D^2A^T(AD^2A^T)^{-1}B\beta$$

and

$$\Delta y = -(AD^2A^T)^{-1}B\beta,$$

with β *defined as the unique solution of*

$$\max\left\{-\frac{p}{2}\beta^T V\beta + \sum_{i=1}^{p}\log\beta_i\right\}, \tag{13}$$

and

$$\gamma = V\beta.$$

Proof. Let $\lambda \in R^n$ and σ^2 be the multipliers associated with the constraints of Problem (11). The optimality conditions are

$$\begin{aligned}\beta^{-1} + B^T\lambda &= 0,\\ A^T\lambda - \sigma^2 D^{-2}\Delta x &= 0,\\ A\Delta x + B\beta &= 0,\\ \sigma^2(1 - \left\|D^{-1}\Delta x\right\|) &= 0.\end{aligned}$$

From the definition of Δx, one immediately sees that $A\Delta x + B\beta = 0$. Letting $\lambda = -p(AD^2A^T)^{-1}B\beta$ and $\sigma^2 = p$, we have

$$A^T\lambda = -pA^T(AD^2A^T)^{-1}B\beta = \sigma^2 D^{-2}\Delta x.$$

This proves the second relation. To prove the first relation, we shall use the optimality condition for Problem (13). However, we must check first that (13) has a bounded optimum. In Lemma 4.1 we proved that $B\beta = 0$ has no nonzero nonnegative solution. Thus, for all $\beta \geq 0$, $\beta \neq 0$, one has

$$\beta^T V\beta = \beta^T B^T(AD^2A^T)^{-1}B\beta > 0.$$

This proves that the objective $-\frac{p}{2}\beta^T V\beta + \sum_{i=1}^{p}\log\beta_i$ is bounded above and Problem (13) has a unique optimum.

The optimality condition for Problem (13) is

$$-pV\beta + \beta^{-1} = 0.$$

Replacing β^{-1} by $pV\beta$ we get the identity

$$pV\beta + B^T\lambda = pB^T(AD^2A^T)^{-1}B\beta - pB^T(AD^2A^T)^{-1}B\beta \equiv 0.$$

It remains to check that $\left\|D^{-1}\Delta x\right\| = 1$. Indeed,

$$\left\|D^{-1}\Delta x\right\|^2 = \beta^T B^T(AD^2A^T)^{-1}B\beta = \beta^T V\beta$$

and

$$\beta^T V \beta = \frac{1}{p}\beta^T \beta^{-1} = 1.$$

Now let us consider Problem (12). The optimality conditions are

$$\begin{aligned} \gamma^{-1} - \mu &= 0, \\ \gamma + B^T \Delta y &= 0, \\ B\mu + \sigma^2 (AD^2A^T)\Delta y &= 0, \\ \sigma^2 (1 - \left\| DA^T \Delta y \right\|) &= 0, \end{aligned}$$

where $\mu \in R^p$ and $\sigma^2 \in R_+$ are the multipliers associated with the two constraints.

We want to show that $\mu = p\beta$ and $\sigma^2 = p$ are the optimal multipliers, where β is the optimal solution of Problem (13). Solving for Δy, one gets

$$\Delta y = -\frac{1}{\sigma^2}(AD^2A^T)^{-1}B\mu = -(AD^2A^T)^{-1}B\beta.$$

Now

$$0 = \gamma + B^T \Delta y = \gamma - B^T (AD^2A^T)^{-1} B\beta = \gamma - V\beta.$$

Remembering the optimality condition for β, one may replace $\gamma = V\beta$ by $(p\beta)^{-1}$, and thus check that the first optimality condition $\gamma^{-1} = p\beta = \mu$ holds.

Finally,

$$1 = \Delta y^T (AD^2A^T)\Delta y = \beta^T V \beta = \left\| DA^T \Delta y \right\|$$

proves that with our choice of multipliers, the last optimality condition also holds. □

Remark 4.1. If V is nonsingular, γ is also the unique solution of

$$\max \left\{ -\frac{1}{2} p \gamma^T V^{-1} \gamma + \sum_{i=1}^{p} \log \gamma_i \right\}. \tag{14}$$

We can now give an explicit formula for the restoration direction. Noting that

$$\Delta s = -\frac{1}{p} A^T \Delta y = A^T (AD^2A^T)^{-1} B\beta,$$

we have the new primal-dual pair

$$\tilde{x}(\alpha) = \begin{pmatrix} x + \alpha \Delta x \\ \alpha\beta \end{pmatrix} = \begin{pmatrix} x - \alpha D^2 A^T (AD^2A^T)^{-1} B\beta \\ \alpha\beta \end{pmatrix}, \tag{15}$$

$$\tilde{s}(\alpha) = \begin{pmatrix} s + \alpha \Delta s \\ \alpha\gamma \end{pmatrix} = \begin{pmatrix} s + \alpha A^T (AD^2A^T)^{-1} B\beta \\ \alpha V\beta \end{pmatrix}, \tag{16}$$

and

$$\tilde{y}(\alpha) = \begin{pmatrix} y + \alpha \Delta y \end{pmatrix} = \begin{pmatrix} y - \alpha (AD^2A^T)^{-1} B\beta \end{pmatrix}. \tag{17}$$

Remark 4.2. We note a significant dissymmetry between the primal and dual directions:

(1) any positive value of β, say $\beta = e$, gives a primal feasible direction, but
(2) $\beta > 0$ does not guarantee $\gamma = V\beta > 0$; however, if V is nonsingular, then taking $\beta = pV^{-1}\widehat{\gamma}$, with $\widehat{\gamma} > 0$, gives a feasible dual direction.

Different stepsizes (α_P, α_D) could be used in the primal-dual space.

Note that, by construction, $\left\|D^{-1}\Delta x\right\| = 1$ and $\|D\Delta s\| = 1$, and that if $\lambda = p\beta$, then $D^{-1}\Delta x = D\Delta s$. At the optimum direction, one has $p\gamma\beta = e$.

The computation of β requires solving the nonlinear optimization problem (13). Since the function $F(\beta) = -\sum_{i=1}^{p} \log \beta_i + \frac{p}{2}\beta^T V\beta$ is self-concordant, it can easily be minimized by classical Newton schemes. We postpone to a later section the discussion on the complexity estimate for getting approximate solutions.

For the sake of a simpler presentation we shall assume in our analysis of ACCPM that the minimizers are exact. However, this is not the case in practice and we must be concerned with the impact of errors on β and γ on the performance of ACCPM. This discussion is also postponed to a later section. Below, we sketch the result that enables an easy extension of our analysis of ACCPM with multiple cuts in the case of inexact computations of β and γ.

The convergence analysis of section 5 relies on the following properties:

(i) $\left\|D^{-1}\Delta x\right\| = \beta^T V\beta = 1$,
(ii) $\|D\Delta s\| = \frac{1}{p^2}\gamma^{-T} V\gamma^{-1} = 1$,
(iii) $p\beta\gamma = e$.

If we can guarantee that the solutions satisfy $p\beta\gamma \approx e$ and $\frac{1}{p^2}\gamma^{-T} V\gamma^{-1} \approx 1 \approx \beta^T V\beta$, then the convergence result on ACCPM is essentially unaffected, while the proofs need only minor adjustments.

We give here a theorem that stipulates the condition that must be met by β and γ to carry the analysis with inexact minimizers. In a later section we shall show that classical interior point schemes make it possible to meet the condition.

LEMMA 4.3. *Assume $\beta > 0$ and $\|p\beta(V\beta) - e\| \le \eta$. Let $\gamma = V\beta$. Then*

$$(1-\eta)e \le p\beta\gamma \le (1+\eta)e$$

and

$$1 - \eta \le \beta^T V\beta = \beta^T\gamma \le 1 + \eta.$$

In particular, $\gamma = V\beta > 0$ if $\eta < 1$.

Proof. The first set of inequalities follows directly from the assumption and the definition of γ. These inequalities also imply that $\gamma = V\beta > 0$ if $\eta < 1$.

Multiplying these inequalities by e^T one gets

$$p(1-\eta) \le p\beta^T V\beta \le p(1+\eta). \qquad \square$$

5. Convergence analysis. We now assume that (x, s) is a pair of θ-centers and that Δx and Δs are computed as in section 4 with β and γ being the exact minimizers of problems (13) and (14). We assume that the computations are done with either the primal, the dual, or the primal-dual scaling.

LEMMA 5.1. *Independently of the specific scaling matrix D (primal, dual, or primal-dual), one has, for any $\alpha < 1 - \theta$, $\left\|\alpha X^{-1}\Delta x\right\| < 1$ and $\left\|\alpha S^{-1}\Delta s\right\| < 1$.*

Proof. By construction $\left\|D^{-1}\Delta x\right\| = \|D\Delta s\| = 1$. From Lemma 3.6, for any primal, dual, or primal-dual scaling D, we have $\left\|X^{-1}\Delta x\right\| \le \frac{1}{1-\theta}\left\|D^{-1}\Delta x\right\| = \frac{\alpha}{1-\theta} < 1$. The proof is the same in the dual case. $\square$

Remark 5.1. The above result can be sharpened by considering separately the three different scaling matrices D. However, we prefer the weaker result since it allows a single formulation for the three cases.

LEMMA 5.2. *The following inequalities hold:*

$$|c^T\Delta x + y^T B\beta - e^T X^{-1}\Delta x| \le \frac{\theta}{1-\theta}$$

and

$$|e^T S^{-1}\Delta s| \le \frac{\theta}{1-\theta}.$$

Proof. From $B\beta = -A\Delta x$, one has

$$c^T\Delta x + y^T B\beta = c^T\Delta x - y^T A\Delta x = e^T(S\Delta x).$$

Thus,

$$\begin{aligned} |c^T\Delta x + y^T B\beta - e^T X^{-1}\Delta x| &= |e^T(S - X^{-1})\Delta x| \\ &= |(sx - e)^T X^{-1}\Delta x| \\ &\le \|e - sx\| \left\|X^{-1}\Delta x\right\| \\ &\le \frac{\theta}{1-\theta}. \end{aligned}$$

To prove the second statement, we note that $x^T\Delta s = 0$ since $Ax = 0$. Thus

$$\begin{aligned} |e^T S^{-1}\Delta s| &= |e^T(S^{-1} - X)\Delta s| \\ &= |(sx - e)^T S^{-1}\Delta s| \\ &\le \|e - sx\| \left\|S^{-1}\Delta s\right\| \\ &\le \frac{\theta}{1-\theta}. \qquad \square \end{aligned}$$

In view of the above lemmas, we can bound the potentials $\tilde\varphi_P$ and $\tilde\varphi_D$ at the new pair of points $(\tilde x(\alpha), \tilde s(\alpha))$.

LEMMA 5.3. *For any $0 < \alpha < 1 - \theta$, the new potentials satisfy*

$$\tilde\varphi_P(\tilde x(\alpha)) \ge \varphi_P(x) + p\log\alpha + \alpha + \log\left(1 - \frac{\alpha}{1-\theta}\right) + \sum_{i=1}^{p}\log\beta_i, \tag{18}$$

$$\tilde\varphi_D(\tilde s(\alpha)) \ge \varphi_D(s) + p\log\alpha + \alpha + \log\left(1 - \frac{\alpha}{1-\theta}\right) + \sum_{i=1}^{p}\log\gamma_i, \tag{19}$$

and

$$\tilde\varphi_{PD}(\tilde x(\alpha), \tilde s(\alpha)) \ge \varphi_{PD}(x,s) + 2p\log\alpha + 2\alpha + 2\log\left(1 - \frac{\alpha}{1-\theta}\right) - p\log p. \tag{20}$$

Proof. Let us prove first the inequality on the primal potential. At the updated point $\tilde{x}(\alpha)$ the potential is

$$\begin{aligned}\tilde{\varphi}_P(\tilde{x}(\alpha)) &= -\tilde{c}^T\tilde{x}(\alpha) + \sum_{i=1}^{m}\log x_i(\alpha) + \sum_{i=1}^{p}\log\alpha\beta_i \\ &= -c^Tx - \alpha c^T\Delta x - \alpha y^TB\beta + \sum_{i=1}^{m}\log x_i(\alpha) + \sum_{i=1}^{p}\log\alpha\beta_i \\ &= \varphi_P(x) - \alpha c^T\Delta x + \sum_{i=1}^{m}\log(1+\alpha x_i^{-1}(\Delta x)_i) - \alpha y^TB\beta + \sum_{i=1}^{p}\log\alpha\beta_i.\end{aligned}$$

Let $h_P = \alpha x^{-1}\Delta x$. By Lemma 5.1 $\|h_P\| < 1$. We can apply Lemma 3.3 to get

$$\sum_{i=1}^{m}\log(1+\alpha x_i^{-1}(\Delta x)_i) \geq \alpha e^Tx^{-1}\Delta x + \|h_P\| + \log(1-\|h_P\|).$$

Then, by Lemma 5.2,

$$\alpha e^Tx^{-1}\Delta x - \alpha c^T\Delta x - \alpha y^TB\beta \geq -\frac{\alpha\theta}{1-\theta}.$$

Since $t+\log(1-t)$ is decreasing, we can bound $\|h_P\|+\log(1-\|h_P\|)$ by $\frac{\alpha}{1-\theta}+\log(1-\frac{\alpha}{1-\theta})$ and get

$$\tilde{\varphi}_P(\tilde{x}(\alpha)) \geq \varphi_P(x) + \alpha + \log\left(1-\frac{\alpha}{1-\theta}\right) + \sum_{i=1}^{p}\log\alpha\beta_i.$$

Now let us prove the dual case. We have

$$\begin{aligned}\tilde{\varphi}_D(\tilde{s}(\alpha)) &= \sum_{i=1}^{m}\log s_i(\alpha) + \sum_{i=1}^{p}\log\alpha\gamma_i \\ &= \varphi_D(s) + \sum_{i=1}^{m}\log(1+\alpha s_i^{-1}(\Delta s)_i) + \sum_{i=1}^{p}\log\alpha\gamma_i.\end{aligned}$$

Let $h_D = \alpha S^{-1}\Delta s$. By Lemma 5.1 $\|h_D\| < 1$. We can apply Lemma 3.3 to get

$$\sum_{i=1}^{m}\log(1+\alpha s_i^{-1}(\Delta s)_i) \geq \alpha e^Ts^{-1}\Delta s + \|h_D\| + \log(1-\|h_D\|).$$

Since by Lemma 5.2

$$\alpha e^Ts^{-1}\Delta s \geq -\frac{\alpha\theta}{1-\theta}$$

we obtain, by putting the inequalities together, the same result as in the primal case

$$\tilde{\varphi}_D(\tilde{s}(\alpha)) \geq \varphi_D(s) + \alpha + \log\left(1-\frac{\alpha}{1-\theta}\right) + \sum_{i=1}^{p}\log\alpha\gamma_i.$$

To conclude the proof of the theorem, we just sum the inequalities on $\tilde{\varphi}_P$ and $\tilde{\varphi}_D$ and use $\beta\gamma = \frac{1}{p}e$ to get

$$\tilde{\varphi}_{PD}(\tilde{x}(\alpha),\tilde{s}(\alpha)) \geq \varphi_{PD}(x,s) + 2p\log\alpha + 2\alpha + 2\log\left(1-\frac{\alpha}{1-\theta}\right) - p\log p. \qquad \square$$

5.1. Recovering the new analytic center. The complexity of the restoration procedure is given by the next theorem.

THEOREM 5.4. *The number of Newton steps to compute the updated θ-analytic center is bounded by*

$$\nu = \frac{-p-\rho}{\sigma} = O(p\log(p+1)),$$

where

$$\rho = \frac{2\theta^2}{1-\theta^2} + 2\alpha + 2p\log\alpha + 2\log\left(1-\frac{\alpha}{1-\theta}\right) - p\log p,$$

and, depending on the Newton scheme,

$$\sigma = \sigma_P,\ \sigma_D,\ \text{or } \sigma_{PD}.$$

Proof. To bound the number of Newton steps, we compute the optimality gap

$$\Delta\tilde{\varphi}_{PD} = (\tilde{\varphi}_P^c + \tilde{\varphi}_D^c) - \tilde{\varphi}_{PD}(\tilde{x}(\alpha), \tilde{s}(\alpha))$$

for the sum of the primal-dual potentials. On the one hand,

$$\tilde{\varphi}_P^c + \tilde{\varphi}_D^c = -(m+p).$$

On the other hand, we can write

$$\tilde{\varphi}_{PD}(\tilde{x}(\alpha), \tilde{s}(\alpha)) \geq \varphi_{PD}(x,s) + 2\alpha + 2p\log\alpha + 2\log\left(1-\frac{\alpha}{1-\theta}\right) - p\log p.$$

Finally,

$$\varphi_{PD}(x,s) \geq \varphi_P(x^c) + \varphi_D(s^c) - \frac{2\theta^2}{1-\theta^2} = -m - \frac{2\theta^2}{1-\theta^2}.$$

Hence,

$$\begin{aligned}&\tilde{\varphi}_{PD}(\tilde{x}(\alpha), \tilde{s}(\alpha))\\ &\geq -m + \frac{2\theta^2}{1-\theta^2} + 2\alpha + 2p\log\alpha + 2\log\left(1-\frac{\alpha}{1-\theta}\right) - p\log p.\end{aligned}$$

Thus

$$\Delta\tilde{\varphi}_{PD} \leq -p-\rho.$$

Using Theorem 3.4 and the above bound on the potential variation we conclude the proof of the theorem. ∎

5.2. Convergence of ACCPM with multiple cuts. The next lemma is a first step on bounding the number of calls to the oracle.

THEOREM 5.5. *For all $0 < \alpha < 1-\theta$*

$$\tilde{\varphi}_D^c \leq \varphi_D^c + \sum_{i=1}^{p} \log\tau_i + \kappa(\theta,\alpha,p),$$

with

$$\kappa(\alpha,\theta,p)=\frac{\theta^2}{1-\theta^2}-\alpha-\log\left(1-\frac{\alpha}{1-\theta}\right)-p\log\alpha-p+p\log p,$$

and τ is the vector whose components are the square roots of the diagonal elements of V.

Proof. The first inequality uses $\tilde{\varphi}_P^c \geq \tilde{\varphi}_P(\tilde{x}(\alpha))$, the duality on potential, and Lemma 2.4 to yield

$$\begin{aligned}-\tilde{\varphi}_D^c &= (m+p)+\tilde{\varphi}_P^c\\ &\geq (m+p)+\tilde{\varphi}_P(\tilde{x}(\alpha))\\ &\geq m+p+\varphi_P(x)+p\log\alpha+\alpha+\log\left(1-\frac{\alpha}{1-\theta}\right)+\sum_{i=1}^{p}\log\beta_i. \qquad (21)\end{aligned}$$

We now need to deal with the contribution of the new variables

$$\sum_{i=1}^{p}\log\beta_i.$$

Since β solves (13), we have $\beta^T V\beta=1$ and

$$\begin{aligned}\sum_{i=1}^{p}\log\beta_i-\frac{p}{2}&=\max_{\beta'}\left\{\sum_{i=1}^{p}\log\beta'_i-\frac{p}{2}{\beta'}^T V\beta'\right\},\\ &\geq\sum_{i=1}^{p}\log\beta'_i-\frac{p}{2}{\beta'}^T V\beta',\end{aligned}$$

for any arbitrary β'.

Let us define the vector τ by

$$\tau_i=\sqrt{a_{m+i}^T(AX^2A^T)^{-1}a_{m+i}}.$$

Note that $\tau^2=\mathrm{diag}V$ while the off-diagonal terms of V are

$$\tau_{ij}=a_{m+i}^T(AX^2A^T)^{-1}a_{m+j}.$$

The off-diagonal elements satisfy

$$|\tau_{ij}|\leq\tau_i\tau_j.$$

Those properties are typical of a variance-covariance matrix. Let us choose

$$\beta'=\frac{\tau^{-1}}{\sqrt{\tau^{-T}V\tau^{-1}}}.$$

Then

$${\beta'}^T V\beta'=1.$$

The matrix $R=\mathrm{diag}(\tau^{-1})V\mathrm{diag}(\tau^{-1})$ is a correlation matrix: its coefficients are bounded in absolute value by 1, and

$$\tau^{-T}V\tau^{-1}=e^TRe\leq p^2.$$

Thus

$$\begin{aligned}\sum_{i=1}^{p} \log \beta_i &\geq \sum_{i=1}^{p} \log \beta_i' \\ &= -\sum_{i=1}^{p} \log \tau_i - p \log \sqrt{\tau^{-T} V \tau^{-1}} \\ &\geq -\sum_{i=1}^{p} \log \tau_i - p \log p. \qquad (22)\end{aligned}$$

Using Corollary 3.2 we have

$$\varphi_P(x) \geq \varphi_P^c - \frac{\theta^2}{1-\theta^2} = -\varphi_D^c - m - \frac{\theta^2}{1-\theta^2}. \qquad (23)$$

Putting together (21), (22), and (23) yields

$$\tilde{\varphi}_D^c \leq \varphi_D^c + \frac{\theta^2}{1-\theta^2} - p - \alpha - \log\left(1 - \frac{\alpha}{1-\theta}\right) + p \log \frac{p}{\alpha} + \sum_{i=1}^{p} \log \tau_i. \qquad \square$$

The bound

$$\kappa(\alpha, \theta, p) = \frac{\theta^2}{1-\theta^2} - p - \alpha - \log\left(1 - \frac{\alpha}{1-\theta}\right) + p \log \frac{p}{\alpha}$$

can be analyzed by selecting, somewhat arbitrarily, $\alpha = 1/\sqrt{2}$ and $\theta = .25$, guaranteeing $\alpha \leq 1 - \theta$ but also

$$\begin{aligned}\kappa(\alpha, \theta, p) &= \frac{\theta^2}{1-\theta^2} - p - \alpha - \log\left(1 - \frac{\alpha}{1-\theta}\right) + p \log \frac{p}{\alpha} \\ &\leq p \log(p+1); \qquad (24)\end{aligned}$$

this is exactly the same result as in [19], but with a rather different derivation, as we show that this inequality is actually achieved at the iterate obtained by the restoration step.

Remark 5.2. If the p cuts generated are identical, then the correlation matrix R is the rank-one matrix ee^T. Otherwise for the optimal β^*

$$\sum_{i=1}^{p} \log \beta_i^* + \sum_{i=1}^{p} \log \tau_i + p \log p \qquad (25)$$

may be significantly greater than 0 and speed the convergence in practice, even though this does not appear to affect the worst case complexity bound.

5.3. Convergence of ACCPM. The convergence analysis uses the proof given in [19], for the case of multiple cuts.

Denote

$$P = \varphi_D(s^c) = \max\{\varphi_D(s) \,:\, s \in \mathcal{F}_D\}$$

and let P^k be the same value after k calls to the oracle, that is, after adding $m_k - 2n = \sum_{j=0}^{k-1} p_j$ cuts, where p_j denotes the number of cuts added at iteration j. By Theorem 5.5 and the observation (24) the following inequality holds:

$$P^{k+1} \leq P^0 + \sum_{j=0}^{k} \sum_{i=1}^{p_j} \log \tau_i^j + \sum_{j=0}^{k} p_j \log(p_j + 1).$$

Theorem 10 of [19] can be used here, where $\bar{p} \leq n$ denotes the maximum number of cuts generated by any call to the oracle.

THEOREM 5.6. *The algorithm stops with a solution as soon as k satisfies*

$$\frac{\varepsilon^2}{(\bar{p}+1)^2} \geq \frac{\frac{n}{2} + \frac{18n^2}{15} \log(1 + \frac{m_{k+1}}{8n^2})}{m_{k+1}}.$$

Furthermore the number of damped Newton steps per call to the oracle is $O(\bar{p} \log(\bar{p} + 1))$. The number of cutting planes generated is at most $O^*(\frac{\bar{p}^2 n^2}{\varepsilon^2})$.[1]

The assumption that $\bar{p} \leq n$ is not required in the proof of [19], and in fact $\bar{p} = O(n)$ would still lead to $O^*(\frac{\bar{p}^2 n^2}{\varepsilon^2})$ cutting planes. (This would only impact the constant.)

6. Computing the optimal direction of restoration. The restoration direction requires the solution of the concave problem

$$\max \left\{ F(\beta) = -\frac{1}{2} p \beta^T V \beta + \sum_{i=1}^{p} \log \beta_i \right\}.$$

We note that in the computation of the restoration direction a significant absence of symmetry occurs: it is easy to give a feasible value for β, say $\beta = \frac{e}{\sqrt{e^T V e}}$ or $\beta = \frac{\tau^{-1}}{\sqrt{\tau^{-T} V \tau^{-1}}}$, that gives a feasible solution to the problem of finding a feasible direction, but, in general, this is not the case for the dual side. τ is the vector whose components are the square roots of the diagonal elements of V.

If V is invertible, then the dual direction could also be computed by maximizing

$$G(\gamma) = -\frac{p}{2} \gamma^T V^{-1} \gamma + \sum_{i=1}^{p} \log \gamma_i.$$

A good starting value for γ could also be given, say $\gamma = \frac{e}{\sqrt{e^T V^{-1} e}}$ or $\gamma = \frac{\tau_D^{-1}}{\sqrt{\tau_D^{-T} V^{-1} \tau_D^{-1}}}$, where τ_D is the vector whose components are the square roots of the diagonal elements of V^{-1}.

The following bounds on $F(\beta)$ will be useful in the computation of complexity estimate of a Newton method to solve (13).

THEOREM 6.1. *For*

$$\beta^0 = \frac{\tau^{-1}}{\sqrt{\tau^{-T} V \tau^{-1}}},$$

$$F(\beta^0) \geq -\sum_{i=1}^{p} \log \tau_i - p \log p - \frac{p}{2},$$

[1] The notation O^* indicates that lower-order terms are ignored.

and

$$\max_{\beta>0} F(\beta) - p\log p - p/2 - p\log\left(\frac{\varepsilon(1-\theta)}{(m+1)(1+\theta)}\right).$$

Proof. The inequality on $F(\beta^0)$ was derived in the proof of Theorem 5.5. See (22).

Let us construct an upper bound on $F(\beta^*)$, where β^* denotes the optimal solution of problem (13), and $\gamma^* = V\beta^*$. From the optimality condition

$$F(\beta^*) + \sum_{i=1}^{p} \log \gamma_i^* = \sum_{i=1}^{p} \log(\beta_i^*\gamma_i^*) - (p/2)\beta^{*T} V \beta^* = -p\log p - p/2.$$

Hence,

$$F(\beta^*) = -p\log p - p/2 - \max\left\{\sum_{i=1}^{p} \log\gamma_i : B^T\Delta y + \gamma = 0, \left\|DA^T\Delta y \le 1\right\|\right\}.$$

By Lemma 3.8, an homothety of Dikin's ellipsoid contains the current set of localization, i.e.,

$$\mathcal{F}_D \subset \left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le \frac{1+\theta}{1-\theta}(m+1)\right\}.$$

By assumption (2.1) and the fact that the algorithm has not terminated, a sphere of radius ε is contained in $\mathcal{F}_D$. Then perfroming an homothety in the y-space, with as a center the current approximate analytic center, and with ratio $\frac{1-\theta}{(m+1)(1+\theta)}$, we conclude that

$$\left\{\Delta s \,:\, \Delta s = -A^T\Delta y, \|D\Delta s\| \le 1\right\} \cap \mathcal{F}_D$$

contains a sphere of radius $\frac{\varepsilon(1-\theta)}{(m+1)(1+\theta)}$. Denoting by y_c the center of this sphere, and selecting $\gamma = -B^T(y_c - y)$ one has

$$-\sum_{i=1}^{p} \log\gamma_i \le -p\log\left(\frac{\varepsilon(1-\theta)}{(m+1)(1+\theta)}\right)$$

with $\left\|DA^T(y_c - y)\right\| \le 1$.

And thus

$$F(\beta^*) \le -p\log p - p/2 - p\log\left(\frac{\varepsilon(1-\theta)}{(m+1)(1+\theta)}\right). \qquad \square$$

If V is invertible, one can derive alternative upper bounds on $F(\beta^*)$ as follows: Using

$$F(\beta) + G(\gamma) \le F(\beta^*) + G(\gamma^*) = -p\log p - p,$$

we have

$$\begin{aligned} F(\beta^*) &\le -p\log p - p - G(\gamma), \quad (\forall \gamma > 0) \\ &= -p\log p - p - \sum_{i=1}^{p} \log\gamma_i + (p/2)\gamma^T V^{-1}\gamma \\ &\le -p\log p - p/2 + (p/2)\log(e^T V^{-1} e) \quad (\text{setting } \gamma = e/\sqrt{e^T V^{-1} e}). \end{aligned}$$

If instead of $\gamma = e$ we set $\gamma = \frac{\tau_D^{-1}}{\sqrt{\tau_D^{-T} V^{-1} \tau_D^{-1}}}$, then

$$F(\beta^*) \leq p \log p - p/2 + \sum_{j=1}^{p} \log(\tau_D)_j + (p/2) \log(\tau_D^{-T} V^{-1} \tau_D^{-1}).$$

The bounds on $F(\beta^0)$ and $F(\beta^*)$ are used to derive a complexity estimate for the computation of an approximate optimal solution. Using the fact that the function F is self-concordant [15], we can resort to a potential increase scheme. The scheme uses the Newton direction

$$-[F''(\beta)]^{-1} F'(\beta).$$

Let us denote $\|u\|_H = \sqrt{-u^T H u}$ the norm of an arbitrary vector u in the metric induced by the negative definite matrix H. The norm $\|F'(\beta)\|_{[F''(\beta)]^{-1}}$ plays a critical role in the analysis. The potential increase scheme is based on an extension of Lemma 3.3. The proof can be found in the unpublished lecture notes [14]. (The proof is also made available in [16].)

LEMMA 6.2. *Let $\Delta\beta$ be such that $\|\Delta\beta\|_{[F''(\beta)]^{-1}} < 1$. Then,*

$$F(\beta + \Delta\beta) \leq F(\beta) + \Delta\beta^T F'(\beta) - t - \log(1 - t)$$

with $t = \|\Delta\beta\|_{[F''(\beta)]^{-1}}$.

Assume now $\|F'(\beta)\|_{[F''(\beta)]^{-1}} \geq \eta$, for a fixed $0 < \eta < 1$. Let

$$\Delta\beta = -[F''(\beta)]^{-1} F'(\beta)$$

and $\alpha = (1 + \|F'(\beta)\|_{[F''(\beta)]^{-1}})^{-1}$. Then $\alpha\Delta\beta$ satisfies the condition of the above lemma. Thus,

$$F(\beta + \alpha\Delta\beta) \leq F(\beta) + \alpha\Delta\beta^T F'(\beta) - \alpha \|F'(\beta)\|_{[F''(\beta)]^{-1}} + \log\left(1 + \|F'(\beta)\|_{[F''(\beta)]^{-1}}\right).$$

Since $\Delta\beta^T F'(\beta) = -\alpha \|F'(\beta)\|^2_{[F''(\beta)]^{-1}}$, we have

$$F(\beta + \alpha\Delta\beta) \leq F(\beta) - \sigma,$$

where $\sigma = \|F'(\beta)\|_{[F''(\beta)]^{-1}} - \log(1 + \|F'(\beta)\|_{[F''(\beta)]^{-1}})$. One easily shows that $\sigma \geq \eta - \log(1 + \eta)$ is bounded from below by an absolute constant.

The complexity estimate for the potential increase scheme follows directly from the above analysis and a bound on the achievable potential increase $(F(\beta^*) - F(\beta^0))$.

THEOREM 6.3. *Let $\beta^0 = \frac{\tau^{-1}}{\sqrt{\tau^{-T} V \tau^{-1}}}$. The potential increase algorithm applied to the maximization of F produces a point β such that $\|F'(\beta)\|_{[F''(\beta)]^{-1}} \leq \eta < 1$ in a number of iterations not greater than*

$$\left\lceil \frac{\sum_{i=1}^{p} \log \tau_i - p \log\left(\frac{\varepsilon(1-\theta)}{(m+1)(1+\theta)}\right) - p + p \log p}{\eta - \log(1 + \eta)} \right\rceil.$$

Formula (26) involves the unknown quantity τ_j and does not provide a workable bound on the number of iterations needed to compute the optimal restoration direction

after each call to the oracle. However using the long step argument of [19], we can bound the cumulative number of such steps by $m_{k^*}\log(1/\varepsilon)$, where k^* is the number of calls to the oracle at termination, and $m_{k^*} = O^*(\frac{p^2n^2}{\varepsilon^2})$.

Remark 6.1. Looking at every iteration individually, and using the fact that $A^Ty \le c$ contains the cutting planes $0 \le y \le e$, we can assert that

$$(AS^{-2}A^T)^{-1} \prec (Y^{-2} + (I-Y)^{-2})^{-1} \prec (4I+4I)^{-1} \prec \frac{1}{8}I,$$

and hence

$$\tau_j^2 = a_j^T(AS^{-2}A^T)^{-1}a_j \le \frac{1}{8}\|a\|^2 = \frac{1}{8}.$$

This indicates that, in practice, the number of iterations needed at each iteration to compute the optimal β should not increase with the number of cutting planes.

Remark 6.2. The number of iterations needed to compute this approximate optimal direction is polynomial in the data, as $\log m$ is polynomial in the data.

It remains to prove that the potential increase scheme yields a solution β that meets the proximity condition $\|p\beta(V\beta) - e\| \le \theta$ used in Theorem 4.3. In other words, we must show that for η small enough the condition $\|F'(\beta)\|_{[F''(\beta)]^{-1}} \le \eta$ implies $\|p\beta(V\beta) - e\| \le \theta$. To this end, we adapt some results and proofs of [3] developed for quadratic programming.

We then relate a few critical norms.

LEMMA 6.4. *Let* $\Delta\beta = -[F''(\beta)]^{-1}F'(\beta) = (\mathrm{diag}(\beta^{-2}) + pV)^{-1}(\beta^{-1} - pV\beta)$. *The following inequality holds:*

$$\left\|\beta^{-1}\Delta\beta\right\| \le \|F'(\beta)\|_{[F''(\beta)]^{-1}} \le \|p\beta(V\beta) - e\|.$$

Proof. Since

$$\begin{aligned}\|F'(\beta)\|^2_{[F''(\beta)]^{-1}} &= (pV\beta - \beta^{-1})^T(\mathrm{diag}(\beta^{-2}) + V)^{-1}(pV\beta - \beta^{-1}) \\ &= \Delta\beta^T\left(\mathrm{diag}(\beta^{-2}) + pV\right)\Delta\beta \\ &\ge \Delta\beta^T\mathrm{diag}(\beta^{-2})\Delta\beta = \left\|\beta^{-1}\Delta\beta\right\|^2.\end{aligned}$$

This proves the left-hand side inequality.

As V is positive semidefinite, one has

$$(\mathrm{diag}(\beta^{-2}) + pV)^{-1} \preceq (\mathrm{diag}(\beta^{-2}))^{-1} = \mathrm{diag}(\beta^2).$$

Therefore,

$$\begin{aligned}\|F'(\beta)\|^2_{[F''(\beta)]^{-1}} &\le (pV\beta - \beta^{-1})^T\mathrm{diag}(\beta^2)(pV\beta - \beta^{-1}) \\ &= (p\beta(V\beta) - e)^T(p\beta(V\beta) - e) = \|(p\beta(V\beta) - e)\|^2. \qquad \square\end{aligned}$$

We can now prove the main result of the section.

LEMMA 6.5. *Assume* $\|F'(\beta)\|_{[F''(\beta)]^{-1}} \le \eta < 1$ *and let*

$$\Delta\beta = -F''(\beta)]^{-1}F'(\beta) = (\mathrm{diag}(\beta)^2 + pV)^{-1}(\beta^{-1} - pV\beta);$$

Then $\beta^+ = \beta + \Delta\beta > 0$. *Besides,*

$$\left\|F'(\beta^+)\right\|_{[F''(\beta^+)]^{-1}} \le \left\|p\beta^+(V\beta^+) - e\right\| \le \|F'(\beta)\|^2_{[F''(\beta)]^{-1}}.$$

Proof. Since $\left\|\beta^{-1}\Delta\beta\right\| \leq \|F'(\beta)\|_{[F''(\beta)]^{-1}} < 1$, then $\beta^+ = \beta(e + \beta^{-1}\Delta\beta) > 0$. Moreover,

$$\left\|p\beta^+(V\beta^+) - e\right\| = \|p(\beta + \Delta\beta)(V(\beta + \Delta\beta)) - e\|\,.$$

From

$$\left(\mathrm{diag}(\beta^{-2}) + pV\right)\Delta\beta = \beta^{-1} - pV\beta$$

we get

$$pV(\beta + \Delta\beta) = -\mathrm{diag}(\beta^{-2})\Delta\beta + \beta^{-1}.$$

We conclude that

$$\begin{aligned}\left\|p\beta^+(V\beta^+) - e\right\| &= \left\|(\beta + \Delta\beta)(-\mathrm{diag}(\beta^{-2})\Delta\beta + \beta^{-1}) - e\right\| \\ &= \left\|(\beta^{-1}\Delta\beta)^2\right\| \leq \left\|\beta^{-1}\Delta\beta\right\|^2. \qquad \square\end{aligned}$$

The above lemma shows that once the condition $\|F'(\beta)\|_{[F''(\beta)]^{-1}} \leq \eta < 1$ is met, one more Newton step is enough to generate a point satisfying

$$\left\|p(\beta^+)(V\beta^+) - e\right\| \leq \left\|\beta^{-1}\Delta\beta\right\|^2 \leq \|F'(\beta)\|^2_{[F''(\beta)]^{-1}} \leq \eta^2.$$

Thus, by Lemma 4.3, the point $\gamma^+ = V\beta^+ > 0$ satisfies

$$(1 - \eta^2)e \leq p\beta^+\gamma^+ \leq (1 + \eta^2)e.$$

7. Conclusion. In this paper, we define an efficient direction to restore primal and dual feasibility and centrality after adding p new central cuts simultaneously. The direction is efficient in the sense that it maximizes the product of the new variables brought into the primal or the dual potentials, under the constraints that the other variables remain within Dikin's ellipsoid. The computation of the optimal direction takes place in a space of dimension p equal to the number of cuts added at a given iteration. If p is sufficiently smaller than n, then significant gains in efficiency can be expected.

The analysis has been derived under the assumption that the cuts are central. If deep cuts are present, which is to be expected in practice, primal feasibility can always be recovered, but dual feasibility appears difficult to achieve in general, except by the use of a primal Newton method. One could then extend the long step argument of [8] in the case of one deep cut to multiple deep cuts.

The implementation of ACCPM [11] uses $\beta = \frac{1}{p}e$. Other choices using the variance-covariance matrix V, if it is invertible, have been proposed in [10], and the analysis of this paper actually strengthens that line of thinking.

Both the heuristic and optimal choices for β and γ need to be tested in practice, and extensions to multiple deep cuts deserve a more thorough study.

REFERENCES

[1] D. S. Atkinson and P. M. Vaidya, *A cutting plane algorithm that uses analytic centers*, Math. Programming, 69 (1995), pp. 1–43.

[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, 69 (1995), pp. 45–73.
[3] D. den Hertog, *Interior Point Approach to Linear, Quadratic and Convex Programming*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1994.
[4] J.-L. Goffin, J. Gondzio, R. Sarkissian, and J.-P. Vial, *Solving nonlinear multicommodity flows problems by the analytic center cutting plane method*, Math. Programming, 76 (1997), pp. 131–154.
[5] J.-L. Goffin, A. Haurie, and J.-P. Vial, *Decomposition and nondifferentiable optimization with the projective algorithm*, Management Sci., 38 (1992), pp. 284–302.
[6] J.-L. Goffin, Z.-Q. Luo, and Y. Ye, *Complexity analysis of an interior cutting plane for convex feasibility problems*, SIAM J. Optim., 6 (1996), pp. 638–652.
[7] J.-L. Goffin and F. Sharifi Mokhtarian, *Using the primal dual infeasible Newton method in the analytic center method for problems defined by deep cutting planes*, J. Optim. Theory Appl., 101 (1999), pp. 35–58.
[8] J.-L. Goffin and J.-P. Vial, *Shallow, deep and very deep cuts in the analytic center cutting plane method*, Math. Program., 84 (1999), pp. 89–103.
[9] J.-L. Goffin and J.-P. Vial, *A two-cut approach in the analytic center cutting plane method*, Math. Methods Oper. Res., 49 (1999), pp. 149–169.
[10] J. Gondzio, *Warm start of the primal–dual method applied in the cutting plane scheme*, Math. Programming, 83 (1998), pp. 125–143.
[11] J. Gondzio, O. du Merle, R. Sarkissian, and J.-P. Vial, *ACCPM–A library for convex optimization based on an analytic center cutting plane method*, European J. Oper. Res., 94 (1996), pp. 206–211.
[12] J. E. Mitchell and M. J. Todd, *Solving combinatorial optimization problems using Karmarkar's algorithm*, Math. Programming, 56 (1992), pp. 245–284.
[13] Y. Nesterov, *Cutting plane algorithms from analytic centers: Efficiency estimates*, Math. Programming, 69 (1995), pp. 149–176.
[14] Yu. Nesterov, *Introductory Lectures on Convex Optimization*, manuscript, Louvain, Belgium, 1996.
[15] Yu. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.
[16] Yu. Nesterov and J.-Ph. Vial, *Homogeneous analytic center cutting plane methods for convex problems and variational inequalities*, SIAM J. Optim., 9 (1999), pp. 707–728.
[17] C. Roos, T. Terlaky, and J.-P. Vial, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, Wiley-Intersci. Ser. Discrete Math. Optim., John Wiley & Sons, Chichester, UK, 1997.
[18] Y. Ye, *A potential reduction algorithm allowing column generation*, SIAM J. Optim., 2 (1992), pp. 7–20.
[19] Y. Ye, *Complexity analysis of the analytic center cutting plane method that uses multiple cuts*, Math. Programming, 78 (1997), pp. 85–104.
[20] Y. Ye, *Interior Point Algorithms: Theory and Analysis*, John Wiley & Sons, New York, 1997.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 289–307

SIMULATED ANNEALING WITH AN OPTIMAL FIXED TEMPERATURE*

MARK FIELDING†

Abstract. Contrary to conventional belief, it turns out that in some problem instances of moderate size, fixed temperature simulated annealing algorithms based on a heuristic formula for determining the optimal temperature can be superior to algorithms based on cooling. Such a heuristic formula, however, often seems elusive. In practical cases considered we include instances of traveling salesman, quadratic assignment, and graph partitioning problems, where we obtain results that compare favorably to the ones known in the literature.

Key words. simulated annealing, fixed temperatures

AMS subject classifications. 90C27, 65K05

PII. S1052623499363955

1. Introduction.

It has been seen, in Cohn and Fielding [6], that simulated annealing with a suitably chosen fixed temperature performs well for many instances of the traveling salesman problem (TSP). Investigated here is the existence of a (problem dependent) optimal fixed temperature, T_{opt}, and how this depends on the time available and the quality of solution required. We determine optimal fixed temperatures, allowing a predetermined number of iterations, to yield near-optimal solutions, and we compare the performance of fixed temperature simulated annealing against that of Aarts' algorithm [1] based on a cooling schedule.

Optimal fixed temperatures are experimentally determined for instances of the TSP, quadratic assignment (QAP), and graph partitioning (GPP) problems. Predictability of the optimum fixed temperature is also considered in the given applications.

The potential usefulness of using a fixed temperature algorithm is mentioned in Kirkpatrick [13], where it is considered suitable for small problem instances. We find the performance of simulated annealing with a fixed temperature to deteriorate as the size of problems increases, while for some problem instances it turns out to perform better than a fast cooling schedule. It is shown in Connolly [7] that fixed temperature schedules outperform fast cooling schedules for both small and large instances of the QAP.

Let $f(x)$ denote the value of the *objective* (*cost*) *function* associated with *configuration* x (a feasible solution to the problem instance at hand). Let $f_{\min}$ be the value of the objective function associated with global minima. Let S be the set of all feasible solutions.

At each iteration of simulated annealing a random perturbation is made to the current solution, x, giving rise to a set, $N(x)$, of *neighbors*. A neighbor, $y \in N(x)$, is accepted as the next configuration, with probability $\exp(-(f(y) - f(x))/T)$, and x is to otherwise remain.

*Received by the editors November 8, 1999; accepted for publication (in revised form) April 18, 2000; published electronically September 27, 2000.
http://www.siam.org/journals/siopt/11-2/36395.html
†Department of Mathematics and Statistics, The University of Melbourne 3010, Victoria, Australia (markf@ms.unimelb.edu.au).

2. Selected applications.

2.1. The TSP. The TSP is commonly used in investigating the performance and behavior of simulated annealing. The TSP is used in two founding papers on simulated annealing, Kirkpatrick, Gellat, and Vecchi [12] and Černý [5] and is more thoroughly followed up in Kirkpatrick [13] and Aarts, Korst, and van Laarhoven [3].

In the TSP we are given n cities and our task is to find the shortest path through each and all of the cities, returning to the city where the path arbitrarily commenced. Between each pair of cities is given a distance, although this could be, for example, time or cost of travel. We will consider symmetric cases of the TSP, where the distance from one city to another is equal to the distance in the reverse. Distances may be, among others, Euclidean (calculated from Cartesian coordinates of the cities), geographic (calculated from coordinates on the earth's surface), or (specified) road distances.

Let $p(i)$ denote the ith city to be visited, and let $d_{c_1\,c_2}$ denote the distance between cities c_1 and c_2. The objective function is the length of a given path and is given by

$$f = \sum_{i=1}^{n-1} d_{p(i)p(i+1)} \; + \; d_{p(n)p(1)}.$$

2.1.1. Implementation details. In applying simulated annealing to the TSP, the following method will be used. The cities are numbered $1, \ldots, n$ arbitrarily. A feasible solution is a given path and is stored as an n-entry array, representing the order in which the cities are to be visited (with multiple array assignments corresponding to the same path). The total number of possible paths is

$$|S| = \frac{(n-1)!}{2}.$$

An initial path is given by an arbitrary sequence of the numbers 1 to n, and for our purposes it is chosen randomly.

A neighbor of a given path, for a symmetric TSP, is generated by choosing two cities randomly, say, the ith and the jth city visited, and reversing the order in which the cities $i+1$ (modulo n) up to j are visited. This is called a 2-*opt* move.

The 2-opt move leads to a neighborhood size of

$$|N(x)| = \frac{n(n-3)}{2},$$

if only neighbors that differ from the existing solution are allowed. (Choosing j as following or preceding i would result in no change to the path.) The following pseudocode gives a possible way of generating i and j:

```
k = integer(random()*n*(n-3));
i = (k mod n)+1;
j = ((i+1+(k div n)) mod n)+1;
```

it requires only one random number to generate each pair.

The distances, either explicitly given or calculated, are stored as a 2-dimensional array and are assigned integer values as instructed in TSPLIB [18]. The change in the objective function, resulting from a 2-opt move, is given by

$$\Delta f = -d_{p(i)p(i+1)} - d_{p(j)p(j+1)} + d_{p(i)p(j)} + d_{p(i+1)p(j+1)}.$$

TABLE 2.1
Selected instances of the TSP.

Instance	n	$f_{\min}$	Source
`gr48`	48	5046	[9], [19]
`kt57`	57	12955	[11]
`eil76`	76	538	[17], [19]
`kroA100`	100	21282	[14], [19]
`gr120`	120	6942	[8], [19]
`pr152`	152	73682	[17], [19]
`kroA200`	200	29368	[14], [19]
`pr264`	264	49135	[17], [19]
`lin318`	318	42029	[15], [19]
`gr442`	442	5069	[9], [19]

2.1.2. Selected TSP instances. Ten TSP instances have been selected, ranging from 48 to 442 cities. Among the instances chosen are those examined in Aarts and van Laarhoven [2]. The selected problem instances are shown in Table 2.1, along with the source of each problem. Except for `kt57`, these instances are available from the website TSPLIB [19]. Note that `lin318` is treated as a TSP rather than a Hamiltonian circuit. Also, `gr442` is taken in the form given in Grötschel and Holland [9] rather than as it appears in TSPLIB as `pcb442` (where it differs in scale and precision).

2.2. The QAP. The QAP occurs in many contexts, often considered as assigning n facilities to n locations. Let $p(i)$ denote the location to which facility i is to be assigned. Between each pair of facilities, i and j, there is given a *flow*, a_{ij}, and between each pair of locations, k and l, there is given a distance, b_{kl}. The objective function is given by

$$f = \sum_i \sum_j a_{ij}\, b_{p(i)p(j)}.$$

There may also be considered an additional cost $c_{i\,p(i)}$ of assigning facility i to location $p(i)$. This cost shall not be considered here. We shall also restrict our attention to symmetric cases of the QAP, that is, where the distance matrix is symmetric.

2.2.1. Implementation details. A feasible solution to the QAP is a given allocation of facilities to locations. This is stored as an n-entry array where the ith entry represents the location to which facility i is located, i.e., $p(i)$. The flow between facilities, and the distances between locations, are stored as 2-dimensional arrays. An initial allocation of facilities may be arbitrarily chosen, and for our purposes it is chosen randomly. The number of possible allocations totals

$$|S| = n!.$$

A neighbor, of a given solution x, is generated by randomly choosing two facilities, i and j, and interchanging their locations. This yields a neighborhood size of

$$|N(x)| = \frac{n(n-1)}{2}.$$

The change in the objective function simplifies to

$$\Delta f = 2 \sum_{k \neq i,j} (a_{kj} - a_{ki})\,(b_{p(k)p(i)} - b_{p(k)p(j)}).$$

TABLE 2.2
Selected instances of the QAP.

Instance	n	$f_{\min}$	Source
nug15	15	1150	[16], [4]
rou15	15	354210	[4]
nug20	20	2570	[16], [4]
nug30	30	(6124)	[16],[4]
kra30a	30	(88900)	[4]
wil50	50	(48816)	[21],[4]
wil100	100	(273038)	[21], [4]
sko100a	100	(152002)	[20],[4]

2.2.2. Selected QAP instances. Eight QAP instances have been gained from the website QAPLIB [4]. The instances are given in Table 2.2. The value given for $f_{\min}$ is given in brackets when it is only the best found solution in the literature, not a proven global minimum.

2.3. The GPP. In the GPP, a graph (V, E) is given, where V is a set of n vertices and E is the set of edges. With each edge (i, j) is assigned a weight. The aim is to allocate the vertices of the graph to a given number of equally sized groups in such a way that the sum of the weights of all edges crossing between groups is minimized.

A simple case of the GPP, considered here, is when a graph is to be partitioned into two equally sized groups, V_1 and V_2, and the objective function is given by the total number of edges crossing between the two groups.

In the application of simulated annealing to such GPPs, Johnson et al. [10] find it advantageous to extend the set of feasible solutions to allow unequal partitions while adding a penalty to the objective function to encourage equally sized groups. The updated objective function is given by

$$f = |\{(i,j) \in E : \ i \in V_1 \text{ and } j \in V_2\}| + \alpha(|V_1| - |V_2|)^2, \tag{2.1}$$

where α is a suitably chosen constant. This means that a solution which is near optimal according to the updated objective function need not be a (true) feasible solution. If the final solution is not feasible, then, repeatedly, a vertex in the larger group is to be transferred to the smaller such that the true objective function increases as little as possible, until the graph is equally partitioned.

Our investigation into the application of simulated annealing to the GPP shall be regarding its ability to locate a partition, equal or otherwise, that minimizes the updated objective function (2.1). We shall also fix α at 0.05, the value found suitable in [10].

2.3.1. Implementation details. The vertices of a graph are arbitrarily numbered $1, \ldots, n$. We have stored a graph as a 2-dimensional array, where the ith row lists the vertices that are connected by an edge to the ith vertex. The degree of a vertex is the number of edges joined to it. A given configuration is represented by an n-entry array with the ith entry denoting to which group vertex i has been allocated. The number of possible configurations is

$$|S| = 2^n,$$

which differs from the number of equally partitioned solutions.

TABLE 2.3
Generated instances of the GPP.

Instance	n	Average degree		$f_{\min}$
		Expected	Actual	
rand124	124	5	4.9	(58.8)
rand250	250	5	5.2	(126.0)
geom250	250	30	25.8	(248.8)
rand500	500	5	5.0	(223.0)
rand500b	500	10	10.2	(695.0)
geom500	500	20	19.0	(174.8)
geom500b	500	40	34.8	(578.8)
rand1000	1000	5	5.2	(483.0)
rand1000b	1000	20	20.3	(3410.0)
geom1000	1000	10	9.3	(30.0)

A neighbor of a given configuration is generated by randomly selecting a vertex, and swapping to which group it belongs. This yields a neighborhood size of

$$|N(x)| = n.$$

The resulting change in the objective function, moving vertex i from V_1 to V_2, is given by

$$\begin{aligned}\Delta f = &|\{j \in V_1 : \ (i,j) \in E\}| - |\{j \in V_2 : \ (i,j) \in E\}| \\ &- 4\,\alpha\,(|V_1| - |V_2| - 1).\end{aligned}$$

Note that $((|V_1| - 1) - (|V_2| + 1))^2 = (|V_1| - |V_2|)^2 - 4(|V_1| - |V_2| - 1)$.

2.3.2. Selected GPP instances. Ten GPP instances have been generated according to the method described in Johnson et al. [10], where two types of graphs are considered, *geometric* and *random*. In the geometric case, points (vertices) are randomly generated in a unit square, and any two distinct vertices within distance d from each other are allocated an edge. In the random case, edges are added between each pair of distinct vertices independently with probability p. In each case, d or p are chosen to achieve a specified expected average degree D,

$$p = \frac{D}{n-1},$$

and

$$d = \sqrt{\frac{D}{n\,\pi}}.$$

The generated problems are listed in Table 2.3. For these instances the global minima are not known, and the value for $f_{\min}$ for each problem instance is given in brackets to show that it is merely the best solution obtained here.

3. Fast cooling schedules. In practice, cooling schedules tend to 0 too quickly to allow reaching global minima solutions with probability even close to 1. Two fast cooling schedules are utilized here, Aarts' cooling schedule [1] and the geometric cooling schedule [12]. Aarts' cooling schedule is used to determine the number of iterations appropriate for reaching near optimal solutions. The geometric schedule has been selected as a simple, easy-to-implement cooling schedule, potentially to assist in determining suitable fixed temperatures.

Aarts [1]: Temperature is held fixed during each loop of $R = |N(x)|$ iterations. At the end of each loop the temperature is dropped according to the rule

$$T_{k+1} = T_k \Big/ \left(1 + \frac{T_k \log(1+\delta)}{3\sigma_k}\right)$$

for some small real number δ ($\delta = 0.1$ is recommended in [1]). Also, σ_k is the standard deviation of the values of the cost function observed during the kth loop of the algorithm.

Geometric [12]: Temperature is again held fixed during each loop of $R = |N(x)|$ iterations. At the end of each loop the temperature is dropped according to the rule

$$T_{k+1} = \alpha\, T_k$$

for some $\alpha < 1$ (say, $0.8 \leq \alpha < 1$). We found the number of iterations performed at each loop had little effect on the algorithm's performance, provided the value of α is adjusted appropriately to give the same overall rate of cooling.

An initial temperature, T_0, is typically determined to yield a specified acceptance of proposed moves, say, 95%. In this paper, T_0 is determined by trial and error and is specified for each problem instance considered.

4. Determining N. The number of iterations to be allowed in the fixed temperature algorithms is denoted by N.[1] We wish to choose an N, as in [6], for each problem instance, appropriately large for simulated annealing to find near optimal solutions.

We have chosen Aarts' algorithm to determine N, although alternative algorithms could have been used. Aarts' algorithm does not require the determining of the cooling parameter δ for each individual problem instance. 100 runs of Aarts' algorithm are performed with the parameter setting ($\delta = 0.1$) recommended by Aarts and others [1, 2, 3]. In choosing N, we consider the number of iterations taken until first visiting the best solution found in each run. The maximum of these is taken, after removing outliers. An outlier is taken as a value more than 1.5 times the interquartile range greater than the third quartile.

5. Searching for an optimal fixed temperature.

5.1. Which optimal fixed temperature? As discussed in [6], determining an optimal temperature schedule may depend on which optimality criterion is adopted. We discuss here various optimality criteria in determining an optimal fixed temperature schedule. The influence of these criteria is then investigated experimentally with a 100-city TSP instance.

An optimal fixed temperature may be chosen experimentally, by running simulated annealing a number of times at each of a number of fixed temperatures and determining which temperature is best, according to an appropriate optimality criterion.

If it is our goal to find a global minimum in the shortest possible time, then an optimal fixed temperature might be considered one yielding a stopping time with

$$\mathbb{E}[\tau(T_{\text{opt}})] = \min_{T \in [0,\infty]} \mathbb{E}[\tau(T)], \tag{5.1}$$

[1] Note that N has been used in denoting the set of neighbors of x, $N(x)$, so N is used here.

where $\tau(T)$ is the time until reaching a global minimum using a fixed temperature T and a random initial state and $[0, \infty]$ is the set of nonnegative reals in union with positive infinity. Note that $T = 0$ relates to *iterative improvement*, where only transitions to improved solutions are accepted, yielding $\mathbb{E}[\tau] = \infty$ (providing local minima that are not global minima exist), and $T = \infty$ is the case where all proposed transitions are accepted.

If a global minimum is unlikely to be reached in a reasonable time frame, then we may simply consider reaching near-optimal solutions. In such a case, τ in (5.1) might represent the time until reaching a configuration x with objective function $f(x) \leq f_{\min} + \epsilon$. The temperature that is deemed optimal might then depend on ϵ.

In Connolly [7] is demonstrated the existence of optimal fixed temperatures for a number of QAP instances. Such a temperature is identified by considering the best solution in a given number of iterations, N, and taking the fixed temperature that on average yields the best result. Connolly determines the fixed temperature to yield the optimal value of

$$\mathbb{E}\left[\min\{f(X_1), \ldots, f(X_{\mathrm{N}})\}\right], \tag{5.2}$$

where X_t is the configuration visited in the tth iteration. The temperature deemed optimal under such a scheme may then depend on the choice of N.

An optimal fixed temperature may alternatively be considered one which yields the maximum probability of reaching a global (or near global) minimum within N iterations. That is,

$$P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) = \max_{T \in [0,\infty]} P(\tau(T) \leq \mathrm{N}). \tag{5.3}$$

If after N iterations the temperature is to be set to zero, allowing the algorithm to quickly settle in a local minimum, then this may also influence which temperature is deemed optimal.

5.2. A 100-city TSP example. We now investigate the influence of the above-mentioned criteria in determining an optimal fixed temperature for the 100-city TSP, `kroA100`. We determine optimal fixed temperatures according to (5.1), with respect to reaching global minima, and within 1 and 2 percent of global minima. We estimate the three corresponding values for $\mathbb{E}[\tau(T_{\mathrm{opt}})]$. With N set to each of these values we then go on to determine optimal fixed temperatures according to (5.2) and (5.3). These results are obtained by running 100 runs of simulated annealing at each of a number of fixed temperatures, under each scheme. The range of temperatures to be considered is determined experimentally, with temperatures outside this range yielding increasingly inferior results. The results of these runs are shown graphically in Figure 5.1.

In determining optimal fixed temperatures according to (5.1), the results are summarized as follows:

Reaching global minimum,

$$T_{\mathrm{opt}} \in (38, 41), \quad \mathbb{E}[\tau(T_{\mathrm{opt}})] \approx 100{,}000{,}000.$$

Reaching within 1% of global minimum:

$$T_{\mathrm{opt}} \in (45, 50), \quad \mathbb{E}[\tau(T_{\mathrm{opt}})] \approx 2{,}000{,}000.$$

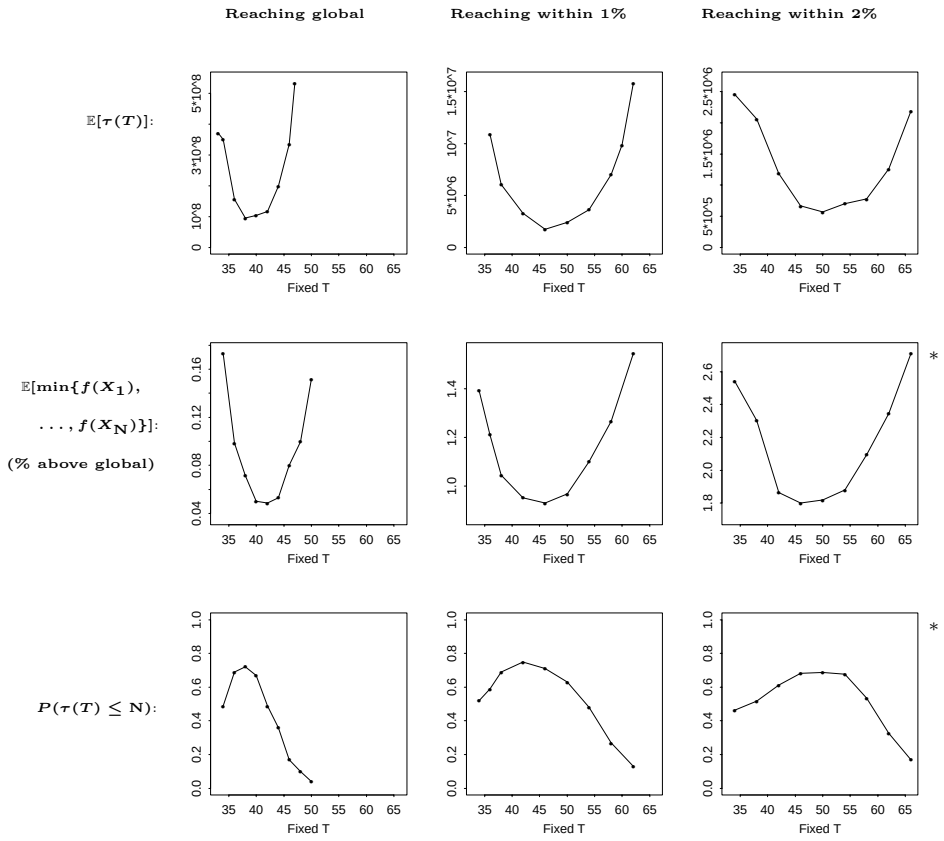


FIG. 5.1. *Locating the optimal fixed temperature for* `kroA100` *under various criteria, with respect to times until reaching global minima and within* 1% *and* 2% *of global minima. Graphs show estimates based on* 100 *runs (*500 *runs) of simulated annealing at each of the various fixed temperatures.*

Reaching within 2% of global minimum:

$$T_{\text{opt}} \in (48, 53), \quad \mathbb{E}[\tau(T_{\text{opt}})] \approx 600{,}000.$$

It is apparent that the temperature found optimal according to (5.1) depends on the required quality of solution.

In determining optimal fixed temperatures according to (5.2), the runs of simulated annealing are repeated, each terminating after first setting the temperature to zero after N iterations, for the respective values of N. The results are summarized as follows:

For N = 100,000,000,

$$T_{\text{opt}} \in (40, 43), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.0005\, f_{\min}.$$

For N = 2,000,000,

$$T_{\text{opt}} \in (43, 46), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.009\, f_{\min}.$$

For N = 600,000,

$$T_{\text{opt}} \in (45, 50), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.018\, f_{\min}.$$

It is apparent that the optimal temperature according to (5.2) depends on the choice of N.

Figure 5.1 also shows estimates for $P(\tau(T) < \mathrm{N})$, where $\tau(T)$ is the time until reaching the global minimum and within 1 and 2 percent of the global minimum, for N set to 100,000,000, 2,000,000, and 600,000,000, respectively. The results are summarized as follows:

For N = 100,000,000 and τ the time until reaching the global minimum,

$$T_{\mathrm{opt}} \in (36, 38), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.71.$$

For N = 2,000,000 and τ the time until reaching within 1% of the global minimum,

$$T_{\mathrm{opt}} \in (40, 44), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.77.$$

For N = 600,000 and τ the time until reaching within 2% of the global minimum,

$$T_{\mathrm{opt}} \in (45, 50), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.69.$$

It is apparent that the temperature found optimal according to (5.3) depends on the required quality of solution.

6. Predicting the optimal fixed temperature.

6.1. The TSP. For the TSP, values for T_{opt} are given in Table 6.1, along with various parameters which may assist in predicting T_{opt}. The optimal fixed temperatures are determined experimentally by running 100 trials of simulated annealing at each of a number of fixed temperatures, using a fixed number of iterations, N, and noting, or interpolating to determine, the temperature which yields the best solution on average. Aarts' algorithm is used to determine N, as described in section 4, using an initial temperature T_0.

TABLE 6.1

Parameters to help predict T_{opt}, for various instances of the TSP. Values for T_{opt} are experimentally determined allowing N *iterations.*

Instance	T_0	N	T_{opt}	n	$f_{\min}$
gr48	2800	509760	20	48	5046
kt57	6000	857223	40	57	12955
eil76	200	1795441	1.4	76	538
kroA100	11700	4243750	46	100	21282
gr120	2900	7104240	11	120	6942
pr152	44500	14640064	75	152	73682
kroA200	11800	29509991	34	200	29368
pr264	32500	67095121	37.5	264	49135
lin318	11800	102173400	26	318	42029
gr442	2420	242935584	2.2	442	5069

	$\epsilon\%$	χ	f_{best}	T_{best}
gr48	3.5	0.0104	5088	14.2
kt57	3.9	0.0082	13021	18.2
eil76	4.2	0.0040	559	0.8
kroA100	3.7	0.0043	21449	16.5
gr120	3.6	0.0030	7156	2.6
pr152	3.0	0.0031	74983	18.3
kroA200	4.6	0.0025	29890	7.7
pr264	2.4	0.0007	50156	15.6
lin318	4.4	0.0013	43220	3.2
gr442	3.8	0.0005	5223	1.0

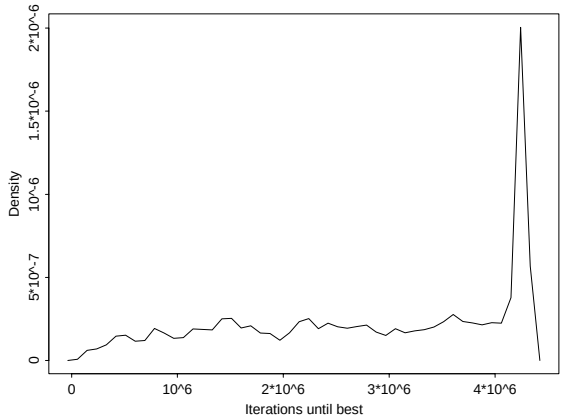


FIG. 6.1. *Iterations until reaching the best solution in each run of* 100 *runs performed of fixed temperature simulated annealing using* T_{opt}*, for* `kroA100`*.*

For each problem instance, estimates for the acceptance ratio, $\chi = \chi(T_{\text{opt}})$, are given at the optimal fixed temperature, along with $\epsilon\%$, the average value of the objective function (as percentage above global minima) observed at this temperature.

There is some uncertainty as to how estimates for χ and $\epsilon\%$ would best be gained. If (quasi) equilibrium[2] is not easily obtained at the optimal fixed temperature, then differing methods may yield differing estimates. For example, running a cooling schedule before continuing with a fixed temperature, and then observing average behavior, may yield different results to a usual fixed-temperature algorithm. By looking at the occurrences of the best solution found in each run it is seen that this is, roughly, consistently likely to occur throughout the course of a fixed temperature algorithm, suggesting that equilibrium at T_{opt} is easily obtained. This is seen in Figure 6.1, with a smoothed histogram of the iterations taken until reaching the best found solution in each fixed temperature run for the problem instance `kroA100`, using $T = T_{\text{opt}}$. A peak at N = 4243750 iterations corresponds to the temperature being set to zero. It is apparent that a *burn-in* period of up to 1 million iterations is required, to allow the algorithm to exhibit equilibrium.

Results for $\epsilon\%$ and χ are obtained after allowing a burn-in period of $1,000$ loops of $R = |N(x)|$ iterations. Upon reaching this many iterations the algorithms are continued for $1,000$ further loops in gaining estimates. These values are merely obtained in order to investigate the potential for such parameters in determining an optimal fixed temperature. In practice, more efficient means of estimation may be used.

From the 10 problem instances considered, the average value of the objective function is 3.7% above the global minimum. The average acceptance ratio is 0.0038, with the desirable value for χ tending to decrease as the size of the TSP instances increases.

Also given in Table 6.1 is the temperature, T_{best}, at which the best visited solution, f_{best}, occurred in one run of simulated annealing based on a geometric cooling schedule, with cooling parameter $\alpha = 0.95$ and initial temperature T_0. It was deter-

[2]It is described in Černý [5] how, at a fixed temperature, simulated annealing will tend toward an equilibrium, where the average of observed values of the objective function tends toward an equilibrium value.

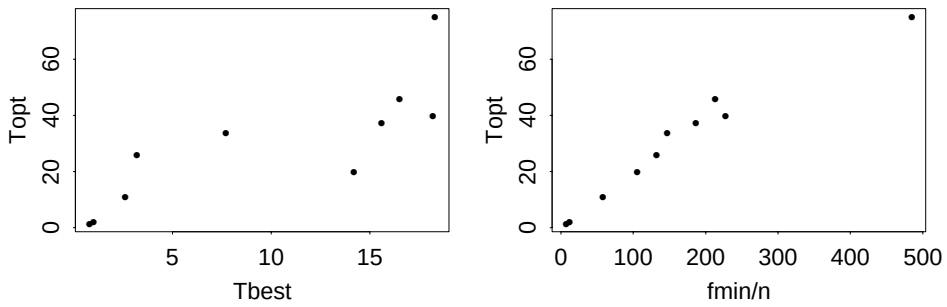


FIG. 6.2. *Relationship between T_{opt} and T_{best}, as well as between T_{opt} and f_{min}, for the TSP.*

mined by Connolly [7] that a suitable fixed temperature may be chosen by running simulated annealing with a fast cooling schedule and observing the temperature at which the best solution found first occurred. Indeed, while Connolly suggests using his values for T_{best} exactly, we find only that there exists a weak relationship between T_{opt} and T_{best} (see Figure 6.2).

In using T_{best} to make predictions of T_{opt}, the results of the problem instances yield on average

$$\hat{T}_{\text{opt}} = 3.4 \times T_{\text{best}}.$$

A reliable way to predict the optimal fixed temperature in the TSP is by a formula developed next (and noted in [6]). From Figure 6.2, it is apparent that a strong relationship exists between T_{opt} and the value of

$$\frac{f_{\text{min}}}{n}.$$

The ratios between T_{opt} and f_{min}/n are

$$0.190,\ 0.176,\ 0.198,\ 0.216,\ 0.190,\ 0.155,\ 0.232,\ 0.201,\ 0.197,\ 0.192,$$

and on average we have

$$\hat{T}_{\text{opt}} = 0.19 \times \frac{f_{\text{min}}}{n}. \tag{6.1}$$

In practice, the optimal solution to a given problem instance need not be known, and Table 6.1 also yields on average

$$\hat{T}_{\text{opt}} = 0.19 \times \frac{f_{\text{best}}}{n}.$$

It is interesting to note that the relationship in (6.1) describes a link between the optimal fixed temperature and the average link length in the global minimum solution, with

$$f = \sum_{i=1}^{n-1} d_{p(i)p(i+1)} \ + \ d_{p(n)p(1)}.$$

TABLE 6.2
Predictions of the optimal fixed temperature for various instances of the TSP.

Instance	T_{opt}	by (6.1)	by T_{best}	by $\epsilon\%$	by χ
gr48	20	20	48	21	8
kt57	40	43	62	40	28
eil76	1.4	1.3	2.7	1.3	1.3
kroA100	46	40	56	47	42
gr120	11	11	8.8	11	12
pr152	75	92	62	85	85
kroA200	34	22	26	32	44
pr264	37.5	35	53	42	60
lin318	26	25	11	21	44
gr442	2.2	2.2	3.4	2.2	5.3

Note that the change in objective function due to a 2-opt move is given by the subtraction and addition of links,

$$\Delta f = -d_{p(i)p(i+1)} - d_{p(j)p(j+1)} + d_{p(i)p(j)} + d_{p(i+1)p(j+1)},$$

and the optimal fixed temperature appears proportional to the average link length in the optimal path.

The transition probabilities between solutions depend on the ratio of the change in path length to the temperature:

$$P(\text{transition accepted}) = \exp(-\Delta f/T).$$

The long-term probability of being found in particular configurations, by the theory of Markov chains, is determined by such transition probabilities. This may give insight into developing approximate formulae in determining suitable fixed temperatures in other applications.

6.2. Performance of T_{opt} predictions in the TSP. To assess the merits of T_{opt} predictions, given in Table 6.2, a χ^2 *goodness-of-fit* test will be employed,

$$\sum \frac{(o-e)^2}{e} \stackrel{\text{d}}{=} \chi^2_{k-1},$$

where o are the observed values of T_{opt} and e are the expected values under a given method of prediction. The value obtained is to be compared against a χ^2 probability distribution with $k-1$ degrees of freedom, with k the number of values considered. The parameters χ and ϵ are used to predict T_{opt} by experimentally determining the temperatures yielding the same values that were observed on average.

For the four methods of prediction considered, (6.1), and by T_{best}, $\epsilon\%$, and χ, the χ^2 values obtained are 5.76, 58.6, 3.05, 44.7, respectively. The 5% upper tale of the χ^2_9 distribution occurs after 16.9, so that only (6.1), or by using $\epsilon\%$, yields satisfactory results.

It is apparent that predictions based on χ tend to be too small for the smaller instances while too large with the larger instances. The desirable value for χ is seen to decrease as the size of problems increases, so that using only an average value for χ would be expected to lead to such a result. Indeed, we shall see that different applications, as well as different problem structures, have different desirable values for χ. This will also be seen to be the case with $\epsilon\%$, which suggests that monitoring χ and $\epsilon\%$ would not generally be suitable in determining a suitable fixed temperature.

TABLE 6.3

Performance of the optimal fixed temperature for various instances of the TSP, against Aarts' algorithm. Percentage above global minimum of best solution on average in 100 *runs of the algorithms. Parentheses are used with repeated values due to accurate predictions.*

	% above global		
Instance	Aarts	with T_{opt}	by (6.1)
gr48	0.93	0.20	(0.20)
kt57	0.99	0.59	0.68
eil76	2.26	0.39	0.39
kroA100	0.78	0.55	0.60
gr120	1.83	0.85	(0.85)
pr152	0.73	0.59	0.68
kroA200	1.40	1.58	1.66
pr264	1.11	0.84	0.86
lin318	1.73	2.20	2.28
gr442	1.66	1.99	(1.99)

We gauge the performance of of fixed-temperature simulated annealing, and (6.1), against Aarts' algorithm. Table 6.3 gives the results of 100 runs of each algorithm. The fixed temperature algorithms are allowed N iterations, as determined with Aarts' algorithm. The given values are subject to variation, and of interest is the difference between using T_{opt} and (6.1) compared with using Aarts' algorithm.

It can be seen that as the size of problem instances increases, the relative performance of fixed-temperature algorithms decreases. For the smaller instances, fixed-temperature algorithms show to perform far better. Using (6.1) to predict values for T_{opt} yields consistently favorable results.

Although (6.1) closely describes the optimal temperatures for the given instances, it need not perform as well for TSP instances in general. It is the experience of the author, however, that (6.1) performs well in determining suitable fixed temperatures for TSPs in general.

For the TSP, fixed-temperature simulated annealing shows to outperform fast cooling in problems of less than 150 cities.

6.3. The QAP. For the QAP, values for T_{opt}, along with various parameters to assist in predicting T_{opt}, are given in Table 6.4, as carried out with the TSP.

The average value for $\epsilon\%$ is 3% and for χ is 0.03. Again, the value for χ tends to decrease as the size of problems increases, but this tends to be the case here also with $\epsilon\%$. Predictions of T_{opt} based on $\epsilon\%$ and χ will therefore not be further considered for the QAP.

In Figure 6.3, a strong relationship between T_{opt} and $f_{\min}/n$ is not found with the QAP. We consider instead insight gained from the case with the TSP. For the QAP, the objective function is given by

$$f = \sum_i \sum_j a_{ij}\, b_{p(i)p(j)},$$

which relates the average value of $a_{ij}\, b_{p(i)p(j)}$ times the square of the size of the problem instance. The change in objective function relates to the addition and subtraction of terms $a_{ij}\, b_{p(i)p(j)}$, with, when interchanging facilities i and j,

$$\Delta f = 2 \sum_{k \neq i,j} \left[-a_{ki}\, b_{p(k)p(i)} - a_{kj}\, b_{p(k)p(j)} + a_{ki}\, b_{p(k)p(j)} + a_{kj}\, b_{p(k)p(i)} \right].$$

TABLE 6.4

Parameters to help predict T_{opt}, for various instances of the QAP. Values for T_{opt} are experimentally determined allowing N *iterations.*

Instance	T_0	N	T_{opt}	n	$f_{\min}$
nug15	360	15691	8.0	15	1150
rou15	96000	13627	2700	15	354210
nug20	525	35360	9.5	20	2570
nug30	780	121313	10.5	30	(6124)
kra30a	16500	122621	300	30	(88900)
wil50	1550	568395	12	50	(48816)
wil100	2700	3894148	24	100	(273038)
sko100a	2550	3824669	18	100	(152002)

	$\epsilon\%$	χ	f_{best}	T_{best}
nug15	4.1	0.048	1174	6.6
rou15	7.0	0.049	371986	999.2
nug20	3.3	0.033	2604	3.3
nug30	1.9	0.022	6180	4.2
kra30a	5.6	0.048	91000	126.2
wil50	0.5	0.015	49014	3.0
wil100	0.6	0.018	273674	6.0
sko100a	0.6	0.009	152532	1.6

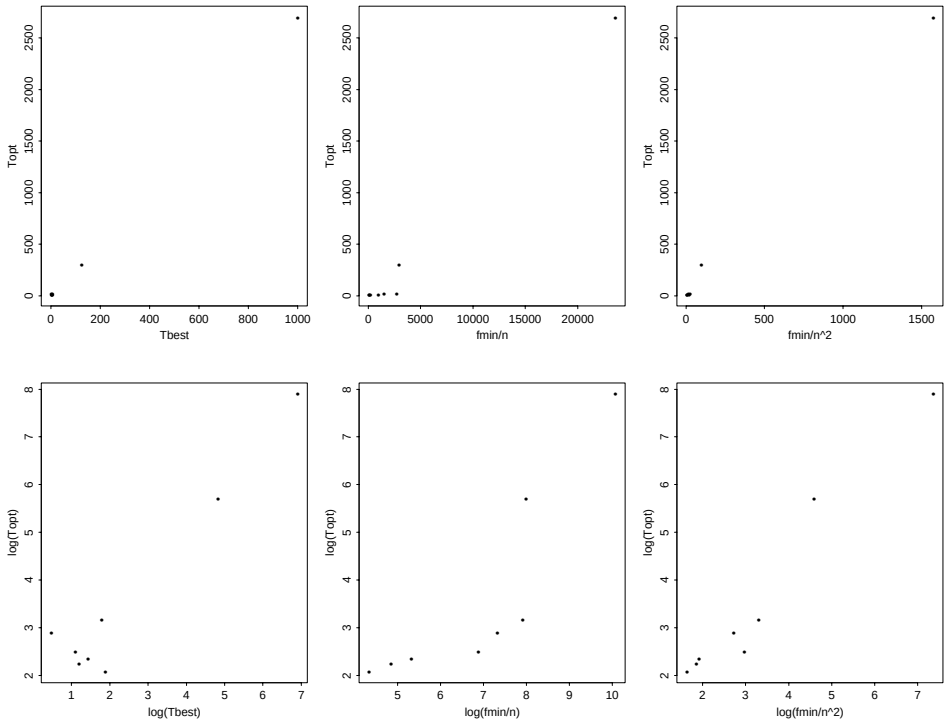


FIG. 6.3. *Relationship between T_{opt} and T_{best}, as well as between T_{opt} and the optimum value of the objective function, for the QAP.*

TABLE 6.5
Predictions of the optimal fixed temperature for various instances of the QAP.

Instance	T_{opt}	by (6.2)	by T_{best}
`nug15`	8.0	7.7	18.5
`rou15`	2700	2361	2798
`nug20`	9.5	9.6	9.2
`nug30`	10.5	10.2	11.8
`kra30a`	300	148	353
`wil50`	12	29	8.4
`wil100`	24	41	17
`sko100a`	18	23	4.5

We therefore examine, in analogy with the case for the TSP, T_{opt} relating to

$$\frac{f_{\min}}{n^2}.$$

Figure 6.3 considers graphically the relationship between T_{opt} and $f_{\min}/n^2$, as well as those between T_{opt} and $f_{\min}/n$ and between T_{opt} and T_{best}. There appears a relationship between T_{opt} and $f_{\min}/n^2$, although it is not as strong as the analogous relationship seen with the TSP. Only a weak relationship is seen between T_{opt} and T_{best}.

Looking at the average ratio between T_{opt} and $f_{\min}/n^2$, we get

$$\hat{T}_{\text{opt}} = 1.5 \times \frac{f_{\min}}{n^2}. \tag{6.2}$$

Predictions may be gained by the average ratio between T_{opt} and T_{best},

$$\hat{T}_{\text{opt}} = 2.8\, T_{\text{best}},$$

found after excluding (outlying) `sko100a`.

6.4. Performance of T_{opt} predictions in the QAP. Table 6.5 shows predictions for T_{opt} that would result, with various methods of prediction.

Although a relationship is apparent between T_{opt} and $f_{\min}/n^2$, this does not lead to a reliable method of predicting T_{opt}. Predictions that would result give a χ^2 value of 222.9. Between $\log(T_{\text{opt}})$ and $\log(f_{\min}/n^2)$, we have the ratios

$$1.275,\ 1.073,\ 1.210,\ 1.226,\ 1.242,\ 0.836,\ 0.961,\ 1.062,$$

with an average of 1.1. By this log-relationship we would get the predictions

$$6.0,\ 3287,\ 7.7,\ 8.2,\ 156,\ 26,\ 38,\ 20,$$

with a χ^2 value of 252.4. Neither method gives a satisfactory explanation for the variation in T_{opt}, with a 5% critical value for a χ^2_7 distribution of 14.1.

The relationship given with T_{best} yields a χ^2 value of 62.4, which is also deemed not satisfactory.

With the QAP from Table 6.6, fixed-temperature simulated annealing appears to outperform the fast cooling schedule for instances of size less than 50 facilities to be allocated.

TABLE 6.6

Performance of the optimal fixed temperature for various instances of the QAP, against Aarts' algorithm. Percentage above global minimum of best solution on average in 100 *runs of the algorithms.*

	% above global	
Instance	Aarts	with T_{opt}
nug15	1.30	0.38
rou15	3.41	1.81
nug20	1.48	0.45
nug30	1.01	0.49
kra30a	2.46	1.94
wil50	0.18	0.27
wil100	0.12	0.28
sko100a	0.22	0.37

TABLE 6.7

Parameters to help predict T_{opt}*, for various instances of the GPP. Values for* T_{opt} *are experimentally determined allowing* N *iterations.*

Instance	T_0	N	T_{opt}	n	degree	$f_{\min}$
rand124	22.0	489499	0.50	124	4.9	(58.8)
rand250	25.5	1440837	0.45	250	5.2	(126.0)
geom250	44.0	1028574	6.0	250	25.8	(248.8)
rand500	28.0	4229417	0.40	500	5.0	(223.0)
rand500b	33.5	4269441	0.50	500	10.2	(695.0)
geom500	43.0	3245665	5.0	500	19.0	(174.8)
geom500b	55.0	3104347	8.0	500	34.8	(578.8)
rand1000	31.0	12064888	0.35	1000	5.2	(483.0)
rand1000b	46.0	12460153	0.70	1000	20.3	(3410.0)
geom1000	36.0	10232663	2.0	1000	9.3	(30.0)

	$\epsilon\%$	χ	f_{best}	T_{best}
rand124	10.0	0.10	63.0	0.23
rand250	6.7	0.08	136.0	0.11
geom250	84.2	0.16	248.8	2.25
rand500	6.0	0.05	250.0	0.17
rand500b	2.4	0.04	712.0	0.17
geom500	214.3	0.16	273.8	1.31
geom500b	82.9	0.13	585.8	1.12
rand1000	5.8	0.04	531.0	0.09
rand1000b	1.3	0.04	3461.2	0.22
geom1000	900.2	0.10	200.0	0.36

6.5. The graph partitioning problem. Table 6.7 shows values for T_{opt} and values for various parameters to help in its prediction (as previously described) for the 10 GPP instances considered.

Figure 6.4 considers a possible relationship between T_{opt} and $f_{\min}/n$ and a possible relationship between T_{opt} and T_{best}. For the GPP instances considered, the average ratio between T_{opt} and T_{best} gives

$$\hat{T}_{\text{opt}} = 3.8 \times T_{\text{best}}.$$

The average acceptance ratio for the 10 problem instances is 0.09, though there is a tendency for the geometric graph instances to relate to larger acceptance ratios, at T_{opt}, than the random instances. The acceptance ratio therefore will not be further considered toward predicting T_{opt} with the GPP.

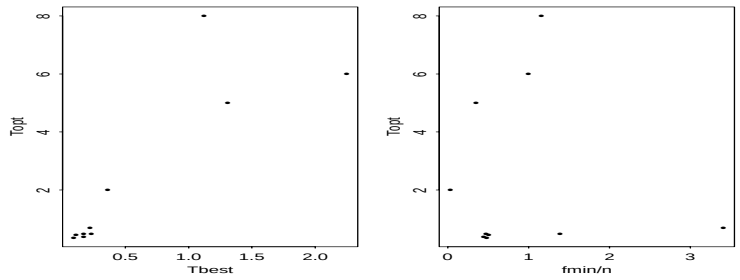


FIG. 6.4. *Investigating possible relationships toward helping predict T_{opt} with the GPP.*

TABLE 6.8
Predictions of the optimal fixed temperature for various instances of the GPP.

Instance	T_{opt}	by T_{best}
rand124	0.50	0.87
rand250	0.45	0.42
geom250	6.0	8.6
rand500	0.40	0.65
rand500b	0.50	0.65
geom500	5.0	5.0
geom500b	8.0	4.3
rand1000	0.35	0.34
rand1000b	0.70	0.84
geom1000	2.0	1.4

TABLE 6.9
Performance of the optimal fixed temperature for various instances of the GPP, against Aarts' algorithm. Best solution on average for 100 *runs of each algorithm, given as the percentage above the best found solution.*

	% above best	
Instance	Aarts	with T_{opt}
rand124	0.76	0.14
rand250	0.40	0.02
geom250	0.55	0.89
rand500	0.97	1.25
rand500b	0.53	0.50
geom500	1.11	7.88
geom500b	0.82	1.94
rand1000	1.58	2.05
rand1000b	0.16	0.56
geom1000	132.55	237.43

The average percentage above global minima, at T_{opt}, varies greatly in the GPP instances considered, with the geometric instances yielding far larger values than the random graph instances, so that $\epsilon\%$ is also not further considered in predicting T_{opt}.

The variation in T_{opt} for the GPP instances may largely be explained by classifying the instances as geometric and random graphs, and the optimal fixed temperature with GPP instances shows to depend largely on problem structure.

6.6. Performance of T_{opt} predictions in the GPP. For predicting T_{opt} by T_{best}, predictions that would result, for the instances considered, are given in Table 6.8. The χ^2 goodness-of-fit value obtained is 62.4. With a critical χ^2_9 value of 19.9,

using T_{best} does not give a satisfactory explanation for the observed variation in T_{opt}.

Consider comparing the performance of fixed-temperature simulated annealing against Aarts' algorithm for the GPP instances considered. Again (see Table 6.9), we see a tendency for fixed-temperature simulated annealing to outperform the fast cooling schedule for smaller sized problems, while its relative performance deteriorates as the size of problems increases.

For the GPP, the performance of fixed-temperature simulated annealing shows to outperform fast cooling for problem instances with less than 500 vertices.

7. Concluding remarks. In using a fixed-temperature simulated annealing algorithm, we have investigated the importance of determining a suitable fixed temperature. We have seen that the temperature deemed most suitable depends on criteria used. Optimal fixed temperatures have been determined experimentally, and investigated also is the predictability of such a temperature.

A formula for fixed temperature prediction has been developed for the TSP. Such did not show to follow directly with other applications considered, although insight gained has also been seen to follow in the QAP, while not yielding satisfactory results.

We have seen that fixed-temperature simulated annealing tends to outperform the use of a fast cooling schedule, for TSP instances of size less than 150 cities, for QAP instances of size less than 50 facilities, and for GPP instances of size less than 500 vertices.

REFERENCES

[1] E. H. L. Aarts and P. J. M. van Laarhoven, *Statistical cooling: A general approach to combinatorial optimization problems*, Philips J. Res., 40 (1985), pp. 193–226.
[2] E. H. L. Aarts and P. J. M. van Laarhoven, *Simulated Annealing: Theory and Applications*, Reidel, Dordrecht, the Netherlands, 1987.
[3] E. H. L. Aarts, J. H. M. Korst, and P. J. M. van Laarhoven, *A quantitative analysis of the simulated annealing algorithm: A case study for the traveling salesman problem*, J. Statist. Phys., 50 (1988), pp. 187–206.
[4] R. E. Burkard, S. E. Karisch, and F. Rendl, *QAPLIB—A Quadratic Assignment Problem Library*, available online from http://fmatbhp1.tu-graz.ac.at/~karisch/qaplib/
[5] V. Černý, *Thermodynamical approach to travelling salesman problem: An efficient simulation algorithm*, J. Optim. Theory Appl., 45 (1985), pp. 41–51.
[6] H. Cohn and M. Fielding, *Simulated annealing: Searching for an optimal temperature schedule*, SIAM J. Optim., 9 (1999), pp. 779–802.
[7] D. T. Connolly, *An improved annealing scheme for the QAP*, European J. Oper. Res., 46 (1990), pp. 93–100.
[8] M. Grötschel, *Polyedrische Charakterisierungen Kombinatorischer Optimie-rungsprobleme*, Hain, Meisenheim am Glan, 1977.
[9] M. Grötschel and O. Holland, *Solution of large-scale symmetric traveling salesman problems*, Math. Programming, 51 (1991), pp. 141–202.
[10] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, *Optimization by simulated annealing: An experimental evaluation; Part* 1, *Graph Partitioning*, Oper. Res., 37 (1989), pp. 865–892.
[11] R. L. Karg and G. L. Thompson, *A heuristic approach to solving traveling salesman problems*, J. Management Sci., 10 (1964), pp. 225–248.
[12] S. Kirkpatrick, C. D. Gellat, and M. P. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.
[13] S. Kirkpatrick, *Optimization by simulated annealing: Quantitative studies*, J. Statist. Phys., 34 (1983), pp. 975–986.
[14] P. D. Krolak, W. Felts, and G. Marble, *A man-machine approach toward solving the traveling salesman problem*, Commun. ACM, 14 (1971), pp. 327–334.
[15] S. Lin and B. W. Kernighan, *An effective heuristic algorithm for the traveling salesman problem*, J. Oper. Res., 21 (1973), pp. 498–516.

[16] C. E. Nugent, R. F. Vollman, and J. Ruml, *An experimental comparison of techniques for the assignment of facilities to locations*, Oper. Res., 16 (1968), pp. 150–173.
[17] 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.
[18] G. Reinelt, *TSPLIB—A traveling salesman problem library*, ORSA J. Comp., 3 (1991), pp. 376–384. Also available from online http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html
[19] G. Reinelt, *TSPLIB*, available online from http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html
[20] J. Skorin-Kapov, *Tabu search applied to the quadratic assignment problem*, ORSA J. Comp., 2 (1990), pp. 33–45.
[21] M. R. Wilhelm and T. L. Ward, *Solving quadratic assignment problems by simulated annealing*, IIE Trans., 19 (1987), pp. 107–119.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 308–319

THE MULTIPLE SUBSET SUM PROBLEM*

ALBERTO CAPRARA†, HANS KELLERER‡, AND ULRICH PFERSCHY‡

Abstract. In the *multiple subset sum problem* (MSSP) items from a given ground set are selected and packed into a given number of identical bins such that the sum of the item weights in every bin does not exceed the bin capacity and the total sum of the weights of the items packed is as large as possible.

This problem is a relevant special case of the multiple knapsack problem, for which the existence of a *polynomial-time approximation scheme* (PTAS) is an important open question in the field of knapsack problems. One main result of the present paper is the construction of a PTAS for MSSP.

For the bottleneck case of the problem, where the minimum total weight contained in any bin is to be maximized, we describe a 2/3-approximation algorithm and show that this is the best possible approximation ratio. Moreover, PTASs are derived for the special cases in which either the number of bins or the number of different item weights is constant.

We finally show that, even for the case of only two bins, no fully PTAS exists for both versions of the problem.

Key words. multiple subset sum problem, approximation scheme, knapsack problem

AMS subject classifications. 90C27, 90C10, 90C59

PII. S1052623498348481

1. Introduction.

Knapsack problems are among the most widely studied problems in combinatorial optimization; see, e.g., the book by Martello and Toth [8] and the recent survey by Pisinger and Toth [9]. For many problems of the knapsack family, optimal and approximation algorithms have been studied in the literature. In particular, the status of approximability, i.e., the existence of (fully) polynomial-time approximation schemes (PTASs), is settled for most of these problems. A notable exception is the *multiple knapsack problem*, which is the generalization of the classical 0-1 *knapsack problem* in which m knapsacks with different capacities are used for packing the items. For this problem, the existence of a PTAS is one of the most important open questions in the area, whereas the best known polynomial-time approximation algorithm has a worst-case performance ratio of 1/2. This algorithm returns the better of the following two solutions: The first solution is computed by considering the m knapsacks in decreasing order of capacities and packing in each knapsack the largest unpacked item that fits into it, whereas the second solution is obtained by setting to 0 the fractional variables in the solution of the continuous relaxation of the natural integer linear programming formulation of the problem (where one forbids the assignment of items to knapsacks into which they do not fit).

The present contribution concerns the *multiple subset sum problem* (MSSP), which is a generalization of the classical *subset sum* problem considering m instead of one knapsack, and at the same time a relevant special case of the multiple knapsack problem, where profits are equal to weights and all knapsacks have the same capacity. Among our results is a PTAS for MSSP.

*Received by the editors November 23, 1998; accepted for publication (in revised form) May 22, 2000; published electronically September 27, 2000.

http://www.siam.org/journals/siopt/11-2/34848.html

†DEIS, University of Bologna, viale Risorgimento 2, I-40136 Bologna, Italy (acaprara@deis.unibo.it). This author was partially supported by CNR and MURST, Italy.

‡Department of Statistics and Operations Research, University of Graz, Universitätsstr. 15, A-8010 Graz, Austria (hans.kellerer@kfunigraz.ac.at, pferschy@kfunigraz.ac.at)

MSSP is formally defined as follows. We are given a set $N := \{1, \dots, n\}$ of *items*, each item i having a positive integer *weight* w_i, and a set $M := \{1, \dots, m\}$ of identical *bins* with positive integer *capacity* c. The objective is to select a subset of items of maximum total weight that can be packed in the bins. The problem can be formulated as the following integer linear program:

$$\text{maximize} \sum_{j \in M} \sum_{i \in N} w_i x_{ij} \tag{1}$$

$$\text{subject to} \sum_{i \in N} w_i x_{ij} \le c, \quad j \in M, \tag{2}$$

$$\sum_{j \in M} x_{ij} \le 1, \quad i \in N, \tag{3}$$

$$x_{ij} \in \{0, 1\}, \quad i \in N,\ j \in M. \tag{4}$$

Without loss of generality we assume $w_i \le c$ for all $i = 1, \dots, n$, and $n \ge m$; otherwise the problem is trivially solved. The *load* of a bin i will be denoted by ℓ_i and represents the overall weight of the items packed in the bin.

The problem is strongly $\mathcal{NP}$-hard as it is an optimization version of the strongly $\mathcal{NP}$-complete 3-*partitioning problem*; see [5]. Note also that an optimal algorithm for MSSP solves the decision versions of both the *bin packing problem* and the *multiprocessor scheduling problem* (also denoted by $P||C_{max}$) as defined in [5].

We will also consider the *bottleneck* version of MSSP, namely, B-MSSP, in which the objective is to maximize the minimum load of a bin. Formally, the problem reads

$$\text{maximize} \quad \min_{j \in M} \sum_{i \in N} w_i x_{ij} \tag{5}$$

subject to (2)–(4). Clearly, also this version of the problem is strongly $\mathcal{NP}$-hard by reduction from the 3-partitioning problem; see again [5].

Obviously, MSSP and B-MSSP deal with tasks which may occur in many practical applications, e.g., in logistics, cutting and packing, portfolio optimization, etc. In the following we will briefly describe a real-world problem from a company producing objects of marble, which was pointed out by Wirsching [10]. Every week a shipment of m marble slabs from a quarry is received by the company. These slabs have a uniform size and are much longer than they are wide. The company produces different products, which first have to be cut from the marble slabs and are then further processed. In particular, each product requires a piece with a specified length from a marble slab. Depending on current stock, the company prepares a list of products that it would be interested to produce. Out of this list, some products should be selected and cut from the slabs so that the total amount of wasted marble is minimized. Clearly, this problem can be formulated as an MSSP.

In this paper we consider polynomial-time approximation algorithms for MSSP and B-MSSP, analyzing their worst-case approximation guarantee. Given an instance of MSSP or B-MSSP, we will denote by z^* the optimal solution value and by z^H the value of the solution returned by a heuristic algorithm H. Similarly, for an item set $S \subset N$, z_S^* and z_S^H will denote the optimal and heuristic solution values, respectively, for the instance restricted to item set S. Consider a value $\varepsilon \in (0, 1)$. We say that H is a $(1 - \varepsilon)$-*approximation algorithm* if $z^H \ge (1 - \varepsilon) z^*$ for all instances.

Note that it is rather easy to derive a 1/2-approximation algorithm for MSSP and B-MSSP. For example, if the items are packed in decreasing order of weights in any bin in which they fit, either all items will be packed in the end, or the load of every bin will be at least $1/2\,c$.

A PTAS is an algorithm taking as input a problem instance and a value $\varepsilon \in (0,1)$, delivering a solution of value $z^H \geq (1-\varepsilon)z^*$, and running in time polynomial in the size of the encoded instance. If the running time is also polynomial in $\frac{1}{\varepsilon}$, the algorithm is called a *fully polynomial-time approximation scheme* (FPTAS). It is well known [5] that no FPTAS can exist for a strongly $\mathcal{NP}$-hard problem unless $\mathcal{P} = \mathcal{NP}$. Note that after completion of this paper Chekuri and Khanna [2] presented a PTAS for the multiple knapsack problem.

In section 2 we will show that MSSP has a PTAS. In the original technical report [1] we also present a nontrivial 3/4-approximation algorithm with $O(m^2+n)$ running time.

As to B-MSSP, in section 3 we illustrate an $O(n \log n)$-time 2/3-approximation algorithm and show that this is the best approximation achievable in polynomial time unless $\mathcal{P} = \mathcal{NP}$. For the special cases in which either the number of bins or the number of distinct item weights is constant, B-MSSP has PTASs, as proved in section 4. In that section we also show that, even if the number of the bins is 2, neither MSSP nor B-MSSP can have an FPTAS unless $\mathcal{P} = \mathcal{NP}$.

2. A PTAS for MSSP. In this section we show that MSSP has a PTAS, which is the best approximation scheme one could hope for, as the problem is $\mathcal{NP}$-hard in the strong sense. The scheme we propose uses the techniques of *small item elimination* and *item grouping*, which have already been used in the context of bin packing. Nevertheless, these two techniques are apparently not sufficient for the derivation of a scheme, since in MSSP some items may clearly remain unpacked in a solution, which is not the case for bin packing. In our PTAS we use in addition a technique of *item preprocessing*, aimed at selecting only a (small) fraction of the large items before applying item grouping. Below, we give a detailed description of the PTAS we propose, hereafter called H^ε.

Given the required accuracy $\varepsilon \in (0,1)$, define $\tilde{\varepsilon} := \varepsilon/3$, and partition the item set N into the set $L := \{i \in N : w_i > \tilde{\varepsilon}c\}$ of *large* items and $S := \{i \in N : w_i \leq \tilde{\varepsilon}c\}$ of *small* items. The next lemma shows that every polynomial-time $(1-\tilde{\varepsilon})$-approximation algorithm for MSSP restricted to large items can be extended to a polynomial-time approximation algorithm for general MSSP with the same worst-case guarantee.

LEMMA 1. *Any polynomial-time $(1-\tilde{\varepsilon})$-approximation algorithm for MSSP restricted to large items yields a polynomial-time $(1-\tilde{\varepsilon})$-approximation algorithm for general MSSP.*

Proof. Denote by H the $(1-\tilde{\varepsilon})$-approximation algorithm for MSSP restricted to large items. Consider an instance of MSSP, and apply algorithm H to the instance corresponding to the set L of large items. Let z_L^* denote the value of the optimal solution for this instance, and by z_L^H the value of the solution found by H. By assumption, $z_L^H \geq (1-\tilde{\varepsilon})z_L^*$. The construction of an approximate solution for all items is straightforward: After applying H, simply assign the small items to the bins in a greedy way, i.e., each small item is packed, if possible, into any bin in which it fits.

If there are items of S which were not packed by this procedure, all bins have load of more than $(1-\tilde{\varepsilon})c$ and we are finished. Hence assume that all small items are packed into the bins, let z^H denote the value of the heuristic solution obtained, and

define $s := \sum_{i \in S} w_i$. We have $z^* \leq z_L^* + s$, and therefore

$$z^H = z_L^H + s \geq (1-\tilde{\varepsilon}) z_L^* + s \geq (1-\tilde{\varepsilon})(z^* - s) + s \geq (1-\tilde{\varepsilon}) z^*. \qquad \square$$

Lemma 1 allows one to initially get rid of the small items, which are reconsidered only at the end of the algorithm. At the same time, one can consider only the large items when the performance of an algorithm is analyzed. This is the so-called *small item elimination* phase mentioned above. If $|L| \leq m$, then the instance associated with L is solved trivially, and we immediately get a PTAS. Therefore, in the remainder of the section we will assume $|L| > m$.

After having (temporarily) removed the small items, we perform the following *item preprocessing* phase, aimed at reducing the number of items from n to a value in $O(m)$. We partition item set L into subsets I_j, $j = 1, \ldots, \lceil \frac{1}{\tilde{\varepsilon}} \rceil - 1$, each containing the items whose weight is in $(j\tilde{\varepsilon}c, (j+1)\tilde{\varepsilon}c]$. Let

$$\sigma_j := \left\lceil \frac{1}{j\tilde{\varepsilon}} \right\rceil - 1.$$

If $|I_j| \leq 2m\sigma_j$, we select all items in I_j; otherwise we select only the $m\sigma_j$ largest and the $m\sigma_j$ smallest. We will show that by neglecting the items which are not selected the optimal solution value is not affected too much. Let R be the set of the items selected in this phase, noting that

$$r := |R| \leq \sum_{j=1}^{\lceil \frac{1}{\tilde{\varepsilon}} \rceil - 1} 2m\sigma_j = \sum_{j=1}^{\lceil \frac{1}{\tilde{\varepsilon}} \rceil - 1} 2m \left(\left\lceil \frac{1}{j\tilde{\varepsilon}} \right\rceil - 1 \right) \leq 2m \frac{1}{\tilde{\varepsilon}} \left(\ln \frac{1}{\tilde{\varepsilon}} + 1 \right), \tag{6}$$

so $r \in O(m)$ if $\tilde{\varepsilon}$ is fixed.

The following lemma allows us to consider only the items in R in our PTAS. Let z_L^* and z_R^* denote the optimal solution value of MSSP for the instances corresponding to L and R, respectively.

LEMMA 2. $z_R^* \geq (1-\tilde{\varepsilon}) z_L^*$.

Proof. Note that for $j = 1, \ldots, \lceil \frac{1}{\tilde{\varepsilon}} \rceil - 1$, $(\sigma_j + 1) j\tilde{\varepsilon}c \geq c$, i.e., each bin in every solution contains at most σ_j items in I_j, as the weight of each item in I_j is strictly larger than $j\tilde{\varepsilon}c$. Consider an optimal solution for instance L with a bin k with load ℓ_k containing some item $i \in I_j \setminus R$. Then there exists at least one item i_s among the $m\sigma_j$ smallest and one item i_b among the $m\sigma_j$ largest in I_j which are both unpacked in the solution. If exchanging i with i_b yields a feasible solution, perform this exchange, without decreasing ℓ_k. Otherwise, exchange i with i_s: As $\ell_k - w_i + w_{i_b} > c$, the new load will be $\ell_k - w_i + w_{i_s} = \ell_k - w_i + w_{i_b} - w_{i_b} + w_{i_s} > c - \tilde{\varepsilon}c$, as $w_{i_b} - w_{i_s} < \tilde{\varepsilon}c$. Applying this exchanges as long as items in $L \setminus R$ are packed clearly yields a solution for instance R whose value is at least $(1-\tilde{\varepsilon}) z_L^*$, since, after each exchange, the load of the bin involved is either not decreased or at least equal to $(1-\tilde{\varepsilon})c$. $\square$

For technical reasons we now distinguish between two cases. If $m \leq 3/\tilde{\varepsilon}^2$, then recalling (6) r is bounded by $\frac{6}{\tilde{\varepsilon}^3}(\ln \frac{1}{\tilde{\varepsilon}} + 1)$ which is a constant for fixed $\tilde{\varepsilon}$. Hence, there is only a constant number of feasible packings for item set R, and the optimal one can be found in constant time by complete enumeration. Considering Lemma 2, this straightforward approach yields a PTAS.

There remains the more difficult case $m > 3/\tilde{\varepsilon}^2$ to be considered. In this case we now perform *item grouping* on set R. To group the items of R into subsets we define

$$k := \lfloor m\tilde{\varepsilon}^2 \rfloor$$

and let p and q be such that $r = pk + q$ and $1 \le q \le k$. Note that $k \ge 3$. We next show that

$$p \le 3\frac{1}{\tilde{\varepsilon}^3}\left(\ln\frac{1}{\tilde{\varepsilon}} + 1\right).$$

Assume otherwise: Obviously, using the assumption $p > 3\frac{1}{\tilde{\varepsilon}^3}\left(\ln\frac{1}{\tilde{\varepsilon}} + 1\right)$ we would have from (6)

$$m\tilde{\varepsilon}^2 - 1 \le \left\lfloor m\tilde{\varepsilon}^2 \right\rfloor = k \le \frac{r}{p} < \frac{2m\frac{1}{\tilde{\varepsilon}}\left(\ln\frac{1}{\tilde{\varepsilon}} + 1\right)}{3\frac{1}{\tilde{\varepsilon}^3}\left(\ln\frac{1}{\tilde{\varepsilon}} + 1\right)} = \frac{2}{3}\, m\tilde{\varepsilon}^2.$$

This is a contradiction for $m\tilde{\varepsilon}^2 > 3$. Hence for fixed $\tilde{\varepsilon}$, p is bounded by a constant.

Denote by $1, \dots, r$ the items in R and assume they are sorted according to increasing weights, i.e., $w_1 \le w_2 \le \cdots \le w_r$. Partition R into the $p+1$ subsets $R_i := \{ik+1, \dots, (i+1)k\}$, $i = 0, \dots, p-1$, and $R_p := \{pk+1, \dots, r\}$. Define the MSSP instance $\mathcal{I}$ with r items where for every $i = 0, \dots, p-1$ there are k items of equal weight $v_i := w_{(i+1)k}$, and q items have weight $v_p := w_r$. This instance can be solved to optimality in polynomial time for fixed $\tilde{\varepsilon}$ as the number of distinct item weights is $p + 1 = O(1)$, as follows.

We call *feasible bin type* a vector t with $p+1$ nonnegative integer entries $t_0, \dots, t_p$, such that $t_i \le k$ for $i = 0, \dots, p-1$, $t_p \le q$, and $\sum_{i=0}^{p} t_i v_i \le c$. Let $t^{(1)}, t^{(2)}, \dots, t^{(f)}$ be an enumeration of all feasible bin types and let $t^{(j)} = (t_0^{(j)}, t_1^{(j)}, \dots, t_p^{(j)})$ denote the jth vector in this enumeration. Moreover, let $\lambda_j := \sum_{i=0}^{p} t_i^{(j)} v_i$ denote the sum of item weights which corresponds to feasible bin type $t^{(j)}$. Note that the capacity constraint implies $t_i \le \left\lceil \frac{1}{\tilde{\varepsilon}} \right\rceil - 1$ for each entry i of a feasible bin type, which yields

$$f \le \left(\left\lceil \frac{1}{\tilde{\varepsilon}} \right\rceil - 1\right)^{p+1}.$$

Hence, f is a constant for $\tilde{\varepsilon}$ fixed. This would be enough to show that instance $\mathcal{I}$ can be solved in polynomial time. On the other hand, the (standard) considerations below show that this instance $\mathcal{I}$ can be solved in $O(m)$ time.

Consider the following integer linear program, which can be used to solve the MSSP instance defined above.

$$\text{maximize} \quad \sum_{j=1}^{f} \lambda_j x_j \tag{7}$$

$$\text{subject to} \quad \sum_{j=1}^{f} t_i^{(j)} x_j \le k, \quad i = 0, \dots, p-1, \tag{8}$$

$$\sum_{j=1}^{f} t_p^{(j)} x_j \le q, \tag{9}$$

$$\sum_{j=1}^{f} x_j \le m, \tag{10}$$

$$x_j \ge 0 \text{ integer}, \quad j = 1, \dots, f. \tag{11}$$

As the number f of variables is constant if $\tilde{\varepsilon}$ is fixed, this integer linear program can be solved in time polynomial in the number of constraints by applying Lenstra's algorithm [7]. Note that the number of constraints is also constant, whereas the maximum

size of a coefficient is in $O(m)$. As the running time of Lenstra's algorithm is polynomial in the number of constraints and in the logarithm of the size of the coefficients, we have that the overall running time of this algorithm is in $O(m)$, although it is doubly exponential in $\frac{1}{\tilde{\varepsilon}}$. We also note that the analogous approaches for bin packing do not have to insist in solving to optimality the counterpart of the integer linear program above but can resort to the solution of the associated linear programming relaxation, requiring a time which is only singly exponential in $\frac{1}{\tilde{\varepsilon}}$. Apparently, the same does not hold in this case, i.e., we have to stick to the integrality condition for the variables.

The solution obtained for instance $\mathcal{I}$ is converted into a solution for item set R (and hence item set L) by replacing, say, the s_i items of weight v_i in the solution, $i = 0, \ldots, p-1$, by items $(i+1)k, (i+1)k-1, \ldots, (i+1)k-s_i+1$ and the s_p items of weight v_p by items $r, r-1, \ldots, r-s_p+1$. The lemma below evaluates the quality of the solution obtained, whose value is denoted by z_L^H.

LEMMA 3. $z_L^H \geq (1-2\tilde{\varepsilon})z_R^*$.

Proof. Let t_1 denote the value of the optimal solution of the MSSP instance $\mathcal{I}$ with a fixed number of distinct item weights. The proof is divided into two parts. In the first part we show that z_L^H is not too far from t_1; in the second we prove that t_1 is close to z_R^*.

Let s be the cardinality of this solution, and let $y_1, \ldots, y_s$ be the weights of the items packed, in increasing order. Analogously, let $x_1, \ldots, x_s$ be the weights of the items packed by the heuristic solution for item set R, again in increasing order. For notational convenience, let $v_{-1} := 0$ throughout the proof.

Observe that $x_{j+k} \geq y_j$ for $j = 1, \ldots, s-k$. This ensures that $t_1 - z_L^H$ is not larger than k times the largest weight in the instance. Hence,

$$t_1 - z_L^H \leq kv_p \leq kc \leq m\tilde{\varepsilon}^2 c \leq \tilde{\varepsilon} z_R^*,$$

as $z_R^* \geq m\tilde{\varepsilon}c$ because the assumption that $m \leq |L|$ implies $m \leq |R|$, and hence at least m large items are packed in the optimal solution. This concludes the first part of the proof.

Now consider the optimal solution for instance R, and let r_i^* be the number of items in R_i packed by this solution, $i = 0, \ldots, p$. Furthermore, let t_2 be the value of the optimal solution of the instance in which there are exactly r_i^* items of weight v_{i-1} for $i = 0, \ldots, p$. Observing that the weight of each item in R_i is not smaller than v_{i-1} for $i = 0, \ldots, p$, it follows that in this latter solution all the items are packed. Moreover, by definition, $t_1 \geq t_2$, as t_1 is the solution of an instance defined by a wider item set.

Let $r := \sum_{i=0}^{p} r_i$, and, as above, $y_1, \ldots, y_r$ and $x_1, \ldots, x_r$ denote the weights (in decreasing order) of the items packed by the solutions of value t_2 and z_R^*, respectively. One has $y_{j+k} \geq x_j$ for $j = 1, \ldots, r-k$, as $r_i^* \leq k$ for $i = 0, \ldots, p$. Hence, by the same considerations as above, the relation

$$z_R^* - t_2 \leq \tilde{\varepsilon} z_R^*$$

holds. Therefore we have

$$z_L^H \geq t_1 - \tilde{\varepsilon} z_R^* \geq t_2 - \tilde{\varepsilon} z_R^* \geq (1-\tilde{\varepsilon})z_R^* - \tilde{\varepsilon} z_R^* = (1-2\tilde{\varepsilon})z_R^*,$$

and the proof is complete. □

Finally, the solution obtained for item set L is completed by adding small items in a greedy way. We summarize the various steps of Algorithm H^{ε} in Figure 1.

ALGORITHM H^ε.

Initialization:
Given the required accuracy ε, compute $\tilde{\varepsilon}$ and partition the item set into sets S and L.
If $|L| \leq m$, then pack each item in L into an empty bin and **goto** Phase 3
Phase 1: *Apply item preprocessing to set L obtaining set R*
Phase 2:
If $m \leq 3/\tilde{\varepsilon}^2$, then
    Compute the optimal solution for item set R by complete enumeration
Else
    Apply item grouping to set R, solve to optimality the instance obtained (with a fixed number of distinct item weights), and define a feasible solution for item set R from the optimal solution obtained
Phase 3: *Pack the items in S in a greedy way*

FIG. 1. *Outline of Algorithm H^ε.*

THEOREM 4. *Algorithm H^ε is a PTAS for MSSP running in $O(n+m)$ time.*

Proof. The approximation guarantee follows from Lemmas 1–3 and the definition of $\tilde{\varepsilon}$. In particular, for the case $m > 3/\tilde{\varepsilon}^2$, after Phase 2 the solution for item set L has value $z^H = z_L^H$. From Lemmas 2 and 3,

$$z_L^H \geq (1-2\tilde{\varepsilon})z_R^* \geq (1-2\tilde{\varepsilon})(1-\tilde{\varepsilon})z_L^*.$$

Hence, $z_L^H \geq (1-\varepsilon)z_L^*$ as long as

$$(1-2\tilde{\varepsilon})(1-\tilde{\varepsilon}) \geq (1-\varepsilon),$$

which is satisfied by the choice $\tilde{\varepsilon} := \varepsilon/3$. Finally, as $\tilde{\varepsilon} \leq \varepsilon$, by Lemma 1 the $(1-\varepsilon)$ approximation for item set L carries over to the overall instance after Phase 3.

The running time is polynomial; in particular the optimal solution for the items with rounded weights can be solved in $O(m)$ time for ε fixed, as described above. By using standard techniques, in particular computing the k-largest element of a set in linear time, it is easy to verify that the running time of H^ε is *linear* in n and m for ε fixed. ∎

3. Approximation algorithms for B-MSSP. Whereas for MSSP one can find a solution which is arbitrarily close to the optimum in polynomial time, as shown in section 2, for the case of B-MSSP it is easy to see the following.

THEOREM 5. *There does not exist any polynomial-time $(2/3+\delta)$-approximation algorithm for B-MSSP for any $\delta > 0$ unless $\mathcal{P} = \mathcal{NP}$.*

Proof. Consider the following well-known NP-complete problem, called 3-*partition* (3-PART); see [5]. One is given $3p+1$ numbers $a_1, \ldots, a_{3p}, b$ such that $\sum_{i=1}^{3p} a_i = pb$ and $b/4 < a_i \leq b/2$ for $i = 1, \ldots, 3p$. The objective is to partition $\{1, \ldots, 3p\}$ into sets $S_1, \ldots, S_p$ such that $|S_j| = 3$ and $\sum_{i \in S_j} a_i = b$ for $j = 1, \ldots, p$. Given an instance of 3-PART, we can define a B-MSSP instance with $3p$ items and p bins, with item weights $a_i + \gamma$, $i = 1, \ldots, 3p$, and bin capacity $3\gamma + b$. If $\gamma > 3\overline{a}$, with $\overline{a} := \max_{i=1}^{3p} a_i$, at most three items can be packed into each bin. Then, if the original 3-PART instance has a solution, the optimal value of the B-MSSP instance constructed is $3\gamma + b$; otherwise its optimal value is at most $2\gamma + 2\overline{a} \leq 2\gamma + b$. Hence, assuming the existence of a polynomial-time $(2/3+\delta)$-approximation algorithm for B-MSSP, by taking γ sufficiently large we get a polynomial-time algorithm for 3-PART. ∎

ALGORITHM $H^{\frac{2}{3}}$.

Initialization:
Partition the item set N into L, N_2, and S computing ℓ, n_2, s
Pack each item in L into an empty bin, $m_1 := \ell$

Phase 1: *Pack the items in* N_2 If $n_2 \leq m - m_1$, pack each item in N_2 into an empty bin and **goto** Phase 2
$max_2 := n_2 - (m - m_1)$ (*upper bound on the number of pairs*)
Let T contain the at most max_2 smallest items in N_2
Compute the maximum number m_2 of pairs obtainable from items in T and pack each pair into an empty bin
Pack each of the $m - m_1 - m_2$ largest items in N_2 into an empty bin

Phase 2: *Pack the small items* While $S \neq \emptyset$ do
Let i be the largest item in S
Let b be the bin with smallest load ℓ_b
If $\ell_b \geq 2/3\, c$, **stop**
Pack item i in bin b
End while

FIG. 2. *Algorithm* $H^{\frac{2}{3}}$.

We now illustrate a simple 2/3-approximation algorithm for B-MSSP called $H^{\frac{2}{3}}$. We partition the item set N into the set $L := \{i | w_i \geq 2/3\, c\}$ of *large* items, the set $N_2 := \{i | c/3 < w_i < 2/3\, c\}$, and the set $S := \{i | w_i \leq c/3\}$ of *small* items. Let $\ell := |L|$, $n_2 := |N_2|$, and $s := |S|$.

We start with an informal description of the algorithm. First of all, the items in L are packed into $m_1 := \ell$ bins. For the remaining $m - m_1$ bins, the main principle is to guarantee that as many bins as possible contain at least one item in N_2. Subject to this constraint, the algorithm packs as many pairs of items in N_2 as possible into bins. Accordingly, if $n_2 \leq m - m_1$, then all the items in N_2 are packed into separate bins. Otherwise, the maximum number of pairs that one would pack is $n_2 - (m - m_1)$. The algorithm computes the number of pairs, say, m_2, that can be obtained from the $2(n_2 - (m - m_1))$ smallest items in N_2. Then the m_2 pairs obtained, as well as the $m - m_1 - m_2$ largest items in N_2, are packed into separate bins. After having packed the large items, the small ones are considered in decreasing order of weight and each item is packed into the bin with smallest weight. This is the well-known *longest processing time* (LPT) rule used in scheduling problems with identical parallel machines. We stop as soon as each bin has a load not smaller than $2/3\, c$.

In Figure 2 we give a pseudocode description of our algorithm. In particular, the computation of the maximum number of pairs for a given item set T can be performed in $O(|T| \log |T|)$ time by standard techniques and will not be described in detail.

The following result was proved in [3] in the context of assigning a set of jobs to m identical processors in order to maximize the earliest processor completion time. We restate it in terms of B-MSSP.

LEMMA 6. *Suppose* $c = \infty$. *Then, by assigning the items in decreasing order of weights, each to the bin with smallest load, the B-MSSP solution value obtained is at least* $(3m - 1)/(4m - 2)$ *times the optimal one.* □

Using this result we can prove that the approximation guarantee of Algorithm $H^{\frac{2}{3}}$ is the best possible (unless $\mathcal{P} = \mathcal{NP}$).

THEOREM 7. *Algorithm $H^{\frac{2}{3}}$ is a 2/3-approximation algorithm for B-MSSP.*

Proof. If $\ell_i \geq 2/3\, c$ for each bin i in the $H^{\frac{2}{3}}$ solution, the claim is clearly true. We will assume that this is not the case, which implies in particular that all the items in S are packed by $H^{\frac{2}{3}}$. We will also assume $L = \emptyset$. Indeed, if $L \neq \emptyset$, then for the instance defined by item set $N \setminus L$ and with $m - \ell$ bins the optimal solution is not worse than z^* and the value of the solution returned by $H^{\frac{2}{3}}$ is unchanged.

Let z^H denote the value of the $H^{\frac{2}{3}}$ solution, R be the set of bins in this solution containing at most one item in N_2, and $r := |R|$. By the assumption above and the structure of Phase 2, all the items in S are packed in some bin in R. Let q be the number of items in N_2 packed in the bins in R, and denote by $x_1, \ldots, x_q$ the weights of these items, in decreasing order.

We now claim that there exists a set R^* of r bins in the optimal solution where $p \leq q$ items in N_2 are packed, and such that the weights of these items, say, $y_1, \ldots, y_p$, in decreasing order, satisfy $y_i \leq x_i$ for $i = 1, \ldots, p$. If the number of pairs of items in N_2 packed by the optimal solution is not larger than m_2, the number of pairs packed by $H^{\frac{2}{3}}$, then the claim follows immediately as the largest items in N_2 are packed alone in a bin by $H^{\frac{2}{3}}$. Otherwise, if the optimal solution contains more pairs of items in N_2, say, $m_2^* > m_2$, then it also contains at least $m_2^* - m_2$ more bins with no item in N_2 packed, as the number of pairs in $H^{\frac{2}{3}}$ is the maximum that guarantees that as many bins as possible contain at least one item in N_2.

Now, consider the B-MSSP instance in which the capacity is equal to ∞, the number of bins is r, and the item set is given by the items packed by $H^{\frac{2}{3}}$ in the bins in R, whose weights, in decreasing order, we denote by $x_1, \ldots, x_t$. Let $\tilde{z}$ denote the optimal solution value for this instance. Clearly, $\tilde{z} \geq z^*$, as the items packed by the bins in R^* by the optimal solution, whose weights are (in decreasing order) $y_1, \ldots, y_s$, satisfy $s \leq t$ and $y_i \leq x_i$ for $i = 1, \ldots, s$. Moreover, the packing of the bins in R by $H^{\frac{2}{3}}$ yields a solution $z^H \geq (3r-1)/(4r-2)\, \tilde{z} \geq 3/4\, z^*$ by Lemma 6, which concludes the proof. ∎

THEOREM 8. *Algorithm $H^{\frac{2}{3}}$ runs in $O(n \log n)$ time and requires $O(n)$ space.*

Proof. The space complexity is clearly $O(n)$. As to the time complexity, computing the pairs requires sorting $O(m)$ smallest items in N_2, i.e., $O(m \log m)$ time. Moreover, Phase 2 requires sorting the $O(n)$ items in S, in $O(n \log n)$ time. As to the determination of the bin with smallest load in every iteration, one may store the bin loads in a heap data structure. The initialization of this data structure requires $O(m \log m)$ time, whereas each of the $O(n)$ iterations of the loop requires $O(\log m)$ time. All the remaining operations can be performed in $O(n)$ time. ∎

4. PTASs for special cases of B-MSSP. In practical applications the number of distinct item weights may happen to be small, and hence may be considered as a constant. Also the number of bins m may be bounded by a constant. In this section we will show that in both cases there exists a PTAS for B-MSSP. As the PTAS presented in section 2, we expect the PTASs below to be rather impractical. However, their existence gives motivation for finding practical approximation algorithms with an approximation ratio better than the 2/3 one presented in the previous section for general B-MSSP.

Given $\varepsilon \in (0, 1)$ and $\tilde{c} \leq c$, partition the item set N into the set $L_{\tilde{c}} := \{i \in N : w_i > \varepsilon \tilde{c}\}$ of $\tilde{c}$-*large* items and the set $S_{\tilde{c}} := \{i \in N : w_i \leq \varepsilon \tilde{c}\}$ of $\tilde{c}$-*small* items. If $\tilde{c} = c$, then $L := L_c$ is the set of *large* items and $S := S_c$ is the set of *small* items. Let $s_{\tilde{c}}$ denote the total sum of the $\tilde{c}$-small item weights and set $s := s_c$.

For a given assignment of $\tilde{c}$-large items to the m bins the *$\tilde{c}$-residual capacity*, $\tilde{c}_{RC}$ for short, is defined as

$$\tilde{c}_{RC} := \sum_{i=1}^{m} \max\{0, \tilde{c} - \lambda_i\},$$

where λ_i is the overall weight of the $\tilde{c}$-large items packed in bin i.

The following lemma is the counterpart of Lemma 1 in section 2.

LEMMA 9. *Suppose a procedure $P_L(\tilde{c})$ working on the $\tilde{c}$-large items is available, which finds in polynomial time, for each $\tilde{c} \le z^*$, an assignment of a subset of the $\tilde{c}$-large items to the m bins such that $\tilde{c}_{RC} \le s_{\tilde{c}}$. Then, this procedure yields a polynomial-time $(1-\varepsilon)$-approximation algorithm for B-MSSP.*

Proof. Let $P(\tilde{c})$ be a procedure that first applies procedure $P_L(\tilde{c})$ to the $\tilde{c}$-large items, and then completes the solution by packing in arbitrary order the $\tilde{c}$-small items into bins with load smaller than $(1-\varepsilon)\tilde{c}$. In the final solution, no bin containing small items will have load greater than $\tilde{c}$. Moreover, as $\tilde{c}_{RC} \le s_{\tilde{c}}$, all bins will have load at least $(1-\varepsilon)\tilde{c}$. Accordingly, for each $\tilde{c} \in [0, c]$, if there is a solution of B-MSSP with value at least $\tilde{c}$, then $P(\tilde{c})$ finds a solution of value at least $(1-\varepsilon)\tilde{c}$. This immediately implies that applying binary search on $\tilde{c}$ between 0 and c, calling $P(\tilde{c})$ at each iteration, yields a polynomial-time $(1-\varepsilon)$-approximation algorithm for B-MSSP. □

Below we show that the PTASs for the special cases mentioned above follow from this lemma in a straightforward way.

We first describe a PTAS, called H_1^ε, for the case in which the number m of bins is constant. Algorithm H_1^ε performs binary search on $\tilde{c}$ between 0 and c. For each value of $\tilde{c}$ note that in order to derive a procedure $P_L(\tilde{c})$ as in Lemma 9, one is not interested in packing in a bin a subset of items P such that $\sum_{j \in P \setminus \{i\}} w_j \ge \tilde{c}$ for some $i \in P$. Hence, we define a *$\tilde{c}$-feasible assignment* to a bin b corresponding to packing in b either no item, a single item, or an item subset P, $|P| \ge 2$, such that $\sum_{j \in P} w_j \le \min\{2\tilde{c}, c\}$. Note that each $\tilde{c}$-feasible assignment packs at most $\lfloor \frac{2}{\varepsilon} \rfloor$ $\tilde{c}$-large items. Algorithm H_1^ε considers all $O(n^{\frac{2}{\varepsilon}m})$ different possibilities to assign the $\tilde{c}$-large items to the m bins yielding $\tilde{c}$-feasible assignments. If m is constant, this is a polynomial number of possibilities. Moreover, if $\tilde{c} \le z^*$, at least one of these will satisfy $\tilde{c}_{RC} \le s_{\tilde{c}}$. The solution is then completed as illustrated in the proof of Lemma 9. The above discussion proves the following

THEOREM 10. *If the number of bins m is constant, Algorithm H_1^ε is a PTAS for B-MSSP.* □

We will now show a PTAS, called H_2^ε, for the case in which the number of distinct item weights is fixed. As is customary in this case, we formulate B-MSSP for $\tilde{c}$-large items as an integer linear program with a constant number of integer variables.

Let $\tilde{c} \le c$ be given, and suppose we have p distinct weights $w_1, \ldots, w_p$ with $w_i > \varepsilon\tilde{c}$ for $i = 1, \ldots, p$. Let n_i be the number of items with weight w_i, $1 \le i \le p$; hence $\sum_{i=1}^{p} n_i \le n$. We call *$\tilde{c}$-feasible bin type* a vector t with p nonnegative integer entries $t_1, \ldots, t_p$, such that either t has only one nonzero entry $t_i = 1$ with $w_i \ge \tilde{c}$ or

$$\sum_{i=1}^{p} t_i w_i \le \min\{2\tilde{c}, c\}$$

holds. Let $t^{(1)}, t^{(2)}, \ldots, t^{(f)}$ be an enumeration of all $\tilde{c}$-feasible bin types and let $t^{(j)} := (t_1^{(j)}, t_2^{(j)} \ldots, t_p^{(j)})$ denote the jth vector in this enumeration. Moreover, let

$\lambda_j := \sum_{i=1}^{p} t_i^{(j)} w_i$ denote the sum of item weights which corresponds to feasible bin type $t^{(j)}$, and define $\delta_j := \max\{0, \tilde{c} - \lambda_j\}$. As in section 2, f is in $O((p + \frac{2}{\varepsilon})^p)$, as $\lfloor \frac{2}{\varepsilon} \rfloor$ is an upper bound on the sum of the entries of a $\tilde{c}$-feasible bin type.

The following system of linear inequalities in integer variables checks whether there is an assignment of the $\tilde{c}$-large items such that $\tilde{c}_{RC} \leq s_{\tilde{c}}$. It has a solution for each $\tilde{c} \leq z^*$.

$$\sum_{j=1}^{f} t_i^{(j)} x_j \leq n_i, \quad i = 1, \ldots, p, \tag{12}$$

$$\sum_{j=1}^{f} x_j \leq m, \tag{13}$$

$$\sum_{j=1}^{f} \delta_j x_j \leq s_{\tilde{c}}, \tag{14}$$

$$x_j \geq 0 \text{ integer}, \quad j = 1, \ldots, q. \tag{15}$$

As in section 2 the number of variables and constraints is constant, and hence a solution, if any, to this system can be found in polynomial time by applying Lenstra's algorithm [7].

Accordingly, H_2^{ε} performs binary search on $\tilde{c}$, solving to optimality the B-MSSP instance restricted to the $\tilde{c}$-large items, and then adds small items in a greedy way. The theorem below follows immediately from Lemma 9.

THEOREM 11. *If the number of distinct item weights is constant, Algorithm H_2^{ε} is a PTAS for B-MSSP.* □

We conclude this section with a negative result concerning the efficient approximability of MSSP and B-MSSP when m is fixed.

THEOREM 12. *If the number of bins m is equal to 2, neither MSSP nor B-MSSP admits an FPTAS unless $\mathcal{P} = \mathcal{NP}$.*

Proof. Consider the following problem, called the *equal cardinality partition* (E-PART), which is known to be NP-complete (cf. problem [SP12] in [5]). Given a finite set I of even cardinality n containing positive integer numbers a_i, $1 \leq i \leq n$, with $\sum_{i=1}^{n} a_i = 2A$, determine if there exists a partitioning of I into two subsets I_1, I_2 such that $|I_1| = |I_2| = n/2$ and $\sum_{i \in I_1} a_i = \sum_{i \in I_2} a_i = A$.

Given an instance of E-PART, define an instance of MSSP with two bins, n items, the ith having weight $w_i = 2A + a_i$ $(i = 1, \ldots, n)$, and bin capacity $c = A(n + 1)$. Clearly, E-PART has a solution if and only if the optimal solutions of MSSP and B-MSSP pack all items in the bins, namely, $n/2$ per bin, and the overall load of the bins is $2A(n + 1)$. Observe that $n/2$ is the maximum number of items that can be packed in a bin. Any solution of MSSP or B-MSSP that is not optimal leaves at least one item unpacked, and hence the overall load of the bins is at most $2An$, and the smallest load of a bin at most An. Hence, the approximation ratio of any nonoptimal solution is at most

$$\frac{2An}{2A(n+1)} = 1 - \frac{1}{n+1}$$

for MSSP and

$$\frac{An}{A(n+1)} = 1 - \frac{1}{n+1}$$

for B-MSSP. If an FPTAS existed for either problem, we would get a polynomial $(1-\varepsilon)$-approximation algorithm also for $\varepsilon < \frac{1}{n+1}$, which would yield an optimal solution for the instance above whenever the original E-PART instance has a solution. This is clearly impossible unless $\mathcal{P} = \mathcal{NP}$. □

Acknowledgments. We would like to thank Günther Wirsching from the Katholische Universität Eichstätt for pointing out the real-world application described in section 1 and two anonymous referees for their helpful comments.

REFERENCES

[1] A. Caprara, H. Kellerer, and U. Pferschy, *The Multiple Subset Sum Problem*, Technical Report 12/1998, Faculty of Economics, University of Graz, Graz, Austria, 1998.

[2] C. Chekuri and S. Khanna, *On multi-dimensional packing problems*, in Proceedings of the Tenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'99), 1999.

[3] J. Csirik, H. Kellerer, and G.J. Woeginger, *The exact LPT method for maximizing the minimum completion time*, Oper. Res. Lett., 11 (1992), pp. 281–287.

[4] W. Fernandez de la Vega and G.S. Lueker, *Bin packing can be solved within* $1+\varepsilon$ *in linear time*, Combinatorica, 1 (1981), pp. 349–355.

[5] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, San Francisco, 1979.

[6] N. Karmarkar and R.M. Karp, *An efficient approximation scheme for the one-dimensional bin packing problem*, in Proceedings of the 23rd Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, Los Alamitos, CA, 1982, pp. 312–320.

[7] H.W. Lenstra, *Integer programming with a fixed number of variables*, Math. Oper. Res., 8 (1983), pp. 538–548.

[8] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implementations*, John Wiley, Chichester, UK, 1990.

[9] D. Pisinger and P. Toth, *Knapsack problems*, in Handbook of Combinatorial Optimization, D.Z. Du and P. Pardalos, eds., Kluwer Academic Publishers, Norwell, MA, 1998, pp. 1–89.

[10] G. Wirsching, *personal communication,* Katholische Universität Eichstätt, 1998.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 320–340

A STRONGLY POLYNOMIAL ROUNDING PROCEDURE YIELDING A MAXIMALLY COMPLEMENTARY SOLUTION FOR $P_*(\kappa)$ LINEAR COMPLEMENTARITY PROBLEMS*

TIBOR ILLÉS†, JIMING PENG‡, CORNELIS ROOS‡, AND TAMÁS TERLAKY§

Abstract. We deal with linear complementarity problems (LCPs) with $P_*(\kappa)$ matrices. First we establish the convergence rate of the complementary variables along the central path. The central path is parameterized by the barrier parameter μ, as usual. Our elementary proof reproduces the known result that the variables on or close to the central path fall apart in three classes in which these variables are $\mathcal{O}(1)$, $\mathcal{O}(\mu)$, and $\mathcal{O}(\sqrt{\mu})$, respectively. The constants hidden in these bounds are expressed in or bounded by the input data. All this is preparation for our main result: a strongly polynomial rounding procedure. Given a point with sufficiently small complementarity gap and which is close enough to the central path, the rounding procedure produces a maximally complementary solution in at most $\mathcal{O}(n^3)$ arithmetic operations.

The result implies that interior point methods (IPMs) not only converge to a complementary solution of $P_*(\kappa)$ LCPs, but, when furnished with our rounding procedure, they can also produce a maximally complementary (exact) solution in polynomial time.

Key words. linear complementarity problems, $P_*(\kappa)$ matrices, error bounds on the size of the variables, optimal partition, maximally complementary solution, rounding procedure

AMS subject classification. 90C33

PII. S1052623498336590

1. Introduction.

In this paper we deal with a class of linear complementarity problems (LCPs),

$$(LCP) \qquad s(x) := Mx + q \geq 0, \quad x \geq 0, \quad xs(x) = 0, \tag{1.1}$$

where M is an $n \times n$ real matrix, $q \in \mathbb{R}^n$, and $xs(x)$ denotes the coordinatewise product of the vectors x and $s(x)$. We say that an algorithm solves (LCP) if either it produces a vector x satisfying the constraints of (LCP) or it provides a certificate that no such vector exist. In the first case we say that x solves (LCP).

The vector x is a *strictly complementary solution* of (LCP) if it solves (LCP) and $x + s(x) > 0$. Contrary to linear optimization (LO) [27], in general no strictly complementary solution exists for (LCP): there might exist pairs of complementary variables x_i and $s_i(x)$ that are both zero in all solutions of the (LCP). Complementary solutions with the maximal number of nonzero coordinates will be referred to as *maximally complementary solutions*. The existence of maximally complementary

*Received by the editors March 26, 1998; accepted for publication (in revised form) April 17, 2000; published electronically September 27, 2000. The last three authors were supported by the Dutch Organization for Scientific Research, NWO, through SWON-grant 613-304-200 for the project High Performance Methods for Mathematical Optimization.

http://www.siam.org/journals/siopt/11-2/33659.html

†Department of Operations Research, Eötvös Loránd University, Budapest, Hungary (illes@cs.elte.hu). This author kindly acknowledges a one-year research fellowship at Delft University of Technology. His research was supported by the Hungarian National Research Fund OTKA T 019492 and FKFP 0152/1999.

‡Faculty of Technical Mathematics and Informatics, Delft University of Technology, P.O. Box 5031, Mekelweg 4, 2628 CD, Delft, The Netherlands (j.peng@its.tudelft.nl, c.roos@its.tudelft.nl).

§Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada (terlaky@cas.mcmaster.ca). This author was partially supported by the Hungarian National Research Fund OTKA T 019492.

solutions follows from the convexity of the solution set, proved by Cottle, Pang, and Venkateswaran in [4]. Kojima et al. [18] under some additional assumptions showed that solutions on the central path converge to a maximally complementary solution of (LCP).

All known algorithms for solving (LCP) need some assumption on the matrix M. So do interior point methods (IPMs) as well. IPMs for solving (LCP) are widely studied in the last decade. A survey on recent results is written by Yoshise [36]. Kojima, Mizuno, and Yoshise [15] presented a polynomial time algorithm that produces an exact solution for LCPs where M is positive semidefinite. The same authors [16] established an $\mathcal{O}(\sqrt{n}L)$ iteration bound[1] for a *potential reduction* algorithm. Ji, Potra, and Huang [11] developed a polynomial, $\mathcal{O}(\sqrt{n}L)$ *predictor-corrector* method for positive semidefinite LCPs under the assumption that the sequence of iterates generated by their interior point algorithm converges to a strictly complementary solution. Later, Ye and Anstreicher [34] proved the same iteration bound, $\mathcal{O}(\sqrt{n}L)$ for predictor-corrector methods, removing the assumption given in [11]. In 1991, Kojima et al. [18] extended all the previously known results to the wider class of so-called $P_*(\kappa)$ LCPs and unified the theory of LCPs from the viewpoint of interior point methods. Jansen, Roos, and Terlaky [9] introduced a family of *primal-dual affine-scaling* algorithms for positive semidefinite LCPs. These results were recently extended to LCPs with $P_*(\kappa)$ matrices by Illés, Roos, and Terlaky [8]. The iteration bound of those algorithms are $\mathcal{O}((1+4\kappa)n\log x_0^T s(x_0)/\epsilon)$, where x_0 is the initial iterate and ϵ is the *complementarity gap* $x^T s(x)$ at termination.

Interior point methods need an interior feasible point to start with. Among others, Ji, Potra, and Sheng [12] studied the initialization problem and proposed a predictor-corrector method for solving the $P_*(\kappa)$ LCPs from infeasible starting points. Kojima, Mizuwo, and Yoshise [17] and Kojima et al. [18] gave a big-M construction that allows one to solve the problem in one phase.

The aim of this paper is twofold. First, we derive some bounds on the magnitude of the variables in the vicinity of the central path,[2] when the complementarity gap is small enough. Second, a strongly polynomial rounding procedure is presented that provides a maximally complementary (exact) solution from any interior point solution that is in a certain neighborhood of the central path and for which the complementarity gap is sufficiently small.

For deriving results on the magnitude of the variables in a given neighborhood of the central path we use some known results from the theory of error bounds for systems of linear inequalities [21]. The theory of error bounds goes back to the early fifties [7]; for recent developments we refer to the survey paper [25] and the references therein. For LCPs, a well-known local error bound is given by Robinson [26] which says that there exists a constant $\epsilon > 0$ and $\tau > 0$ such that

$$\operatorname{dist}(x, \Gamma^*) \leq \tau \| \min(x, s(x)) \| \tag{1.2}$$

for all x satisfying $\| \min(x, s(x)) \| \leq \epsilon$, where Γ^* denotes the solution set of (LCP) in $\mathbb{R}^n_+$, $\operatorname{dist}(x, \Gamma^*) = \min_{y \in \Gamma^*} \|y - x\|$ and the minimum $\min(x, s(x))$ is taken coordinatewise. By using the properties of the central path and some results on error bounds of Cook et al. [3] and Mangasarian and Shiau [21], we derive some bounds on these constants in terms of the input data if x is on or close to the central path. To the

[1] L is the binary input length of the problem [18].

[2] The central path is defined in the usual way. See section 2.

best of our knowledge, this is the first result yielding easily to calculate bounds for these constants in the study of LCPs.

The bounds on the magnitude of the variables along the central path depend on the dimension n of the problem, on the parameter κ and the barrier parameter μ that parameterizes the central path, and on two condition numbers σ_{LCP} and ν_{LCP} of (LCP). The condition number σ_{LCP} is closely related to that defined by Ye [35] and studied by Vavasis and Ye [31] for polyhedra with real number data and slightly modified by Roos, Terlaky, and Vial [27] for the case of LO problems. The second condition number, ν_{LCP}, will be introduced later. Other condition numbers for LCP are defined in [19, 32]. It will be shown in a quite elementary way that in a given neighborhood of the central path the variables fall apart in three classes and their magnitudes are $\mathcal{O}(1)$, $\mathcal{O}(\mu)$, and $\mathcal{O}(\sqrt{\mu})$, respectively, provided the parameter μ is sufficiently small.

The rounding procedure we describe for (LCP) resembles the one presented in the papers [33, 22] and in the book [27]. We show that IPMs with a rounding procedure terminate in a finite (polynomial) number of iterations and yield a maximally complementary solution.

There are some other methods [15, 19, 10, 24, 29] in the literature that generate an exact solution to (LCP) in $\mathcal{O}(n^3L)$ iterations, but those are different from ours. For example, Kojima, Mizuno, and Yoshise [15] in Appendix B of their paper presented a method which leads to a basic solution of the LCPs, thus not providing a maximally complementary solution. In [24], Monteiro and Wright considered the local convergence of IPMs for monotone LCP. By using the error bound results in [7, 21], they also obtained some estimates about the magnitude of the variables in the neighborhood of the central path. When the updated point is sufficiently close to the solution set, then an orthogonal projection is employed to get an exact solution. If the iterate is "close enough" to the solution set, the projected point is maximally complementary. Similar results were also reported by Ji and Potra [10]. However, the complexity of the algorithms in [10, 24] depends on some constants which they do not relate to the input data of the problem and thus, in the above-mentioned papers, there is no proven guarantee that these methods yield a maximally complementary solution in polynomial time.

Stoer, Wechs, and Mizuno in [29] modified their original algorithm to obtain an exact solution in the following way: using asymptotically correct estimates of the optimal partition of the variables they have introduced a linear system; the solution of this system will be an exact solution if and only if the estimation is good enough. They have no explicit criteria for the estimated partition to be good enough to get an exact solution.

An explicit bound on the number of necessary iterations to produce the optimal partition will be computed (Theorem 4.3) in our paper. Furthermore, we show that the optimal partition can be identified from a less accurate solution, while we need to have a solution with smaller duality gap when we want to compute an exact maximally complementary solution (Theorem 5.1). Finally, we state the complexity of the Dikin affine-scaling algorithm to produce an exact solution of the (LCP) (see Theorem 5.2). As we will see later, the bounds (i.e., the complexity estimate of computing a maximally complementary solution) depend only on the input data of the problem. We do not claim that the accuracy which is theoretically needed to start our rounding procedure is practically reachable. However, from a theoretical point of view our results improve all the previously known results related to rounding procedures. The

first time an explicit polynomial bound is computed on the necessary accuracy when a successful rounding procedure can be started. Similarly to the earlier results, for rounding only the solution of a linear system is needed. Furthermore, our bounds exhibit the difficulty of obtaining an exact solution for a sufficient (LCP), in general.

Our rounding procedure generates an exact maximally complementary solution, while the procedure presented in [15] produces a complementary basic solution, which, in general, is not maximally complementary. Having a maximally complementary solution, a complementary basic solution can be computed in strongly polynomial time by using the basis identification procedure presented by Berkelaar et al. [2]. However, no strongly polynomial algorithm is known to generate a maximally complementary solution from a complementary basic solution.

The paper is organized as follows. Some preliminary results are discussed in section 2. In section 3 the optimal partition is defined and the related concept of maximally complementary solutions. We introduce two condition numbers for (LCP) and derive local bounds on the magnitude of the variables on the central path. The main result in this section describes how the optimal partition can be determined if the barrier parameter μ is small enough. In section 4 we generalize the results of section 3 to points that are close to the central path, so-called approximate centers, and we show that the optimal partition can be identified from x if x belongs to a certain neighborhood of the central path and if $x^T s(x)$ is small enough; such a vector x can be obtained in polynomial time by any interior point method. Section 5 presents a strongly polynomial rounding procedure that yields a maximally complementary solution. Some concluding remarks close the paper in section 6.

Throughout, we shall use $\|\cdot\|_p$ $(p \in [1,\infty])$ to denote the p-norm on $\mathbb{R}^n$, with $\|\cdot\|$ denoting the Euclidean norm $\|\cdot\|_2$. E will denote the identity matrix, e will be used to denote the vector which has all its components equal to 1. Given an n-dimensional vector x, we denote by X the $n \times n$ diagonal matrix whose diagonal entries are the coordinates x_j of x. If $x, s \in \mathbb{R}^n$, then $x^T s$ denotes the dot product of the two vectors. Further, xs and x^α for $\alpha \in \mathbb{R}$ will denote the vectors resulting from coordinatewise operations. For any matrix $A \in \mathbb{R}^{m\times n}$, A_i, $A_{\cdot j}$ are the ith row and the jth column of A, respectively. Furthermore,

$$\pi(A) := \prod_{j=1}^{n} \|A_{\cdot j}\|.$$

For any index set $J \subset \{1, 2, \ldots, m\}$, $|J|$ denotes the cardinality J and $A_J \in \mathbb{R}^{|J|\times n}$ the submatrix of A whose rows are indexed by elements in J. Moreover, if $K \subset \{1, 2, \ldots, n\}$, $A_{JK} \in \mathbb{R}^{|J|\times|K|}$ denotes the submatrix of A_J whose columns are indexed by elements in K.

2. Preliminaries. For further use we first recall some well-known results and definitions. The reader may consult [5, 6, 18, 36] for proofs and details.

A matrix $M \in \mathbb{R}^{n\times n}$ is a $P_*(\kappa)$ matrix if

$$(1+4\kappa) \sum_{i\in I_+(x)} x_i[Mx]_i + \sum_{i\in I_-(x)} x_i[Mx]_i \geq 0 \quad \text{for all } x \in \mathbb{R}^n, \tag{2.1}$$

where

$$I_+(x) := \{1 \leq i \leq n \ : \ x_i[Mx]_i > 0\}, \quad I_-(x) := \{1 \leq i \leq n \ : \ x_i[Mx]_i < 0\},$$

and κ is a nonnegative real number. Note that the index sets $I_+(x)$ and $I_-(x)$ depend not only on x but on the matrix M as well. Equivalently (2.1) can also be stated as

$$x^T Mx \geq -4\kappa \sum_{i \in I_+(x)} x_i [Mx]_i.$$

The matrix M is a P_* matrix if it is a $P_*(\kappa)$ matrix for some nonnegative κ:

$$P_* = \bigcup_{\kappa \geq 0} P_*(\kappa).$$

One easily verifies that M is a $P_*(0)$ matrix if and only if M is positive semidefinite. Furthermore, if M is $P_*(\overline{\kappa})$ for some $\overline{\kappa} \geq 0$, then M is $P_*(\kappa)$ for all $\kappa \geq \overline{\kappa}$.

The class of *sufficient* matrices (SU) was introduced by Cottle, Pang, and Venkateswaran [4]. A matrix $M \in \mathbb{R}^{n \times n}$ is *column sufficient* if for all $x \in \mathbb{R}^n$,

$$X(Mx) \leq 0 \Rightarrow X(Mx) = 0$$

and is *row sufficient* if M^T is column sufficient. The matrix M is *sufficient* if it is both row and column sufficient. Recently, Väliaho [30] proved that $P_* = SU$.

The sets of feasible and positive feasible vectors are denoted, respectively, by

$$\Gamma = \{x \ : \ x \geq 0, \, s(x) \geq 0\},$$
$$\Gamma^0 = \{x \ : \ x > 0, \, s(x) > 0\},$$

and the set of solutions of (LCP) by

$$\Gamma^* = \{x \ : \ x \geq 0, \ s(x) \geq 0, \ xs(x) = 0\}.$$

It is known (cf. [18], Theorem 4.6) that if $M \in P_*$ and $\Gamma \neq \emptyset$, then $\Gamma^* \neq \emptyset$. Further, if $\Gamma^0 \neq \emptyset$, then Γ^* is compact;[3] moreover for every $\mu > 0$ there exists a unique $x \in \Gamma^0$ such that

$$xs(x) = \mu e.$$

In other words, assuming that Γ^0 is nonempty the *central path*

$$\mathcal{C} := \{x \in \Gamma^0 \ : \ xs(x) = \mu e \text{ for some } \mu > 0\}$$

exists. Kojima et al. [18] showed that the assumption $\Gamma^0 \neq \emptyset$ can be made without loss of generality. Hence we may assume that the central path $\mathcal{C}$ exists. The central path $\mathcal{C}$ is a one-dimensional smooth curve that leads to a solution of (LCP) when μ approaches 0.[4]

We insert here the following technical lemma that will be used at several places below.

LEMMA 2.1. *Let x be a solution of the equation $Dx = d$, where D is an integral and nonzero $m \times n$ matrix and d is an integral vector. If J denotes the support of x and the columns of $D_{.J}$ are linearly independent, then*

$$\frac{1}{\Delta(D)} \leq |x_j| \leq \Delta(D) \, \|d\|_1, \quad j \in J.$$

[3]The key observation for this is Lemma 4.5 of Kojima et al. [18].

[4]For further details we refer to Chapters 2 and 4 in [18]. See also [23, 28].

Here $\Delta(D)$ denotes the largest absolute value of the determinants of the square submatrices of D. The right inequality holds also if d is not integral.

Proof. For completeness we include a proof here. Let x be as in the lemma and let the index set K be such that D_{KJ} is a nonsingular square submatrix of D; such K exists because the columns in $D_{.J}$ are linearly independent. Now we have $D_{KJ}x_J = d_K$, and hence, by Cramer's rule,[5]

$$x_j = \frac{det\left(D_{KJ}^{(j)}\right)}{det(D_{KJ})} \quad \text{for all } j \in J, \tag{2.2}$$

where $D_{KJ}^{(j)}$ denotes the matrix arising when the jth column in D_{KJ} is replaced by d_K. Since the denominator in the above quotient is at least 1 we obtain

$$|x_j| \leq \det\left(D_{KJ}^{(j)}\right), \quad j \in J. \tag{2.3}$$

When we evaluate the last determinant to its jth column, while using that each square submatrix is also a square submatrix of D, the right-hand side inequality follows. For the left inequality we use that d is integral; since $x_j \neq 0$ this implies that the numerator in (2.2) is at least one. ∎

COROLLARY 2.2. *If D is integral and the columns of D are all nonzero, then, under the assumptions of Lemma* 2.1*,*

$$\frac{1}{\pi(D)} \leq |x_j| \leq \pi(D)\,\|d\|, \quad j \in J.$$

The right inequality holds also if d is not integral.

Proof. If the columns of D are all nonzero, then the left inequality in Lemma 2.1 remains valid if we replace $\Delta(D)$ by $\pi(D)$. This is immediate from the well-known Hadamard inequality for determinants and because D and d are integral. The inequality at the right follows by applying Hadamard's inequality to (2.3).[6] ∎

3. The optimal partition and two condition numbers. In the rest of this paper we assume that $M \in P_*(\kappa)$ for some $\kappa \geq 0$. This implies that the matrix M is sufficient.

3.1. Optimal partition. Let us denote the index set $\{1, 2, \ldots, n\}$ by I and define the sets

$$\begin{aligned} B &:= \{i \in I \ : \ x_i > 0 \text{ for some } x \in \Gamma^*\}, \\ N &:= \{i \in I \ : \ s_i(x) > 0 \text{ for some } x \in \Gamma^*\}, \\ T &:= \{i \in I \ : \ x_i = s_i(x) = 0 \text{ for all } x \in \Gamma^*\}. \end{aligned}$$

We show that these index sets are disjoint and $B \cup N \cup T = I$, i.e., they form the so-called *optimal partition* of the index set I with respect to (LCP).

LEMMA 3.1 (see [18]). *The index sets B, N, and T form a partition of the index set I.*

Proof. From the definition of the sets B, N, and T it is obvious that $B \cap T = \emptyset$, $N \cap T = \emptyset$, and $I = B \cup N \cup T$. Let us assume that $B \cap N \neq \emptyset$. Then there exist

[5] The idea of using Cramer's rule in this way was applied first by Khachiyan in [13].

[6] The idea for deriving bounds from Hadamard's inequality is due to Klafszky and Terlaky [14] (in Hungarian).

$x', x'' \in \Gamma^*$ such that $x'_j > 0, s_j(x') = 0, x''_j = 0$, and $s_j(x'') > 0$ for some $j \in I$. Let us denote $x := x' - x'', s' = s(x'), s'' = s(x''), s := s' - s''$, and $X := \mathrm{diag}(x)$. It is easy to see that

$$Xs = XMx \leq 0$$

and

$$x_j s_j = x_j (Mx)_j = (x'_j - x''_j)(s'_j - s''_j) = -x'_j s''_j < 0,$$

which contradicts the column sufficiency of the matrix M. □

COROLLARY 3.2. *Let x' and x'' solve (LCP), so $x', x'' \in \Gamma^*$. Then $x's(x'') = 0$ and $x''s(x') = 0$.*

Proof. The definition of the classes B and N implies $\{i \in I \ : \ x'_i > 0\} \subset B$ and $\{i \in I \ : \ s_i(x'') > 0\} \subset N$. Since $B \cap N = \emptyset$, it follows that x' and $s(x'')$ are complementary. The proof for x'' and $s(x')$ is analogous. □

COROLLARY 3.3. *The solution set Γ^* is convex.*

Proof. Let $x', x'' \in \Gamma^*$ and $\lambda \in [0, 1]$. If $x := \lambda x' + (1 - \lambda)x''$, then $x \geq 0$ and $s(x) = \lambda s(x') + (1 - \lambda)s(x'') \geq 0$. Thus $x \in \Gamma$. Further, Corollary 3.2 gives that $xs(x) = 0$, from which $x \in \Gamma^*$. □

A solution $x \in \Gamma^*$ is called *maximally complementary* if $x_B > 0$ and $s_N(x) > 0$. Since Γ^* is convex (and polyhedral) a maximally complementary solution exists.[7]

From now on we assume that $\Gamma^0 \neq \emptyset$. If the ith column of M is zero, then the P_* property implies that the ith row is zero as well. Therefore $s_i(x) = q_i$ in that case, for every x. Hence, if $q_i < 0$, then (LCP) is infeasible. If $q_i \geq 0$, then the constraint $s_i(x)$ is always satisfied and we may reduce the problem by removing the ith row and column of M. Thus we will assume that all columns of M are nonzero. When $q = 0$, then the (LCP) has a trivial solution $(x = 0)$. Therefore, without loss of generality, we further assume that $q \neq 0$.[8]

Our goal is to find the optimal partition of the index set and, finally, to round off to a maximally complementary solution. In fact, we will show that given $x(\mu)$ we can find the optimal partition provided μ is small enough. To this end we need to give bounds for the size of the variables along the central path. In the next two sections we obtain such bounds in terms of two condition numbers for (LCP).

3.2. The first condition number for (LCP). In this section we introduce our first *condition number* of (LCP).

This is done as in, e.g., Roos, Terlaky, and Vial [27] and Andersen and Ye [1] for LO problems. Since $\Gamma^0 \neq \emptyset$, Γ^* is nonempty and compact (see section 2), so the following two numbers are well defined:

$$\sigma^x_{LCP} := \min_{i \in B} \max_{x \in \Gamma^*} \{x_i\}, \qquad \sigma^s_{LCP} := \min_{i \in N} \max_{x \in \Gamma^*} \{s_i(x)\}.$$

By convention we take $\sigma^x_{LCP} = \infty$ if B is empty and $\sigma^s_{LCP} = \infty$ if N is empty; thus both σ^x_{LCP} and σ^s_{LCP} are positive. If B is nonempty, then σ^x_{LCP} is finite and if N

[7] The convexity of Γ^* is proved in another way in [4, Theorems 5 and 6, pp. 240–241]. Furthermore it is shown that Γ^* is a polyhedron.

[8] It may be noted that in this paper we find a maximally complementary solution of (LCP) under the assumptions $\Gamma^0 \neq \emptyset$ and $q \neq 0$. If $q = 0$, we have the trivial solution $x = 0$, but this solution will in general not be maximally complementary. The case $q = 0$ is interesting in itself. For example, if M is skew-symmetric, it covers LO and there exists a strictly complementary solution [27]; the other extreme occurs if M is a positive definite matrix (e.g., if M is the identity matrix): then $B = N = \emptyset$ and $T = I$.

is nonempty, then σ^s_{LCP} is finite. Since $q \neq 0$ it cannot happen that both B and N are empty; thus under the *interior point condition* ($\Gamma^0 \neq \emptyset$), at least one of the two numbers is finite. As a consequence, the number

$$\sigma_{LCP} := \min\{\sigma^x_{LCP}, \sigma^s_{LCP}\}$$

is positive and finite. One can easily verify that σ_{LCP} can also be written as

$$\sigma_{LCP} := \min_{i \in B \cup N} \max_{x \in \Gamma^*} \{x_i + s_i(x)\}.$$

In general, we have to solve a problem without knowing its condition number σ_{LCP}. In such cases there is a cheap way to get a lower bound for σ_{LCP} if the problem data (M, q) are integral. We proceed by deriving such a lower bound.

LEMMA 3.4. *If M and q are integral, then $\sigma_{LCP} \geq \frac{1}{\pi(M)}$.* □

Proof. For any vector $x \in \Gamma$ we have, with $s = s(x)$,

$$\begin{pmatrix} s_B \\ s_N \\ s_T \end{pmatrix} = \begin{pmatrix} M_{BB} & M_{BN} & M_{BT} \\ M_{NB} & M_{NN} & M_{NT} \\ M_{TB} & M_{TN} & M_{TT} \end{pmatrix} \begin{pmatrix} x_B \\ x_N \\ x_T \end{pmatrix} + \begin{pmatrix} q_B \\ q_N \\ q_T \end{pmatrix}. \tag{3.1}$$

Further, $x \in \Gamma^*$ holds if and only if $x_N = 0$, $x_T = s_T = 0$, $s_B = 0$. This is equivalent to

$$\begin{pmatrix} M_{BB} & 0_{BN} \\ M_{NB} & -E_{NN} \\ M_{TB} & 0_{TN} \end{pmatrix} \begin{pmatrix} x_B \\ s_N \end{pmatrix} = \begin{pmatrix} -q_B \\ -q_N \\ -q_T \end{pmatrix}, \quad x_B \geq 0, s_N \geq 0. \tag{3.2}$$

Any maximally complementary solution x yields a positive solution of this system. In order to get a lower bound on σ_{LCP} we need to derive a lower bound on the maximal value of each coordinate of the vector $z := (x_B, s_N)$ when this vector runs through all possible solutions of (3.2). For each i we know that there exists a solution z with $z_i > 0$. Hence there exists a basic solution z of (3.2) with $z_i > 0$. Therefore, Corollary 2.2 yields the following lower bound on the biggest coordinate of z.

$$\max_{x \in \Gamma^*} z_i \geq \frac{1}{\pi\left(M_{.B}\right)}.$$

Since $\pi(M) \geq \pi\left(M_{.B}\right)$, the lemma follows. □

Now we are ready to estimate the size of the variables $x_i, s_i(x)$ when x lies on the central path, i.e., $xs(x) = \mu e$, and $i \in B$ or $i \in N$. We denote $s(\mu) := s\left(x(\mu)\right)$.

THEOREM 3.5. *For any positive μ one has*

$$\begin{aligned} x_i(\mu) &\geq \frac{\sigma_{LCP}}{n(1+4\kappa)},\ i \in B, & x_i(\mu) &\leq \frac{n\mu(1+4\kappa)}{\sigma_{LCP}},\ i \in N, \\ s_i(\mu) &\leq \frac{n\mu(1+4\kappa)}{\sigma_{LCP}},\ i \in B, & s_i(\mu) &\geq \frac{\sigma_{LCP}}{n(1+4\kappa)},\ i \in N. \end{aligned}$$

Proof. We first consider the case $i \in N$. Let us assume that $\overline{x} \in \Gamma^*$ and $\overline{s} := s(\overline{x})$.

Taking into consideration that $M \in P_*(\kappa)$ and $x(\mu), s(\mu), \overline{x}, \overline{s} \geq 0$, we get

$$\begin{aligned}(x(\mu)-\overline{x})^T(s(\mu)-\overline{s}) &= (x(\mu)-\overline{x})^T M(x(\mu)-\overline{x}) \\ &\geq -4\kappa \sum_{i\in I_+(x(\mu)-\overline{x})} (x(\mu)-\overline{x})_i [M(x(\mu)-\overline{x})]_i \\ &= -4\kappa \sum_{i\in I_+(x(\mu)-\overline{x})} (x(\mu)-\overline{x})_i (s(\mu)-\overline{s})_i \\ &= -4\kappa \sum_{i\in I_+(x(\mu)-\overline{x})} ((x(\mu)s(\mu))_i - (x(\mu)\overline{s})_i - (\overline{x}s(\mu))_i + (\overline{x}\overline{s})_i) \\ &\geq -4\kappa \sum_{i\in I_+(x(\mu)-\overline{x})} (x(\mu)s(\mu))_i \\ &\geq -4\kappa n\mu. \qquad (3.3)\end{aligned}$$

The last inequality holds because $x(\mu)s(\mu) = \mu e$ and $\overline{x} \in \Gamma^*$. On the other hand

$$(x(\mu)-\overline{x})^T(s(\mu)-\overline{s}) = n\mu - x(\mu)^T\overline{s} - \overline{x}^T s(\mu).$$

Combining this with (3.3) we have

$$x(\mu)^T\overline{s} + s(\mu)^T\overline{x} \leq n\mu(1+4\kappa),$$

which implies

$$x_i(\mu)\overline{s}_i \leq x(\mu)^T\overline{s} \leq n\mu(1+4\kappa) \qquad \text{for all } i \in I. \qquad (3.4)$$

Now if $i \in N$ and $(\overline{x}, \overline{s})$ is a maximally complementarity solution, then by definition $\overline{s}_i \geq \sigma_{LCP}$. Dividing by $\overline{s}_i$ in (3.4) we obtain

$$x_i(\mu) \leq \frac{n\mu(1+4\kappa)}{\overline{s}_i} \leq \frac{n\mu(1+4\kappa)}{\sigma_{LCP}}. \qquad (3.5)$$

Since $x_i(\mu)s_i(\mu) = \mu$, it also follows that

$$s_i(\mu) \geq \frac{\sigma_{LCP}}{n(1+4\kappa)} \qquad \text{for all } i \in N.$$

This proves the second and fourth inequality in the lemma. The first and third inequalities for $i \in B$ are obtained from (3.4) analogously. □

3.3. The second condition number for (*LCP*). In this section we derive bounds that will help us control the variables $x_i(\mu)$ and $s_i(\mu)$ if $i \in T$. Before dealing with the main theorem in this section we review some results about systems of linear inequalities and equalities.

Let $A \in \mathbb{R}^{m\times n}$ and $C \in \mathbb{R}^{k\times n}$ be two real matrices. For given $b \in \mathbb{R}^m$ and $d \in \mathbb{R}^k$, consider the following system of linear inequalities

$$Ax \leq b, \quad Cx = d. \qquad (3.6)$$

Cook et al. [3] and Mangasarian and Shiau [21] studied the Lipschitz continuity of solutions of (3.6) with respect to right-hand-side perturbations of (3.6). We will use a variant of those results. For completeness, we give a simple proof, similar to that presented in [3]. We further assume that both A and C do not contain a zero row.

LEMMA 3.6. *Let the system* (3.6) *have nonempty feasible sets* Γ^1 *and* Γ^2 *for the right-hand side* (b^1, d^1) *and* (b^2, d^2)*, respectively. For each* $x^1 \in \Gamma^1$ *there exists an* $x^2 \in \Gamma^2$ *such that*

$$\|x^1 - x^2\|_\infty \leq \nu(A; C) \left\| \begin{pmatrix} b^1 - b^2 \\ d^1 - d^2 \end{pmatrix} \right\|_\infty, \tag{3.7}$$

where[9]

$$\nu(A; C) := \max_{u,v} \left\{ \left\| \begin{pmatrix} u \\ v \end{pmatrix} \right\|_1 \;\middle|\; \begin{array}{c} A^T u + C^T v = z - y,\ e^T(z+y) = 1,\ u, y, z \geq 0, \\ \textit{the columns of } (A^T, C^T) \textit{ corresponding to nonzero} \\ \textit{elements of } (u, v) \textit{ are linearly independent} \end{array} \right\}.$$

Proof. We are interested in finding t such that $t = \left\|x - x^1\right\|_\infty$, with $x \in \Gamma^2$, is minimal. This amounts to solving the linear minimization problem

$$\min_x \left\{ t \;:\; Ax \leq b^2,\ Cx = d^2,\ te + x \geq x^1,\ te - x \geq -x^1 \right\}. \tag{3.8}$$

Note that this problem is feasible, since $\Gamma^2 \neq \emptyset$, and bounded. Hence, the optimal value t^* is equal to the optimal value of the dual problem of (3.8). This gives

$$t^* = \max \left\{ u^T b^2 + v^T d^2 + y^T x^1 - z^T x^1 \;:\; \begin{array}{c} A^T u + C^T v + y - z = 0, \\ e^T(z+y) = 1, \\ y, z \geq 0,\ u \leq 0 \end{array} \right\}. \tag{3.9}$$

Let (u, v, y, z) be an optimal solution of this problem. Recalling that $x^1 \in \Gamma^1$ and $u \leq 0$, from the definition of t^* we get

$$\begin{aligned} t^* &= u^T b^2 + v^T d^2 + (y - z)^T x^1 \\ &= u^T b^2 + v^T d^2 - \left(A^T u + C^T v\right)^T x^1 \\ &= u^T b^2 + v^T d^2 - u^T A x^1 - v^T C x^1 \\ &\leq u^T b^2 + v^T d^2 - u^T b^1 - v^T d^1 \\ &= u^T \left(b^2 - b^1\right) + v^T \left(d^2 - d^1\right) \\ &\leq \left\| \begin{pmatrix} u \\ v \end{pmatrix} \right\|_1 \left\| \begin{pmatrix} b^2 - b^1 \\ d^2 - d^1 \end{pmatrix} \right\|_\infty. \end{aligned}$$

Hence, the proof will be complete if we show that (3.9) has an optimal solution (u, v, y, z) such that the columns of (A^T, C^T) corresponding to the nonzero components of (u, v) are linearly independent. This can be shown as follows. Suppose to the contrary that the columns corresponding to the nonzero coordinates of (u, v) are dependent; then there exist nonzero vectors $\bar{u}$ and $\bar{v}$ such that $A^T \bar{u} + C^T \bar{v} = 0$ and $\bar{u}_i = 0$ if $u_i = 0$ and $\bar{v}_i = 0$ if $v_i = 0$. Let us define $w(\lambda) := (u, v, y, z) + \lambda(\bar{u}, \bar{v}, 0, 0)$; then $w(\lambda)$ is feasible for (3.9) for all λ satisfying $u + \lambda \bar{u} \leq 0$. From the definition of $\bar{u}$ and $\bar{v}$ one can easily conclude that there is a closed interval $[\alpha, \beta]$ with $\alpha < 0 < \beta$ (here we allow $\alpha = -\infty$ and $\beta = \infty$) such that $w(\lambda)$ is feasible for (3.9) for any

[9] This definition is a slight modification of the one given by Mangasarian and Shiau [21]. They simply require $\|A^T u + C^T v\|_1 = 1$, not using the variables y and z. Our definition has the advantage that the feasible region of the optimization problem defining $\nu(A; C)$ is a polyhedral set.

$\lambda \in [\alpha, \beta]$. Hence we necessarily have $\bar{u}^T b^2 + \bar{v}^T d^2 = 0$—otherwise a contradiction with the optimality of $w(0)$ would arise. As a consequence, $w(\lambda)$ is optimal for all $\lambda \in [\alpha, \beta]$. Clearly, by choosing λ appropriately, we can obtain a solution of (3.9) with fewer nonzero coordinates. By repeating this procedure we obtain a solution (u, v, y, z) of (3.9) for which the columns of (A^T, C^T) corresponding to the nonzero components of (u, v) are linearly independent. For such a solution by the definition of $\nu(A; C)$ we have

$$\left\| \begin{pmatrix} u \\ v \end{pmatrix} \right\|_1 \leq \nu(A; C),$$

because $(\hat{u}, \hat{v}, \hat{y}, \hat{z}) = (-u, -v, z, y)$ is feasible for the optimization problem in the definition of $\nu(A, C)$ with

$$\left\| \begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} \right\|_1 = \left\| \begin{pmatrix} u \\ v \end{pmatrix} \right\|_1.$$

This completes the proof. □

We proceed by deriving a lower bound for $\nu(A; C)$.

LEMMA 3.7. *One has*

$$\nu(A; C) \geq \frac{1}{\min_{i,j} \left(\|a_i\|_1, \|c_j\|_1 \right)},$$

where a_i runs through the rows of A and c_j through the rows of C.

Proof. Let a denote the ith row of A. Then, if e_i denotes the ith unit vector, one has $\left\|A^T e_i\right\|_1 = \|a\|_1$. Hence, assuming $a \neq 0$, taking $u = e_i / \|a\|_1, v = 0, z_j = a_j / \|a\|_1$ if $a_j \geq 0, y_j = -a_j / \|a\|_1$ if $a_j < 0$, and all remaining entries of y and z equal to zero, the quadruple (u, v, y, z) is feasible for the maximization problem defining $\nu(A; C)$. Therefore, $\nu(A; C) \geq \|u\|_1 = 1 / \|a\|_1$. A similar argument yields that $\nu(A; C) \geq 1 / \|c\|_1$ for each row c of C, and hence the lemma follows. □

An upper bound for $\nu(A; C)$ can be derived if all the entries of A and C are integral.

LEMMA 3.8. *For integer A, C one has*

$$\nu(A; C) \leq n\Delta(A^T; C^T) \leq n\pi(A^T, C^T). \tag{3.10}$$

Proof. Let (u, v, y, z) be a feasible solution for the maximization problem in the definition of $\nu(A; C)$. Let $w^T = (u^T, v^T)$, $\bar{A} = (A^T, C^T)$. Then $\bar{A}w = z - y$. Since the columns of $\bar{A}$ corresponding to nonzero elements of w are linearly independent, we may apply Lemma 2.1, which yields

$$\|w\|_\infty \leq \Delta\left(\bar{A}\right) \|z - y\|_1 \leq \Delta\left(\bar{A}\right).$$

The last inequality follows since $\|z - y\|_1 \leq \|z + y\|_1 = 1$. Since $\|w\|_1 \leq n\|w\|_\infty$ the first inequality in the lemma follows from this. The rest of the lemma follows from the Hadamard inequality for determinants. Hence the proof is complete. □

We now are going to apply Lemma 3.8 to a second condition number for (LCP), which enables us to bound the variables along the central path. This second condition number, denoted as ν_{LCP}, depends on the input matrix M and the optimal partition (B, N, T). It is defined as follows.

DEFINITION 3.9. *Let I_1, I_2 be a partition of the index set I such that $B \subseteq I_1$ and $N \subseteq I_2$. Let us define*

$$\nu_{LCP} := \max_{I_1 \cup I_2 = I} \nu \left[\begin{pmatrix} -E_{I_1} & 0 \\ 0 & -E_{I_2} \end{pmatrix}; \begin{pmatrix} M & -E \\ E_{I_2} & 0 \\ 0 & E_{I_1} \end{pmatrix} \right].$$

If the matrix M is integral, then we can give a lower bound and an easily computable upper bound for ν_{LCP}.

LEMMA 3.10. *If M is integral, then*

$$1 \leq \nu_{LCP} \leq \max_{I_1 \cup I_2 = I} n\Delta \left[\begin{pmatrix} -E_{I_1} & 0 \\ 0 & -E_{I_2} \end{pmatrix}; \begin{pmatrix} M & -E \\ E_{I_2} & 0 \\ 0 & E_{I_1} \end{pmatrix} \right] = n\Delta(M) \leq n\pi(M).$$

Proof. The first inequality is immediate from Lemma 3.7, the second inequality follows from Lemma 3.8, the equality is obvious, and the last inequality is Hadamard's inequality. □

Now we are ready to state our main theorem in this section.

THEOREM 3.11. *If*

$$\mu < \frac{\sigma_{LCP}^2}{n^2(1+4\kappa)^2}, \tag{3.11}$$

then

$$\frac{\sqrt{\mu}}{\nu_{LCP}} \leq x_i(\mu),\, s_i(\mu) \leq \nu_{LCP}\sqrt{\mu}, \quad i \in T.$$

Proof. When (3.11) holds, one can easily verify that

$$x_i(\mu) \geq \frac{\sigma_{LCP}}{n(1+4\kappa)} > \frac{n\mu(1+4\kappa)}{\sigma_{LCP}} \geq s_i(\mu) \text{ for all } i \in B$$

and

$$s_i(\mu) \geq \frac{\sigma_{LCP}}{n(1+4\kappa)} > \frac{n\mu(1+4\kappa)}{\sigma_{LCP}} \geq x_i(\mu) \text{ for all } i \in N.$$

Letting

$$I_1 = \{i \,:\, x_i(\mu) \geq s_i(\mu)\}, \quad I_2 = \{i \,:\, x_i(\mu) < s_i(\mu)\},$$

we have $B \subset I_1$ and $N \subset I_2$ if (3.11) holds. Hence, defining

$$H(x) := \min(x, s(x)),$$

we have

$$H_i(x(\mu)) = \begin{cases} s_i(\mu) & \text{if } i \in I_1, \\ x_i(\mu) & \text{if } i \in I_2. \end{cases}$$

From the fact that $H(x(\mu)) = \min(x(\mu), s(\mu))$ and $x_i(\mu)s_i(\mu) = \mu$ we conclude that $H_i(x(\mu)) \leq \sqrt{\mu}$.

Consider the following linear system:

$$\begin{array}{rcll} Mx - s & = & -q, & \\ x_{I_2} & = & 0, & -x_{I_1} \leq 0, \\ s_{I_1} & = & 0, & -s_{I_2} \leq 0. \end{array} \tag{3.12}$$

It is easy to see that the feasible set of the system (3.12) is the solution set Γ^* of (LCP). Let this set play the role of Γ^2 in Lemma 3.6. Further, let the solution set of the following linear system play the role of Γ^1:

$$\begin{array}{rcll} Mx - s & = & -q, & \\ x_{I_2} & = & H_{I_2}(x(\mu)), & -x_{I_1} \leq 0, \\ s_{I_1} & = & H_{I_1}(x(\mu)), & -s_{I_2} \leq 0. \end{array} \tag{3.13}$$

Clearly Γ^1 is not empty, because $x(\mu)$ satisfies (3.13). Now it follows from Lemma 3.6 that there exists a solution x^* of (3.12), i.e., $x^* \in \Gamma^*$, such that

$$\left\| \begin{pmatrix} x^* - x(\mu) \\ s^* - s(\mu) \end{pmatrix} \right\|_\infty \leq \nu \left[\begin{pmatrix} -E_{I_1} & 0 \\ 0 & -E_{I_2} \end{pmatrix}; \begin{pmatrix} M & -E \\ E_{I_2} & 0 \\ 0 & E_{I_1} \end{pmatrix} \right] \|H(x(\mu))\|_\infty.$$

Using the definition of ν_{LCP} and $H_i(x(\mu)) \leq \sqrt{\mu}$ it follows that

$$\left\| \begin{pmatrix} x^* - x(\mu) \\ s^* - s(\mu) \end{pmatrix} \right\|_\infty \leq \nu_{LCP}\sqrt{\mu}.$$

Since $x_i^* = 0$ for $i \in T$, we conclude that for all $i \in T \cap I_1$ one has

$$\frac{\sqrt{\mu}}{\nu_{LCP}} \leq s_i(\mu) \leq \sqrt{\mu} \leq x_i(\mu) \leq \nu_{LCP}\sqrt{\mu}.$$

Similarly for all $i \in T \cap I_2$, it holds that

$$\frac{\sqrt{\mu}}{\nu_{LCP}} \leq x_i(\mu) \leq \sqrt{\mu} \leq s_i(\mu) \leq \nu_{LCP}\sqrt{\mu}.$$

This proves the theorem. □

3.4. Finding the optimal partition. In Table 3.1 we show the results of the last two theorems (Theorem 3.5 and Theorem 3.11).

These results have an important consequence. If μ is so small that

$$\frac{n\mu(1+4\kappa)}{\sigma_{LCP}} < \frac{\sqrt{\mu}}{\nu_{LCP}}$$

and

$$\nu_{LCP}\sqrt{\mu} < \frac{\sigma_{LCP}}{n(1+4\kappa)},$$

TABLE 3.1
Local bounds for the variables on the central path.

	$i \in B$	$i \in N$	$i \in T$
$x_i(\mu)$	$\geq \frac{\sigma_{LCP}}{n(1+4\kappa)}$	$\leq \frac{n\mu(1+4\kappa)}{\sigma_{LCP}}$	$\frac{\sqrt{\mu}}{\nu_{LCP}} \leq x_i(\mu) \leq \nu_{LCP}\sqrt{\mu}$
$s_i(\mu)$	$\leq \frac{n\mu(1+4\kappa)}{\sigma_{LCP}}$	$\geq \frac{\sigma_{LCP}}{n(1+4\kappa)}$	$\frac{\sqrt{\mu}}{\nu_{LCP}} \leq s_i(\mu) \leq \nu_{LCP}\sqrt{\mu}$

then we have a complete separation of the variables. Both inequalities give the same bound on μ, namely,

$$\mu < \frac{{\sigma_{LCP}}^2}{\nu_{LCP}^2 n^2 (1+4\kappa)^2}. \tag{3.14}$$

This means that if a point on the central path is given such that (3.14) holds, then we can determine the optimal partition (B, N, T) of (LCP).

Unfortunately, in practice we may not assume that we can calculate points on the central path exactly. Practical algorithms generate points in the vicinity of the central path. Therefore, in the next section we deal with the situation that a point x is given in an appropriate neighborhood of the central path. We will show that if x is close enough to $x(\mu)$, with μ small enough, we also have a complete separation of the variables into the three different classes B, N, and T. This will imply that all path-following IPMs eventually produce iterates that are suitable for identifing the optimal partition of (LCP).

4. Optimal partition identification from approximate centers. In this section we generalize the results of the previous section to the case where a point x is given in a specific neighborhood of the central path. On the central path all the coordinates of the vector $xs(x)$ are equal. This suggests that a good measure of centrality could be the ratio of the smallest and largest coordinate. If we bound this ratio, then a neighborhood of the central path is obtained. We therefore use the *measure of centrality*[10]

$$\delta_c(x) := \frac{\max(xs(x))}{\min(xs(x))},$$

where $\max(xs(x))$ denotes the largest coordinate of $xs(x)$ and $\min(xs(x))$ denotes the smallest one.

4.1. Finding the optimal partition from approximate centers. We first generalize the results of Theorems 3.5 and 3.11 to the case where x is not on the central path $\mathcal{C}$.

[10]This measure of centrality is introduced by Ling [20] and used in [8, 9]. The same measure of centrality is used throughout Roos, Terlaky, and Vial [27].

LEMMA 4.1. *Let $x \in \Gamma^0$ and $s = s(x)$. If $\delta_c(x) \le \tau$, for some $\tau > 1$, and $\mu := \frac{x^T s(x)}{n}$, then one has*

$$x_i \ge \frac{\sigma_{LCP}}{\tau n(1+4\kappa)},\ i \in B, \qquad x_i \le \frac{(1+4\kappa)n\mu}{\sigma_{LCP}},\ i \in N,$$
$$s_i \le \frac{(1+4\kappa)n\mu}{\sigma_{LCP}},\ i \in B, \qquad s_i \ge \frac{\sigma_{LCP}}{\tau n(1+4\kappa)},\ i \in N.$$

If, further,

$$\mu \le \frac{\sigma_{LCP}^2}{\tau n^2(1+4\kappa)^2}, \tag{4.1}$$

then

$$\frac{\sqrt{\mu}}{\tau\sqrt{\tau}\nu_{LCP}} \le x_i,\ s_i \le \sqrt{\tau}\nu_{LCP}\sqrt{\mu},\ i \in T.$$

Proof. The proof uses essentially the same arguments as the proofs of Theorems 3.5 and 3.11. The arguments leading to (3.5) in the proof of Theorem 3.5 are still valid, so

$$x_i \le \frac{(1+4\kappa)x^T s}{\sigma_{LCP}} = \frac{(1+4\kappa)n\mu}{\sigma_{LCP}} \quad \text{for } i \in N. \tag{4.2}$$

The rest of the proof is a little complicated by the fact that x is not on the central path but only in a certain neighborhood of the central path. If $\delta_c(x) \le \tau$, then there are α, $\beta \in (0, \infty)$ such that

$$\alpha e \le xs \le \beta e \quad \text{with} \quad \frac{\beta}{\alpha} = \tau. \tag{4.3}$$

These inequalities replace the identity $x_i(\mu)s_i(\mu) = \mu$ used in the proof of Theorem 3.5. Due to the left inequality in (4.3) we also have $x_i s_i \ge \alpha$ for all i. Hence using (4.2) we must have

$$s_i \ge \frac{\alpha\sigma_{LCP}}{(1+4\kappa)x^T s}.$$

The right inequality in (4.3) gives $x^T s \le n\beta$, and thus

$$s_i \ge \frac{\alpha\sigma_{LCP}}{n\beta(1+4\kappa)} = \frac{\sigma_{LCP}}{\tau n(1+4\kappa)}.$$

This proves the second and fourth inequality in the lemma. The proof of the first and third inequalities can be obtained in the same way and is therefore left to the reader.

To prove the last statement of the lemma, we notice that for the current point (x, s), it obviously holds that

$$\frac{x_i s_i}{\mu} = \frac{n x_i s_i}{x^T s} \ge \frac{n \min(xs)}{n \max(xs)} \ge \frac{1}{\tau} \quad \text{for all } i = 1, 2, \ldots, n$$

and

$$\frac{x_i s_i}{\mu} = \frac{n x_i s_i}{x^T s} \le \frac{n \max(xs)}{n \min(xs)} \le \tau \quad \text{for all } i = 1, 2, \ldots, n.$$

TABLE 4.1
Local estimates for variables belonging to index sets B, N, and T if $\delta_c(x) \le \tau$.

$\mu = \frac{x^T s(x)}{n}$	$i \in B$	$i \in N$	$i \in T$
x_i	$\ge \frac{\sigma_{LCP}}{\tau n(1+4\kappa)}$	$\le \frac{(1+4\kappa)n\mu}{\sigma_{LCP}}$	$\frac{\sqrt{\mu}}{\tau\sqrt{\tau}\nu_{LCP}} \le x_i \le \sqrt{\tau}\nu_{LCP}\sqrt{\mu}$
$s_i(x)$	$\le \frac{(1+4\kappa)n\mu}{\sigma_{LCP}}$	$\ge \frac{\sigma_{LCP}}{\tau n(1+4\kappa)}$	$\frac{\sqrt{\mu}}{\tau\sqrt{\tau}\nu_{LCP}} \le s_i \le \sqrt{\tau}\nu_{LCP}\sqrt{\mu}$

Letting $H(x) = \min(x, s)$, the above two inequalities give

$$[H(x)]_i \le \sqrt{\tau}\sqrt{\mu} \quad \text{for all } i = 1, 2, \ldots, n.$$

Following arguments similar to those in the proof of Theorem 3.11, one can easily derive the conclusion. □

In Table 4.1 we show the results of the above lemma.

We conclude that the partition (B, N, T) can be identified if $x^T s(x)$ is so small that

$$\frac{(1+4\kappa)n\mu}{\sigma_{LCP}} < \frac{\sqrt{\mu}}{\tau\sqrt{\tau}\nu_{LCP}}$$

and

$$\sqrt{\tau}\nu_{LCP}\sqrt{\mu} < \frac{\sigma_{LCP}}{\tau n(1+4\kappa)}.$$

It is easy to verify that both inequalities give the same bound for μ; thus for complete separation of the variables we need

$$\mu < \frac{\sigma_{LCP}^2}{n^2\tau^3\nu_{LCP}^2(1+4\kappa)^2}. \tag{4.4}$$

Therefore we may state without further proof our main result.

THEOREM 4.2. *Let $x \in \Gamma^0$ be such that $\delta_c(x) \le \tau$, for some $\tau > 1$, and $\mu = \frac{x^T s(x)}{n}$. If (4.4) is true, then, with $s = s(x)$, the optimal partition of (LCP) follows from*

$$T = \left\{ i \ : \ \frac{\sqrt{\mu}}{\tau\sqrt{\tau}\nu_{LCP}} \le x_i,\ s_i \le \sqrt{\tau}\nu_{LCP}\sqrt{\mu} \right\},$$

$$B = \{i \notin T \ : \ x_i > s_i\}, \quad \textit{and} \quad N = \{i \notin T \ : \ x_i < s_i\}. \qquad \square$$

4.2. Complexity of finding the optimal partition. In this section we assume that we have given a point $x^{(0)} \in \Gamma^0$ close to the central path (i.e., $\delta_c(x^{(0)}) \le \tau$ for some $\tau > 1$). We define μ^0 by $n\mu^0 = \left(x^{(0)}\right)^T s^{(0)}$. Starting at x^0, interior point methods for solving (LCP) need $\mathcal{O}(\sqrt{n}\log(n\mu^0/\epsilon))$ iterations (see, e.g., [12, 16, 18, 34]) or $\mathcal{O}(n\log(n\mu^0/\epsilon))$ iterations (see, e.g., [8]) to generate a point x such that $\delta_c(x) \le \tau$ and $x^T s(x) \le \epsilon$. The first bound holds for methods with small updates of the barrier parameter, whereas the second bound is typical for methods using large updates, and also for methods using a Dikin-type affine-scaling direction. Hence, by

substituting the value of ϵ according to Theorem 4.2, we can get iteration bounds to identify the optimal partition.

The above will be illustrated below for the Dikin affine-scaling algorithm presented in [8]. If $n \geq 4$, this algorithm, with $\tau = 2$, requires at most

$$3(1+4\kappa)n \log \frac{n\mu^0}{\epsilon} \tag{4.5}$$

iterations to generate a point x such that $\delta_c(x) \leq 2$ and $x^T s(x) \leq \epsilon$.

THEOREM 4.3. *Starting at a point $x^{(0)} \in \Gamma^0$ with $\delta_c\left(x^{(0)}\right) \leq 2$, and $n \geq 4$, the Dikin affine-scaling algorithm reveals the optimal partition after at most*

$$3(1+4\kappa)n \log \frac{8n^2(1+4\kappa)^2\nu_{LCP}^2\mu^0}{\sigma_{LCP}^2} \leq 3(1+4\kappa)n \log \left(8n^4(1+4\kappa)^2\pi(M)^4\mu^0\right)$$

iterations.

Proof. The expression (4.5) gives the number of iterations to reach an ϵ-solution. With μ as in Theorem 4.2 and $\epsilon = n\mu$ we obtain the first bound. The inequality follows by using the upper bound for ν_{LCP} in Lemma 3.10 and the lower bound for σ_{LCP} in Lemma 3.4. □

Similar results can be derived for other polynomial IPMs.

5. Rounding to a strictly complementary solution. We just established that the optimal partition of (LCP) can be found after a polynomial number of iterations with any known path-following IPMs for $P_*(\kappa)$ LCPs. The required number of iterations depends on the starting point $x^{(0)}$, the parameter κ, and the condition numbers ν_{LCP} and σ_{LCP}. Our ultimate goal is not only to find the optimal partition but also to find an exact and maximally complementary solution of (LCP). Assuming that the optimal partition (B, N, T) has been determined, with B nonempty,[11] we describe a rounding procedure that can be applied to any sufficiently centered positive vector x with $x^T s(x)$ small enough, and the rounding procedure yields a vector $\tilde{x}$ such that (3.2) is satisfied and $\tilde{x}_B > 0, s_N(\tilde{x}) > 0$. As might be expected, the accuracy that was sufficient to find the optimal partition is not enough to perform the rounding procedure. In Theorem 5.1 we will give a bound on the complementary gap that provides sufficient accuracy for our rounding procedure. The rounding procedure yields a maximally complementary solution in strongly polynomial time. Finally, the number of iterations required to reach the necessarily small complementarity gap is bounded by Theorem 5.2.

5.1. Rounding procedure. Let $x \in \Gamma^0, s = s(x)$ be given and assume that the optimal partition (B, N, T) is known. Now we want to compute $\Delta x_B, \Delta s_N$ such that

$$x_B + \Delta x_B > 0 \quad \text{and} \quad s_N + \Delta s_N > 0 \tag{5.1}$$

and

$$\begin{pmatrix} 0 \\ s_N + \Delta s_N \\ 0 \end{pmatrix} = \begin{pmatrix} M_{BB} & M_{BN} & M_{BT} \\ M_{NB} & M_{NN} & M_{NT} \\ M_{TB} & M_{TN} & M_{TT} \end{pmatrix} \begin{pmatrix} x_B + \Delta x_B \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} q_B \\ q_N \\ q_T \end{pmatrix}, \tag{5.2}$$

[11] If $B = \emptyset$, then $x = 0$ and $s = q$ is the only possible solution. The vector $(0, q)$ solves the problem if and only if $q \geq 0$.

because then $\tilde{x} := (x_B + \Delta x_B, 0, 0) \in \Gamma^*$, and this solution is maximally complementary. Since x and s satisfy (3.1), we may subtract (3.1) from (5.2), leading to the system

$$\begin{pmatrix} -s_B \\ \Delta s_N \\ -s_T \end{pmatrix} = \begin{pmatrix} M_{BB} & M_{BN} & M_{BT} \\ M_{NB} & M_{NN} & M_{NT} \\ M_{TB} & M_{TN} & M_{TT} \end{pmatrix} \begin{pmatrix} \Delta x_B \\ -x_N \\ -x_T \end{pmatrix}, \tag{5.3}$$

which is thus equivalent to (5.2). We can rewrite this as

$$\begin{pmatrix} M_{BB} & 0_{BN} \\ M_{NB} & -E_{NN} \\ M_{TB} & 0_{TN} \end{pmatrix} \begin{pmatrix} \Delta x_B \\ \Delta s_N \end{pmatrix} = \begin{pmatrix} M_{BN}x_N + M_{BT}x_T - s_B \\ M_{NN}x_N + M_{NT}x_T \\ M_{TN}x_N + M_{TT}x_T - s_T \end{pmatrix}. \tag{5.4}$$

We conclude that we can round x to a maximally complementary solution $\tilde{x}$ if we can find a solution $(\Delta x_B, \Delta s_N)$ of (5.4) that satisfies (5.1). We show below that if $x^T s(x) = n\mu$ is small enough and x is close enough to the central path, then such $(\Delta x_B, \Delta s_N)$ can be found by Gaussian elimination.

It may be useful to point out that the analysis below works out well because the variables x_T, x_N, s_B, and s_T that occur in the right-hand side of (5.4) are "small" if μ is small enough and x is close enough to the central path. These variables are bounded above by Lemma 4.1. Since x_B and s_N are "large," by the same lemma, it is therefore not surprising that (5.4) admits a solution such that (5.1) holds.

THEOREM 5.1. *Assume that M is integral. Let $x \in \Gamma^0$ be such that $\delta_c(x) \le \tau = 2$. If*

$$\mu < \frac{\sigma_{LCP}^2}{8n^3(1+4\kappa)^2\nu_{LCP}^2 \|M\|_\infty^2 \pi(M)^2}, \tag{5.5}$$

then the rounding procedure yields a maximally complementary solution in at most $\mathcal{O}(n^3)$ arithmetic operations.

Proof. To keep the expressions simple we introduce the following notation:

$$A := \begin{pmatrix} M_{BB} & 0_{BN} \\ M_{NB} & -E_{NN} \\ M_{TB} & 0_{TN} \end{pmatrix}, \ \Delta z := \begin{pmatrix} \Delta x_B \\ \Delta s_N \end{pmatrix}, \text{ and } r := \begin{pmatrix} M_{BN}x_N + M_{BT}x_T - s_B \\ M_{NN}x_N + M_{NT}x_T \\ M_{TN}x_N + M_{TT}x_T - s_T \end{pmatrix}.$$

Then (5.4) becomes

$$A\Delta z = r. \tag{5.6}$$

When solving (5.6) by Gaussian elimination, which needs $\mathcal{O}(n^3)$ arithmetic operations, we obtain a solution such that the columns of A corresponding to its support are linearly independent. Hence, using Corollary 2.2,

$$\|\Delta z\|_\infty \le \pi(A)\|r\| = \pi(M_{\cdot B})\|r\| \le \pi(M)\|r\|. \tag{5.7}$$

We proceed by estimating $\|r\|$. We use the trivial inequality $\|r\| \le \sqrt{n}\|r\|_\infty$ and

$$\|r\|_\infty \le \left\| \begin{pmatrix} M_{BN} & M_{BT} & -E_B & 0 \\ M_{NN} & M_{NT} & 0 & 0 \\ M_{TN} & M_{TT} & 0 & -E_T \end{pmatrix} \right\|_\infty \left\| \begin{pmatrix} x_N \\ x_T \\ s_B \\ s_T \end{pmatrix} \right\|_\infty. \tag{5.8}$$

Observe that the value of μ given by (5.5) satisfies the hypothesis of Theorem 4.2. Therefore, we have a complete separation of the variables. As a consequence, all entries in the vectors x_N, x_T, s_B, and s_T are bounded above by $\sqrt{\tau}\nu_{LCP}\sqrt{\mu}$. Hence, the infinity norm of the concatenation of these vectors, which appears at the right in (5.8), is bounded above by this number. Obviously the infinity norm of the matrix in (5.8) is bounded above by the infinity norm of M. Thus we find

$$\|r\| \leq 2n\sqrt{n}\nu_{LCP}(1+4\kappa)\sqrt{\mu}\,\|M\|_\infty .$$

Substitution in (5.7) yields

$$\|\Delta z\|_\infty \leq \sqrt{n}\nu_{LCP}\sqrt{\tau}\sqrt{\mu}\,\|M\|_\infty \pi(M). \tag{5.9}$$

Using the lower bound of Lemma 4.1 (with $\tau = 2$) for the entries of x_B and s_N, we conclude that the rounding procedure certainly yields a maximally complementary solution if

$$\sqrt{2n}\nu_{LCP}\sqrt{\mu}\,\|M\|_\infty \pi(M) < \frac{\sigma_{LCP}}{2n(1+4\kappa)}.$$

This inequality is equivalent to

$$\sqrt{\mu} < \frac{\sigma_{LCP}}{2\sqrt{2}n\sqrt{n}\nu_{LCP}(1+4\kappa)\,\|M\|_\infty \pi(M)},$$

which yields the bound for μ in the theorem. This completes the proof. □

5.2. Complexity of finding an exact solution. We apply the results of the previous section to estimate the number of iterations required by the Dikin affine-scaling algorithm to reach the state where the rounding procedure yields a maximally complementary solution. Without further proof we may state our final result.

THEOREM 5.2. *Starting at a point $x^{(0)} \in \Gamma^0$ with $\delta_c\left(x^{(0)}\right) \leq 2$, and $n \geq 4$, the Dikin affine-scaling algorithm requires at most*

$$3(1+4\kappa)n\,\log\frac{8n^3(1+4\kappa)^2\nu_{LCP}^2\,\|M\|_\infty^2\,\pi(M)^2\,\mu^0}{\sigma_{LCP}^2}$$

iterations to generate a point x at which the rounding procedure produces a maximally complementary solution. □

6. Concluding remarks. The aim of this paper was to show that one can determine a maximally complementary solution of (LCP) in polynomial time, thus extending a well-known result for LO (cf. Roos, Terlaky, and Vial [27]). We assumed that $\Gamma^0 \neq \emptyset$, $q \neq 0$ and that a starting point $x^{(0)} \in \Gamma^0$ is given. Under these assumptions we could derive the desired result.

A crucial point in the analysis is the convergence rate along the central path of the variables in the index set T, which is $\mathcal{O}(\sqrt{\mu})$. All known proofs of this result use a corollary of Robinson [26] related to the theory of polyhedral multifunctions. In section 3 we presented a new and relatively simple proof.

In the analysis we need two condition numbers for $P_*(\kappa)$ LCPs, both of which appear in the achieved iterations bound. Both numbers were bounded by expressions in the input data. Using Theorems 3.11 and 4.1 we showed that if $x \in \Gamma^0$ is sufficiently close to the central path and $x^T s(x)$ is sufficiently small, then we can identify the optimal partition and compute a maximally complementary solution by using Gaussian

elimination (Theorem 5.1). Similar bounds were presented in [18, 15] to generate a complementary basic solution of (LCP).

The number of iterations to obtain the accuracy necessary to run the rounding procedure is computed for Dikin affine-scaling algorithm [8] in Theorem 5.2. Similar results for other known IPMs can be obtained as well.

REFERENCES

[1] E. D. Andersen and Y. Ye, *Combining interior-point and pivoting algorithms for linear programming*, Management Science, 42 (1996), pp. 1719–1731.

[2] A. B. Berkelaar, B. Jansen, C. Roos, and T. Terlaky, *Basis- and partition identification for quadratic programming and linear complementarity problems*, Math. Programming, 86 (1999), pp. 261–282.

[3] W. Cook, A. M. H. Gerards, A. Schrijver, and É. Tardos, *Sensitivity results in integer linear programming*, Math. Programming, 34 (1986), pp. 251–264.

[4] 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.

[5] R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.

[6] D. den Hertog, C. Roos, and T. Terlaky, *The linear complementarity problem, sufficient matrices and the criss-cross method*, Linear Algebra Appl., 187 (1993), pp. 1–14.

[7] A. J. Hoffman, *On approximate solutions of systems of linear inequalities*, J. Research Nat. Bur. Standards, 49 (1952), pp. 263–265.

[8] T. Illés, C. Roos, and T. Terlaky, *Polynomial affine-scaling algorithms for $P_*(\kappa)$ linear complementarity problems*, in Recent Advances in Optimization, Proceedings of the 8th French–German Conference on Optimization, Trier, 1996, Lecture Notes in Econom. Math. Systems 452, P. Gritzmann, R. Horst, E. Sachs, and R. Tichatschke, eds., Springer-Verlag, New York, 1997, pp. 119–137.

[9] B. Jansen, C. Roos, and T. Terlaky, *A family of polynomial affine scaling algorithms for positive semidefinite linear complementarity problems*, SIAM J. Optim., 7 (1997), pp. 126–140.

[10] J. Ji and F. Potra, *Tapia indicators and finite termination of infeasible-interior-point methods for degenerate LCP*, in Lectures in Appl. Math. 32, 1996, pp. 443–454.

[11] J. Ji, F. Potra, and S. Huang, *A predictor-corrector method for linear complementarity problems with polynomial complexity and superlinear convergence*, J. Optim. Theory Appl., 84 (1995), pp. 187–199.

[12] J. Ji, F. Potra, and R. Sheng, *A predictor-corrector method for the P_*-matrix LCP from infeasible starting points*, Optim. Methods Softw., 6 (1995), pp. 109–126.

[13] L. G. Khachiyan, *A polynomial algorithm in linear programming*, Dokl. Akad. Nauk SSSR, 244 (1979), pp. 1093–1096 (in Russia). Translated into English in Soviet Mathematics Doklady, 20 (1979), pp. 191–194.

[14] E. Klafszky and T. Terlaky, *Az ellipszoid módszerről*, Szigma, 20 (1988), pp. 196–108.

[15] 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.

[16] M. Kojima, S. Mizuno, and A. Yoshise, *An $\mathcal{O}(\sqrt{n}L)$ iteration potential reduction algorithm for linear complementarity problems*, Math. Programming, 50 (1991), pp. 331–342.

[17] M. Kojima, S. Mizuno, and A. Yoshise, *A little theorem of the big $\mathcal{M}$ in interior point algorithms*, Math. Programming, 59 (1993), pp. 361–375.

[18] 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.

[19] M. Kojima, N. Megiddo, and Y. Ye, *An interior point potential reduction algorithm for the linear complementarity problem*, Math. Programming, 54 (1992), pp. 267–279.

[20] P. D. Ling, *A New Proof of Convergence for the New Primal-Dual Affine Scaling Interior-Point Algorithm of Jansen, Roos and Terlaky*, Technical Report, University of East Anglia, Norwich, UK, 1993.

[21] 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.

[22] S. Mehrotra and Y. Ye, *On finding the optimal facet of linear programs*, Math. Programming, 62 (1993), pp. 497–515.

[23] R. D. C. Monteiro and T. Tsuchyia, *Limiting behavior of the derivatives of certain trajectories associated with a monotone horizontal linear complementarity problem*, Math. Oper. Res., 21 (1996), pp. 793–814.
[24] R. D. C. Monteiro and S. Wright, *Local convergence of interior-point algorithms for degenerate monotone LCP*, Comput. Optim. Appl., 3 (1994), pp. 131–155.
[25] J. S. Pang, *Error bounds in mathematical programming*, Math. Programming, 79 (1997), pp. 299–332.
[26] S. M. Robinson, *Some continuity properties of polyhedral multifunctions*, Math. Programming Stud., 14 (1981), pp. 206–214.
[27] C. Roos, T. Terlaky, and J.-Ph. Vial, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley, New York, 1997.
[28] J. Stoer, and M. Wechs, *Infeasible-Interior-Point Paths for Sufficient Linear Complementarity Problems and Their Analyticity*, Math. Programming, 83 (1998), pp. 407–423.
[29] J. Stoer, M. Wechs, and S. Mizuno, *High order infeasible-interior-point methods for solving sufficient linear complementarity problems*, Math. Oper. Res., 23 (1998), pp. 832–862.
[30] H. Väliaho, *P_*-matrices are just sufficient*, Linear Algebra Appl., 239 (1996), pp. 103–108.
[31] S. A. Vavasis and Y. Ye, *Condition numbers for polyhedra with real number data*, Oper. Res. Lett., 17 (1995), pp. 209–214.
[32] Y. Ye and P. M. Pardalos, *A class of linear complementarity problems solvable in polynomial time*, Linear Algebra Appl., 152 (1991), pp. 3–17.
[33] Y. Ye, *On the finite convergence of interior-point algorithms for linear programming*, Math. Programming, 57 (1992), pp. 325–335.
[34] Y. Ye and K. Anstreicher, *On quadratic and $\mathcal{O}(\sqrt{n}L)$ convergence of a predictor-corrector algorithm for LCP*, Math. Programming, 62 (1993), pp. 537–552.
[35] Y. Ye, *Toward probabilistic analysis of interior-point algorithms for linear programming*, Math. Oper. Res., 19 (1994), pp. 38–52.
[36] A. Yoshise, *Complementarity problems*, in Interior Point Methods of Mathematical Programming, T. Terlaky, ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996, pp. 297–367.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 341–363

ON SMOOTHING METHODS FOR THE P_0 MATRIX LINEAR COMPLEMENTARITY PROBLEM*

XIAOJUN CHEN† AND YINYU YE‡

Abstract. In this paper, we propose a Big-Γ smoothing method for solving the P_0 matrix linear complementarity problem. We study the trajectory defined by the augmented smoothing equations and global convergence of the method under an assumption that the original P_0 matrix linear complementarity problem has a solution. The method has been tested on the P_0 matrix linear complementarity problem with unbounded solution set. Preliminary numerical results indicate the robustness of the method.

Key words. linear complementarity problem, P_0 matrix, smoothing algorithm

AMS subject classifications. 65H10, 90C30, 90C33

PII. S1052623498335080

1. Introduction.

In this paper we consider the linear complementarity problem (LCP)

$$t^T s = 0, \quad s = Mt + q, \quad \text{and} \quad t, s \geq 0,$$

where M is an $n \times n$ P_0 matrix and q is an n-dimensional vector. A matrix $M \in R^{n\times n}$ is called a P_0 matrix if

$$\max_{i:t_i\neq 0}\{t_i(Mt)_i\} \geq 0 \quad \text{for all} \quad t \in R^n, t \neq 0.$$

An LCP is called a P_0 matrix LCP if the matrix M is a P_0 matrix. The class of the P_0 matrix LCP includes the monotone LCP and the P matrix LCP. The P_0 matrix LCP has been studied extensively under additional conditions [5, 11].

A differentiable function on R^n is called a P_0 function if its Jacobian is a P_0 matrix at every point in R^n. A nonlinear complementarity problem (NCP) is called a P_0 function NCP if the involved function is a P_0 function. Kojima, Megiddo, and Noma [10] proved the existence of a trajectory in the interior of the feasible set of the P_0 function NCP under some additional conditions. Their results influenced the development of interior point methods and noninterior point methods and led several continuation methods for solving P_0 function NCP.

Recently, Facchinei and Kanzow [6] applied regularization methods for solving a continuously differentiable P_0 function NCP under the following assumption.

Assumption 1.1. The solution set of the P_0 function NCP is nonempty and bounded.

This assumption is weaker than that used by Kojima and colleagues in [10, 11]. Moreover, it includes the monotone NCP with an interior point and the P_0 and R_0 NCP [5]. After Facchinei–Kanzow's encouraging work, several algorithms and

*Received by the editors March 5, 1998; accepted for publication (in revised form) March 7, 2000; published electronically September 27, 2000.

http://www.siam.org/journals/siopt/11-2/33508.html

†Department of Mathematics and Computer Science, Shimane University, Matsue 690-8504, Japan (chen@math.shimane-u.ac.jp). This author was supported in part by the Scientific Research grant from the Ministry of Education, Science, Sport, and Culture of Japan.

‡Department of Management Sciences, University of Iowa, Iowa City, IA (yyye@dollar.biz.uiowa.edu). This author was supported in part by NSF grants DMI-9522507 and DMS-9703490.

theoretical results on regularization methods for the P_0 function NCP have been developed [15, 16, 18] under Assumption 1.1. In particular, Ravindran and Gowda [16] generalized the results of Facchinei and Kanzow [6] to a continuous P_0 function variational inequality problem with box constraints. Facchinei and Kanzow [6] gave a counterexample to show that it is not possible to remove the boundedness assumption of the solution set for regularization methods for solving the P_0 matrix LCP and the P_0 function NCP.

In this paper, we study a "Big-Γ" smoothing method for the P_0 matrix LCP under the following assumption, which removes the boundedness assumption of the solution set from Assumption 1.1.

Assumption 1.2. The P_0 matrix LCP has a solution.

Big-Γ interior point methods have been studied for solving the monotone LCP [12]. The methods add one inequality, with a positive number Γ as the right-hand-side bound, to bound the variables of the problem. If this inequality contains an original solution, then the augmented problem has a solution and it is also a solution to the original problem. One can always set Γ sufficiently big such that the inequality does contain a solution, assuming that it exists. However, the techniques used in Big-Γ interior point methods heavily rely on the monotone property, which cannot be carried over from the monotone LCP to the P_0 matrix LCP. One difference, for example, is that the existence of an interior feasible point implies the bounded solution set for the monotone LCP, but it is not held for the P_0 matrix LCP.

Example 1.1.

$$M = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix} \quad \text{and} \quad q = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

It is not difficult to verify that M is a P_0-matrix and that this LCP has a strictly feasible point $(t, s) = (1, 1, 1, 1, 1, 1)^T$. However, the solution set of the LCP contains the unbounded line $(t, s) = (t_1, 0, 0, 0, 0, 1)^T$ for all $t_1 \geq 0$.

The generalization of Big-Γ methods to the P_0 matrix LCP is nontrivial; see [11]. In order to make the Big-Γ smooth paths and their neighborhood be bounded, we have to slightly destroy the P_0 property. Furthermore, in contrast with the trajectory analysis given by Kojima, Megiddo, and Noma [10], the existence of sufficiently short central path is not guaranteed under Assumption 1.2; see Example 2.1.

In section 2, we establish the existence of the Big-Γ smooth trajectory, which leads to a solution of the problem. In section 3, we propose an algorithm for tracing the trajectory numerically and show the global convergence property of the algorithm. We tested the algorithm on the P_0 matrix LCP with unbounded solution sets. Numerical results reported in section 4 indicate the robustness of the algorithm.

We use $\|\cdot\|$ to denote $\|\cdot\|_\infty$. We use e for a vector with all entries equal to 1 and I for a diagonal matrix with all diagonal entries equal to 1. We denote the solution set of LCP(M, q) by $S_0(M, q)$.

2. A Big-Γ smoothing model. Let

$$N = \begin{pmatrix} M & r & 0 \\ 0 & 1 & 0 \\ -e^T & -1 & -1 \end{pmatrix}, \qquad p = \begin{pmatrix} q \\ 0 \\ \Gamma \end{pmatrix},$$

where $r = e - Me - q$ and $\Gamma \geq n + 5$ is sufficiently big.

Let $x = (t, \theta, \alpha) \in R^{n+2}$ and $y = (s, \eta, \beta) \in R^{n+2}$. We consider the LCP$(N, p)$

$$x^T y = 0, \quad y = Nx + p, \quad \text{and} \quad x, y \geq 0.$$

By the construction of the model, we have the following lemma whose simple proof is omitted.

LEMMA 2.1.

1. *LCP(N, p) has a feasible interior point*

$$t = e, \quad \theta = 1, \quad \alpha = 1,$$
$$s = e, \quad \eta = 1, \quad \beta = \Gamma - (n+2) \geq 1.$$

2. *If (t^*, s^*) is a solution of LCP(M, q) with $e^T t^* \leq \Gamma$, then*

$$(t^*, 0, \Gamma - e^T t^*, s^*, 0, 0) \quad \text{and} \quad (t^*, 0, 0, s^*, 0, \Gamma - e^T t^*)$$

are solutions of LCP(N, p).

3. *If LCP(N, p) has a solution, then we have that in every complementarity solution $(t^*, \theta^*, \alpha^*, s^*, \eta^*, \beta^*)$ of LCP(N, p), (t^*, s^*) is a solution of LCP(M, q), $\eta^* = \theta^* = 0$ and $\alpha^* + \beta^* = \Gamma - e^T t^*$.*

4. *The feasible set of LCP(N, p) is bounded.*

Notice that N is not a P_0 matrix, since $N_{n+2,n+2} = -1$. Although we can easily construct a P_0 matrix that satisfies Results 1–3 of Lemma 2.1, e.g., set $N_{n+2,n+2} = 1$, the resulting LCP may have an unbounded solution set. The authors believe that it is hard to construct a Big-Γ model for the P_0 matrix LCP that has both the P_0 property and the boundedness of the solution set. This contrasts with the monotone LCP, for which we always can construct a Big-Γ model having a bounded solution set without loss of the monotone property [12, 20].

Nevertheless, the matrix N is a block lower triangular matrix and its first block is an $(n+1) \times (n+1)$ P_0 matrix; i.e.,

$$N = \begin{pmatrix} \tilde{N} & 0 \\ -e^T & -1 \end{pmatrix},$$

where

$$\tilde{N} := \begin{pmatrix} M & r \\ 0 & 1 \end{pmatrix}.$$

We will often use this fact later.

In what follows, for simplicity, we use $z := (x, y)$. It is easy to verify that the LCP(N, p) is equivalent to the following system of nonsmooth equations:

$$H_0(z) := \begin{pmatrix} Nx + p - y \\ x - \max(x - y, 0) \end{pmatrix} = 0. \tag{2.1}$$

To define a smoothing approximation function of H_0, we employ two density functions

$$\rho_1(\mu) = \frac{2}{(\mu^2 + 4)^{\frac{3}{2}}}$$

and

$$\rho_2(\mu) = \begin{cases} \frac{1}{4} & \text{if } -4 \le \mu \le 0, \\ 0 & \text{otherwise.} \end{cases}$$

Let

$$\Psi_i(x_i, y_i, \varepsilon) := x_i - \int_{-\infty}^{(x_i - y_i)/\varepsilon} (x_i - y_i - \varepsilon\mu)\rho_1(\mu)d\mu, \qquad i = 1, 2, \ldots, n+1,$$

and

$$\Psi_{n+2}(x_{n+2}, y_{n+2}, \varepsilon) := x_{n+2} - \int_{-\infty}^{(x_{n+2} - y_{n+2})/\varepsilon} (x_{n+2} - y_{n+2} - \varepsilon\mu)\rho_2(\mu)d\mu.$$

Calculating the integral, we obtain

$$\Psi_i(x_i, y_i, \varepsilon) = \frac{1}{2}\left(x_i + y_i - \sqrt{(x_i - y_i)^2 + 4\varepsilon^2}\right), \qquad i = 1, 2, \ldots, n+1,$$

and

$$\Psi_{n+2}(x_{n+2}, y_{n+2}, \varepsilon) = \begin{cases} x_{n+2}, & x_{n+2} - y_{n+2} \le -4\varepsilon, \\ y_{n+2} - 2\varepsilon, & x_{n+2} - y_{n+2} \ge 0, \\ y_{n+2} - \frac{1}{8\varepsilon}(x_{n+2} - y_{n+2})^2 - 2\varepsilon & \text{otherwise.} \end{cases}$$

Each component of Ψ is continuously differentiable [3]. Moreover, by Lemma 2.4 in [8], for $i = 1, 2, \ldots, n+1$,

$$|\Psi_i(x_i, y_i, \varepsilon_1) - \Psi_i(x_i, y_i, \varepsilon_2)| \le \left(\int_{-\infty}^{\infty} |\mu|\rho_1(\mu)d\mu\right)|\varepsilon_1 - \varepsilon_2| = 2|\varepsilon_1 - \varepsilon_2|,$$

and for $i = n+2$,

$$|\Psi_i(x_i, y_i, \varepsilon_1) - \Psi(x_i, y_i, \varepsilon_2)| \le \left(\int_{-\infty}^{\infty} |\mu|\rho_2(\mu)d\mu\right)|\varepsilon_1 - \varepsilon_2| = 2|\varepsilon_1 - \varepsilon_2|.$$

Therefore, letting $\varepsilon > 0$, we see that the smoothing approximation function defined by

$$H(z, \varepsilon) := \begin{pmatrix} Nx + p - y \\ \Psi(x, y, \varepsilon) \end{pmatrix}$$

is continuously differentiable in $R^{2(n+2)}$. Moreover,

$$\|H(z, \varepsilon_1) - H(z, \varepsilon_2)\| \le 2|\varepsilon_1 - \varepsilon_2| \quad \text{for} \quad z \in R^{2(n+2)} \tag{2.2}$$

and

$$\|H(z, \varepsilon) - H_0(z)\| \le 2\varepsilon \quad \text{for} \quad z \in R^{2(n+2)}. \tag{2.3}$$

We will show that for every $\varepsilon > 0$ the system

$$H(z, \varepsilon) = 0 \tag{2.4}$$

has at most two solutions, and the two solutions differ possibly in the (x_{n+2}, y_{n+2}) components. Under certain conditions, the solutions of (2.4) form two paths that never cross each other and converge to two solutions of (2.1).

To study the existence of the trajectory, we first consider the following closed set:

$$D(\varepsilon) = \{z \ : \ \|H(z,\varepsilon)\| \le c\varepsilon\},$$

where $c > 2\sqrt{2(n+2)}$ is a constant.

LEMMA 2.2. *Suppose that LCP(N,p) has a solution z^*. Then for every $\varepsilon > 0$, the set $D(\varepsilon)$ is nonempty and compact. Moreover, for every $z \in D(\varepsilon)$,*

$$x_i + c\varepsilon \ge 0, \qquad y_i + c\varepsilon \ge 0, \tag{2.5}$$

and

$$(x_i + c\varepsilon)(y_i + c\varepsilon) \le 2(c+1)(|x_i| + |y_i|)\varepsilon + (4 + c^2)\varepsilon^2 \tag{2.6}$$

for $i = 1, 2, \ldots, n+2$.

Proof. At the solution z^*, we have

$$\|H(z^*,\varepsilon)\| \le \|H(z^*,\varepsilon) - H_0(z^*)\| + \|H_0(z^*)\| = 2\varepsilon.$$

Hence $z^* \in D(\varepsilon)$ and so $D(\varepsilon)$ is nonempty.

Suppose $z \in D(\varepsilon)$. Then we have

$$\|H(z,\varepsilon)\| = \left\| \begin{pmatrix} Nx + p - y \\ \Psi(x,y,\varepsilon) \end{pmatrix} \right\| \le c\varepsilon.$$

Set $u = \Psi(x,y,\varepsilon)$. Then $u \in R^{n+2}$ satisfies

$$c\varepsilon \ge u_i \ge -c\varepsilon, \qquad i = 1, 2, \ldots, n+2.$$

By construction of Ψ (cf. Lemma 2.1 in [2]), we have

$$\Psi(x-u, y-u, \varepsilon) = 0. \tag{2.7}$$

Since ρ_1 is continuous, symmetric, and has an infinite support, by Theorem 2.1 in [4], we have

$$x_i - u_i > 0, \quad y_i - u_i > 0, \quad i = 1, 2, \ldots, n+1.$$

Using $c\varepsilon \ge -u_i$, we obtain (2.5) for $i = 1, 2, \ldots, n+1$. Now we show (2.5) for $i = n+2$. By the definition of Ψ and (2.7), we have the following equalities:

$$\begin{aligned} 0 &= \Psi_{n+2}(x_{n+2} - u_{n+2}, y_{n+2} - u_{n+2}, \varepsilon) \\ &= x_{n+2} - u_{n+2} - \int_{-\infty}^{(x_{n+2} - y_{n+2})/\varepsilon} (x_{n+2} - y_{n+2} - \varepsilon\mu)\rho_2(\mu)d\mu. \end{aligned}$$

This implies that $x_{n+2} - u_{n+2} \ge 0$, since the integral part is nonnegative. To show $y_{n+2} - u_{n+2} \ge 0$, we assume on the contrary that $y_{n+2} < u_{n+2}$. Then $x_{n+2} - y_{n+2} > x_{n+2} - u_{n+2} \ge 0$ and

$$\begin{aligned} x_{n+2} - u_{n+2} &= \int_{-\infty}^{(x_{n+2} - y_{n+2})/\varepsilon} (x_{n+2} - y_{n+2} - \varepsilon\mu)\rho_2(\mu)d\mu \\ &= \frac{1}{4}\int_{-4}^{0} (x_{n+2} - y_{n+2} - \varepsilon\mu)d\mu \\ &= x_{n+2} - y_{n+2} + 2\varepsilon, \end{aligned}$$

which implies

$$y_{n+2} - u_{n+2} = 2\varepsilon > 0.$$

This contradicts the assumption that $y_{n+2} - u_{n+2} < 0$. Hence we have

$$x_{n+2} - u_{n+2} \geq 0 \quad \text{and} \quad y_{n+2} - u_{n+2} \geq 0.$$

Using $c\varepsilon \geq -u_{n+2}$, we obtain (2.5) for $i = n + 2$.

Now we show (2.6). Suppose that $x_i \leq y_i$. Then we have

$$x_i - u_i - \max(x_i - y_i, 0) = \min(x_i - u_i, y_i - u_i) = x_i - u_i.$$

From (2.3) and (2.7),

$$x_i - u_i = |\Psi_i(x_i - u_i, y_i - u_i, \varepsilon) - (x_i - u_i)| \leq 2\varepsilon.$$

By a simple manipulation, we obtain

$$\begin{aligned}
&(x_i + c\varepsilon)(y_i + c\varepsilon) \\
&= (x_i - u_i)(y_i - u_i) + (x_i + y_i)(u_i + c\varepsilon) - u_i^2 + c^2\varepsilon^2 \\
&= (x_i - u_i)(y_i - x_i) + (x_i - u_i)^2 + |x_i + y_i|(u_i + c\varepsilon) - u_i^2 + c^2\varepsilon^2 \\
&\leq 2(|x_i| + |y_i|)\varepsilon + 4\varepsilon^2 + 2|x_i + y_i|c\varepsilon + c^2\varepsilon^2 \\
&\leq 2(1 + c)(|x_i| + |y_i|)\varepsilon + (4 + c^2)\varepsilon^2,
\end{aligned}$$

where the first inequality follows from

$$0 \leq x_i - u_i \leq 2\varepsilon \quad \text{and} \quad -c\varepsilon \leq u_i \leq c\varepsilon.$$

The proof is similar for the case $x_i \geq y_i$.

Now we show $D(\varepsilon)$ is bounded. Using

$$c\varepsilon \geq H_{n+2}(z, \varepsilon) \geq -c\varepsilon,$$

we have

$$\begin{aligned}
c\varepsilon &\geq (Nx + p - y)_{n+2} \\
&= -e^T x + \Gamma - y_{n+2} \\
&\geq -c\varepsilon.
\end{aligned}$$

This implies

$$\Gamma + c\varepsilon \geq e^T x + y_{n+2} \geq \Gamma - c\varepsilon.$$

Since x_i and y_i are bounded below by (2.5), x and y cannot go to ∞, and so $D(\varepsilon)$ is bounded. □

THEOREM 2.3. *Suppose that LCP(N, p) has a solution z^*. Then for every $\varepsilon > 0$, there is a $z_\varepsilon \in R^{2(n+2)}$ such that*

$$H'(z_\varepsilon, \varepsilon)^T H(z_\varepsilon, \varepsilon) = 0 \tag{2.8}$$

and

$$\|H(z_\varepsilon, \varepsilon)\| \leq c\varepsilon. \tag{2.9}$$

Proof. By Lemma 2.2, for every $\varepsilon > 0$, $D(\varepsilon)$ is nonempty and bounded. Let

$$\theta(z) = \frac{1}{2}\|H(z,\varepsilon)\|_2^2.$$

Since $H(\cdot,\varepsilon)$ is continuously differentiable and $D(\varepsilon)$ is nonempty and compact, θ has a global minimum point z_ε in $D(\varepsilon)$. Recalling that $\|\cdot\| := \|\cdot\|_\infty$, $z^* \in D(\varepsilon)$, and $H_0(z^*) = 0$, we have

$$\begin{aligned}
\|H(z_\varepsilon,\varepsilon)\| &\le \|H(z_\varepsilon,\varepsilon)\|_2 \\
&\le \|H(z^*,\varepsilon)\|_2 \\
&\le \sqrt{2(n+2)}\|H(z^*,\varepsilon)\| \\
&\le \sqrt{2(n+2)}(\|H_0(z^*)\| + 2\varepsilon) \\
&= 2\varepsilon\sqrt{2(n+2)} \\
&< c\varepsilon.
\end{aligned}$$

Hence, (2.9) holds. Moreover, this implies $z_\varepsilon \in \text{int } D(\varepsilon)$. By Theorem 4.1.3 in [13], we have

$$\theta'(z_\varepsilon) = H'(z_\varepsilon,\varepsilon)^T H(z_\varepsilon,\varepsilon) = 0.$$

This completes the proof. □

Theorem 2.3, together with Lemma 2.1, shows that if the LCP(M,q) has a solution, then there is a Γ such that the LCP(N,p) has a solution and for every $\varepsilon > 0$ the system (2.8) has a solution. Clearly, if $H'(z_\varepsilon,\varepsilon)$ is nonsingular, then z_ε is a solution of (2.4). Now we give a necessary and sufficient condition for the nonsingularity.

LEMMA 2.4. *Let M be a P_0 matrix. Then $H'(z,\varepsilon)$ is nonsingular at $z \in R^{2(n+2)}$ if and only if $x_{n+2} - y_{n+2} \neq -2\varepsilon$.*

Proof. Let

$$d_i(x_i - y_i,\varepsilon) = \int_{-\infty}^{(x_i-y_i)/\varepsilon} \rho_1(\mu)d\mu, \qquad i = 1,2,\ldots,n+1$$

and

$$d_{n+2}(x_{n+2} - y_{n+2},\varepsilon) = \int_{-\infty}^{(x_{n+2}-y_{n+2})/\varepsilon} \rho_2(\mu)d\mu.$$

Let

$$D_{n+1}(x-y,\varepsilon) = \text{diag}(d_1(x_1 - y_1,\varepsilon),\ldots,d_{n+1}(x_{n+1} - y_{n+1},\varepsilon))$$

and

$$D(x-y,\varepsilon) = \text{diag}(D_{n+1}(x-y,\varepsilon), d_{n+2}(x_{n+2} - y_{n+2},\varepsilon)).$$

By the definition of $H(z,\varepsilon)$,

$$H'(z,\varepsilon) = \begin{pmatrix} N & -I \\ I - D(x-y,\varepsilon) & D(x-y,\varepsilon) \end{pmatrix}.$$

It is well known that $H'(z,\varepsilon)$ is nonsingular if and only if $I - D(x-y,\varepsilon)(I-N)$ is nonsingular.

Notice that $\tilde{N}$ is a P_0 matrix, and

$$I - D(x-y,\varepsilon)(I-N) = \begin{pmatrix} I_{n+1} - D_{n+1}(x-y,\varepsilon)(I_{n+1}-\tilde{N}) & 0 \\ -d_{n+2}(x_{n+2}-y_{n+2},\varepsilon)e^T & 1-2d_{n+2}(x_{n+2}-y_{n+2},\varepsilon) \end{pmatrix}.$$

Since $\mathrm{supp}\{\rho_1\} = R$ and $\tilde{N}$ is a P_0 matrix, $I_{n+1} - D_{n+1}(x-y,\varepsilon)(I_{n+1}-\tilde{N})$ is nonsingular [8].

Hence $H'(z,\varepsilon)$ is nonsingular if and only if $d_{n+2}(x_{n+2}-y_{n+2},\varepsilon) \neq \frac{1}{2}$. By the definition of $d_{n+2}(x_{n+2}-y_{n+2},\varepsilon)$, we have $d_{n+2}(x_{n+2}-y_{n+2},\varepsilon) \neq \frac{1}{2}$ if and only if $x_{n+2} - y_{n+2} \neq -2\varepsilon$. This completes the proof of the lemma. □

LEMMA 2.5. *Suppose that M is a P_0 matrix. Then for every $\varepsilon > 0$, (2.4) has at most two solutions, and any two solutions differ possibly in the (x_{n+2}, y_{n+2}) components. Moreover, a solution z_ε of (2.4) is unique if and only if $H'(z_\varepsilon,\varepsilon)$ is singular.*

Proof. Let $\tilde{x} = (x_1,\ldots,x_{n+1}), \tilde{y} = (y_1,\ldots,y_{n+1}), \tilde{z} = (\tilde{x},\tilde{y})$, and

$$\tilde{\Psi}(\tilde{z},\varepsilon) = (\Psi_1(x_1,y_1,\varepsilon),\ldots,\Psi_{n+1}(x_{n+1},y_{n+1},\varepsilon))^T.$$

That is, $\tilde{x}$, $\tilde{y}$, and $\tilde{\Psi}$ are the first $n+1$ components of x, y, and Ψ, respectively. If z is a solution of (2.4), then $\tilde{z}$ is a solution of

$$\begin{pmatrix} \tilde{N}\tilde{x} + p - \tilde{y} \\ \tilde{\Psi}(\tilde{z},\varepsilon) \end{pmatrix} = 0. \tag{2.10}$$

Since $\tilde{N}$ is a P_0 matrix and $\tilde{\Psi}$ is given by ρ_1, by Theorem 2.3 in [4], $\tilde{z}_\varepsilon$ is the unique solution of (2.10). Hence any two solutions of (2.4) differ possibly at the (x_{n+2}, y_{n+2}) components.

Now we show (2.4) has at most two solutions. Since a solution $\tilde{z}$ of (2.10) is unique, we only need to show that the system of the remaining equations in (2.4),

$$\begin{pmatrix} \Gamma - e^T\tilde{x} - x_{n+2} - y_{n+2} \\ \Psi_{n+2}(x_{n+2},y_{n+2},\varepsilon) \end{pmatrix} = 0,$$

has at most two solutions.

Substituting $y_{n+2} = \Gamma - e^T\tilde{x} - x_{n+2}$ into the second equation, we obtain

$$\psi(x_{n+2}) := \Psi_{n+2}(x_{n+2}, \Gamma - e^T\tilde{x} - x_{n+2}, \varepsilon) = 0.$$

The function $\psi : R \to R$ is a polynomial of degree 2 in the interval

$$\left[\frac{1}{2}(\Gamma - e^T\tilde{x}) - 2\varepsilon,\ \frac{1}{2}(\Gamma - e^T\tilde{x})\right]$$

and linear outside of this interval. Furthermore, ψ is monotonically decreasing in

$$\left[\frac{1}{2}(\Gamma - e^T\tilde{x}) - \varepsilon,\ \infty\right)$$

and monotonically increasing in

$$\left(-\infty,\ \frac{1}{2}(\Gamma - e^T\tilde{x}) - \varepsilon\right].$$

Therefore, ψ has at most two zero points. Moreover, a zero point x_{n+2} of ψ is unique if and only if

$$x_{n+2} = \frac{1}{2}(\Gamma - e^T\tilde{x}) - \varepsilon = \frac{1}{2}(y_{n+2} + x_{n+2}) - \varepsilon.$$

Hence the system of (2.4) has at most two solutions, and a solution of (2.4) is unique if and only if $x_{n+2} - y_{n+2} = -2\varepsilon$. By Lemma 2.4, a solution of (2.4) is unique if and only if H' is singular at this solution. □

LEMMA 2.6. *Suppose that M is a P_0 matrix and the solution set of the LCP(M,q) is nonempty and bounded. Then there exist $\Gamma > 0$ and $\varepsilon^0 > 0$ such that LCP(N,p) has a solution and for every $\varepsilon \in (0,\varepsilon^0]$, $H'(z,\varepsilon)$ are nonsingular for all $z \in D(\varepsilon^0)$.*

Proof. Since the solution set $S_0(M,q)$ is bounded, we can choose $\Gamma > 0$ satisfying

$$\Gamma > 4e^T t \qquad \text{for all} \quad (t,s) \in S_0(M,q). \tag{2.11}$$

Then from Lemma 2.1, the solution set of the LCP(N,p) is given by

$$S_0(N,q) = \{(t,0,\Gamma - e^T t, s, 0, 0), (t,0,0,s,0,\Gamma - e^T t) : \ (t,s) \in S_0(M,q)\}.$$

Hence for a solution $(t^*,s^*) \in S_0(M,q)$, $z^{*,1} = (t^*,0,\Gamma - e^T t^*, s^*, 0, 0)$ and $z^{*,2} = (t^*,0,0,\ s^*,0,\Gamma - e^T t^*)$ are solutions of LCP(N,p) and

$$\max_{z\in S_0(N,p)} \min(|x-y-x^{*,1}+y^{*,1}|_{n+2}, |x-y-x^{*,2}+y^{*,2}|_{n+2}) = \max_{z\in S_0(N,p)} |e^T(t^*-t)| \le \frac{\Gamma}{4}.$$

By the continuity of $H(z,\varepsilon)$ on ε, for such Γ there exists $\varepsilon^0 \in (0, \frac{\Gamma}{8})$ such that for all $z \in D(\varepsilon^0)$,

$$\max_{z\in D(\varepsilon^0)} \min(|x - y - x^{*,1} + y^{*,1}|_{n+2}, |x - y - x^{*,2} + y^{*,2}|_{n+2}) \le \frac{\Gamma}{2}.$$

Let $z \in D(\varepsilon^0)$. Without loss of generality we may assume $|x-y-x^{*,1}+y^{*,1}|_{n+2} \le |x - y - x^{*,2} + y^{*,2}|_{n+2}$. Then

$$\begin{aligned}
&|x_{n+2} - y_{n+2}| \\
&\ge |x^{*,1}_{n+2} - y^{*,1}_{n+2}| - |x_{n+2} - y_{n+2} - x^{*,1}_{n+2} + y^{*,1}_{n+2}| \\
&\ge \Gamma - e^T t^* - \max_{z\in D(\varepsilon^0)} \min(|x - y - x^{*,1} + y^{*,1}|_{n+2}, |x - y - x^{*,2} + y^{*,2}|_{n+2}) \\
&\ge \Gamma - \frac{\Gamma}{4} - \frac{\Gamma}{2} = \frac{\Gamma}{4} > 2\varepsilon^0.
\end{aligned}$$

By Lemma 2.4, $H'(z,\varepsilon)$ is nonsingular for $\varepsilon \in (0,\varepsilon^0]$ and $z \in D(\varepsilon^0)$. □

THEOREM 2.7. *Suppose that M is a P_0 matrix and the solution set of the LCP(M,q) is nonempty and bounded. Then there exist $\Gamma > 0$ and $\varepsilon^0 > 0$ such that*

1. *for every $\varepsilon \in (0,\varepsilon^0]$, the system (2.4) has only two solutions $z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$, which are continuous in ε and never cross each other;*
2. *$z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$ converge to two solutions of LCP(N,q) as $\varepsilon \to 0$.*

Proof. 1. By Theorem 2.3 and Lemma 2.6 there exist $\Gamma > 0$ satisfying (2.11) and $\varepsilon^0 > 0$ such that LCP(N,p) has a solution and for every $\varepsilon \in (0,\varepsilon^0]$, the system (2.4) has a solution in $D(\varepsilon^0)$. Then using Lemma 2.5, the system (2.4) has only two solutions $z_\alpha(\varepsilon), z_\beta(\varepsilon) \in D(\varepsilon^0)$, and $H'(z_\alpha(\varepsilon),\varepsilon)$ and $H'(z_\beta(\varepsilon),\varepsilon)$ are nonsingular.

By the implicit Theorem 5.2.4 in [13], $z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$ are continuous in $\varepsilon \in (0, \varepsilon^0]$.

Now we show that $z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$ never cross each other. Assume on the contrary that there is an $\tilde{\varepsilon} \in (0, \varepsilon^0]$ such that $z_\alpha(\tilde{\varepsilon}) = z_\beta(\tilde{\varepsilon})$. Then by Lemma 2.5, $H'(z_\alpha(\tilde{\varepsilon}), \tilde{\varepsilon})$ is singular. This is a contradiction, since $z_\alpha(\tilde{\varepsilon}) \in D(\varepsilon^0)$ and for every $z \in D(\varepsilon^0)$, $H'(z, \varepsilon)$ is nonsingular. Hence $z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$ never cross each other, which forms two paths.

2. Since $z_\alpha(\varepsilon)$, $z_\beta(\varepsilon) \subset D(\varepsilon^0)$, and $D(\varepsilon^0)$ is bounded, $z_\alpha(\varepsilon)$ and $z_\beta(\varepsilon)$ has limiting points, respectively, as $\varepsilon \to 0$. By (2.9) in Theorem 2.3, every limiting point is a solution of LCP(N, p). We assume that for some sequence $\epsilon_k \to 0$, $z_\alpha(\varepsilon_k) \to z^*_\alpha$, and $z_\beta(\varepsilon_k) \to z^*_\beta$.

From $H(z_\alpha(\varepsilon), \varepsilon) = H(z_\beta(\varepsilon), \varepsilon) = 0$, we have

$$(x_\alpha(\varepsilon), y_\alpha(\varepsilon)) := z_\alpha(\varepsilon) \geq 0, \qquad (x_\beta(\varepsilon), y_\beta(\varepsilon)) := z_\beta(\varepsilon) \geq 0$$

and

$$(x_\alpha(\varepsilon))_i (y_\alpha(\varepsilon))_i = (x_\beta(\varepsilon))_i (y_\beta(\varepsilon))_i = \varepsilon^2, \qquad i = 1, 2, \ldots, n+1.$$

Moreover, without loss of generality, we may assume

$$(x_\alpha(\varepsilon))_{n+2} \geq \frac{\Gamma}{4} + 2\varepsilon, \qquad (y_\alpha(\varepsilon))_{n+2} \leq 2\varepsilon$$

and

$$(x_\beta(\varepsilon))_{n+2} \leq 2\varepsilon, \qquad (y_\beta(\varepsilon))_{n+2} \geq \frac{\Gamma}{4} + 2\varepsilon.$$

(See the proof of Lemmas 2.2 and 2.6.)

Consider the two sets

$$E_1 := \Big\{ (x, y, \varepsilon) : \varepsilon > 0, x \geq 0, y = Nx + p \geq 0, x_{n+2} \geq \frac{\Gamma}{4} + 2\varepsilon, y_{n+2} \leq 2\varepsilon, \\ x_i y_i = \varepsilon^2, i = 1, 2, \ldots, n+1 \Big\}$$

and

$$E_2 := \Big\{ (x, y, \varepsilon) : \varepsilon > 0, x \geq 0, y = Nx + p \geq 0, x_{n+2} \leq 2\varepsilon, y_{n+2} \geq \frac{\Gamma}{4} + 2\varepsilon, \\ x_i y_i = \varepsilon^2, i = 1, 2, \ldots, n+1 \Big\}.$$

The sets E_1 and E_2 are semialgebraic and $E_1 \cap E_2 = \emptyset$. Furthermore, $z_\alpha(\varepsilon) \subset E_1, z^*_\alpha(\varepsilon) \in E_1, z_\beta(\varepsilon) \subset E_2$, and $z^*_\beta(\varepsilon) \in E_2$. Hence by the similar argument in the proof of Theorem 5.2 in [19] (also see the proof of Theorem 4.4 in [10]), we claim that $z_\alpha(\varepsilon) \to z^*_\alpha(\varepsilon)$ and $z_\beta(\varepsilon) \to z^*_\beta(\varepsilon)$, as $\varepsilon \to 0$. □

Now we consider a special class of the P_0 matrix LCP whose solution set is not empty. Let

$$M = \begin{pmatrix} M_1 & M_{12} \\ 0 & M_2 \end{pmatrix}, \qquad q = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}, \tag{2.12}$$

where $M_1 \in R^{n_1 \times n_1}$ is a P_0 matrix, $M_{12} \in R^{n_1 \times n_2}$, $M_2 \in R^{n_2 \times n_2}$ is a monotone matrix, $q_1 \in R^{n_1}$, $q_2 \in R^{n_2}$, and $n_1 + n_2 = n$.

It is easy to see that M is a P_0 matrix and in every solution $(t_1^*, t_2^*, s_1^*, s_2^*)$ of the LCP(M, q), $(t_2^*, s_2^*) \in S_0(M_2, q_2)$.

We denote the set of all maximal complementarity solutions (the number of positive components in t_2 and s_2 is maximal) of LCP(M_2, q_2) by $\hat{S}_0(M_2, q_2)$. If LCP(M_2, q_2) has a strictly complementarity solution, then $\hat{S}_0(M_2, q_2)$ is the set of all strictly complementarity solutions of LCP(M_2, q_2).

Let us use the standard index set notation

$$T = \{i : \ (t_2^*)_i > 0 = (s_2^*)_i = (M_2 t_2^* + q_2)_i, (t_2^*, s_2^*) \in \hat{S}_0(M_2, q_2)\},$$
$$S = \{i : \ (t_2^*)_i = 0 < (s_2^*)_i = (M_2 t_2^* + q_2)_i, (t_2^*, s_2^*) \in \hat{S}_0(M_2, q_2)\}.$$

To simplify illustration, we assume $T \cup S = \{1, 2, \ldots, n_2\}$; i.e., LCP$(M_2, q_2)$ has a strictly complementarity solution.

Assumption 2.1.

(i) M is a matrix defined by (2.12).

(ii) LCP(M_2, q_2) has a strictly complementarity solution, and $S_0(M_2, q_2)$ is bounded.

(iii) For every $(t_2^*, s_2^*) \in \hat{S}_0(M_2, q_2)$, LCP$(M_1, M_{12} t_2^* + q_1)$ has a solution and

$$\hat{S}_0(M, q) := \{(t_1^*, t_2^*, s_1^*, s_2^*) : (t_1^*, s_1^*) \in S_0(M_1, M_{12} t_2^* + q_1), (t_2^*, s_2^*) \in \hat{S}_0(M_2, q_2)\}$$

is bounded.

(iv) $(r_2)_{i \in T} = (e - M_2 e - q_2)_{i \in T} = 0$.

Example 1.1 satisfies Assumption 2.1 here:

$$M_1 = (0), \quad M_{12} = (1, 0), \quad q_1 = (0),$$
$$M_2 = \begin{pmatrix} 0 & 1 \\ -1 & 1 \end{pmatrix}, \qquad q_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$
$$S_0(M_2, q_2) = \{(\tau, 0, 0, 1 - \tau) : \ 0 \le \tau \le 1\},$$
$$\hat{S}_0(M_2, q_2) = \{(\tau, 0, 0, 1 - \tau) : \ 0 < \tau < 1\},$$
$$S_0(M, q) = \{(t_1, 0, 0, 0, 0, 1) : \ t_1 \ge 0\} \cup \{(0, t_2, 0, t_2, 0, 1 - t_2) : \ 0 < t_2 \le 1\},$$
$$\hat{S}_0(M, q) = \{(0, t_2, 0, t_2, 0, 1 - t_2) : \ 0 < t_2 < 1\},$$

and $T = \{1\}$.

Although the solution set $S_0(M, q)$ is unbounded, its subset $\hat{S}_0(M, q)$ is bounded.

THEOREM 2.8. *Under Assumption* 2.1*, the conclusion of Theorem* 2.7 *holds.*

Proof. We first consider the problem LCP(M_2, q_2). By Theorem 2.7, there exist Γ_2 and ε_2^0 such that two paths $z_{\alpha_2}(\varepsilon)$ and $z_{\beta_2}(\varepsilon)$ for $\varepsilon \in (0, \varepsilon_2^0]$ exist and converge to two solutions of the corresponding problem LCP(N_2, p_2),

$$z_{\alpha_2} = (\bar{t}_2, 0, \Gamma_2 - e^T \bar{t}_2, \bar{s}_2, 0, 0) \quad \text{and} \quad z_{\beta_2} = (\bar{t}_2, 0, 0, \bar{s}_2, 0, \Gamma_2 - e^T \bar{t}_2).$$

By Lemma 2.1, $(\bar{t}_2, \bar{s}_2)$ is a solution of LCP(M_2, q_2).

Now we show $(\bar{t}_2, \bar{s}_2)$ is a strictly complementarity solution. Choose any element $(t_2^*, s_2^*) \in \hat{S}_0(M_2, q_2)$.

By Lemma 2.5, for every $\varepsilon \in (0, \varepsilon_2]$, the two solutions of (2.4) are identical in the (t_2, s_2) components. Moreover, by Lemma 2.1 in [9] and the construction of the Big-Γ model, we have

$$\theta_2 = \beta_2 = \varepsilon$$

and

$$(t_2)_i(s_2 + \varepsilon r_2)_i = (t_2)_i(M_2 t_2 + q_2 + \varepsilon r_2)_i = \varepsilon^2.$$

Hence, by (iv) of Assumption 2.1,

$$\begin{aligned}
&\sum_{i\in T} \varepsilon^2 + \varepsilon^2 \sum_{i\in S} \frac{(s_2)_i}{(s_2 + \varepsilon r_2)_i} \\
&= \sum_{i\in T} (t_2)_i(s_2 + \varepsilon r_2)_i + \sum_{i\in S} \frac{(t_2)_i(s_2)_i(s_2 + \varepsilon r_2)_i}{(s_2 + \varepsilon r_2)_i} \\
&= t_2^T s_2 \geq s_2^T t_2^* + t_2^T s_2^* \\
&= \sum_{i\in T} (s_2)_i(t_2^*)_i + \sum_{i\in S} (t_2)_i(s_2^*)_i \\
&= \sum_{i\in T} \frac{(s_2 + \varepsilon r_2)_i(t_2)_i(t_2^*)_i}{(t_2)_i} + \sum_{i\in S} \frac{(s_2 + \varepsilon r_2)_i(t_2)_i(s_2^*)_i}{(s_2 + \varepsilon r_2)_i} \\
&= \varepsilon^2 \sum_{i\in T} \frac{(t_2^*)_i}{(t_2)_i} + \varepsilon^2 \sum_{i\in S} \frac{(s_2^*)_i}{(s_2 + \varepsilon r_2)_i},
\end{aligned}$$

where the inequality uses the monotone property of M_2. Therefore, we have

$$\sum_{i\in T} \frac{(t_2^*)_i}{(t_2)_i} + \sum_{i\in S} \frac{(s_2^* - s_2)_i}{(s_2 + \varepsilon r_2)_i} \leq \sum_{i\in T} 1.$$

This implies that $(t_2)_{i\in T}$ and $(s_2)_{i\in S}$ cannot go to zero. Hence $(\bar{t}_2, \bar{s}_2) \in \hat{S}_0(M_2, q_2)$.

Now we claim that there exist Γ satisfying

$$\Gamma > 4e^T t \quad \text{for all} \quad t \in \hat{S}_0(M, q) \tag{2.13}$$

and $\varepsilon^0 > 0$ such that the conclusion of Theorem 2.7 holds. Indeed, since M is block triangular, we claim that the system

$$\hat{H}(t_1, x_{n+2}, s_1, y_{n+2}, \varepsilon) := \begin{pmatrix} M_1 t_1 + M_{12} t_2 + \varepsilon r_1 + q_1 - s_2 \\ -e^T(t_1 + t_2) - \varepsilon - x_{n+2} - y_{n+2} + \Gamma \\ \Psi_1((t_1 - s_1)_1, \varepsilon) \\ \cdots \\ \Psi_{n_1}((t_1 - s_1)_{n_1}, \varepsilon) \\ \Psi_{n+2}(x_{n+2} - y_{n+2}, \varepsilon) \end{pmatrix} = 0$$

has two solutions for all $\varepsilon \in (0, \varepsilon^0]$.

Assume on the contrary that this claim is not true. By (iii) of Assumption 2.1, the solution set $S_0(M_1, M_{12}\bar{t}_2 + q_1)$ is nonempty and bounded. There is $\varepsilon_0 > 0$ such

that $S_0(M_1, M_{12}t_2 + q_1)$ is nonempty for all t_2 satisfying $\|t_2 - \bar{t}_2\| \le \varepsilon_0$ [7, 16]. Hence by Theorem 2.3 for all $\varepsilon \in (0, \varepsilon_0)$, there is a $z^1_\varepsilon := (t_1, x_{n+2}, s_1, y_{n+2})_\varepsilon$ such that

$$\hat{H}'(z^1_\varepsilon, \varepsilon)^T \hat{H}(z^1_\varepsilon, \varepsilon) = 0$$
$$\|\hat{H}(z^1_\varepsilon, \varepsilon)\| \le c\varepsilon \le c\varepsilon_0.$$

However, by our assumption, there exists a sequence $\{\varepsilon_k\}$ with $\varepsilon_{k-1} \le \varepsilon_k \le \varepsilon_0$ and $\varepsilon_k \to 0$ such that $\hat{H}'(z^1_{\varepsilon_k}, \varepsilon_k)$ is singular. Then by Lemma 2.4, at the point $z^1_{\varepsilon_k}$,

$$-2\varepsilon_k = x_{n+2} - y_{n+2} \le 2x_{n+2} + e^T(t_1 + t_2) + \varepsilon_k - \Gamma + c\varepsilon_k.$$

Since $D(\varepsilon_0)$ is bounded, by passing to a subsequence, we may assume that $z^1_{\varepsilon_k} \to (\bar{t}_1, \bar{t}_2, \bar{x}_{n+2}, \bar{s}_1, \bar{s}_2, \bar{y}_{n+2})$. By (2.5) and (2.6), $(\bar{t}_1, \bar{t}_2, \bar{s}_1, \bar{s}_2) \in \hat{S}_0(M, q)$ and $\bar{x}_{n+2} = \bar{y}_{n+2} = 0$. Hence we have

$$e^T(\bar{t}_1 + \bar{t}_2) \ge \Gamma.$$

This contradicts (2.13). This completes the proof. □

Let us end this section by the following example, which shows the following:

1. the central path can be very short when the solution set is bounded;
2. the central path may not exist for all $\varepsilon > 0$ if the solution set is unbounded.

However, the Big-Γ smooth paths exist for all $\varepsilon \in (0, 1]$.

Example 2.1. Let

$$M = \begin{pmatrix} 0 & -1 \\ 0 & 1 \end{pmatrix}, \qquad q = \begin{pmatrix} \delta \\ -\frac{1}{2} \end{pmatrix}, \qquad \text{where} \quad \delta \ge \frac{1}{2}.$$

It is easy to verify that M is a P_0 matrix and the LCP(M, q) has a unique solution

$$(t^*, s^*) = \left(0, \frac{1}{2}, \delta - \frac{1}{2}, 0\right) \quad \text{if} \quad \delta > \frac{1}{2}.$$

However, for any $\delta^0 > 0$ the complementarity level set

$$\left\{(t, s) \ : \ t^T s \le \delta^0 + \delta\left(\delta - \frac{1}{2}\right),\ s = Mt + q, (t, s) > 0\right\}$$

contains the unbounded line

$$\left(k, \delta - \frac{\delta^0}{k}, \frac{\delta^0}{k}, \delta - \frac{\delta^0}{k} - \frac{1}{2}\right)$$

for all $k > \delta^0/(\delta - \frac{1}{2})$.

The central path

$$\{(t, s) \ : \ t_i s_i = \varepsilon,\ i = 1, 2,\ s = Mt + q, (t, s) > 0\}$$
$$= \left\{\left(\frac{\varepsilon}{\delta - t_2}, t_2\right),\ t_2 = \frac{1 + \sqrt{1 + 16\varepsilon}}{4} < \delta\right\}$$

does not exist for $\varepsilon \ge \delta(\delta - \frac{1}{2})$.

If $\delta = \frac{1}{2}$, this problem has a unbounded solution set

$$\left\{ \left(t_1, \frac{1}{2}, 0, 0 \right) : \ t_1 \geq 0 \right\}.$$

In this case the central path does not exist for all $\varepsilon > 0$.

Now we show that if we choose $\Gamma \geq n + 5 = 7$, then for all $\varepsilon \leq 1$, (2.4) has a solution; i.e., the Big-Γ smooth paths exist for all $\varepsilon \leq 1$.

By Lemma 2.1 in [9] and Lemma 2.2, at the Big-Γ smooth paths, $(x, y) \geq 0$ and

$$\begin{aligned} x_1 y_1 = x_1(\delta - x_2 + (2-\delta)x_3) &= \varepsilon^2, \\ x_2 y_2 = x_2 \left(-\frac{1}{2} + x_2 + \frac{1}{2} x_3 \right) &= \varepsilon^2, \\ x_3 y_3 = x_3^2 &= \varepsilon^2, \\ y_4 = \Gamma - x_1 - x_2 - x_3 - x_4, & \\ x_4 = \int_{-\infty}^{(x_4 - y_4)/\varepsilon} (x_4 - y_4 - \varepsilon\mu)\rho_2(\mu) d\mu. & \end{aligned}$$

We can calculate the point on the Big-Γ smooth paths

$$x_3 = y_3 = \varepsilon \leq 1,$$

$$x_2 = \frac{1 - \varepsilon + \sqrt{(1-\varepsilon)^2 + 16\varepsilon^2}}{4} \leq 1,$$

$$x_1 = \frac{\varepsilon^2}{\delta - x_2 + (2-\delta)\varepsilon} \leq 1,$$

$$y_1 = \delta - x_2 + (2-\delta)x_3, \qquad y_2 = x_2 + \frac{1}{2}x_3 - \frac{1}{2},$$

$$x_4 = \Gamma - x_1 - x_2 - x_3 - 2\varepsilon \geq 2\varepsilon, \qquad y_4 = 2\varepsilon,$$

or

$$x_4 = 0, \qquad y_4 = \Gamma - x_1 - x_2 - x_3 \geq 4\varepsilon,$$

where we use $\rho(\mu) = 0$ for $\mu \notin [-4, 0]$ to calculate x_4 and y_4.

There are two Big-Γ smooth paths that are bounded, continuous in ε, never cross each other, and converge to two solutions

$$z^{*,1} = \left(0, \frac{1}{2}, 0, \Gamma - \frac{1}{2}, \delta - \frac{1}{2}, 0, 0, 0 \right)$$

and

$$z^{*,2} = \left(0, \frac{1}{2}, 0, 0, \delta - \frac{1}{2}, 0, 0, \Gamma - \frac{1}{2} \right)$$

as $\varepsilon \to 0$. Both of the above equations contain the solution of the original LCP(M, q): $(x_1^{*,1}, x_2^{*,1}, y_1^{*,1}, y_2^{*,1}) = (x_1^{*,2}, x_2^{*,2}, y_1^{*,2}, y_2^{*,2}) = (0, \frac{1}{2}, \delta - \frac{1}{2}, 0)$.

3. Algorithm and its convergence. In this section we propose an algorithm and prove its global convergence.

ALGORITHM 3.1. *Given $\sigma \in (0,1)$ and $\alpha_i \in (0,1)$ for $i = 1, 2$.*

Step 0 (*Initial Step*)
Choose x^0, y^0, ε^0 such that $\|H(z^0, \varepsilon^0)\| \le c\varepsilon^0$ and $H'(z^0, \varepsilon^0)$ is nonsingular.

Step 1 (*Newton Step*)
If $H(z^k, \varepsilon^k) = 0$, set $z^{k+1} = z^k$ and go to Step 3. Otherwise, let Δz^k solve the equation

$$H(z^k, \varepsilon^k) + H'(z^k, \varepsilon^k)\Delta z^k = 0. \tag{3.1}$$

Step 2 (*Line Search*)
Let λ_k be the maximum of the values $1, \alpha_1, \alpha_1^2, \ldots$ such that

$$\|H(z^k + \lambda_k \Delta z^k, \varepsilon^k)\| \le (1 - \sigma\lambda_k)\|H(z^k, \varepsilon^k)\|. \tag{3.2}$$

Set $z^{k+1} = z^k + \lambda_k \Delta z^k$.

Step 3 (ε *Reduction*)
Let γ_k be the maximum of the values $\alpha_2^2, \alpha_2^3, \ldots$ such that

$$\|H(z^{k+1}, (1 - \gamma_k)\varepsilon^k)\| \le (1 - \gamma_k)c\varepsilon_k. \tag{3.3}$$

If $x_{n+2}^{k+1} - y_{n+2}^{k+1} \ne -2(1 - \gamma_k)\varepsilon^k$, set $\varepsilon^{k+1} = (1 - \gamma_k)\varepsilon^k$. Otherwise, set $\varepsilon^{k+1} = (1 - \alpha_2\gamma_k)\varepsilon^k$.

Algorithm 3.1 is similar to the smoothing method introduced by Burke and Xu [1]. The main difference is that the definition of ε^{k+1} in Step 3 ensures the nonsingularity of $H'(z^{k+1}, \varepsilon^{k+1})$ for the P_0 matrix LCP(M, q).

It is easy to verify that if $y^0 = Nx^0 + p$, then $y^k = Nx^k + p$ for all $k \ge 0$.

The following lemma shows that we can easily find a starting point (z^0, ε^0) satisfying these conditions in the initial step of Algorithm 3.1.

LEMMA 3.1. *Suppose that M is a P_0 matrix. Let*

$$x^0 = (e, 1, 0), \quad y^0 = (e, 1, \Gamma - (n+1)), \quad \frac{1}{c+1} \le \varepsilon^0 \le 1. \tag{3.4}$$

Then $y^0 = Nx^0 + p \ge 0$, $\|H(z^0, \varepsilon^0)\| \le c\varepsilon^0$, and $H'(z^0, \varepsilon^0)$ is nonsingular.

Proof. Obviously $y^0 = Nx^0 + p$. Since $\Gamma \ge n + 5$, $y^0 \ge 0$. Thus $H_i(z^0, \varepsilon^0) = 0, i = 1, 2, \ldots, n+2$. By a simple calculation, we have

$$\Psi_i(x_i^0, y_i^0, \varepsilon^0) = \frac{1}{2}\left(x_i + y_i - \sqrt{(x_i - y_i)^2 + 4\varepsilon^2}\right) = 1 - \varepsilon^0, \quad i = 1, 2, \ldots, n+1,$$

and

$$\Psi_{n+2}(x_{n+2}^0, y_{n+2}^0, \varepsilon^0) = x_{n+2} = 0.$$

Hence (z^0, ε^0) satisfies $\|H(z^0, \varepsilon^0)\| \le c\varepsilon^0$.

Moreover, from $\Gamma \ge n + 5$ and $\varepsilon^0 \le 1$, we have

$$x_{n+2}^0 - y_{n+2}^0 = n + 1 - \Gamma \le -4 < -2\varepsilon^0.$$

Therefore, $H'(z^0, \varepsilon^0)$ is nonsingular by Lemma 2.4. ∎

THEOREM 3.2. *If M is a P_0 matrix, then Algorithm* 3.1 *is well defined, and the sequence $\{z^k\}$ satisfies*

$$\|H(z^k, \varepsilon^k)\| \le c\varepsilon^k. \tag{3.5}$$

Proof. We prove this theorem by induction.

For $k = 0$, by Lemma 3.1, $z^0 = (e, 1, 0, e, 1, \Gamma-(n+1))$ satisfies (3.5) and $H'(z^0, \varepsilon^0)$ is nonsingular.

We suppose that z^k satisfies (3.5) and $H'(z^k, \varepsilon^k)$ is nonsingular. Then Step 1 is well defined. If $H(z^k, \varepsilon^k) \neq 0$, then $\Delta z^k \neq 0$. Hence Δz^k is a strictly descent direction of $\|H(\cdot, \varepsilon^k)\|$ at z^k, and so the line search procedure is finite by construction in Step 2.

Step 3 is well defined since if $H(z^k, \varepsilon^k) = 0$, then $z^{k+1} = z^k$ and

$$\|H(z^{k+1}, \varepsilon^k)\| = 0 < c\varepsilon^k.$$

Otherwise, by the construction of Step 2,

$$\|H(z^{k+1}, \varepsilon^k)\| < \|H(z^k, \varepsilon^k)\| \le c\varepsilon^k,$$

which implies that there is a finite number $\gamma_k > 0$ such that (3.3) holds.

By the construction of Step 3, $H'(z^{k+1}, \varepsilon^{k+1})$ is nonsingular.

Now we show that (3.5) holds at $(z^{k+1}, \varepsilon^{k+1})$.

If $x^{k+1}_{n+2} - y^{k+1}_{n+2} \neq -2(1-\gamma_k)\varepsilon^k$, then by construction of Step 3, (3.5) holds. Hence we need only consider the case

$$x^{k+1}_{n+2} - y^{k+1}_{n+2} = -2(1 - \gamma_k)\varepsilon^k;$$

i.e.,

$$\varepsilon^{k+1} = (1 - \alpha_2\gamma_k)\varepsilon^k.$$

Notice that

$$\varepsilon^k > (1 - \alpha_2\gamma_k)\varepsilon^k > (1 - \gamma_k)\varepsilon^k$$

and Step 3 provides that

$$\begin{aligned} c\varepsilon^{k+1} &\ge c(1 - \alpha_2\gamma_k)\varepsilon^k \\ &\ge c(1 - \gamma_k)\varepsilon_k \\ &\ge H(z^{k+1}, (1 - \gamma_k)\varepsilon_k) \\ &\ge -c(1 - \gamma_k)\varepsilon^k. \end{aligned} \tag{3.6}$$

By Result 3 of Proposition 2.1 in [2], for $i = 1, 2, \ldots, n+1$, $\Psi_i(x_i^{k+1}, y_i^{k+1}, \cdot)$ is strictly decreasing with respect to ε, which gives

$$\begin{aligned} c\varepsilon^{k+1} &\ge \Psi_i(x_i^{k+1}, y_i^{k+1}, (1 - \gamma_k)\varepsilon^k) \\ &> \Psi_i(x_i^{k+1}, y_i^{k+1}, \varepsilon^{k+1}) \\ &= \Psi_i(x_i^{k+1}, y_i^{k+1}, (1 - \alpha_2\gamma_k)\varepsilon^k) \\ &\ge \Psi_i(x_i^{k+1}, y_i^{k+1}, (1 - \gamma_k)\varepsilon^k) + (\alpha_2 - 1)\gamma_k\varepsilon_k \end{aligned}$$

$$
\begin{aligned}
&> -c(1-\gamma_k)\varepsilon^k - (1-\alpha_2)\gamma_k\varepsilon^k \\
&= -c\varepsilon_k + c\gamma_k\varepsilon^k - \gamma_k\varepsilon^k + \alpha_2\gamma_k\varepsilon^k \\
&\geq -c\varepsilon^k + \alpha_2(c-1)\gamma_k\varepsilon^k + \alpha_2\gamma_k\varepsilon^k \\
&> -c\varepsilon^k + c\alpha_2\gamma_k\varepsilon^k \\
&= -c\varepsilon^{k+1},
\end{aligned}
$$

where the third inequality follows from Result 4 of Proposition 2.1 in [2] (also see [8]), and the fourth inequality follows from (3.6). Hence (3.5) holds for $i = 1, 2, \ldots, n+1$.

Let $w = x_{n+2}^{k+1} - y_{n+2}^{k+1}$, and

$$\phi(\varepsilon) = \int_{-\infty}^{\frac{w}{\varepsilon}} (w - \varepsilon\mu)\rho_2(\mu)d\mu.$$

Then $\omega = -2(1-\gamma_k)\varepsilon^k < 0$ and

$$\phi'(\varepsilon) = -\int_{-\infty}^{\frac{w}{\varepsilon}} \mu\rho_2(\mu)d\mu \geq 0.$$

Hence ϕ is a monotonically increasing function. This implies that Ψ_{n+2} is monotonically decreasing and Lipschitz continuous with respect to the parameter ε. Hence we obtain

$$
\begin{aligned}
c\varepsilon^{k+1} &\geq \Psi_{n+2}(x_{n+2}^{k+1}, y_{n+2}^{k+1}, (1-\gamma_k)\varepsilon^k) \\
&\geq \Psi_{n+2}(x_{n+2}^{k+1}, y_{n+2}^{k+1}, (1-\alpha_2\gamma_k)\varepsilon^k) \\
&\geq \Psi_{n+2}(x_{n+2}^{k+1}, y_{n+2}^{k+1}, (1-\gamma_k)\varepsilon^k) - 2(1-\alpha_2)\gamma_k\varepsilon^k \\
&\geq -c(1-\gamma_k)\varepsilon^k - 2(1-\alpha_2)\gamma_k\varepsilon_k \\
&= -c\varepsilon^k + (c - 2(1-\alpha_2))\gamma_k\varepsilon^k \\
&= -c\varepsilon^k + c\alpha_2\gamma_k\varepsilon_k + (c-2)(1-\alpha_2)\gamma_k\varepsilon^k \\
&\geq -c(1-\alpha_2\gamma_k)\varepsilon^k \\
&= -c\varepsilon^{k+1},
\end{aligned}
$$

where the third inequality follows from that Ψ_{n+2} is Lipschitz continuous with the Lipschitz constant 2 and $c \geq 2$. (See (2.2).)

Therefore, we have

$$c\varepsilon^{k+1} \geq H_i(z^{k+1}, \varepsilon^{k+1}) \geq -c\varepsilon^{k+1} \quad \text{for} \quad i = n+3, \ldots, 2(n+2).$$

By the definition of H, the parameter ε is not involved in the first $n+2$ components of H; i.e.,

$$H_i(z^{k+1}, \varepsilon^k) = H_i(z^{k+1}, \varepsilon^{k+1}) \text{ for } i = 1, 2, \ldots, n+2.$$

Hence (3.5) holds. □

THEOREM 3.3. *Suppose that M is a P_0 matrix. Then Algorithm* 3.1 *is well defined. Let $\{(z^k, \varepsilon^k)\}$ be a sequence generated by Algorithm* 3.1.

1. *$\{z^k\}$ remains in the bounded set $D(\varepsilon^0)$, and $\{\varepsilon^k\}$ decreases monotonically in R_{++}.*

2. *If an accumulation point $(\bar{z}, \bar{\varepsilon})$ of $\{(z^k, \varepsilon^k)\}$ satisfies $\bar{x}_{n+2} - \bar{y}_{n+2} \neq -2\bar{\varepsilon}$ or $\bar{\varepsilon} = 0$, then*

$$\lim_{k\to\infty} \varepsilon^k = 0 \quad \textit{and} \quad \lim_{k\to\infty} H(z^k, \varepsilon^k) = 0, \tag{3.7}$$

and for all accumulation points $\{\hat{z}, \hat{\varepsilon}\}$ of $\{z^k, \varepsilon^k\}$,

$$\hat{\varepsilon} = 0 \quad \textit{and} \quad H(\hat{z}, \hat{\varepsilon}) = H_0(\hat{z}) = 0. \tag{3.8}$$

Proof. First we show that

$$D(\varepsilon^k) \subseteq D(\varepsilon^{k-1}) \quad \text{for} \quad \varepsilon^k \leq \varepsilon^{k-1}.$$

Suppose that $z \in D(\varepsilon^k)$. Then

$$\begin{aligned} \|H(z, \varepsilon^{k-1})\| &\leq \|H(z, \varepsilon^{k-1}) - H(z, \varepsilon^k)\| + \|H(z, \varepsilon^k)\| \\ &\leq 2(\varepsilon^{k-1} - \varepsilon^k) + c\varepsilon^k \\ &= c\varepsilon^{k-1} - (c-2)(\varepsilon^{k-1} - \varepsilon^k) \\ &\leq c\varepsilon^{k-1}, \end{aligned}$$

where we use $c \geq 2$ and (2.2). Hence $z \in D(\varepsilon^{k-1})$. This, together with Theorem 3.2, implies that the sequence generated by Algorithm 3.1 remains in $D(\varepsilon^0)$.

By Theorem 3.2, Algorithm 3.1 is well defined. Furthermore, by construction of Algorithm 3.1,

$$0 < \varepsilon^{k+1} \leq (1 - \alpha_2\gamma_k)\varepsilon^k < \varepsilon^k.$$

Hence $\{\varepsilon^k\}$ is a monotonically decreasing sequence, and there is $\bar{\varepsilon}$ such that

$$\lim_{k\to\infty} \varepsilon^k = \bar{\varepsilon}.$$

If $\bar{\varepsilon} = 0$, then from Theorem 3.2 we have (3.7). Moreover, since $\{\varepsilon^k\}$ is a monotonically decreasing sequence, (3.5) and (3.7) imply (3.8).

Suppose on the contrary that $\bar{\varepsilon} > 0$. Then this implies $\gamma_k \to 0$.

Since $\{z^k\}$ remains in the bounded set $D(\varepsilon^0)$, taking a subsequence if necessary, we may assume that the sequence $\{z^k\}$ converges to some $\bar{z}$. Based on the ε reduction step, we have

$$\left\| H\left(z^k, \left(1 - \frac{1}{\alpha_2^2}\gamma_{k-1}\right)\varepsilon^{k-1}\right)\right\| > \left(1 - \frac{1}{\alpha_2^2}\gamma_{k-1}\right) c\varepsilon^{k-1}.$$

Since $\gamma_k \to 0$, by passing to limits, we have

$$\|H(\bar{z}, \bar{\varepsilon})\| \geq c\bar{\varepsilon} > 0. \tag{3.9}$$

Since $\bar{x}_{n+2} - \bar{y}_{n+2} \neq -2\bar{\varepsilon}$, $H'(\bar{z}, \bar{\varepsilon})$ is nonsingular by Lemma 2.4. Hence we can find a unique solution $\Delta\bar{z}$ of the linear equations in Step 1. Furthermore, from (3.9)

it is a strictly descent direction for $\|H(\cdot,\bar{\varepsilon})\|$ at $\bar{z}$. As a result, the corresponding linear search step length $\bar{\lambda}$ and ε reduction step length $\bar{\gamma}$ are both bounded below by a positive constant. Notice that the function H and its Jacobian H' are continuous in a neighborhood of $(\bar{z},\bar{\varepsilon})$. It follows that Δz^k converges to $\Delta\bar{z}$ and therefore γ_k must be uniformly bounded below by some positive constant for all large k. This contradicts the assumption that $\gamma_k \to 0$. Hence we must have $\varepsilon^k \to 0$, and so (3.7) and (3.8) hold. □

COROLLARY 3.4. *Suppose that the solution set $S_0(M,q)$ of the P_0 matrix LCP(M,q) is nonempty and bounded. Then there exist $\Gamma > 0$ and $\varepsilon^0 > 0$ such that the sequence $\{z^k\}$ generated by Algorithm* 3.1 *is bounded and its limiting points are solutions of LCP(N,p).*

Proof. By Lemma 2.6, there exists $\Gamma > 0$ and $\varepsilon^0 > 0$ such that $H'(z,\varepsilon)$ is nonsingular for all $z \in D(\varepsilon^0)$.

Hence at any accumulation point $(\bar{z},\bar{\varepsilon})$ generated by Algorithm 3.1, $\bar{x}_{n+2}-\bar{y}_{n+2} \neq -2\bar{\varepsilon}$. By Theorem 3.3, we complete the proof. □

COROLLARY 3.5. *Under Assumption* 2.1*, the conclusion of Corollary* 3.4 *holds.*

Proof. The proof is similar to that of Theorem 2.8. It is sufficient to show that any limit point of the sequence $\{t_2^k, s_2^k\}$ generated by Algorithm 3.1 for solving LCP(M_2,q_2) is a strictly complementarity solution.

Let

$$u^k = \Psi(t_2^k, s_2^k, \varepsilon^k).$$

By Corollary 3.4 and (ii) of Assumption 2.1, any limit point of $\{t_2^k, s_2^k\}$ is a solution of LCP(M_2,q_2), and

$$\|u^k\| \le c\varepsilon^k \to 0.$$

Notice that we have

$$s_2 - u^k = M_2(t_2^k - u^k) + \varepsilon^k r_2 + q_2 + (M_2 - I)u^k, \tag{3.10}$$

$$t_2^k - u^k > 0, \quad s_2^k - u^k > 0, \quad (s_2^k - u^k)_i(t_2 - u^k)_i = (\varepsilon^k)^2. \tag{3.11}$$

Let

$$q_2(\varepsilon^k) = q_2 + (M_2 - I)u^k + \varepsilon^k r_2.$$

The boundedness of $S_0(M_2,q_2)$ implies there is $k_0 \ge 0$ such that for all $k \ge k_0$, $S_0(M_2,q_2(\varepsilon^k))$ is nonempty [17]. Since $(t_2^k - u^k, s_2^k - u^k)$ is an interior point of LCP$(M_2,q_2(\varepsilon^k))$, the monotone property of M_2 implies $S_0(M_2,q_2(\varepsilon^k))$ is bounded. Moreover, since $S_0(M_2,q_2)$ has a strictly complementarity solution and $\|q_2(\varepsilon^k)-q_2\| \le (\|r_2\| + c\|M - I\|)\varepsilon^k \to 0$, there is $k_1 \ge k_0$ such that for all $k \ge k_1$, $S_0(M_2,q_2(\varepsilon^k))$ has a strictly complementarity solution.

Therefore, by Theorem 2.8 there is ε^{k_1} such that for $\varepsilon \in (0,\varepsilon^{k_1})$ the smooth path for LCP$(M_2,q_2(\varepsilon^k))$ exists and leads to a strictly complementarity solution.

By (3.10) and (3.11), $(t_2^k - u^k, s_2^k - u^k)$ is on the path. Therefore, using $\|u^k\| \to c\varepsilon^k \to 0$ again, we complete this proof. □

We can restart Algorithm 3.1 when $\bar{x}_{n+2} - \bar{y}_{n+2} = -2\bar{\varepsilon}$. In particular, we have the following proposition.

PROPOSITION 3.6. *Suppose $H'(z,\varepsilon)$ is singular. Then for $\hat{\Gamma} = \Gamma + \varepsilon$ and*

$$\hat{z}_i = z_i,\ i \neq n+2, 2n+2, \quad \hat{x}_{n+2} = x_{n+2} - \frac{1}{2}\varepsilon, \quad \hat{y}_{n+2} = y_{n+2} + \frac{3}{2}\varepsilon,$$

$\hat{H}'(\hat{z},\varepsilon)$ is nonsingular and

$$\hat{H}(\hat{z},\varepsilon) = H(z,\varepsilon), \tag{3.12}$$

where $\hat{H}$ is the function using $\hat{\Gamma}$.

Proof. At the new point

$$\hat{z} = (x_1, \ldots, x_{n+1}, \hat{x}_{n+2}, y_1, \ldots, y_{n+1}, \hat{y}_{n+2})$$

the new LCP$(N, \hat{p})$ with the new $\hat{\Gamma}$ satisfies

$$\hat{H}_i(\hat{z},\varepsilon) = H_i(z,\varepsilon), \qquad i = 1, 2, \ldots, 2(n+1)+1,$$

and

$$\hat{H}_{2(n+2)}(\hat{z},\varepsilon) = \hat{x}_{n+2} = x_{n+2} - \frac{1}{2}\varepsilon = x_{n+2} + \frac{\varepsilon}{4}\int_{-4}^{-2}(2+\mu)d\mu = H_{2(n+2)}(z,\varepsilon),$$

where we use $\hat{x}_{n+2} - \hat{y}_{n+2} \leq -4\varepsilon$ and $\rho_2(\mu) = 0$ for $\mu \leq -4$. Hence we have (3.12). Furthermore, $\hat{H}'(\hat{z},\varepsilon)$ is nonsingular since

$$\hat{x}_{n+2} - \hat{y}_{n+2} = x_{n+2} - y_{n+2} - 2\varepsilon = -4\varepsilon < -2\varepsilon. \qquad \square$$

4. Numerical results. In this section, we report numerical results for testing Algorithm 3.1. These test problems are P_0 matrix LCP with unbounded solution set, which include a random test problem and a Murty-type problem with an unbounded solution set.

Problem 1. Example 1.1.

Problem 2. Example 2.1 with $\delta = \frac{1}{2}$.

Notice that this problem has no interior point.

Problem 3. A Murty-type problem with an unbounded solution set.

$$M = \begin{pmatrix} 1 & 2 & \ldots & 2 & 2 \\ 0 & 1 & \ldots & 2 & 2 \\ \vdots & \vdots & \ddots & & \vdots \\ \vdots & \vdots & & 1 & 2 \\ 0 & 0 & 0 & \ldots & 0 \end{pmatrix}, \qquad q = -\begin{pmatrix} 1 \\ 1 \\ \vdots \\ 1 \\ 0 \end{pmatrix}.$$

The solution set of this problem contains the unbounded line $t = (0, \ldots, 0, t_n)^T$ for all $t_n \geq 0.5$. Moreover, the set of interior points is empty.

Problem 4. A random problem [14].

$$M = \begin{pmatrix} P + D_1 & P + D_2 \\ -I & 0 \end{pmatrix}, \qquad q = -\begin{pmatrix} e \\ 0 \end{pmatrix},$$

where $P = A^T A$, $A \in R^{\frac{n}{2} \times \frac{n}{2}}$ with $0 < a_{ij} < 1$, D_1 and D_2 are diagonal matrices with $1 \leq (D_1)_{ii}, (D_2)_{ii} \leq 3$. This problem is a P_0 matrix LCP and has an engineering application. The solution set of this problem contains the unbounded set

TABLE 4.1
Numerical results of Algorithm 3.1

prob	n	k	k_m	$\|t\| + \|s\|$	$\|H_0\|$
1	3	5	0	1	2.0837e-9
2	2	9	0	0.5	7.2783e-9
3	100	10	2	2	1.9674e-9
4	100	10	7	0.1928	3.9348e-9

$\{t = (0, \ldots, 0, t_{\frac{n}{2}+1}, \ldots, t_n)^T : t_i \geq 1, i = \frac{n}{2} + 1, \ldots, n\}$. Moreover, the set of interior points is empty.

We implemented Algorithm 3.1 in MATLAB using the following parameters:

$$\sigma = 0.125, \quad \alpha_1 = 0.625, \quad \alpha_2 = 0.925, \quad \Gamma = n + 5, \quad c = 2\sqrt{2(n+2)} + 1.$$

Based on Lemma 3.1, we chose the initial point as

$$x^0 = (e, 1, 0), \quad y^0 = (e, 1, \Gamma - (n+1)), \quad \varepsilon^0 = \frac{c+2}{2(c+1)}.$$

We terminate the iteration if $\|H_0(z^k)\| \leq 1.0^{-8}$ or $k \geq 200$.

In Table 4.1, we report our results for these four problems. The columns in Table 4.1 have the following definitions:

prob:	number of test problems,
n :	dimension of test example,
k:	number of iterations,
k_m	the total iteration number of line search steps,
$\|t\| + \|s\|$:	the value $\|t\| + \|s\|$ at the final iterate,
$\|H_0\|$:	the value $\|H_0(z)\|$ at the final iterate.

Remark 4.1. Regularization methods [6, 15, 16, 18] have been used successfully to solve ill-posed problems. However, if the original problem has a solution with $t = 0$, then the regularization method only can generate this solution. In many cases we need a nonzero solution or a strictly complementarity solution if it exists. In such cases, the Big-Γ smoothing method may give a satisfactory solution. Our numerical results show this advantage. In particular, Algorithm 3.1 generated the following final iterates:

Problem 1:	$t = (0, 0.5, 0)$,	$s = (0.5, 0, 0.5)$.
Problem 2:	$t = (0, 0.5)$,	$s = (0, 0)$.
Problem 3:	$t = (0, \ldots, 0, 1, 0)$,	$s = (1, \ldots, 1, 0, 0)$.
Problem 4:	number of components with	$t_i, s_i > 1.0^{-8}$ is n.

All final iterates approach maximal complementarity solutions. We assume that such numerical results are due to the properties of the path. Let us use Example 1.1 to compare the Big-Γ smooth path with the regularization path [6]:

$$\{t(\varepsilon) \ : \ t(\varepsilon) \text{ is a solution of the LCP}(M + \varepsilon I, q), \quad \varepsilon > 0\}.$$

Example 1.1 contains a solution with $t = 0$.

Since $M + \varepsilon I$ is a P matrix for every $\varepsilon > 0$, the regularization path $t(\varepsilon) \equiv (0, 0, 0)$ for every $\varepsilon > 0$.

In the Big-Γ smoothing model for Example 1.1, $r = e - Me - q = 0$. We choose $\Gamma = n + 5 = 8$. At the Big-Γ smooth path,

$$x_i > 0, y_i > 0, \quad y_i = (Nx + p)_i, \quad x_i y_i = \varepsilon^2, \quad i = 1, 2, 3, 4,$$

and

$$x_5 = \int_{-\infty}^{(x_5 - y_5)/\varepsilon} (x_5 - y_5 - \varepsilon\mu)\rho_2(\mu)d\mu, \qquad y_5 = \Gamma - x_1 - x_2 - x_3 - x_4 - x_5.$$

We can calculate this point for $\varepsilon \in (0, 1]$:

$$x_1 = \frac{4\varepsilon^2}{\sqrt{1 + 8\varepsilon^2} + 1} \le 1, \quad x_2 = \frac{\sqrt{1 + 8\varepsilon^2} + 1}{4} \le 1, \quad x_3 = \frac{4\varepsilon^2}{\sqrt{1 + 8\varepsilon^2} + 1} \le 1,$$

$$x_4 = \varepsilon \le 1, \quad \text{and} \quad x_5 = 0, \text{ or } x_5 = \Gamma - x_1 - x_2 - x_3 - x_4 - 2\varepsilon,$$

where we use $\rho_2(\mu) = 0$ for $\mu \notin [-4, 0]$ to calculate x_5. Two Big-Γ smooth paths never cross each other and converge to two solutions:

$$z^{*,1} = \left(0, \frac{1}{2}, 0, 0, \Gamma - \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, 0, 0\right)$$

and

$$z^{*,2} = \left(0, \frac{1}{2}, 0, 0, 0, \frac{1}{2}, 0, \frac{1}{2}, 0, \Gamma - \frac{1}{2}\right),$$

respectively, as $\varepsilon \to 0$. Both of them contain a strictly complementarity solution of the original LCP(M, q)

$$(t^*, s^*) = \left(0, \frac{1}{2}, 0, \frac{1}{2}, 0, \frac{1}{2}\right).$$

Acknowledgment. The authors are grateful to the referees, M. S. Gowda, C. Kanzow, D. Sun, and P. Tseng for their helpful comments.

REFERENCES

[1] J. Burke and S. Xu, *The global linear convergence of a non-interior path-following algorithm for linear complementarity problem*, Math. Oper. Res., 23 (1998), pp. 719–734.
[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] C. Chen and O.L. Mangasarian, *Smoothing methods for convex inequalities and linear complementarity problems*, Math. Programming, 71 (1995), pp. 51–69.
[4] X. Chen and Y. Ye, *On homotopy-smoothing methods for box-constrained variational inequalities*, SIAM J. Control Optim., 37 (1999), pp. 589–616.
[5] R.W. Cottle, J.-S. Pang, and R.E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.
[6] F. Facchinei and C. Kanzow, *Beyond monotonicity in regularization methods for nonlinear complementarity problems*, SIAM J. Control Optim., 37 (1999), pp. 1150–1161.
[7] F. Facchinei and J.-S. Pang, *Total Stability of Variational Inequalities*, Tech. report, Universitá di Roma "La Sapienza," Dipartmento di Informatica e Sistemistica, Via Buonarroti, Roma, Italy, 1998.

[8] 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, PA, 1996, pp. 105–116.

[9] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 851–868.

[10] M. Kojima, N. Megiddo, and T. Noma, *Homotopy continuation methods for nonlinear complementarity problems*, Math. Oper. Res., 16 (1991), pp. 754–774.

[11] M. Kojima, N. Megiddo, T. Noma, and A. Yoshise, *A Unified Approach to Interior Point Algorithms for Linear Complementarity Problems*, Lecture Notes in Computer Science 538, Springer-Verlag, New York, 1991.

[12] M. Kojima, S. Mizuno, and A. Yoshise, *A little theorem of the big M in interior point algorithms*, Math. Programming, 59 (1993), pp. 361–375.

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

[14] P.M. Pardalos, Y. Ye, C. Han, and J.A. Kaliski, *Solution of P_0-matrix linear complementarity problems using a potential reduction algorithm*, SIAM J. Matrix Anal. Appl., 14 (1993), pp. 1048–1060.

[15] H.-D. Qi, *A regularized smoothing Newton method for box constrained variational inequality problems with P_0-functions*, SIAM J. Optim., 10 (1999), pp. 315–330.

[16] G. Ravindran and M.S. Gowda, *Regularization of P_0-Function in Box Variational Inequality Problems*, Research report, Department of Mathematics and Statistics, University of Maryland Baltimore County, Baltimore, 1997.

[17] S. Robinson, *Generalized equations and their solutions*. I, Math. Programming Stud., 10 (1979), pp. 128–141.

[18] D. Sun, *A regularized Newton method for solving nonlinear complementarity problems*, Appl. Math. Optim., 40 (1999), pp. 315–339.

[19] R. Sznajder and M.S. Gowda, *On the limiting behavior of the trajectory of regularized solutions of a P_0-complementarity problem*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998, pp. 371–379.

[20] Y. Ye, *On homogeneous and self-dual algorithms for LCP*, Math. Programming, 76 (1997), pp. 211–222.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 364–379

THE PROXIMAL POINT ALGORITHM WITH GENUINE SUPERLINEAR CONVERGENCE FOR THE MONOTONE COMPLEMENTARITY PROBLEM*

NOBUO YAMASHITA† AND MASAO FUKUSHIMA†

Abstract. In this paper, we consider a proximal point algorithm (PPA) for solving monotone nonlinear complementarity problems (NCP). PPA generates a sequence by solving subproblems that are regularizations of the original problem. It is known that PPA has global and superlinear convergence properties under appropriate criteria for approximate solutions of subproblems. However, it is not always easy to solve subproblems or to check those criteria. In this paper, we adopt the generalized Newton method proposed by De Luca, Facchinei, and Kanzow to solve subproblems and adopt some NCP functions to check the criteria. Then we show that the PPA converges globally provided that the solution set of the problem is nonempty. Moreover, without assuming the local uniqueness of the solution, we show that the rate of convergence is superlinear in a genuine sense, provided that the limit point satisfies the strict complementarity condition.

Key words. nonlinear complementarity problem, proximal point algorithm, genuine superlinear convergence

AMS subject classifications. 47H05, 90C33

PII. S105262349935949X

1. Introduction.

The nonlinear complementarity problem (NCP) [9] is to find a vector $x \in R^n$ such that

$$\text{NCP}(F): \qquad F(x) \geq 0, \quad x \geq 0, \quad \langle x, F(x) \rangle = 0,$$

where F is a mapping from R^n into R^n and $\langle \cdot, \cdot \rangle$ denotes the inner product in R^n. Throughout this paper we assume that F is continuously differentiable and monotone.

Until now, a variety of methods for solving NCP have been proposed and investigated. Among them, the proximal point algorithm (PPA) proposed by Martinet [7] and further studied by Rockafellar [13] is known for its theoretically nice convergence properties. PPA originally was designed to find a vector x satisfying $0 \in T(x)$, where T is a maximal monotone operator. Hence it is applicable to a wide class of problems such as convex programming problems, monotone variational inequality problems, and monotone complementarity problems. In this paper, we focus on PPA for solving monotone complementarity problems. PPA generates a sequence $\{x^k\}$ by solving subproblems that are regularizations of the original problem. For NCP(F), given the current point x^k, PPA obtains the next point x^{k+1} by approximately solving the subproblem

$$F^k(x) \geq 0, \quad x \geq 0, \quad \langle x, F^k(x) \rangle = 0, \tag{1.1}$$

where $F^k : R^n \to R^n$ is defined by

$$F^k(x) := F(x) + c_k(x - x^k) \tag{1.2}$$

*Received by the editors June 7, 1999; accepted for publication (in revised form) February 9, 2000; published electronically September 27, 2000. This work was supported in part by the Scientific Research Grant-in-Aid from the Ministry of Education, Science, Sports and Culture, Japan.
http://www.siam.org/journals/siopt/11-2/35949.html
†Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (nobuo@amp.i.kyoto-u.ac.jp, fuku@amp.i.kyoto-u.ac.jp).

and $c_k > 0$. The mapping F^k is strongly monotone when F is monotone. Hence subproblem (1.1) is expected to be more tractable than the original problem. With appropriate criteria for approximate solutions of subproblems (1.1), PPA has global and superlinear convergence property under mild conditions [6, 13]. However, it is not always easy to check those criteria for general monotone operator problems. In this paper, we will show that, for monotone complementarity problems, some NCP functions turn out to be useful in constructing practical approximation criteria. Another implementation issue is how to solve subproblems efficiently. In the PPA proposed in this paper, we will use the generalized Newton method proposed by De Luca, Facchinei, and Kanzow [2] to solve subproblems (1.1). Since F^k is strongly monotone, we can show that the approximation criteria for each subproblem are attained finitely. The PPA then converges globally provided that the solution set of NCP(F) is nonempty. Moreover, without assuming the local uniqueness of the solution, we can show that the rate of convergence is superlinear. From the practical viewpoint, it is important to estimate computational costs for solving a subproblem at each iteration. We give conditions under which the approximation criteria for the subproblem are eventually fulfilled by a single Newton iteration.

The paper is organized as follows. In section 2, we review some concepts and preliminary results that will be used in the subsequent analysis. In section 3, we describe the proposed PPA for NCP(F). In section 4 we show its convergence properties.

Throughout we adopt the following notation. For $a \in R$, $(a)_+$ denotes $\max\{0, a\}$, and for $x \in R^n$, $[x]_+$ denote the projection of x onto R^n_+, the nonnegative orthant of R^n. For two vectors x and y, $\min\{x, y\}$ denotes the vector whose ith element is $\min\{x_i, y_i\}$. For a vector valued function $F : R^n \to R^m$, $F'(x)$ denotes the Jacobian and $\nabla F(x)$ denotes the transposed matrix of $F'(x)$.

2. Preliminaries. In this section, we first review some mathematical concepts and basic properties of PPA that will be used in the subsequent analysis. We then discuss reformulations of NCP and related results concerning error bounds. Finally, we briefly mention the generalized Newton method for NCP proposed in [2], which will be used to solve subproblems in PPA.

2.1. Mathematical concepts. First we recall some definitions concerning the monotonicity of a mapping from R^n into itself.

DEFINITION 2.1. *The mapping $F : R^n \to R^n$ is called a*

(a) *monotone function if*

$$\langle x - y, F(x) - F(y)\rangle \geq 0 \quad \textit{for all } x, y \in R^n, \tag{2.1}$$

(b) *strongly monotone function with modulus $\mu > 0$ if*

$$\langle x - y, F(x) - F(y)\rangle \geq \mu\|x - y\|^2 \quad \textit{for all } x, y \in R^n. \tag{2.2}$$

From the definition, it is clear that a strongly monotone function is monotone. Moreover, if F is a differentiable monotone function, then $\nabla F(x)$ is positive semidefinite for all $x \in R^n$.

DEFINITION 2.2. *Let $H : R^n \to R^n$ be locally Lipschitz continuous. Then the B subdifferential of H at x is the set of $n \times n$ matrices defined by*

$$\partial_B H(x) := \left\{ \lim_{\substack{x^i \in D_H \\ x^i \to x}} \nabla H(x^i)^T \right\},$$

where $D_H \subseteq R^n$ is the set where H is differentiable.

Note that the Clarke subdifferential of H is defined by

$$\partial H(x) := \text{co}\ \partial_B H(x),$$

where co denotes the convex hull of a set [1].

Next we recall the notion of semismoothness, which lies in between the continuous differentiability and the directional differentiability.

DEFINITION 2.3. *Let $H : R^n \to R^n$ be locally Lipschitz continuous. We say that H is semismooth at x if*

$$\lim_{\substack{V \in \partial H(x+td') \\ d' \to d,\, t \downarrow 0}} Vd' \tag{2.3}$$

exists for all d. Moreover, we say that H is strongly semismooth at x if for any $d \to 0$ and for any $V \in \partial H(x+d)$,

$$Vd - H'(x;d) = O(\|d\|^2),$$

where $H'(x;d)$ denotes the directional derivative of H at x along direction d.

Note that, when H is semismooth at x, the limit (2.3) is equal to the directional derivative $H'(x;d)$.

2.2. Proximal point algorithm. NCP(F) is equivalent to the problem of finding a point x such that

$$0 \in T(x), \tag{2.4}$$

where $T : R^n \to 2^{R^n}$ is the maximal monotone mapping defined by

$$T(x) := F(x) + N(x), \tag{2.5}$$

with $N : R^n \to 2^{R^n}$ being the normal cone mapping for R^n_+ defined by

$$N(x) := \begin{cases} \{y \in R^n \mid \langle x - z,\ y\rangle \geq 0,\ \ \forall z \geq 0\} & \text{if } x \geq 0, \\ \emptyset & \text{otherwise.} \end{cases}$$

With an arbitrary initial point x^0, PPA generates a sequence $\{x^k\}$ converging to a solution of (2.4) by the iterative scheme:

$$x^{k+1} \approx P_k(x^k),$$

where $P_k : R^n \to R^n$ is the mapping defined by $P_k := (I + \frac{1}{c_k}T)^{-1}$, $\{c_k\}$ is a positive sequence, and $x^{k+1} \approx P_k(x^k)$ means that x^{k+1} is an approximation to $P_k(x^k)$. For NCP(F), $P_k(x^k)$ is given by

$$P_k(x^k) = \left(I + \frac{1}{c_k}(F+N)\right)^{-1}(x^k).$$

Hence, we have

$$0 \in F(P_k(x^k)) + c_k(P_k(x^k) - x^k) + N(P_k(x^k)).$$

Therefore, the finding $x^{k+1} \approx P_k(x^k)$ for NCP(F) amounts to approximately solving the following subproblem NCP(F^k): Find $x \in R^n$ such that

$$F^k(x) \geq 0, \qquad x \geq 0, \qquad \langle x, F^k(x) \rangle = 0, \tag{2.6}$$

where F^k is defined by (1.2). Note that when c_k is small, the subproblem is close to the original one. On the other hand, when c_k is large, a solution of the subproblem is expected to lie near x^k, and hence the subproblem is presumably easy to solve.

To ensure convergence of PPA, x^{k+1} has to be located sufficiently near the solution $P_k(x^k)$ of subproblem (1.1). There have been proposed a number of criteria for the approximate solution of the subproblem. Among others, Rockafellar [13] proposed the following two criteria:

Criterion 1. $\|x^{k+1} - P_k(x^k)\| \leq \varepsilon_k, \quad \sum_{k=0}^{\infty} \varepsilon_k < \infty.$

Criterion 2. $\|x^{k+1} - P_k(x^k)\| \leq \eta_k \|x^{k+1} - x^k\|, \quad \sum_{k=0}^{\infty} \eta_k < \infty.$

Note that Criterion 1 guarantees global convergence, while Criterion 2, which is rather restrictive, ensures superlinear convergence of PPA.

THEOREM 2.4. (see [13, Theorem 1]). *Suppose that the sequence* $\{x^k\}$ *is generated by PPA with Criterion* 1 *and that* $\{c_k\}$ *is bounded. If* NCP(F) *has at least one solution, then* $\{x^k\}$ *converges to a solution* x^* *of* NCP(F).

Note that it is not necessary to let $\{c_k\}$ converge to 0 for the global convergence. Therefore, we may keep F^k uniformly strongly monotone, so that subproblems (1.1) are numerically well-conditioned.

On the other hand, if we let $\{c_k\}$ converge to 0, we can expect rapid convergence of PPA. Luque [6, Theorem 2.1] showed superlinear convergence without assuming the local uniqueness of the solution of NCP(F).

THEOREM 2.5. (see [6, Theorem 2.1]). *Suppose that* $\{x^k\}$ *is generated by PPA with Criteria* 1 *and* 2 *and that* $c^k \to 0$. *If there exist positive constants* δ *and* C *such that*

$$\operatorname{dist}(x, \bar{X}) \leq C\|w\| \text{ whenever } x \in T^{-1}(w) \text{ and } \|w\| \leq \delta, \tag{2.7}$$

where $\operatorname{dist}(x, \bar{X})$ *denotes the distance from point* x *to the solution set* $\bar{X}$ *of* NCP(F), *then the sequence* $\{\operatorname{dist}(x^k, \bar{X})\}$ *converges to 0 superlinearly.*

2.3. Reformulations of NCP. NCP can be reformulated as a system of equations in various ways. In this subsection, we review basic properties of two reformulations of NCP that will play a crucial role in solving subproblems of PPA. In the remainder of this section, we deal with the problem NCP($\hat{F}$), where $\hat{F} : R^n \to R^n$ is a certain mapping.

First we consider the function $\phi_{FB} : R^2 \to R$ defined by

$$\phi_{FB}(a, b) := a + b - \sqrt{a^2 + b^2}. \tag{2.8}$$

This function is called the Fischer–Burmeister function and has the following property:

$$\phi_{FB}(a, b) = 0 \iff a \geq 0, b \geq 0, ab = 0.$$

Any function with this property is often called an NCP function. Using the function ϕ_{FB}, we define the mapping $H : R^n \to R^n$ by

$$H(x) := \begin{pmatrix} \phi_{FB}(x_1, \hat{F}_1(x)) \\ \vdots \\ \phi_{FB}(x_n, \hat{F}_n(x)) \end{pmatrix}. \tag{2.9}$$

Then it is straightforward to see that NCP($\hat{F}$) is equivalent to the system of equations

$$H(x) = 0. \tag{2.10}$$

The mapping H is not differentiable at a point x such that $x_i = \hat{F}_i(x) = 0$ for some i. However, when $\hat{F}$ is continuously differentiable, H is locally Lipschitz, and hence it has the B subdifferential everywhere. Though it is not necessarily easy to calculate the B subdifferential of a general locally Lipschitz mapping, De Luca, Facchinei, and Kanzow [2] show that, for the mapping H, an element V of $\partial_B H(x)$ is expressed as

$$V = D_a + D_b \nabla \hat{F}(x)^T, \tag{2.11}$$

where D_a, D_b are diagonal matrices defined by

$$\begin{aligned} &((D_a)_{ii}, (D_b)_{ii}) \\ &= \begin{cases} \left(1 - \frac{x_i}{\sqrt{x_i^2 + \hat{F}_i(x)^2}}, 1 - \frac{\hat{F}_i(x)}{\sqrt{x_i^2 + \hat{F}_i(x)^2}}\right) & \text{if } (x_i, \hat{F}_i(x)) \neq (0,0), \\ (1-\eta, 1-\xi) & \text{otherwise} \end{cases} \end{aligned} \tag{2.12}$$

and (η, ξ) is a vector satisfying $\eta^2 + \xi^2 = 1$. De Luca, Facchinei, and Kanzow [2] also discuss how to calculate (η, ξ) when $(x_i, \hat{F}_i(x)) = (0,0)$.

The next proposition will be useful in the analysis of the generalized Newton method for solving subproblems of PPA.

PROPOSITION 2.6. *Let M be a positive definite matrix and μ be a positive constant such that*

$$\langle v, Mv \rangle \geq \mu \|v\|^2 \qquad \text{for all } v \in R^n. \tag{2.13}$$

Let $D_a = \operatorname{diag}(a_i)$ and $D_b = \operatorname{diag}(b_i)$ be diagonal matrices whose diagonal elements are nonnegative and satisfy $a_i + b_i \geq d$ for all i, where d is a positive constant. Then we have

$$\inf_{\|v\|=1} \|(D_a + D_b M)v\| \geq \bar{B}\mu,$$

where $\bar{B} = d/(n \max\{1, \|M\|\})$. Moreover, the following inequality holds:

$$\|(D_a + D_b M)^{-1}\| \leq \frac{1}{\bar{B}\mu}.$$

Proof. Let v be an arbitrary vector such that $\|v\| = 1$. Then, since

$$\langle v, Mv \rangle \geq \mu$$

holds by (2.13), there exists an index i such that

$$v_i (Mv)_i \geq \frac{\mu}{n}. \tag{2.14}$$

Since

$$v_i(Mv)_i \le |v_i|\|M\|,$$

it follows from (2.14) that

$$|v_i| \ge \frac{\mu}{n\|M\|}. \tag{2.15}$$

Moreover, (2.14) implies that v_i has the same sign as $(Mv)_i$. Hence, by (2.14) and (2.15), we have

$$\begin{aligned}
|((D_a + D_bM)v)_i| &= a_i|v_i| + b_i|(Mv)_i| \\
&\ge \frac{\mu}{n\|M\|}a_i + \frac{\mu}{n|v_i|}b_i \\
&\ge \frac{\mu}{n\|M\|}a_i + \frac{\mu}{n}b_i \\
&\ge \frac{(a_i + b_i)\mu}{n\max\{1, \|M\|\}} \\
&\ge \bar{B}\mu.
\end{aligned}$$

Consequently, we have

$$\|(D_a + D_bM)v\| \ge \bar{B}\mu.$$

Next we show the last part of the lemma. Note that, under the given assumptions, $D_a + D_bM$ is nonsingular [2, Lemma 5.1]. Since

$$\|(D_a + D_bM)^{-1}\| = \frac{1}{\inf_{\|v\|=1}\|(D_a + D_bM)v\|},$$

it follows that

$$\|(D_a + D_bM)^{-1}\| \le \frac{1}{\bar{B}\mu}. \qquad \square$$

As a direct consequence of this proposition, we have the following corollary.

COROLLARY 2.7. *Suppose that $\hat{F}$ is strongly monotone with modulus μ. Let D_a and D_b be defined by* (2.12). *Then we have*

$$\|(D_a + D_b\nabla\hat{F}(x)^T)^{-1}\| \le \frac{1}{B_1\mu},$$

where $B_1 = (2 - \sqrt{2})/(n\max\{1, \|\nabla\hat{F}(x)\|\})$.

Now we define the function $\Phi_{FB} : R^n \to R$ by

$$\Phi_{FB}(x) := \frac{1}{2}\|H(x)\|^2, \tag{2.16}$$

where H is given by (2.9). We note that Φ_{FB} attains its global minimum at a solution of NCP($\hat{F}$), because NCP($\hat{F}$) is equivalent to (2.10).

LEMMA 2.8. *The mapping $H : R^n \to R^n$ defined by* (2.9) *has the following properties:*

(a) *If $\hat{F}$ is differentiable, then H is semismooth.*

(b) *If $\nabla\hat{F}$ is locally Lipschitz continuous, then H is strongly semismooth.*

(c) *If $\nabla\hat{F}(x)$ is positive definite, then every $V \in \partial_B H(x)$ is nonsingular.*

Proof. Items (a) and (c) are shown in [3]. Item (b) is shown in [14]. □

LEMMA 2.9. *The function $\Phi_{FB} : R^n \to R$ defined by* (2.16) *has the following properties:*

(a) *If $\hat{F}$ is differentiable, then Φ_{FB} is differentiable.*

(b) *If $\hat{F}$ is monotone, then any stationary point of Φ_{FB} is a solution of* NCP($\hat{F}$).

(c) *If $\hat{F}$ is strongly monotone with modulus μ and Lipschitz continuous with constant L, then $\sqrt{\Phi_{FB}(x)}$ provides a global error bound for* NCP($\hat{F}$), *that is,*

$$\|x - \hat{x}\| \le \frac{B_2(L+1)}{\mu}\sqrt{\Phi_{FB}(x)} \quad \text{for all } x \in R^n,$$

where $\hat{x}$ is the unique solution of NCP($\hat{F}$) *and B_2 is a positive constant independent of $\hat{F}$.*

(d) *If $\hat{F}$ is affine and* NCP($\hat{F}$) *has a solution, then $\sqrt{\Phi_{FB}(x)}$ provides a local error bound for* NCP($\hat{F}$), *that is, there exist positive constants B_3 and B_4 such that*

$$\operatorname{dist}(x, \hat{X}) \le B_3\sqrt{\Phi_{FB}(x)} \quad \text{for all } x \in \{y \in R^n \mid \Phi_{FB}(y) \le B_4\},$$

where $\hat{X}$ denotes the solution set of NCP($\hat{F}$).

Proof. Items (a) and (b) are shown in [5]. Item (c) follows from [8, 11]. Item (d) is shown in [4]. □

By using Lemma 2.9(c), we have the following error bound result on a compact set.

COROLLARY 2.10. *Let $S \subseteq R^n$ be a compact set. Suppose that $\hat{F}$ is strongly monotone with modulus μ and Lipschitz continuous with constant L on S. Then $\sqrt{\Phi_{FB}(x)}$ provides an error bound on S, that is, there exists a positive constant B_2 such that*

$$\|x - \hat{x}\| \le \frac{B_2(L+1)}{\mu}\sqrt{\Phi_{FB}(x)} \quad \text{for all } x \in S,$$

where $\hat{x}$ is the unique solution of NCP($\hat{F}$).

In the PPA to be presented in the next section, we will also utilize the following function $\Psi : R^n \to R$, which has a more favorable error bound property than Φ_{FB}:

$$\Psi(x) := \sum_{i=1}^{n} \psi(x_i, \hat{F}_i(x)),$$

where $\psi : R^2 \to R$ is defined by

$$\psi(a, b) := |ab| + |\min\{a, b\}|.$$

Note that ψ is also an NCP function. It is clear that $\Psi(x) \ge 0$ for all x, and $\Psi(x) = 0$ if and only if x is a solution of NCP($\hat{F}$).

The next lemma shows an interesting error bound result for the function Ψ, which will play an important role in section 4. Note that this error bound is valid only on the set R^n_+.

LEMMA 2.11. *Suppose that $\hat{F}$ is strongly monotone with modulus μ. Then we have*

$$\|x-\hat{x}\| \le 2\max\{1,\|x\|\}\sqrt{\frac{\Psi(x)}{\mu}} \quad \textit{for all } x \in \left\{ y \in R^n_+ \mid \Psi(y) \le \frac{\mu}{4} \right\},$$

where $\hat{x}$ is the unique solution of NCP($\hat{F}$).

Proof. Let $x \in R^n_+$ be arbitrary. Since $\hat{F}$ is strongly monotone with modulus μ, we have

$$\begin{aligned}
\mu\|x-\hat{x}\|^2 &\le \langle x-\hat{x}, \hat{F}(x)-\hat{F}(\hat{x})\rangle \\
&= \langle x, \hat{F}(x)\rangle + \langle \hat{x}, -\hat{F}(x)\rangle + \langle \hat{F}(\hat{x}), -x\rangle \\
&\le \sum_{i=1}^n |x_i\hat{F}_i(x)| + \sum_{i=1}^n |\hat{x}_i||(-\hat{F}_i(x))_+| + \sum_{i=1}^n |\hat{F}_i(\hat{x})||(-x_i)_+| \\
&= \sum_{i=1}^n |x_i\hat{F}_i(x)| + \sum_{i=1}^n |\hat{x}_i||(-\hat{F}_i(x))_+|.
\end{aligned}$$

Since

$$(-b)_+ \le |\min\{a,b\}| \quad \text{for all } (a,b) \in R^2,$$

it follows that

$$\begin{aligned}
\mu\|x-\hat{x}\|^2 &\le \sum_{i=1}^n |x_i\hat{F}_i(x)| + \sum_{i=1}^n |\hat{x}_i||(-\hat{F}_i(x))_+| \\
&\le \sum_{i=1}^n \left\{ |x_i\hat{F}_i(x)| + |\hat{x}_i||\min\{x_i, \hat{F}_i(x)\}| \right\} \\
&\le \max\{1,\|\hat{x}\|_\infty\}\Psi(x) \\
&\le \max\{1,\|\hat{x}\|\}\Psi(x).
\end{aligned}$$

Hence we have

$$\begin{aligned}
\|x-\hat{x}\| &\le \sqrt{\frac{\max\{1,\|\hat{x}\|\}}{\mu}\Psi(x)} \\
&\le \max\{1,\|\hat{x}\|\}\sqrt{\frac{\Psi(x)}{\mu}} \\
&\le \max\{1,\|\hat{x}-x\|+\|x\|\}\sqrt{\frac{\Psi(x)}{\mu}}.
\end{aligned}$$

Therefore, if $\|\hat{x}-x\|+\|x\| \le 1$, then the desired inequality holds. If $\|\hat{x}-x\|+\|x\| > 1$, then

$$\left(1-\sqrt{\frac{\Psi(x)}{\mu}}\right)\|x-\hat{x}\| \le \|x\|\sqrt{\frac{\Psi(x)}{\mu}}.$$

Since $1-\sqrt{\Psi(x)/\mu} \ge \frac{1}{2}$ whenever $\Psi(x) \le \frac{\mu}{4}$, we also have the desired inequality. ∎

We note that, unlike Lemma 2.8(c), Lemma 2.11 does not assume the Lipschitz continuity of $\hat{F}$. Moreover, unlike Lemma 2.8(d), the error bound result shown in Lemma 2.11 is explicitly represented in terms of the modulus of strong monotonicity of $\hat{F}$.

2.4. Generalized Newton method. In this section, we review the generalized Newton method for solving NCP proposed by De Luca, Facchinei, and Kanzow [2]. The PPA to be presented in the next section will use this method to solve subproblems.

PROCEDURE 1 *(generalized Newton method for* NCP($\hat{F}$)).

Step 1. Choose a constant $\beta \in (0, \frac{1}{2})$. Let x^0 be an initial point and set $j := 0$.

Step 2. If x^j satisfies a termination criterion, then stop.

Step 3. Choose $V_j \in \partial_B H(x^j)$ and get d^j satisfying

$$V_j d^j = -H(x^j). \tag{2.17}$$

Step 4. If $x^j + d^j$ satisfies the termination criterion, then stop. Otherwise, find the smallest nonnegative integer i_j such that

$$\Phi_{FB}(x^j + 2^{-i_j} d^j) \le (1 - \beta 2^{-i_j}) \Phi_{FB}(x^j).$$

Step 5. Set $x^{j+1} := x^j + 2^{-i_j} d^j$ and $j := j + 1$, and go to Step 2.

Note that Procedure 1 is a slight simplification of the algorithm in [2]. Within the framework of the present paper, however, there is essentially no difference between them, because we only consider the case where $\hat{F}$ is strongly monotone.

For Procedure 1 with the termination criterion ignored, the following convergence result holds.

PROPOSITION 2.12 (see [2]). *Suppose that $\hat{F}$ is differentiable and strongly monotone and that $\nabla \hat{F}$ is Lipschitz continuous around the unique solution $\hat{x}$ of* NCP($\hat{F}$). *Then Procedure* 1 *globally converges to $\hat{x}$ and the rate of convergence is quadratic.*

Since the mappings F^k involved in the subproblems generated by PPA are strongly monotone, Procedure 1 can be applied to these problems effectively.

3. Algorithm and its convergence properties. In this section we describe PPA for NCP(F) and study its convergence properties.

ALGORITHM 1.

Step 1. *Choose parameters $\alpha \in (0,1)$, $c_0 \in (0,1)$ and an initial point $x^0 \in R^n$. Set $k := 0$.*

Step 2. *If x^k satisfies $\Phi_{FB}(x^k) = 0$, then stop.*

Step 3. *Let $F^k : R^n \to R^n$ be defined by* (1.2), *and apply Procedure* 1 *to obtain an approximate solution $\tilde{x}^{k+1}$ of* NCP(F^k) *that satisfies the conditions*

$$\Psi^k([\tilde{x}^{k+1}]_+) \le \frac{c_k^3}{4 \max\{1, \|[\tilde{x}^{k+1}]_+\|\}^2} \tag{3.1}$$

and

$$\sqrt{\Phi_{FB}^k(\tilde{x}^{k+1})} \le c_k^4 \|x^k - \tilde{x}^{k+1}\|, \tag{3.2}$$

where

$$\Psi^k(x) := \sum_{i=1}^n \psi(x_i, F_i^k(x))$$

and

$$\Phi_{FB}^k(x) := \frac{1}{2}\sum_{i=1}^{n} \phi_{FB}(x_i, F_i^k(x))^2.$$

Step 4. *Set* $x^{k+1} := [\tilde{x}^{k+1}]_+$, $c_{k+1} := \alpha c_k$, *and* $k := k+1$. *Go to Step* 2.

Remark 3.1. The condition (3.1) in Step 3 corresponds to Criterion 1 of PPA, while the condition (3.2) corresponds to Criterion 2.

Remark 3.2. If x^k is a solution of NCP(F), then the algorithm stops at Step 2. Otherwise, since $F^k(x^k) = F(x^k)$, x^k is not a solution of NCP(F^k) at Step 3. Moreover, since F^k is strongly monotone, Theorem 2.12 ensures that Procedure 1 can finitely find $\tilde{x}^{k+1}$ satisfying (3.1) and (3.2).

First we show that Algorithm 1 has a global convergence property.

THEOREM 3.1. *Suppose that* NCP(F) *has at least one solution. Then the sequence* $\{x^k\}$ *generated by Algorithm* 1 *converges to a solution* x^* *of* NCP(F).

Proof. It suffices to show that $\{x^k\}$ satisfies the assumption of Theorem 2.4, that is, $\{x^k\}$ satisfies Criterion 1. Since $x^{k+1} = [\tilde{x}^{k+1}]_+$ in Step 4 and $0 < c_k < 1$, we have, by (3.1) in Step 3,

$$\Psi^k(x^{k+1}) \le \frac{c_k^3}{4\max\{1, \|x^{k+1}\|\}^2} \tag{3.3}$$

$$\le \frac{c_k}{4}. \tag{3.4}$$

Since F^k is strongly monotone with modulus c_k, it then follows from Lemma 2.11 and (3.4) that

$$\|x^{k+1} - P_k(x^k)\| \le 2\max\{1, \|x^{k+1}\|\}\sqrt{\frac{\Psi^k(x^{k+1})}{c_k}}, \tag{3.5}$$

where $P_k(x^k)$ is the unique solution of NCP(F^k). By (3.3) and (3.5), we have

$$\|x^{k+1} - P_k(x^k)\| \le c_k.$$

Since $\Sigma_{k=1}^{\infty} c_k < \infty$, it follows from Theorem 2.4 that $\{x^k\}$ converges to a solution of NCP(F). □

Remark 3.3. The sequence $\{c_k\}$ in Algorithm 1 converges to 0. However this property is needed for superlinear convergence, not for global convergence. To see this, consider the algorithm using the condition

$$\Psi^k([\tilde{x}^{k+1}]_+) \le \frac{c_k^2 \eta^k}{4\max\{1, \|[\tilde{x}^{k+1}]_+\|\}^2}$$

with $\eta \in (0,1)$ instead of (3.1). In a similar way to the proof of Theorem 3.1, we can show that the modified algorithm has the global convergence property even if $\{c_k\}$ is bounded away from 0.

Next we give conditions for Algorithm 1 to converge superlinearly. For this purpose, we first show that T defined by (2.5) has the property (2.7) under the following assumption.

Assumption 1. $\|\min\{x, F(x)\}\|$ provides a local error bound for NCP(F), that is, there exist positive constants $\bar{C}$ and $\bar{\delta}$ such that

$$\operatorname{dist}(x, \bar{X}) \le \bar{C}\|\min\{x, F(x)\}\| \quad \text{for all } x \text{ with } \|\min\{x, F(x)\}\| \le \bar{\delta},$$

where $\bar{X}$ denotes the solution set of NCP(F).

Note that when F is affine, Assumption 1 holds by Lemma 2.9(d). On the other hand, when $\nabla F(x)$ is positive definite at any solution x of NCP(F), Assumption 1 holds by Lemma 2.8(c) and [10, Proposition 3].

The following result directly follows from [12, Proposition 3.1]. However, for completeness, we give a proof.

PROPOSITION 3.2. *Let T be the maximal monotone mapping defined by* (2.5). *If Assumption* 1 *holds and the solution set $\bar{X}$ of NCP(F) is nonempty, then there exist positive constants C and δ such that*

$$\operatorname{dist}(x, \bar{X}) \le C\|w\| \quad \text{for all } x \in T^{-1}(w), \ \text{for all } w \text{with } \|w\| \le \delta.$$

Proof. The mapping T defined by (2.5) is expressed as

$$T(x) = T_1(x) \times \cdots \times T_n(x),$$

where $T_i(x) \subseteq R$ is given by

$$T_i(x) = \begin{cases} \{F_i(x)\} & \text{if } x_i > 0, \\ \{F_i(x) + v_i \mid v_i \in (-\infty, 0]\} & \text{if } x_i = 0, \\ \emptyset & \text{otherwise} \end{cases}$$

for $i = 1, \ldots, n$.

Consider a pair (x, w) such that $w \in T(x)$. If $x_i > 0$, we have

$$|w_i| = |F_i(x)| \ge |\min\{x_i, F_i(x)\}|.$$

If $x_i = 0$ and $F_i(x) > 0$, it is clear that

$$|w_i| \ge 0 = |\min\{x_i, F_i(x)\}|.$$

If $x_i = 0$ and $F_i(x) \le 0$, then there exists $v_i \le 0$ such that

$$|w_i| = |F_i(x) + v_i|.$$

Hence we have

$$|w_i| = |F_i(x) + v_i| \ge |F_i(x)| = |\min\{x_i, F_i(x)\}|.$$

Consequently we have

$$\|w\| \ge \|\min\{x, F(x)\}\|.$$

It then follows from Assumption 1 that the desired property holds. □

By using Proposition 3.2, we show that Algorithm 1 has a superlinear rate of convergence.

THEOREM 3.3. *Suppose that Assumption* 1 *holds. Let $\{x^k\}$ be generated by Algorithm* 1. *Then the sequence $\{\operatorname{dist}(x^k, \bar{X})\}$ converges to* 0 *superlinearly.*

Proof. By Theorem 3.1, $\{x^k\}$ is bounded. Hence we may suppose that F^k is uniformly Lipschitz continuous with modulus L on a bounded set containing $\{x^k\}$. It then follows from Corollary 2.10 that there exists a positive constant B_2 such that

$$\|\tilde{x}^{k+1} - P_k(x^k)\| \le \frac{B_2(L+1)}{c_k} \sqrt{\Phi^k_{FB}(\tilde{x}^{k+1})}.$$

Hence by (3.2) in Step 3, we have

$$\|\tilde{x}^{k+1} - P_k(x^k)\| \leq B_2(L+1)c_k^3\|x^k - \tilde{x}^{k+1}\|.$$

Since $\Sigma_{k=1}^{\infty} c_k^3 < \infty$, the last inequality implies that $\{\tilde{x}^k\}$ satisfies Criterion 2. Therefore, by Proposition 3.2 and [6, Theorem 2.1], there exists a constant $C > 0$ such that for sufficiently large k

$$\operatorname{dist}(\tilde{x}^{k+1}, \bar{X}) \leq \frac{C}{(C^2 + (1/c_k)^2)^{\frac{1}{2}}} \operatorname{dist}(x^k, \bar{X}).$$

Noting that $\operatorname{dist}(x^{k+1}, \bar{X}) \leq \operatorname{dist}(\tilde{x}^{k+1}, \bar{X})$, we then have

$$\operatorname{dist}(x^{k+1}, \bar{X}) \leq \frac{C}{(C^2 + (1/c_k)^2)^{\frac{1}{2}}} \operatorname{dist}(x^k, \bar{X}).$$

Since $c_k \to 0$, $\{\operatorname{dist}(x^k, \bar{X})\}$ converges to 0 superlinearly. □

Theorem 3.3 says that the sequence $\{x^k\}$ generated by Algorithm 1 converges to the solution set $\bar{X}$ superlinearly under mild conditions. However, this does not necessarily mean that Algorithm 1 is practically efficient, because it says nothing about computational costs to solve a subproblem at each iteration. So it is important to estimate the number of iterations Procedure 1 spends at each iteration of Algorithm 1. Moreover, it is particularly interesting to see under what conditions Procedure 1 requires just a single iteration. In the next section, we answer this question.

4. Genuine superlinear convergence. In this section we give conditions under which a single Newton step of Procedure 1 for NCP(F^k) attains (3.1) and (3.2) in Step 3 of Algorithm 1, thereby genuine superlinear convergence of Algorithm 1 is ensured.

First we show that (3.1) is implied by (3.2) for sufficiently large k.

LEMMA 4.1. *When k is sufficiently large, if*

$$\sqrt{\Phi_{FB}^k(\tilde{x}^{k+1})} \leq c_k^4\|x^k - \tilde{x}^{k+1}\|$$

holds, then

$$\Psi^k([\tilde{x}^{k+1}]_+) \leq \frac{c_k^3}{4\max\{1, \|[\tilde{x}^{k+1}]_+\|\}^2}$$

also holds.

Proof. Since Ψ^k is uniformly locally Lipschitz continuous and $\{x^k\}$ converges, there exists $L > 0$ such that

$$\Psi^k([\tilde{x}^{k+1}]_+) \leq L\|[\tilde{x}^{k+1}]_+ - P_k(x^k)\| \leq L\|\tilde{x}^{k+1} - P_k(x^k)\|, \tag{4.1}$$

for all k. Moreover, since F is continuous differentiable, F is Lipschitz continuous on a compact set containing $\{x^k\}$. Therefore $\sqrt{\Phi_{FB}^k(x)}$ provides an error bound for NCP(F^k) on the same set by Corollary 2.10, that is, there exists $\tau > 0$ such that

$$\|\tilde{x}^{k+1} - P_k(x^k)\| \leq \frac{\tau}{c_k}\sqrt{\Phi_{FB}^k(\tilde{x}^{k+1})}. \tag{4.2}$$

It then follows from (3.2), (4.1), and (4.2) that there exists a positive constant τ' such that

$$\Psi^k([\tilde{x}^{k+1}]_+) \le \tau' c_k^3 \|\tilde{x}^{k+1} - x^k\|.$$

Since $\{\|\tilde{x}^{k+1} - x^k\|\}$ converges to 0 and since $\{\|[\tilde{x}^{k+1}]_+\|\}$ is bounded, (3.1) holds for sufficiently large k. □

This lemma says that (3.2) implies (3.1) for all k sufficiently large. Therefore, in the remainder of this section, we consider only conditions under which (3.2) is satisfied after a single Newton step for NCP(F^k).

The following lemma indicates the relation between $\|x^k - P_k(x^k)\|$ and $\mathrm{dist}(x^k, \bar{X})$.

LEMMA 4.2. *For sufficiently large k, there exists a constant $B_5 > 0$ such that*

$$\|x^k - P_k(x^k)\| \le \frac{B_5}{c_k} \mathrm{dist}(x^k, \bar{X}).$$

Proof. Let $\bar{x}^k$ be the nearest point in $\bar{X}$ from x^k. Since $\{x^k\}$ is bounded, so is $\{\bar{x}^k\}$. Thus the function $\sqrt{\Phi_{FB}(x)}$ is Lipschitz continuous on a bounded set containing $\{x^k\}$ and $\{\bar{x}^k\}$. Moreover, $\sqrt{\Phi^k_{FB}(x)}$ is also uniformly Lipschitz continuous on the same set. Let $L_1 > 0$ and $L_2 > 0$ be Lipschitz constants of $\sqrt{\Phi_{FB}(x)}$ and $\sqrt{\Phi^k_{FB}(x)}$, respectively. Then we have

$$\begin{aligned}\sqrt{\Phi_{FB}(x^k)} &= \left|\sqrt{\Phi_{FB}(x^k)} - \sqrt{\Phi_{FB}(\bar{x}^k)}\right| \\ &\le L_1 \|x^k - \bar{x}^k\| \\ &= L_1 \mathrm{dist}(x^k, \bar{X}).\end{aligned}$$

It follows from Corollary 2.10 that

$$\begin{aligned}\|x^k - P_k(x^k)\| &\le \frac{B_2(L_2+1)}{c_k} \sqrt{\Phi^k_{FB}(x^k)} \\ &= \frac{B_2(L_2+1)}{c_k} \sqrt{\Phi_{FB}(x^k)}.\end{aligned}$$

Combining the above inequalities and letting $B_5 = B_2(L_2+1)L_1$ yield the desired inequality. □

Next we assume that the strict complementarity is satisfied at the limit point of the generated sequence. The assumption ensures the twice differentiability of H.

Assumption 2.

(a) The limit point x^* of the sequence $\{x^k\}$ generated by Algorithm 1 is nondegenerate, that is, $x^*_i + F_i(x^*) > 0$ holds for all i.

(b) F is a continuously differentiable function with a locally Lipschitzian Jacobian.

For the sake of convenience, we define the mapping $H^k : R^n \to R^n$ by

$$H^k(x) := \begin{pmatrix} \phi_{FB}(x_1, F^k_1(x)) \\ \vdots \\ \phi_{FB}(x_n, F^k_n(x)) \end{pmatrix}.$$

LEMMA 4.3. *Suppose that Assumptions* 1 *and* 2 *hold. Then* H^k *is twice continuously differentiable in a neighborhood of* x^k *for sufficiently large* k*, and there exists a positive constant* B_6 *such that*

$$\|\nabla H^k(x^k)^T(x^k - P_k(x^k)) - H^k(x^k) + H^k(P_k(x^k))\| \le B_6\|x^k - P_k(x^k)\|^2.$$

Proof. Let $\bar{F} : R^{2n+1} \to R^n$ and $\bar{H} : R^{2n+1} \to R^n$ be defined by

$$\bar{F}(x, y, \mu) := F(x) + \mu(x - y),$$
$$\bar{H}(x, y, \mu) := \begin{pmatrix} \phi_{FB}(x_1, \bar{F}_1(x, y, \mu)) \\ \vdots \\ \phi_{FB}(x_n, \bar{F}_n(x, y, \mu)) \end{pmatrix},$$

respectively. Suppose that x^* is the limit point of the sequence $\{x^k\}$. Then, by Assumption 2, there exists a positive constant B_6 such that

$$\left\| \begin{pmatrix} \nabla_x \bar{H}(x, y, \mu) \\ \nabla_y \bar{H}(x, y, \mu) \\ \nabla_\mu \bar{H}(x, y, \mu) \end{pmatrix}^T \begin{pmatrix} x - x' \\ y - y' \\ \mu - \mu' \end{pmatrix} - \bar{H}(x, y, \mu) + \bar{H}(x', y', \mu') \right\|$$
$$\le B_6(\|x - x'\|^2 + \|y - y'\|^2 + |\mu - \mu'|^2), \quad \forall (x, y, \mu), (x', y', \mu') \in N. \tag{4.3}$$

We also note that H^k is twice continuously differentiable near x^k when k is sufficiently large. Since

$$\nabla_x \bar{H}(x, x^k, c_k) = \nabla H^k(x)$$

and since $(x^k, x^k, c_k), (P_k(x^k), x^k, c_k) \in N$ for sufficiently large k, substituting (x^k, x^k, c_k) for (x, y, μ) and $(P_k(x^k), x^k, c_k)$ for (x', y', μ') in (4.3) yields the desired inequality. □

Remark 4.1. For all k, H^k is strongly semismooth, and hence we have

$$\|\nabla H^k(x^k)^T(x^k - P_k(x^k)) - H^k(x^k) + H^k(P_k(x^k))\| \le B_k\|x^k - P_k(x^k)\|^2,$$

where B_k is a constant depending on H^k. If $\{B_k\}$ were bounded, the result stated in Lemma 4.3 would hold without Assumption 2. However, the boundedness of $\{B_k\}$ appears to be difficult to ensure without making an extra assumption.

Now let us denote

$$x_N^k := x^k - V_k^{-1}H^k(x^k), \quad V_k \in \partial_B H^k(x^k). \tag{4.4}$$

Note that x_N^k is a point produced by a single Newton iteration of Procedure 1 for NCP(F^k) with the initial point x^k.

By using Corollary 2.7 and Lemma 4.3, we can show the following key lemma.

LEMMA 4.4. *Suppose that Assumptions* 1 *and* 2 *hold. Then there exists* $B_7 > 0$ *such that*

$$\|x_N^k - P_k(x^k)\| \le \frac{B_7\|x^k - P_k(x^k)\|^2}{c_k}$$

for sufficiently large k*.*

Proof. First note that, by Lemma 4.3, $\nabla H^k(x^k)$ exists and hence $V^k = \nabla H^k(x^k)^T$ for all k sufficiently large. By (2.11), $\nabla H^k(x^k)$ is expressed as $\nabla H^k(x^k) = D_a +$

$\nabla F^k(x^k)^T D_b$. Moreover, $\{\|\nabla F^k(x^k)\|\}$ is bounded. Therefore, by Corollary 2.7 and Lemma 4.3, there exist $B_1 > 0$ and $B_6 > 0$ such that

$$\begin{aligned}\|x_N^k - P_k(x^k)\| &= \|x^k - P_k(x^k) - (\nabla H^k(x^k)^T)^{-1}(H^k(x^k) - H^k(P_k(x^k)))\| \\ &\le \|(\nabla H^k(x^k)^T)^{-1}\|\|\nabla H^k(x^k)^T(x^k - P_k(x^k)) - H^k(x^k) + H^k(P_k(x^k))\| \\ &\le \frac{B_6\|x^k - P_k(x^k)\|^2}{B_1 c_k}\end{aligned}$$

for sufficiently large k. Consequently, letting $B_7 = B_6/B_1$ shows the lemma. □

Now we are in a position to establish the main result of this section.

THEOREM 4.5. *Suppose that Assumptions* 1 *and* 2 *hold. Let* x_N^k *be given by* (4.4). *Then for sufficiently large* k, x_N^k *satisfies the condition* (3.2) *in Step* 3 *of Algorithm* 1, *that is,*

$$\sqrt{\Phi_{FB}^k(x_N^k)} \le c_k^4 \|x_N^k - x^k\|.$$

Proof. Let $\gamma > 0$ be arbitrary. Since $\{\mathrm{dist}(x^k, \bar{X})\}$ converges to 0 superlinearly by Theorem 3.3, we have for sufficiently large k

$$\mathrm{dist}(x^k, \bar{X}) \le \gamma c_k^6.$$

It follows from Lemma 4.2 that

$$\|x^k - P_k(x^k)\| \le \gamma B_5 c_k^5.$$

Then by Lemma 4.4, we have

$$\begin{aligned}\|x_N^k - P_k(x^k)\| &\le \frac{B_7}{c_k}\|x^k - P_k(x^k)\|^2 \\ &\le \gamma B_5 B_7 c_k^4 \|x^k - P_k(x^k)\|.\end{aligned}$$

By the triangle inequality, the last inequality yields

$$\frac{1 - \gamma B_5 B_7 c_k^4}{\gamma B_5 B_7}\|x_N^k - P_k(x^k)\| \le c_k^4 \|x_N^k - x^k\|. \tag{4.5}$$

On the other hand, since $\sqrt{\Phi_{FB}^k(x)}$ is uniformly locally Lipschitz continuous, there exists $L_2 > 0$ such that

$$\sqrt{\Phi_{FB}^k(x_N^k)} \le L_2 \|x_N^k - P_k(x^k)\|.$$

Hence, by (4.5) it suffices to show

$$L_2 \le \frac{1 - \gamma B_5 B_7 c_k^4}{\gamma B_5 B_7}.$$

Since γ is arbitrary, choosing γ sufficiently small yields the last inequality. □

This theorem, along with Theorem 3.3, ensures that Algorithm 1 converges superlinearly in a genuine sense, provided that the limit of the generated sequence $\{x^k\}$ is nondegenerate.

Acknowledgments. We would like to thank Professor Paul Tseng for his valuable comments. We would also like to thank two anonymous referees for their helpful suggestions on the earlier version of this paper.

REFERENCES

[1] F.H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley & Sons, New York, 1983.
[2] 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.
[3] 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.
[4] 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.S. Wormersley, eds., World Scientific Publishers, Singapore, 1995, pp. 88–105.
[5] C. Geiger and C. Kanzow, *On the resolution of monotone complementarity problems,* Comput. Optim. Appl., 5 (1996), pp. 155–173.
[6] F.J. Luque, *Asymptotic convergence analysis of the proximal point algorithm,* SIAM J. Control Optim., 22 (1984), pp. 277–293.
[7] B. Martinet, *Perturbation des méthodes d'opimisation*, RAIRO Anal. Numér., 12 (1978), pp. 153–171.
[8] J.-S. Pang, *A posteriori error bounds for the linearly-constrained variational inequality problem,* Math. Oper. Res., 12 (1987), pp. 474–484.
[9] J.-S. Pang, *Complementarity problems,* in Handbook of Global Optimization, R. Horst and P. Pardalos, eds., Kluwer Academic Publishers, Boston, 1994, pp. 271–338.
[10] J.-S. Pang, and L.Q. Qi, *Nonsmooth equations: Motivation and algorithms,* SIAM J. Optim., 3 (1993), pp. 443–465.
[11] P. Tseng *Growth behavior of a class of merit functions for the nonlinear complementarity problem,* J. Optim. Theory Appl., 89 (1996), pp. 17–37.
[12] P. Tseng, *A modified forward-backward splitting method for maximal monotone mapping,* SIAM J. Control Optim., 38 (2000), pp. 431–446.
[13] R.T. Rockafellar, *Monotone operators and the proximal point algorithm,* SIAM J. Control Optim., 14 (1976), pp. 877–898.
[14] 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. 11, No. 2, pp. 380–388

ON PRIMAL AND DUAL INFEASIBILITY CERTIFICATES IN A HOMOGENEOUS MODEL FOR CONVEX OPTIMIZATION*

ERLING D. ANDERSEN†

Abstract. Andersen and Ye [*Math. Programming*, 84 (1999), pp. 375–399] suggested a homogeneous formulation and an interior-point algorithm for solution of the monotone complementarity problem (MCP). The advantage of the homogeneous formulation is that it always has a solution. Moreover, in the case in which the MCP is solvable or is (strongly) infeasible, the solution provides a certificate of optimality or infeasibility. In this note we demonstrate that if the suggested formulation is applied to the Karush–Kuhn–Tucker optimality conditions corresponding to a convex optimization problem, then an infeasibility certificate provides information about whether the primal or dual problem is infeasible given certain assumptions.

Key words. convex programming, homogeneous, self-dual

AMS subject classifications. 90C25, 90C20

PII. S1052623499353145

1. Introduction.

Most interior-point methods for solving convex optimization problems require that the problem has an optimal solution. Clearly, this assumption is not satisfied if the problem is either primal or dual infeasible. In the linear case this problem is addressed by using a homogeneous and self-dual model which was originally suggested by Goldman and Tucker [3] and later generalized to the monotone complementarity problem (MCP) by Andersen and Ye [2]. This larger class of problems contains all convex optimization problems, because the Karush–Kuhn–Tucker conditions corresponding to a convex optimization problem form an MCP.

The main idea of the homogeneous model is to embed the optimization problem in a slightly larger problem which always has a solution. The optimal solution to the embedded problem indicates whether the original problem has an optimal solution. Moreover, in the case in which the original problem has an optimal solution, the optimal solution to the embedded problem can easily be transformed into an optimal solution to the original problem. In the case where the original problem is (strongly) infeasible, then a certificate for the infeasibility is computed. However, in [2] it is not stated whether an infeasibility certificate indicates primal or dual infeasibility when the homogeneous model is applied to the optimality conditions of a convex optimization problem. The main purpose of the present work is to show that an infeasibility certificate in some cases indicates whether the primal or dual problem is infeasible.

The outline of the paper is as follows. In section 2 we present a homogeneous model for convex optimization and state the main lemma. In section 3 we apply the developed theory to convex quadratic and quadratically constrained optimization problems.

*Received by the editors March 3, 1999; accepted for publication (in revised form) May 31, 2000; published electronically October 11, 2000.
http://www.siam.org/journals/siopt/11-2/35314.html
†Faculty of ITS, TU Delft, SSOR/TWI, Mekelweg 4, 2628 CD Delft, The Netherlands (e.d.andersen@mosek.com).

2. A homogeneous model for convex optimization. The problem of interest is

$$\begin{array}{ll} \text{minimize} & c(x) \\ \text{subject to} & a_i(x) \geq 0, \quad i = 1, \ldots, m, \end{array} \tag{2.1}$$

where $x \in R^n$. The function $c : R^n \to R$ is assumed to be convex, and the component functions $a_i : R^n \to R$, $i = 1, \ldots, m$, are assumed to be concave. All functions are assumed to be once differentiable. Hence, the problem (2.1) minimizes a convex function over a convex set.

Next, define the Lagrange function

$$L(x, y) := c(x) - y^T a(x),$$

and then the Wolfe dual corresponding to (2.1) is

$$\begin{array}{ll} \text{maximize} & L(x, y) \\ \text{subject to} & \nabla_x L(x, y)^T = 0, \\ & y \geq 0. \end{array} \tag{2.2}$$

Combining (2.1) and (2.2) gives the MCP

$$\begin{array}{ll} \text{minimize} & y^T z \\ \text{subject to} & \nabla_x L(x, y)^T = 0, \\ & a(x) \quad = z, \\ & y, z \geq 0, \end{array} \tag{2.3}$$

where $z \in R^m$ is a vector of slack variables. A solution to (2.3) is said to be complementary if the corresponding objective value is zero.

Now when applying the homogeneous model suggested in [1, 2] to this problem we obtain the homogenized MCP

$$\begin{array}{lll} \text{minimize} & z^T y + \tau\kappa & \\ \text{subject to} & \tau \nabla_x L(x/\tau, y/\tau)^T & = 0, \\ & \tau a(x/\tau) & = z, \\ & -x^T \nabla_x L(x/\tau, y/\tau)^T - y^T a(x/\tau) & = \kappa, \\ & z, \tau, y, \kappa \geq 0, & \end{array} \tag{2.4}$$

where τ and κ are two additional variables.

Following [2], we say that (2.4) is asymptotically feasible if and only if a convergent sequence $(x^k, z^k, \tau^k, y^k, \kappa^k)$ exists for $k = 1, 2, \ldots$ such that

$$\lim_{k \to \infty} \begin{pmatrix} \tau^k \nabla_x L(x^k/\tau^k, y^k/\tau^k)^T, \\ \tau^k a(x^k/\tau^k) - z^k, \\ -(x^k)^T \nabla_x L(x^k/\tau^k, y^k/\tau^k)^T - (y^k)^T a(x^k/\tau^k) - \kappa^k \end{pmatrix} = 0 \tag{2.5}$$

and

$$(x^k, z^k, \tau^k, y^k, \kappa^k) \in R^n \times R^m_+ \times R_{++} \times R^m_+ \times R_{++} \quad \forall k, \tag{2.6}$$

where the limit point $(x^*, z^*, \tau^*, y^*, \kappa^*)$ of the sequence is called an asymptotically feasible point. We write R_+ and R_{++} for the nonnegative and positive real line, respectively. If this limit point also satisfies

$$(y^*)^T z^* + \tau^* \kappa^* = 0,$$

it is said to be asymptotically complementary.

THEOREM 2.1. *Equation* (2.4) *is asymptotically feasible, and every asymptotically feasible point is an asymptotically complementary solution.*

Proof. See [1] for the proof. □

Hence, Theorem 2.1 implies that the objective function in (2.4) is redundant, and hence the problem is a feasibility problem.

An asymptotically complementary solution $(x^*, z^*, \tau^*, y^*, \kappa^*)$ is said to be maximally complementary if the number of positive coordinates in $(z^*, \tau^*, y^*, \kappa^*)$ is maximal among asymptotically complementary solutions. Using this definition we can state the following theorem.

THEOREM 2.2. *Let* $(x^*, z^*, \tau^*, y^*, \kappa^*)$ *be any asymptotically feasible and maximally complementary solution to* (2.4). *Equation* (2.3) *has a feasible and complementary solution if and only if* $\tau^* > 0$. *Furthermore, in this case* $(x^*, y^*, z^*)/\tau^*$ *is an optimal solution to* (2.3).

Proof. See [1] for the proof. □

Therefore, in the case $\tau^* > 0$ it can be concluded that (2.1) has an optimal solution. On the other hand if $\kappa^* > 0$, then it can be concluded that a primal-dual optimal solution to (2.1) having zero duality gap does not exist. Moreover, using the following lemma it may be possible to conclude that either the primal or the dual problem is infeasible.

LEMMA 2.3. *Let* $(x^k, z^k, \tau^k, y^k, \kappa^k)$ *be any bounded sequence satisfying* (2.6) *such that*

$$\lim_{k\to\infty} (x^k, z^k, \tau^k, y^k, \kappa^k) = (x^*, z^*, \tau^*, y^*, \kappa^*)$$

is an asymptotically feasible and maximally complementary solution to (2.4). *Given*

$$\lim_{k\to\infty} -(x^k)^T \nabla_x L(x^k/\tau^k, y^k/\tau^k)^T - (y^k)^T a(x^k/\tau^k) = \kappa^* > 0, \tag{2.7}$$

then

$$\lim_{k\to\infty} \sup\ (\nabla a(x^k/\tau^k)(x^k/\tau^k) - a(x^k/\tau^k))^T (y^k) > 0 \tag{2.8}$$

or

$$\lim_{k\to\infty} \sup -\nabla c(x^k/\tau^k) x^k > 0 \tag{2.9}$$

holds true.

Moreover, if

$$\lim_{k\to\infty} \tau^k \nabla c(x^k/\tau^k) = 0, \tag{2.10}$$

then the primal problem (2.1) *is infeasible if* (2.8) *holds and the dual problem* (2.2) *is infeasible if* (2.9) *holds.*

Proof. Note that $\kappa^* > 0$ implies that $\tau^* = 0$ by complementarity. Furthermore, if (2.7) is true, then either (2.8) or (2.9) must be true. Now suppose (2.10) holds.

Assume first that (2.8) holds and the primal problem (2.1) has a feasible solution. Let $\bar{x}$ be any feasible solution, and since a is concave, then

$$a(\bar{x}) \leq a(x^k/\tau^k) + \nabla a(x^k/\tau^k)(\bar{x} - x^k/\tau^k),$$

which leads to the contradiction

$$\begin{aligned} 0 &\le \liminf_{k\to\infty} \ (y^k)^T a(\bar{x}) \\ &\le \liminf_{k\to\infty} \ (y^k)^T (a(x^k/\tau^k) + \nabla a(x^k/\tau^k)(\bar{x} - x^k/\tau^k)) \\ &= \liminf_{k\to\infty} \ -(y^k)^T (\nabla a(x^k/\tau^k)(x^k/\tau^k) - a(x^k/\tau^k)) \\ &< 0. \end{aligned} \tag{2.11}$$

Here the equation follows from the fact that assumption (2.10) and the first equation of (2.5) together imply

$$\lim_{k\to\infty} \nabla a(x^k/\tau^k)^T y^k = 0.$$

Hence, if (2.8) holds, then (2.1) must be infeasible.

Assume next that (2.9) is true and the dual problem (2.2) has a solution denoted $(\bar{x}, \bar{y})$. By convexity we have that

$$\begin{aligned} c(0) &\ge c(x^k/\tau^k) + \nabla c(x^k/\tau^k)(0 - x^k/\tau^k) \quad \forall k, \\ c(x^k/\tau^k) &\ge c(\bar{x}) + \nabla c(\bar{x})(x^k/\tau^k - \bar{x}) \quad \forall k, \\ a(x^k/\tau^k) &\le a(\bar{x}) + \nabla a(\bar{x})(x^k/\tau^k - \bar{x}) \quad \forall k. \end{aligned}$$

This implies

$$\begin{aligned} c(0) - \nabla c(x^k/\tau^k)(0 - x^k/\tau^k) - c(\bar{x}) &\ge c(x^k/\tau^k) - c(\bar{x}) \\ &\ge \nabla c(\bar{x})(x^k/\tau^k - \bar{x}) \\ &= (\nabla a(\bar{x})^T \bar{y})^T (x^k/\tau^k - \bar{x}) \\ &\ge \bar{y}^T (a(x^k/\tau^k) - a(\bar{x})). \end{aligned}$$

Therefore,

$$\begin{aligned} \tau^k (c(0) - \nabla c(x^k/\tau^k)(0 - x^k/\tau^k) - c(\bar{x})) &\ge \tau^k (c(x^k/\tau^k) - c(\bar{x})) \\ &\ge \tau^k \bar{y}^T (a(x^k/\tau^k) - a(\bar{x})). \end{aligned} \tag{2.12}$$

Given the assumptions, we have that

$$\liminf_{k\to\infty} \ \tau^k (c(0) - \nabla c(x^k/\tau^k)(0 - x^k/\tau^k) - c(\bar{x})) < 0$$

and

$$\limsup_{k\to\infty} \ \tau^k \bar{y}^T (a(x^k/\tau^k) - a(\bar{x})) \ge 0,$$

because $\bar{y} \ge 0$ and $\lim_{k\to\infty} \tau^k a(x^k/\tau^k) \ge 0$. Therefore, taking the limit on both sides of (2.12) leads to a contradiction, implying that the dual problem (2.2) is infeasible. ∎

3. Applications. In this section we will show that Lemma 2.3 can be strengthened in the case of quadratic and quadratically constrained optimization problems.

3.1. Quadratically constrained quadratic optimization. A quadratically constrained optimization problem can be stated as

$$\begin{aligned} &\text{minimize} && \tfrac{1}{2} x^T Q^0 x + c^T x \\ &\text{subject to} && \tfrac{1}{2} x^T Q^i x + a_{i:} x \ge b_i, \quad i = 1, \ldots, m, \end{aligned} \tag{3.1}$$

where $a_{i:}$ is the ith row of A. It is assumed Q^0 and $-Q^i \in R^{n\times n}$ are symmetric positive semidefinite matrices $\forall i$'s. Moreover, $A \in R^{m\times n}$ and all other quantities have conforming dimensions. The dual problem corresponding to (3.1) is

$$\begin{aligned} &\text{maximize} \quad b^T y - \left(\frac{1}{2}x^T Q^0 x - \sum_{i=1}^{m} y_i \frac{1}{2} x^T Q^i x\right) \\ &\text{subject to} \quad Q^0 x + c - A^T y - \sum_{i=1}^{m} y_i Q^i x = 0, \\ &\qquad\qquad y \geq 0, \end{aligned} \tag{3.2}$$

and the associated homogeneous model is

$$\begin{aligned} Q^0 x + c\tau - A^T y - \sum_{i=1}^{m} \frac{y_i}{\tau} Q^i x &= 0, \\ \frac{x^T Q^i x}{2\tau} + a_{i:}x - b_i \tau &= z_i,\ i = 1, \ldots, m, \\ -\frac{x^T Q^0 x}{\tau} - c^T x + \sum_{i=1}^{m} \frac{y_i}{2\tau} \frac{x^T Q^i x}{\tau} + b^T y &= \kappa, \\ z, y, \tau, \kappa &\geq 0. \end{aligned} \tag{3.3}$$

Using the special structure of (3.1) and Lemma 2.3 we can state the following lemma.

LEMMA 3.1. *Let* $(x^k, z^k, \tau^k, y^k, \kappa^k)$ *be any bounded sequence satisfying* (2.6) *such that*

$$\lim_{k\to\infty} (x^k, z^k, \tau^k, y^k, \kappa^k) = (x^*, z^*, \tau^*, y^*, \kappa^*)$$

is an asymptotically feasible and maximally complementary solution to (3.3). *Given*

$$\lim_{k\to\infty} \left(-\frac{(x^k)^T Q^0 x^k}{\tau^k} - c^T x^k + \sum_{i=1}^{m} \frac{y_i^k}{2\tau^k} \frac{(x^k)^T Q^i x^k}{\tau^k} + b^T y^k\right) = \kappa^* > 0, \tag{3.4}$$

then at least one of

$$b^T y^* + \lim_{k\to\infty} \sup \sum_{i=1}^{m} \frac{y_i^k}{2\tau^k} \frac{(x^k)^T Q^i x^k}{\tau^k} > 0 \tag{3.5}$$

or

$$c^T x^* < 0 \tag{3.6}$$

holds true. The primal problem (3.1) *is infeasible if* (3.5) *holds. Moreover, the dual problem* (3.2) *is infeasible if* (3.6) *holds.*

Proof. It can be verified that

$$\frac{(x^k)^T Q^0 x^k}{\tau^k} \geq 0 \quad \forall k.$$

Hence it follows that if (3.4) holds, then at least one of the conditions (3.5) or (3.6) is true.

First we prove an intermediate result. We have that

$$\begin{aligned}0 &= \lim_{k\to\infty}(x^k)^T\left(Q^0x^k + c\tau^k - A^Ty^k - \sum_{i=1}^m \frac{y_i^k}{\tau^k}Q^ix^k\right)\\ &= \lim_{k\to\infty}\left((x^k)^T\left(Q^0x^k - \sum_{i=1}^m \frac{y_i^k}{2\tau^k}Q^ix^k + c\tau^k\right) - \sum_{i=1}^m y_i^k\left(\frac{(x^k)^TQ^ix^k}{2\tau^k} + a_{i:}x^k\right)\right)\\ &= \lim_{k\to\infty}(x^k)^T\left(Q^0x^k - \sum_{i=1}^m \frac{y_i^k}{2\tau^k}Q^ix^k\right)\end{aligned} \tag{3.7}$$

because $\lim_{k\to\infty}\tau^k = 0$ and

$$\lim_{k\to\infty} y_i^k\left(\frac{(x^k)^TQ^ix^k}{2\tau^k} + a_{i:}x^k\right) = \lim_{k\to\infty} y_i^k(z_i^k + b_i\tau^k) = 0.$$

Given the convexity assumptions,

$$(x^k)^TQ^0x^k \geq 0.$$

This fact in combination with (3.7) leads to the conclusion that

$$(x^*)^TQ^0x^* = \lim_{k\to\infty}(x^k)^TQ^0x^k = 0 \qquad \text{and} \qquad Q^0x^* = \lim_{k\to\infty} Q^0x^k = 0.$$

This, combined with the facts

$$\lim_{k\to\infty}\left(Q^0x^k + c\tau^k - A^Ty^k - \sum_{i=1}^m \frac{y_i^k}{\tau^k}Q^ix^k\right) = 0$$

and $\lim_{k\to\infty}\tau^k = 0$, leads to the conclusion

$$\lim_{k\to\infty}\left(-A^Ty^k - \sum_{i=1}^m \frac{y_i^k}{\tau^k}Q^ix^k\right) = 0. \tag{3.8}$$

First, assume that (3.5) is the case and (3.1) has a feasible solution $\bar{x}$. Therefore,

$$\begin{aligned}0 &\leq \sum_{i=1}^m y_i^k\left(\frac{1}{2}\bar{x}^TQ^i\bar{x} + a_{i:}\bar{x} - b_i\right)\\ &\leq \sum_{i=1}^m y_i^k\left(\frac{1}{2}\frac{(x^k)^TQ^ix^k}{(\tau^k)^2} + a_{i:}\bar{x} - b_i + (Q^ix^k/\tau^k)^T(\bar{x} - x^k/\tau^k)\right)\\ &= \sum_{i=1}^m y_i^k\left(-\frac{1}{2}\frac{(x^k)^TQ^ix^k}{(\tau^k)^2} - b_i\right) + \bar{x}^T\left(A^Ty^k + \sum_{i=1}^m \frac{y_i^k}{\tau^k}Q^ix^k\right),\end{aligned}$$

where the second inequality follows from the concavity assumption. This fact, in combination with (3.5) and (3.8), gives rise to the contradiction

$$\begin{aligned}0 &\leq \lim_{k\to\infty}\inf\left(\sum_{i=1}^m y_i^k\left(-\frac{1}{2}\frac{(x^k)^TQ^ix^k}{(\tau^k)^2} - b^Ty^k\right) + \bar{x}^T\left(A^Ty^k + \sum_{i=1}^m \frac{y_i^k}{\tau^k}Q^ix^k\right)\right)\\ &= -b^Ty^* - \lim_{k\to\infty}\sup\sum_{i=1}^m \frac{y_i^k}{2\tau^k}\frac{(x^k)^TQ^ix^k}{\tau^k}\\ &< 0.\end{aligned}$$

Hence, given (3.5) then (3.1) must be infeasible.

Second, assume that (3.6) holds and the dual problem has a feasible solution denoted $(\bar{x}, \bar{y})$. Then

$$(x^*)^T \left(Q^0 \bar{x} + c - A^T \bar{y} - \sum_{i=1}^{m} \bar{y}_i Q^i \bar{x} \right) = 0,$$

from which we obtain

$$c^T x^* = (x^*)^T \left(A^T \bar{y} + \sum_{i=1}^{m} \bar{y}_i Q^i \bar{x} \right). \tag{3.9}$$

Since

$$\begin{aligned} \lim_{k\to\infty} \tau^k \left(\frac{(x^k)^T Q^i x^k}{2\tau^k} + a_{i:} x^k - \tau^k b_i \right) &= \lim_{k\to\infty} \tau^k z_i^k \\ &= 0 \end{aligned} \tag{3.10}$$

then

$$\lim_{k\to\infty} (x^k)^T Q^i x^k = (x^*)^T Q^i x^* = 0$$

is true. This fact, in combination with (3.9), leads to the contradiction

$$\begin{aligned} 0 &> c^T x^* \\ &= (x^*)^T A^T \bar{y} \\ &\geq 0 \end{aligned}$$

because $\bar{y} \geq 0$ and

$$\begin{aligned} a_{i:} x^* &= \lim_{k\to\infty} a_{i:} x^k \\ &= \lim_{k\to\infty} \left(z_i^k - \frac{(x^k)^T Q^i x^k}{\tau^k} + b_i \tau^k \right) \\ &\geq 0. \end{aligned}$$

Therefore, we can conclude that if (3.6) holds, then (3.2) is infeasible. □

3.2. Quadratic optimization. An important special case of quadratically constrained optimization is quadratic optimization, i.e., the case where

$$Q^i = 0, \quad i = 1, \ldots, m.$$

In this case the homogeneous model has the form

$$\begin{aligned} Q^0 x + c\tau - A^T y &= 0, \\ Ax - b\tau &= z, \\ -\frac{x^T Q^0 x}{\tau} - c^T x + b^T y &= \kappa, \\ x, \tau, y, \kappa, &\geq 0. \end{aligned} \tag{3.11}$$

Using Lemma 3.1 we can state the following lemma.

LEMMA 3.2. *Let* $(x^k, z^k, \tau^k, y^k, \kappa^k)$ *be any bounded sequence satisfying* (2.6) *such that*

$$\lim_{k\to\infty} (x^k, z^k, \tau^k, y^k, \kappa^k) = (x^*, z^*, \tau^*, y^*, \kappa^*)$$

is an asymptotically feasible and maximally complementary solution to (3.11). *Given*

$$\lim_{k\to\infty} \left(-\frac{(x^k)^T Q^0 x^k}{\tau^k} - c^T x^k + b^T y^k \right) = \kappa^* > 0, \tag{3.12}$$

then

$$b^T y^* > 0 \tag{3.13}$$

or

$$c^T x^* < 0 \tag{3.14}$$

holds true. The primal problem (3.1) *is infeasible if* (3.13) *holds. Moreover, the dual problem* (3.2) *is infeasible if* (3.14) *holds.*

Proof. This follows immediately from Lemma 3.1. From the proof, note that $Q^0 x^* = 0$.

It can be observed that in the case where the primal problem (3.1) is concluded to be infeasible, y^* satisfies

$$A^T y^* = 0, \quad b^T y^* > 0, \quad y^* \geq 0, \tag{3.15}$$

which by Farkas's lemma implies that

$$\{x : Ax \geq b\} = \emptyset, \tag{3.16}$$

i.e., the problem is infeasible. Observe if the dual problem has a feasible solution; then $(0, y^*)$ is a ray along which dual objective value tends to $+\infty$, i.e., the dual problem (3.2) is unbounded.

Similarly, it can be observed that in the case where the dual problem (3.2) is concluded to be infeasible, an x^* is known such that

$$Ax^* \geq 0, \quad Q^0 x^* = 0, \quad c^T x^* < 0, \tag{3.17}$$

which (once again using Farkas's lemma) implies that

$$\{(x, y) : \ Q^0 x + c - A^T y = 0, \ y \geq 0\} = \emptyset.$$

Note also that if the primal problem has a feasible solution, then x^* is a ray along which the primal objective value tends to $-\infty$, i.e., the primal problem (3.1) is unbounded. ∎

Acknowledgments. The author is thankful for comments made by Tamás Terlaky, the anonymous referees, and the associate editor, Mike Todd, regarding this article.

REFERENCES

[1] E. D. Andersen and Y. Ye, *On a homogeneous algorithm for a monotone complementarity problem with nonlinear equality constraints*, in Complementarity and Variational Problems: State of the Art, Proceedings in Applied Mathematics 92, M. C. Ferris and J.-S. Pang, eds., SIAM, Philadelphia, 1997, pp. 1–11.

[2] E. D. Andersen and Y. Ye, *On a homogeneous algorithm for the monotone complementarity problem,* Math. Programming, 84 (1999), pp. 375–399.

[3] 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, 1956, pp. 53–97.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 389–410

A SMOOTHING NEWTON METHOD FOR MINIMIZING A SUM OF EUCLIDEAN NORMS*

LIQUN QI†‡ AND GUANGLU ZHOU†

Abstract. We consider the problem of minimizing a sum of Euclidean norms, $f(x) = \sum_{i=1}^{m} \|b_i - A_i^T x\|$. This problem is a nonsmooth problem because f is not differentiable at a point x when one of the norms is zero. In this paper we present a smoothing Newton method for this problem by applying the smoothing Newton method proposed by Qi, Sun, and Zhou [*Math. Programming*, 87 (2000), pp. 1–35] directly to a system of strongly semismooth equations derived from primal and dual feasibility and a complementarity condition. This method is globally and quadratically convergent. As applications to this problem, smoothing Newton methods are presented for the Euclidean facilities location problem and the Steiner minimal tree problem under a given topology. Preliminary numerical results indicate that this method is extremely promising.

Key words. sum of norms, smoothing Newton method, semismoothness, Euclidean facilities location, shortest networks, Steiner minimum trees

AMS subject classifications. 90C33, 90C30, 65H10

PII. S105262349834895X

1. Introduction.

Consider the problem of minimizing a sum of Euclidean norms (MSNs):

$$\min_{x \in R^n} \sum_{i=1}^{m} \|b_i - A_i^T x\|, \tag{1.1}$$

where $b_1, b_2, \ldots, b_m \in R^d$ are column vectors in the Euclidean d-space, $A_1, A_2, \ldots, A_m \in R^{n \times d}$ are $n \times d$ matrices with each having full column rank, $n \leq m(d-1)$, and $\|r\|$ represents the Euclidean norm $(\sum_{i=1}^{m} r_i^2)^{1/2}$. Let $A = [A_1, A_2, \ldots, A_m]$. In what follows we always assume that A has rank n. Let

$$f(x) = \sum_{i=1}^{m} \|b_i - A_i^T x\|. \tag{1.2}$$

It is clear that $x = 0$ is an optimal solution to problem (1.1) when all of the b_i are zero. Therefore, we assume in the rest of this paper that not all of the b_i are zero. Problem (1.1) is a convex programming problem, but its objective function f is not differentiable at any point x when some $b_i - A_i^T x = 0$. Three special cases of this problem are the Euclidean single facility location (ESFL) problem, the Euclidean multifacility location (EMFL) problem, and the Steiner minimal tree (SMT) problem under a given topology.

Many algorithms have been designed to solve problem (1.1). For the ESFL problem, Weiszfeld [34] gave a simple iterative algorithm in 1937. Later, a number of

*Received by the editors November 30, 1998; accepted for publication (in revised form) March 23, 2000; published electronically October 18, 2000. This work was supported by the Australian Research Council.

http://www.siam.org/journals/siopt/11-2/34895.html

†School of Mathematics, The University of New South Wales, Sydney 2052, Australia (zhou@maths.unsw.edu.au).

‡Current address: Department of Applied Mathematics, Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (maqilq@polyu.edu.hk).

important results were obtained along this line; see [6, 7, 13, 22, 24, 30, 31, 33]. Practical algorithms for solving these problems began with the work of Calamai and Conn [4, 5] and Overton [25], where they proposed projected Newton algorithms with the quadratic rate of convergence. The essential idea of these algorithms is as follows. In each iteration a search direction is computed by Newton's method projected into a linear manifold along which f is locally differentiable. The advantage of this method is the quadratic convergence and the avoidance of approximation techniques for f. However, it is difficult to use this method due to the dynamic structure of the linear manifold into which the method projects the search direction. Every time terms are added and deleted from the active set, the size and the sparse structure of the problem changes.

More recently, Andersen [1] used the HAP idea [13] to smooth the objective function by introducing a perturbation $\varepsilon > 0$ and applied a Newton barrier method for solving this problem. Andersen et al. [3] proposed a primal-dual interior-point method based on the ε-perturbation and presented impressive computational results. Xue and Ye [35, 36] presented polynomial-time primal-dual potential reduction algorithms by transforming this problem into a standard convex programming problem in conic form. However, these methods do not possess second-order convergence.

In recent years, two major reformulation approaches, the nonsmooth approach and the smoothing approach, for solving nonlinear complementarity problems (NCPs) and box constrained variational inequality problems (BVIPs), have been rapidly developed based on NCP and BVIP functions, e.g., see [8, 9, 10, 11, 14, 15, 19, 21, 26, 28, 32, 37, 38] and references therein. In particular, Jiang and Qi [21] and De Luca, Facchinei, and Kanzow [14] proposed globally and superlinearly (quadratically) convergent nonsmooth Newton methods for NCPs, which only require solving a system of linear equations to determine the search direction at each iteration. A globally and superlinearly (quadratically) convergent smoothing Newton method was proposed by Chen, Qi, and Sun in [10], where the authors exploited a Jacobian consistence property and applied this property to an infinite sequence of smoothing approximation functions to get high-order convergence. On the other hand, Hotta and Yoshise [20], Qi, Sun, and Zhou [28], and Jiang [19] proposed smoothing methods for NCPs and BVIPs by treating the smoothing parameter as a variable, in which the smoothing parameter is driven to zero automatically and no additional procedure for adjusting the smoothing parameter is necessary. Some regularized versions of the method in [28] were proposed in [26, 32, 38] for NCPs and BVIPs.

In this paper we present a smoothing Newton method for problem (1.1) by applying the smoothing Newton method proposed by Qi, Sun, and Zhou [28] directly to a system of strongly semismooth equations derived from primal and dual feasibility and a complementarity condition and prove that this method is globally and quadratically convergent. Numerical results indicate that this method is extremely promising.

This paper is organized as follows. In section 2, we transform primal and dual feasibility and a complementarity condition derived from problem (1.1) and its dual problem into a system of strongly semismooth equations. Some smooth approximations to the projection operator on the unit ball are given in section 3. In section 4, we present a smoothing Newton method for solving problem (1.1) and prove that this method is globally and quadratically convergent. In section 5, we discuss applications to the ESFL problem, the EMFL problem, and the SMT problem. In section 6, we present some numerical results. We conclude this paper in section 7.

Concerning notation, for a continuously differentiable function $F : R^n \to R^m$, we

denote the Jacobian of F at $x \in R^n$ by $F'(x)$, whereas the transposed Jacobian is denoted as $\nabla F(x)$. In particular, if $m = 1$, the gradient $\nabla F(x)$ is viewed as a column vector.

Let $F : R^n \to R^m$ be a locally Lipschitzian vector function. By Rademacher's theorem, F is differentiable almost everywhere. Let Ω_F denote the set of points where F is differentiable. Then the B-subdifferential of F at $x \in R^n$ is defined as

$$\partial_B F(x) = \{ \lim_{\substack{x^k \to x \\ x^k \in \Omega_F}} \nabla F(x^k)^T \}, \tag{1.3}$$

while Clarke's generalized Jacobian of F at x is defined as

$$\partial F(x) = \text{conv}\partial_B F(x), \tag{1.4}$$

(see [12, 27, 29]). F is called *semismooth* at x if F is directionally differentiable at x and for all $V \in \partial F(x + h)$ and $h \to 0$,

$$F'(x; h) = Vh + o(\|h\|); \tag{1.5}$$

F is called *p-order semismooth*, $p \in (0, 1]$, at x if F is semismooth at x and for all $V \in \partial F(x + h)$ and $h \to 0$,

$$F'(x; h) = Vh + O(\|h\|^{1+p}); \tag{1.6}$$

F is called *strongly semismooth* at x if F is 1-order semismooth at x. F is called a (strongly) semismooth function if it is (strongly) semismooth everywhere (see [27, 29]). In particular, a PC^2 (piecewise twice continuously differentiable) function is a strongly semismooth function. Here, $o(\|h\|)$ stands for a vector function $e : R^n \to R^m$, satisfying

$$\lim_{h \to 0} \frac{e(h)}{\|h\|} = 0,$$

while $O(\|h\|^2)$ stands for a vector function $e : R^n \to R^m$, satisfying

$$\|e(h)\| \le M\|h\|^2$$

for all h satisfying $\|h\| \le \delta$ and some $M > 0$ and $\delta > 0$.

LEMMA 1.1 (see [29]).

(i) *If F is semismooth at x, then for any $h \to 0$,*

$$F(x + h) - F(x) - F'(x; h) = o(\|h\|);$$

(ii) *if F is p-order semismooth at x, then for any $h \to 0$,*

$$F(x + h) - F(x) - F'(x; h) = O(\|h\|^{1+p}).$$

THEOREM 1.2 (see [16, Theorem 19]). *Suppose that the function $\mathcal{F} : R^n \to R^m$ is p-order semismooth at x and the function $\mathcal{G} : R^m \to R^l$ is p-order semismooth at $\mathcal{F}(x)$. Then the composite function $\mathcal{H} = \mathcal{G} \circ \mathcal{F}$ is p-order semismooth at x.*

For a set $\mathcal{A}$, $|\mathcal{A}|$ denotes the cardinality of the set $\mathcal{A}$. We denote $x^T x$ by x^2, for a vector $x \in R^n$, i.e., $x^2 = \|x\|^2$. For $A \in R^{n \times m}$, $\|A\|$ denotes the induced norm, i.e., $\|A\| = \max\{\|Au\| : u \in R^n, \|u\| = 1\}$. Let I_d denote the $d \times d$ identity matrix. Let $b^T = [b_1^T, \ldots, b_m^T]$, $y = [y_1^T, \ldots, y_m^T]^T \in R^{md}$, $R_+ = \{\varepsilon \in R : \varepsilon \ge 0\}$, and $R_{++} = \{\varepsilon \in R : \varepsilon > 0\}$. Finally, we use $\varepsilon \downarrow 0^+$ to denote the case that a positive scalar ε tends to 0.

2. Some preliminaries. In [1, 3], Andersen et al. studied the duality for problem (1.1) and presented some efficient algorithms for solving it. In this section we will transform three sets of equations—primal feasibility, dual feasibility, and the complementarity condition derived from problem (1.1) and its dual problem—into a system of strongly semismooth equations. This transformation is very important for the method proposed in this paper.

LEMMA 2.1. *Assume that A has rank n. Then the set of solutions to the problem* (1.1) *is bounded.*

Proof. It follows from the assumed rank of A that

$$\min_{\|x\|=1} \|A^T x\| = \tau > 0. \tag{2.1}$$

From (2.1) we obtain

$$\|A^T x\| \geq \tau \|x\|. \tag{2.2}$$

This shows that the set of solutions to the problem (1.1) is bounded. □

The dual of the problem (1.1) has the form (see [1])

$$\max_{y \in Y} b^T y, \tag{2.3}$$

where

$$Y = \left\{ y = [y_1^T, \ldots, y_m^T]^T \in R^{md} \ : \ y_i \in R^d, \ \|y_i\| \leq 1, \ i = 1, \ldots, m; Ay = 0 \right\}. \tag{2.4}$$

THEOREM 2.2 (see [1]). *Let $x \in R^n$, $y \in Y$ and let $x^* \in R^n$, $y^* \in Y$ be optimal solutions to problems* (1.1) *and* (2.3)*, respectively. Then*

$$(a) \ b^T y \leq \sum_{i=1}^m \|b_i - A_i^T x\| \quad \text{(weak duality)}$$

and

$$(b) \ b^T y^* = \sum_{i=1}^m \|b_i - A_i^T x^*\| \quad \text{(strong duality)}.$$

DEFINITION 2.3 (see [1]). *A solution $x \in R^n$ and a solution $y \in Y$ are called ε-optimal to problems* (1.1) *and* (2.3) *if*

$$\sum_{i=1}^m \|b_i - A_i^T x\| - b^T y \leq \varepsilon.$$

From Theorem 2.2 we have that (x^*, y^*) is a pair of optimal solutions to problems (1.1) and (2.3) if and only if (x^*, y^*) is a solution to the following system:

$$\begin{cases} Ay = 0, \\ \|y_i\| \leq 1, \ i = 1, \ldots, m, \\ \displaystyle\sum_{i=1}^m \|b_i - A_i^T x\| - b^T y = 0. \end{cases} \tag{2.5}$$

Suppose that $y \in R^{md}$, satisfying that $Ay = 0$ and $\|y_i\| \le 1, i = 1, 2, \ldots, m$. Then

$$\begin{aligned}
\sum_{i=1}^{m} \|b_i - A_i^T x\| - b^T y &= \sum_{i=1}^{m} \|b_i - A_i^T x\| - \sum_{i=1}^{m} b_i^T y_i \\
&= \sum_{i=1}^{m} \left(\|b_i - A_i^T x\| - b_i^T y_i \right) \\
&= \sum_{i=1}^{m} \left(\|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i + x^T (A_i y_i) \right) \\
&= \sum_{i=1}^{m} \left(\|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i \right) + x^T (Ay) \\
&= \sum_{i=1}^{m} \left(\|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i \right),
\end{aligned}$$

and for $i = 1, 2, \ldots, m$,

$$\|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i \ge 0.$$

So the duality gap is zero if and only if

$$\|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i = 0$$

for $i = 1, \ldots, m$. Then (2.5) is equivalent to

$$\text{(2.6)} \qquad \begin{cases} Ay = 0, \\ \|y_i\| \le 1, \ i = 1, \ldots, m, \\ \|b_i - A_i^T x\| - (b_i - A_i^T x)^T y_i = 0, \ i = 1, \ldots, m. \end{cases}$$

LEMMA 2.4. *Let $r, s \in R^d$. If $\|s\| \le 1$, then $\|r\| = r^T s$ if and only if $r - \|r\| s = 0$.*

Proof. Suppose $\|r\| = r^T s$. If $r = 0$, then $r - \|r\| s = 0$. If $r \ne 0$, then

$$\|r\| = r^T s \le \|r\| \|s\|.$$

So $\|s\| = 1$. Then $(r - \|r\| s)^2 = \|r\|^2 - 2\|r\| r^T s + \|r\|^2 \|s\|^2 = 0$, i.e., $r - \|r\| s = 0$.

On the other hand, if $r = 0$, then $\|r\| = r^T s$. If $r - \|r\| s = 0$ and $r \ne 0$, then $\|s\| = 1$ and $r^T s - \|r\| s^T s = r^T s - \|r\| = 0$, i.e., $\|r\| = r^T s$. ∎

From the above lemma (2.6) is equivalent to

$$\text{(2.7)} \qquad \begin{cases} Ay = 0, \\ \|y_i\| \le 1, \ i = 1, \ldots, m, \\ (b_i - A_i^T x) - \|b_i - A_i^T x\| y_i = 0, \ i = 1, \ldots, m. \end{cases}$$

It follows from (2.7) that if (x^*, y^*) is a pair of optimal solutions to problems (1.1) and (2.3), then for $i = 1, \ldots, m$, either $b_i - A_i^T x^* = 0$ or $\|y_i^*\| = 1$. We say strict complementarity holds at (x^*, y^*) if, for each i, only one of these two conditions holds.

Let $B = \{s \in R^d : \|s\| \le 1\}$ and let $\Pi_B(s)$ be the projection operator onto B.

LEMMA 2.5. *Let $r, s \in R^d$. Then $s = \Pi_B(s + r)$ if and only if $\|s\| \le 1$ and $\|r\| = r^T s$.*

Proof. Suppose that $s = \Pi_B(s + r)$. Then $\|s\| \leq 1$ and

$$r^T(s - s^*) \geq 0 \text{ for any } \ s^* \in B.$$

It follows that $\|r\| = \max_{\|s^*\| \leq 1} r^T s^* \leq r^T s$. So $\|r\| = r^T s$.

On the other hand, if $\|r\| = r^T s$ and $\|s\| \leq 1$, then for any $s^* \in B$,

$$r^T(s - s^*) \geq 0$$

because $\|r\| = \max_{\|s^*\| \leq 1} r^T s^*$. Hence $s = \Pi_B(s + r)$. □

It follows from the above lemma that (2.6) is equivalent to

$$\begin{cases} Ay = 0, \\ \\ y_i - \Pi_B(y_i + b_i - A_i^T x) = 0, \ i = 1, \ldots, m. \end{cases} \tag{2.8}$$

Define $F : R^{n+md} \to R^{n+md}$ by

$$\begin{cases} F_j(x, y) = (Ay)_j, \ j = 1, \ldots, n, \\ \\ F_j(x, y) = y_i - \Pi_B(y_i + b_i - A_i^T x), \\ \qquad\qquad j = n + il, \quad i = 1, \ldots, m, \quad l = 1, \ldots, d. \end{cases} \tag{2.9}$$

Then we have that (x^*, y^*) is a pair of optimal solutions to problems (1.1) and (2.3) if and only if (x^*, y^*) is a solution to the following equation:

$$F(x, y) = 0. \tag{2.10}$$

From Lemma 2.1, (2.3), and (2.4), we have the following.

LEMMA 2.6. *All solutions to* (2.10) *are bounded.*

Clearly, F is not continuously differentiable, but we can prove that it is strongly semismooth.

THEOREM 2.7. *The function F defined in* (2.9) *is strongly semismooth on $R^n \times R^{md}$.*

Proof.

$$\Pi_B(s) = \begin{cases} \dfrac{s}{\|s\|} & \text{if } \|s\| > 1, \\ s & \text{if } \|s\| \leq 1. \end{cases}$$

Then

$$\Pi_B(s) = \frac{s}{\max\{1, \|s\|\}} = \frac{s}{1 + \max\{0, (\|s\| - 1)\}}. \tag{2.11}$$

Since the function h, defined by $h(x) = \|x\|$, where $x \in R^d$, max functions, and linear functions are all strongly semismooth, from Theorem 1.2 F is strongly semismooth on $R^n \times R^{md}$. □

3. Smooth approximations to $\Pi_B(s)$. In this section we will present some smooth approximations to the projection operator $\Pi_B(s)$ and study the properties of these smooth approximations.

In [9], Chen and Mangasarian presented a class of smooth approximations to the function $\max\{0, \cdot\}$. Similarly, we can give a class of smooth approximations to the

projection operator $\Pi_B(s)$ defined in (2.11). For simplicity, throughout this paper we use only the following smooth function to approximate $\Pi_B(s)$, which is based on the neural networks smooth function and defined as follows:

$$\phi(t,s) = \frac{s}{q(t,s)},\ (t,s) \in R_{++} \times R^d, \tag{3.1}$$

where $q(t,s) = t\ln(e^{\frac{1}{t}} + e^{\frac{\sqrt{\|s\|^2+t^2}}{t}})$.

PROPOSITION 3.1. *$\phi(t,s)$ has the following properties:*

(i) *For any given $t > 0$, $\phi(t,s)$ is continuously differentiable;*

(ii) *$\phi(t,s) \in \mathrm{int}B$, for any given $t > 0$;*

(iii) *$|\phi(t,s) - \Pi_B(s)| \le (\ln 2 + 1)t$;*

(iv) *for any given $t > 0$,*

$$\nabla\phi_s(t,s) = \frac{1}{q(t,s)}I_d - \frac{ss^T}{q(t,s)^2(1 + e^{(1-\sqrt{\|s\|^2+t^2})/t})\sqrt{\|s\|^2+t^2}}, \tag{3.2}$$

and $\nabla\phi_s(t,s)$ is symmetric, positive definite and $\|\nabla\phi_s(t,s)\| < 1$;

(v) *for any given $s \in R^d$ and $t > 0$,*

$$\nabla\phi_t(t,s) = -\frac{1}{q^2(t,s)}\left(\ln e(t,s) - \frac{e^{\frac{1}{t}}}{te(t,s)} + \frac{\|s\|^2 e^{\frac{\sqrt{\|s\|^2+t^2}}{t}}}{t\sqrt{\|s\|^2+t^2}e(t,s)}\right)s, \tag{3.3}$$

where $e(t,s) = e^{\frac{1}{t}} + e^{\frac{\sqrt{\|s\|^2+t^2}}{t}}$.

Proof. It is clear that (i) holds. For any $t > 0$, $q(t,s) > \max\{1, \|s\|\}$. So (ii) holds. By Proposition 2.2(ii) in [9],

$$|q(t,s) - \max\{1,\|s\|\}| \le (\ln 2 + 1)t.$$

Hence,

$$\begin{aligned}\|\phi(t,s) - \Pi_B(s)\| &= \frac{\|s\||q(t,s) - \max\{1,\|s\|\}|}{q(t,s)\max\{1,\|s\|\}} \\ &\le |q(t,s) - \max\{1,\|s\|\}| \\ &\le (\ln 2 + 1)t.\end{aligned}$$

By simple computation, (iv) and (v) hold. ∎

Let

$$p(t,s) = \begin{cases} \phi(|t|,s) & \text{if } t \ne 0, \\ \Pi_B(s) & \text{if } t = 0. \end{cases} \tag{3.4}$$

From Proposition 3.1 of [28] and Theorem 1.2 we have the following.

PROPOSITION 3.2. *$p(t,s)$ is a strongly semismooth function on $R \times R^d$.*

It follows from Proposition 3.1 that the following proposition holds.

PROPOSITION 3.3.

(i) *If $\|s^*\| < 1$, then*

$$\lim_{\substack{t^k \downarrow 0^+ \\ s^k \to s^*}} \nabla\phi_s(t^k, s^k) = I_d;$$

(ii) *if* $\|s^*\| > 1$, *then*

$$\lim_{\substack{t^k \downarrow 0^+ \\ s^k \to s^*}} \nabla \phi_s(t^k, s^k) = \frac{1}{\|s^*\|} I_d - \frac{1}{\|s^*\|^3} s^* (s^*)^T,$$

which is symmetric, nonnegative definite, and the norm of this matrix is less than 1 *and the rank of this matrix is* $d - 1$.

4. A smoothing Newton method. In this section we will present a smoothing Newton method for solving problem (1.1) by applying the smoothing Newton method proposed by Qi, Sun, and Zhou [28] directly to the system of strongly semismooth equation (2.10) and prove that this method is globally and quadratically convergent.

Define $G : R \times R^n \times R^{md} \to R^{n+md}$ by

$$\begin{cases} G_j(t,x,y) = (Ay)_j - tx_j, \ j = 1, \ldots, n, \\ \\ G_j(t,x,y) = (y_i)_l - (p(t, y_i + b_i - A_i^T x))_l, \\ \qquad\qquad j = n + il, \quad i = 1, \ldots, m, \quad l = 1, \ldots, d. \end{cases} \tag{4.1}$$

Then G is continuously differentiable at any (t,x,y) with $t \neq 0$ and from Theorem 1.2 and Proposition 3.2 it is strongly semismooth on $R \times R^n \times R^{md}$.

Let $z := (t,x,y) \in R \times R^n \times R^{md}$ and define $H : R \times R^n \times R^{md} \to R^{n+md+1}$ by

$$H(z) := \begin{pmatrix} t \\ G(z) \end{pmatrix}. \tag{4.2}$$

Then H is continuously differentiable at any $z \in R_{++} \times R^n \times R^{md}$ and strongly semismooth at any $z \in R \times R^n \times R^{md}$, and $H(t^*, x^*, y^*) = 0$ if and only if $t^* = 0$ and $F(x^*, y^*) = 0$.

Let $p(t, y + b - A^T x) = [p(t, y_1 + b_1 - A_1^T x)^T, \ldots, p(t, y_m + b_m - A_m^T x)^T]^T$.

LEMMA 4.1. *For any* $z = (t,x,y) \in R_{++} \times R^n \times R^{md}$,

$$H'(z) := \begin{pmatrix} 1 & 0 & 0 \\ -x & -tI_n & A \\ E(z) & P(z)A^T & I_{md} - P(z) \end{pmatrix}, \tag{4.3}$$

where

$$E(z) = \nabla p_t(t, y + b - A^T x), \tag{4.4}$$

and

$$P(z) = \mathrm{Diag}(p'_s(t, y_i + b_i - A_i^T x)), \tag{4.5}$$

and $H'(z)$ *is nonsingular.*

Proof. We have that (4.3) holds by simple computation. For any $z = (t,x,y) \in R_{++} \times R^n \times R^{md}$, in order to prove $H'(z)$ is nonsingular, we need to prove only that

$$M = \begin{pmatrix} -tI_n & A \\ P(z)A^T & I_{md} - P(z) \end{pmatrix}$$

is nonsingular. For any $t > 0$ and $(x,y) \in R^n \times R^{md}$, from Proposition 3.1 $P(z)$ is symmetric positive definite and $\|P(z)\| < 1$. Let $Mg = 0$, where $g = (g_1^T, g_2^T)^T \in R^n \times R^{md}$. Then we have

$$-tI_n g_1 + Ag_2 = 0, \tag{4.6}$$

and

$$P(z)A^T g_1 + (I_{md} - P(z))g_2 = 0. \tag{4.7}$$

From (4.7) we have

$$g_2 = -(I_{md} - P(z))^{-1}P(z)A^T g_1. \tag{4.8}$$

Then

$$(tI_n + A(I_{md} - P(z))^{-1}P(z)A^T)g_1 = 0. \tag{4.9}$$

Let

$$B(z) = tI_n + A(I_{md} - P(z))^{-1}P(z)A^T. \tag{4.10}$$

Then $B(z)$ is an $n \times n$ symmetric positive definite matrix because A has full rank. So $g_1 = 0$. Thus $g = 0$. This implies that M is nonsingular. So $H'(z)$ is nonsingular. □

Choose $\bar{t} \in R_{++}$ and $\gamma \in (0,1)$ such that $\gamma\bar{t} < 1$. Let $\bar{z} := (\bar{t}, 0, 0) \in R \times R^n \times R^{md}$. Define the merit function $\psi : R \times R^n \times R^{md} \to R_+$ by

$$\psi(z) := \|H(z)\|^2.$$

ψ is continuously differentiable on $R_{++} \times R^n \times R^{md}$ and strongly semismooth on $R \times R^n \times R^{md}$. Define $\beta : R_+ \times R^n \times R^{md} \to R_+$ by

$$\beta(z) := \gamma \min\{1, \psi(z)\}.$$

Let

$$\Omega := \{z = (t, x, y) \in R \times R^n \times R^{md} \mid t \geq \beta(z)\bar{t}\ \}.$$

Then, because for any $z \in R \times R^n \times R^{md}$, $\beta(z) \leq \gamma < 1$, it follows that for any $(x, y) \in R^n \times R^{md}$,

$$(\bar{t}, x, y) \in \Omega.$$

ALGORITHM 4.1.

Step 0. *Choose constants* $\delta \in (0,1)$ *and* $\sigma \in (0, 1/2)$. *Let* $z^0 := (\bar{t}, x^0, y^0) \in R_{++} \times R^n \times R^{md}$ *and* $k := 0$.

Step 1. *If* $H(z^k) = 0$, *then stop. Otherwise, let* $\beta_k := \beta(z^k)$.

Step 2. *Compute* $\Delta z^k := (\Delta t^k, \Delta x^k, \Delta y^k) \in R \times R^n \times R^{md}$ *by*

$$H(z^k) + H'(z^k)\Delta z^k = \beta_k \bar{z}. \tag{4.11}$$

Step 3. *Let* j_k *be the smallest nonnegative integer* j *satisfying*

$$\psi(z^k + \delta^j \Delta z^k) \leq [1 - 2\sigma(1 - \gamma\bar{t}\ )\delta^j]\psi(z^k). \tag{4.12}$$

Define $z^{k+1} := z^k + \delta^{j_k}\Delta z^k$.

Step 4. *Replace* k *by* $k+1$ *and go to Step* 1.

Remark. We can solve (4.11) in the following way: Let $\Delta t^k = -t^k + \beta_k \bar{t}$. Solve

(4.13)
$$B(z^k)\Delta x^k = -A(I_{md} - P(z^k))^{-1}(y^k - p^k + \Delta t^k E(z^k)) + (Ay^k - (t^k + \Delta t^k)x^k)$$

to get Δx^k, where $B(z^k)$ is defined in (4.10) and $p^k = p(t^k, y^k + b - A^T x^k)$. Then

$$\Delta y^k = -(I_{md} - P(z^k))^{-1}P(z^k)A^T\Delta x^k - (I_{md} - P(z^k))^{-1}(y^k - p^k + \Delta t^k E(z^k)).$$

Equation (4.13) is an n-dimensional symmetric positive definite linear system.

From Proposition 4.5 of [28] and Lemma 4.1 of [32] we have the following.

PROPOSITION 4.2. *Algorithm* 4.1 *is well defined at the kth iteration and generates an infinite sequence* $\{z^k = (t^k, x^k, y^k)\}$. *Moreover,* $0 < t^{k+1} \le t^k \le \bar{t}$ *and* $z^k \in \Omega$.

For any given $t \in R$, define $\psi_t(x, y) : R^n \times R^{md} \to R^+$ by

$$\psi_t(x, y) = \|G(z)\|^2. \tag{4.14}$$

It is easy to see that for any fixed $t \in R_{++}$, ψ_t is continuously differentiable with the gradient given by

$$\nabla\psi_t(x, y) = 2(G'_{(x,y)}(z))^T G(z), \tag{4.15}$$

where

$$G'_{(x,y)}(z) = \begin{pmatrix} -tI_n & A \\ P(z)A^T & I_{md} - P(z) \end{pmatrix}, \tag{4.16}$$

and $P(z)$ is defined in (4.5). By repeating the proof of Lemma 4.1, $G'_{(x,y)}(z)$ is nonsingular at any point $z = (t, x, y) \in R_{++} \times R^n \times R^{md}$. For any $z = (t, x, y) \in R \times R^n \times R^{md}$,

$$\psi(z) = t^2 + \psi_t(x, y). \tag{4.17}$$

It follows from Lemma 2.6 that we have the following.

LEMMA 4.3. *The set* $\mathcal{S} = \{(x, y) \in R^n \times R^{md} : \psi_0(x, y) = 0\}$ *is nonempty and bounded.*

LEMMA 4.4.

(i) *For any* $t > 0$ *and* $\alpha > 0$, *the level set*

$$L_t(\alpha) = \{(x, y) \in R^n \times R^{md} : \psi_t(x, y) \le \alpha\}$$

is bounded.

(ii) *For any* $0 < t_1 \le t_2$ *and* $\alpha > 0$, *the level set*

$$L_{[t_1,t_2]}(\alpha) = \{(x, y) \in R^n \times R^{md} : \psi_t(x, y) \le \alpha, t \in [t_1, t_2]\}$$

is bounded.

Proof. (i) For any $(x, y) \in L_t(\alpha)$,

$$\psi_t(x, y) = (Ay - tx)^2 + \sum_{i=1}^{m} \left(y_i - p(t, y_i + b_i - A_i^T x)\right)^2 \le \alpha.$$

So

$$\sum_{i=1}^{m} \left(y_i - p(t, y_i + b_i - A_i^T x)\right)^2 \le \alpha, \tag{4.18}$$

and

$$(Ay - tx)^2 \le \alpha. \tag{4.19}$$

From (4.18) y is bounded. It follows from (4.19) that x is bounded. Hence $L_t(\alpha)$ is bounded. Similarly, we can prove that (ii) holds. □

It follows from Lemma 4.4(i) that we have the following.

COROLLARY 4.5. *For any $t > 0$, $\psi_t(x, y)$ is coercive, i.e.,*

$$\lim_{\|(x,y)\| \to +\infty} \psi_t(x, y) = +\infty.$$

THEOREM 4.6.

(i) *An infinite sequence $\{z^k\} \subseteq R \times R^n \times R^{md}$ is generated by Algorithm* 4.1, *and*

$$\lim_{k \to +\infty} H(z^k) = 0 \text{ and } \lim_{k \to +\infty} t^k = 0. \tag{4.20}$$

Hence each accumulation point, say, $z^* = (0, x^*, y^*)$, of $\{z^k\}$ is a solution of $H(z) = 0$, and x^* and y^* are optimal solutions to problems (1.1) *and* (2.3), *respectively.*

(ii) *The sequence $\{z^k\}$ is bounded. Hence there exists at least an accumulation point, say, $z^* = (0, x^*, y^*)$, of $\{z^k\}$ such that x^* and y^* are optimal solutions to problems* (1.1) *and* (2.3), *respectively.*

(iii) *If problem* (1.1) *has a unique solution x^*, then*

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

Proof. The proof of (i) and (ii) is similar to that of Theorem 4.5 in [26], so we omit it. It is follows from (ii) that (iii) holds. □

Let $z^* = (0, x^*, y^*)$ and define

$$A(z^*) = \{\lim H'(t^k, x^k, y^k) : t^k \downarrow 0^+, x^k \to x^* \text{ and } y^k \to y^*\}. \tag{4.21}$$

Clearly, $A(z^*) \subseteq \partial H(z^*)$.

LEMMA 4.7. *If all $V \in A(z^*)$ are nonsingular, then there is a neighborhood $N(z^*)$ of z^* and a constant C such that for any $z = (t, x, y) \in N(z^*)$ with $t \ne 0$, $H'(z)$ is nonsingular and*

$$\|(H'(z))^{-1}\| \le C.$$

Proof. From Lemma 4.1, for any $z = (t, x, y) \in N(z^*)$ with $t \ne 0$, $H'(z)$ is nonsingular. If the conclusion is not true, then there is a sequence $\{z^k = (t^k, x^k, y^k)\}$ with all $t^k \ne 0$ such that $z^k \to z^*$, and $\|(H'(z^k))^{-1}\| \to +\infty$. Since H is locally Lipschitzian, ∂H is bounded in a neighborhood of z^*. By passing to a subsequence, we may assume that $H'(z^k) \to V$. Then V must be singular, a contradiction to the assumption of this lemma. This completes the proof. □

THEOREM 4.8. *Suppose that $z^* = (0, x^*, y^*)$ is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm 4.1 and all $V \in A(z^*)$ are nonsingular. Then the whole sequence $\{z^k\}$ converges to z^* quadratically.*

Proof. First, from Theorem 4.6 z^* is a solution of $H(z) = 0$. Then, from Lemma 4.7, for all z^k sufficiently close to z^*,

$$\|H'(z^k)^{-1}\| = O(1).$$

Because H is strongly semismooth at z^*, from Lemma 1.1, for z^k sufficiently close to z^*,

$$\begin{array}{rcl} \|z^k + \Delta z^k - z^*\| & = & \|z^k + H'(z^k)^{-1}[-H(z^k) + \beta_k \bar{z}] - z^*\| \\ & = & O(\|H(z^k) - H(z^*) - H'(z^k)(z^k - z^*)\| + \beta_k \bar{t}\ ) \\ & = & O(\|z^k - z^*\|^2) + O(\psi(z^k)), \end{array} \tag{4.22}$$

and H is locally Lipschitz continuous near z^*, i.e., for all z^k close to z^*,

$$\psi(z^k) = \|H(z^k)\|^2 = O(\|z^k - z^*\|^2). \tag{4.23}$$

Therefore, from (4.22) and (4.23), for all z^k sufficiently close to z^*,

$$\|z^k + \Delta z^k - z^*\| = O(\|z^k - z^*\|^2). \tag{4.24}$$

By following the proof of Theorem 3.1 in [27], for all z^k sufficiently close to z^*, we have

$$\|z^k - z^*\| = O(\|H(z^k) - H(z^*)\|). \tag{4.25}$$

Hence, for all z^k sufficiently close to z^*, we have

$$\begin{array}{rcl} \psi(z^k + \Delta z^k) & = & \|H(z^k + \Delta z^k)\|^2 \\ & = & O(\|z^k + \Delta z^k - z^*\|^2) \\ & = & O(\|z^k - z^*\|^4) \\ & = & O(\|H(z^k) - H(z^*)\|^4) \\ & = & O(\psi(z^k)^2). \end{array} \tag{4.26}$$

Therefore, for all z^k sufficiently close to z^* we have

$$z^{k+1} = z^k + \Delta z^k. \tag{4.27}$$

From (4.27) and (4.24),

$$\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^2). \tag{4.28}$$

This completes the proof. □

Next, we study under what conditions all the matrices $V \in A(z^*)$ are nonsingular at a solution point $z^* = (0, x^*, y^*)$ of $H(z) = 0$.

PROPOSITION 4.9. *Suppose that $\|b_i - A_i^T x^*\| > 0$ for $i = 1, \ldots, m$. Then all $V \in A(z^*)$ are nonsingular.*

Proof. Because $\|b_i - A_i^T x^*\| > 0$ for $i = 1, \ldots, m$, $\|y_i^*\| = 1$ for $i = 1, \ldots, m$. From (2.8) we have

$$\|y_i^* + b_i - A_i^T x^*\| > 1 \text{ for } i = 1, \ldots, m.$$

Let $s_i = y_i + b_i - A_i^T x$ and $s_i^* = y_i^* + b_i - A_i^T x^*$ for $i = 1, \ldots, m$. It is easy to see that for any $V \in A(z^*)$, there exists a sequence $\{z^k = (t^k, x^k, y^k)\}$ such that

$$V = \begin{pmatrix} 1 & 0 & 0 \\ -x^* & 0 & A \\ E^* & P^* A^T & I_{md} - P^* \end{pmatrix},$$

where

$$E^* = [E_1^*, \ldots, E_m^*]^T,$$

$$(E_i^*)^T = \lim_{\substack{t^k \downarrow 0^+ \\ x^k \to x^* \\ y_i^k \to y_i^*}} \nabla \phi_t(t^k, s_i^k) \quad \text{for } i = 1, \ldots, m,$$

and

$$P^* = \text{Diag}\left(\frac{1}{\|s_i^*\|} I_d - \frac{1}{\|s_i^*\|^3} s_i^* (s_i^*)^T \right).$$

Let

$$M = \begin{pmatrix} 0 & A \\ P^* A^T & I_{md} - P^* \end{pmatrix}.$$

Hence, proving V is nonsingular is equivalent to proving M is nonsingular. Let

$$P_i^* = \frac{1}{\|s_i^*\|} I_d - \frac{1}{\|s_i^*\|^3} s_i^* (s_i^*)^T.$$

From Proposition 3.3, there exists a $d \times d$ matrix B_i^* such that

$$P_i^* = B_i \text{Diag}(\lambda_j^i) B_i^T,$$

where $0 < \lambda_j^i < 1$ for $j = 1, \ldots, d-1$ and $\lambda_d^i = 0$, and $B_i B_i^T = I_d$.

Let $B = \text{Diag}(B_i)$ and $D = \text{Diag}\left(\text{Diag}(\lambda_j^i)\right)$. Then

$$M = \begin{pmatrix} I_n & 0 \\ 0 & B \end{pmatrix} \begin{pmatrix} 0 & AB \\ D(AB)^T & I_{md} - D \end{pmatrix} \begin{pmatrix} I_n & 0 \\ 0 & B^T \end{pmatrix}.$$

Let

$$N = \begin{pmatrix} 0 & AB \\ D(AB)^T & I_{md} - D \end{pmatrix}.$$

Then, proving M is nonsingular is equivalent to proving N is nonsingular.

Let $\bar{B} = \text{Diag}(\bar{B}_i)$, where $\bar{B}_i$, $i = 1, \ldots, m$, is a $d \times (d-1)$ matrix obtained by deleting the dth column of B_i, and $q = [q_1^T, q_2^T]^T = [q_1^T, q_{11}, \ldots, q_{1d}, \ldots, q_{m1}, \ldots, q_{md}]^T \in R^n \times R^{md}$.

Let $Nq = 0$. Then we have

$$ABq_2 = 0, \tag{4.29}$$

and

$$D(AB)^T q_1 + (I_{md} - D)q_2 = 0. \tag{4.30}$$

Let

$$\bar{q}_2 = [q_{11}, \ldots, q_{1(d-1)}, \ldots, q_{m1}, \ldots, q_{m(d-1)}]^T \in R^{m(d-1)},$$

and

$$\bar{D} = \mathrm{Diag}\left(\mathrm{Diag}(\lambda_j^i, j = 1, \ldots, d-1)\right).$$

From (4.30) we have

$$q_{id} = 0 \quad \text{for } i = 1, \ldots, m, \tag{4.31}$$

and

$$D(A\bar{B})^T q_1 + (I_{m(d-1)} - \bar{D})\bar{q}_2 = 0. \tag{4.32}$$

Then, from (4.29) and (4.31),

$$A\bar{B}\bar{q}_2 = 0. \tag{4.33}$$

By following the proof of Lemma 4.1, we have $q_1 = 0$ and $\bar{q}_2 = 0$. Thus $q = 0$. This implies that N is nonsingular. So V is nonsingular. This completes the proof. □

PROPOSITION 4.10. *Let $M_0(z^*) = \{i : \|b_i - A_i^T x^*\| = 0, i = 1, \ldots, m\}$. If $\bar{A} = [A_i, i \in M_0(z^*)]$ is an $n \times n$ nonsingular matrix and $\|y_i^*\| < 1$ for $i \in M_0(z^*)$, then all $V \in A(z^*)$ are nonsingular.*

Proof. Without loss of generality, we suppose that $\|b_i - A_i^T x^*\| = 0$ for $i = 1, \ldots, j$ and $\|b_i - A_i^T x^*\| > 0$ for $i = j+1, \ldots, m$. Then $\|y_i^*\| < 1$ for $i = 1, \ldots, j$ and $\|y_i^*\| = 1$ for $i = j+1, \ldots, m$. From (2.8) we have

$$\|y_i^* + b_i - A_i^T x^*\| > 1, \text{ for } i = j+1, \ldots, m.$$

Let $s_i = y_i + b_i - A_i^T x$ and $s_i^* = y_i^* + b_i - A_i^T x^*$ for $i = 1, \ldots, m$. It is easy to see that for any $V \in A(z^*)$, there exists a sequence $\{z^k = (t^k, x^k, y^k)\}$ such that

$$V = \begin{pmatrix} 1 & 0 & 0 \\ -x^* & 0 & A \\ E^* & P^* A^T & I_{md} - P^* \end{pmatrix},$$

where

$$E^* = [E_1^*, \ldots, E_m^*]^T,$$

$$(E_i^*)^T = \lim_{\substack{t^k \downarrow 0+ \\ x^k \to x^* \\ y_i^k \to y_i^*}} \nabla \phi_t(t^k, s_i^k) \quad \text{for } i = 1, \ldots, m,$$

and

$$P^* = \mathrm{Diag}\left(P_i^*\right),$$

$$P_i^* = I_d \text{ for } i = 1, \ldots, j,$$

$$P_i^* = \frac{1}{\|s_i^*\|} I_d - \frac{1}{\|s_i^*\|^3} s_i^* (s_i^*)^T \text{ for } i = j+1, \ldots, m.$$

Let

$$M = \begin{pmatrix} 0 & A \\ P^* A^T & I_{md} - P^* \end{pmatrix}.$$

Hence, proving V is nonsingular is equivalent to proving M is nonsingular.

Let

$$\tilde{A} = [A_{j+1}, \ldots, A_m],$$

$$D = \text{Diag}\,(P_i^*, i = j+1, \ldots, m)\,,$$

and

$$q = [q_1^T, q_2^T, q_3^T]^T \in R^n \times R^n \times R^{md-n}.$$

Let $Mq = 0$. Then we have

$$\bar{A} q_2 + \tilde{A} q_3 = 0, \tag{4.34}$$

$$\bar{A}^T q_1 = 0, \tag{4.35}$$

and

$$D\tilde{A}^T q_1 + (I_{md-n} - D) q_3 = 0. \tag{4.36}$$

From (4.35) we have $q_1 = 0$. Then, from (4.36), $q_3 = 0$. It follows from (4.34) that $q_2 = 0$. Thus $q = 0$. This implies that M is nonsingular. So V is nonsingular. This completes the proof. □

By combining Theorem 4.8 and Propositions 4.9 and 4.10 we can directly obtain the following results.

THEOREM 4.11. *Suppose that $z^* = (0, x^*, y^*)$ is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm* 4.1. *If $\|b_i - A_i^T x^*\| > 0$ for $i = 1, \ldots, m$, then the whole sequence $\{z^k\}$ converges to z^*, and the convergence is quadratic.*

THEOREM 4.12. *Suppose that $z^* = (0, x^*, y^*)$ is an accumulation point of the infinite sequence $\{z^k\}$ generated by Algorithm* 4.1. *Let $M_0(z^*) = \{i : \|b_i - A_i^T x^*\| = 0, i = 1, \ldots, m\}$. If $\bar{A} = [A_i, i \in M_0(z^*)]$ is an $n \times n$ nonsingular matrix and $\|y_i^*\| < 1$ for $i \in M_0(z^*)$, then the whole sequence $\{z^k\}$ converges to z^* quadratically.*

5. Applications. In this section, we will apply the algorithm proposed in section 4 to solve the ESFL problem, the EMFL problem, and the SMT problem under a given topology.

The ESFL problem. Let $a_1, a_2, \ldots, a_m$ be m ($m \geq 2$) points in R^d, the d-dimensional Euclidean space. Let $\omega_1, \omega_2, \ldots, \omega_m$ be m positive weights. Find a point $x \in R^d$ that minimizes

$$f(x) = \sum_{i=1}^{m} \omega_i \|x - a_i\|. \tag{5.1}$$

This is called the ESFL problem. For more information on this problem, see [23].

The ESFL problem can be easily transformed into a special case of problem (1.1) where $b_i = \omega_i a_i$ and $A_i^T = \omega_i I_d$, $i = 1, 2, \ldots, m$. Therefore, it follows from Theorems 4.6, 4.11, and 4.12 that we have the following theorem.

THEOREM 5.1. *For the ESFL problem, assume that an infinite sequence* $\{z^k\} \subseteq R \times R^d \times R^{md}$ *is generated by Algorithm* 4.1. *Then the following hold:*

(i) *There exists at least an accumulation* $z^* = (0, x^*, y^*)$ *such that* x^* *is an optimal solution to the ESFL problem.*

(ii) *Suppose* $\omega_i \|x^* - a_i\| > 0$ *for* $i = 1, \ldots, M$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

(iii) *Suppose* $\omega_i \|x^* - a_i\| = 0$ *for some* i *and* $\omega_j \|x^* - a_j\| > 0$ *for all* $j \neq i$, *i.e., only the* i*th term is active, and* $\|y_i^*\| < 1$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

The EMFL problem. Let $a_1, a_2, \ldots, a_M$ be M points in R^d, the d-dimensional Euclidean space. Let ω_{ji}, $j = 1, 2, \ldots, N$, $i = 1, 2, \ldots, M$, and υ_{jl}, $1 \leq j \leq l \leq N$, be given nonnegative numbers. Find a point $x = (x_1, x_2, \ldots, x_N) \in R^{dN}$ that minimizes

$$f(x) = \sum_{j=1}^{N} \sum_{i=1}^{M} \omega_{ji} \|x_j - a_i\| + \sum_{1 \leq j \leq l \leq N} \upsilon_{jl} \|x_j - x_l\|. \tag{5.2}$$

This is the so-called EMFL problem. For ease of notation, we assume that $\upsilon_{jj} = 0$ for $j = 1, 2, \ldots, N$ and $\upsilon_{jl} = \upsilon_{lj}$ for $1 \leq j \leq l \leq N$.

To transform the EMFL problem (5.2) into an instance of problem (1.1), we simply do the following. Let $x = (x_1, x_2, \ldots, x_N)$. It is clear that $x \in R^n$ where $n = dN$. For each nonzero ω_{ji}, there is a corresponding term of the Euclidean norm $\|c(\omega_{ji}) - A(\omega_{ji})^T x\|$ where $c(\omega_{ji}) = \omega_{ji} a_i$, and $A(\omega_{ji})^T$ is a row of N blocks of $d \times d$ matrices whose jth block is $\omega_{ji} I_d$ and whose other blocks are zero. For each nonzero υ_{jl}, there is a corresponding term of the Euclidean norm $\|c(\upsilon_{jl}) - A(\upsilon_{jl})^T u\|$ where $c(\upsilon_{jl}) = 0$ and $A(\upsilon_{jl})^T$ is a row of N blocks of $d \times d$ matrices whose jth and lth blocks are $-\upsilon_{jl} I_d$ and $\upsilon_{jl} I_d$, respectively, and whose other blocks are zero. Define the index set $\Sigma = \{1, 2, \ldots, \tau\}$, where the set $\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_\tau\}$ is in one-to-one correspondence with the set of nonzero weights ω_{ji} and υ_{jl}, and then write problem (5.2) as follows.

Find a point $x = (x_1, x_2, \ldots, x_N) \in R^{dN}$ that minimizes

$$f(x) = \sum_{i=1}^{\tau} \|c_i - A_i^T x\|, \tag{5.3}$$

where for $i = 1, 2, \ldots, \tau$, $c_i \in R^d$, and $A_i \in R^{dN \times d}$. Therefore, it follows from Theorems 4.6, 4.11, and 4.12 that we have the following theorem.

THEOREM 5.2. *For the EMFL problem, assume that an infinite sequence* $\{z^k\} \subseteq R \times R^{dN} \times R^{\tau d}$ *is generated by Algorithm* 4.1. *Then the following hold:*

(i) *There exists at least an accumulation* $z^* = (0, x^*, y^*)$ *such that* x^* *is an optimal solution to the EMFL problem.*

(ii) *Suppose* $\|b_i - A_i^T x^*\| > 0$ *for* $i = 1, \ldots, \tau$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

(iii) *Let* $\Sigma_0(x^*) = \{i \in \Sigma : \|b_i - A_i^T x^*\| = 0\}$. *Assume that* $|\Sigma_0(x^*)| = N$, *the matrices* A_i, $i \in \Sigma_0(x^*)$ *are linearly independent and* $\|y_i^*\| < 1$ *for* $i \in \Sigma_0(x^*)$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

The SMT problem. The Euclidean SMT problem is given by a set of points $P = \{p_1, p_2, \ldots, p_N\}$ in the Euclidean plane and asks for the shortest planar straight-line graph spanning P. The solution takes the form of a tree, called the SMT, 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 the degree of each Steiner point is at most 3; see [17]. A full Steiner topology of point set P is a tree graph whose vertex set contains P and $N-2$ Steiner points and where the degree of each vertex in P is exactly 1 and the degree of each Steiner vertex is exactly 3.

Computing an SMT for a given set of N points in the Euclidean plane is NP-hard. However, the problem of computing the shortest network under a given full Steiner topology can be solved efficiently. We can transform this problem into the following problem; see [35] for more detail.

Find a point $x = (x_1, x_2, \ldots, x_{N-2}) \in R^{2N-4}$ that minimizes

$$f(x) = \sum_{i=1}^{m} \|c_i - A_i^T x\|, \tag{5.4}$$

where for $i = 1, 2, \ldots, m$, $c_i \in R^2$, and $A_i \in R^{2(N-2)\times 2}$. Therefore, it follows from Theorems 4.6, 4.11, and 4.12 that we have the following theorem.

THEOREM 5.3. *For the problem of computing the shortest network under a given full Steiner topology, assume that an infinite sequence* $\{z^k\} \subseteq R \times R^{2N-4} \times R^{4N-6}$ *is generated by Algorithm* 4.1. *Then the following hold:*

(i) *There exists at least an accumulation* $z^* = (0, x^*, y^*)$ *such that* x^* *is an optimal solution to the EMFL problem.*

(ii) *Suppose* $\|c_i - A_i^T x^*\| > 0$ *for* $i = 1, \ldots, m$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

(iii) *Let* $M_0(x^*) = \{i : \|b_i - A_i^T x^*\| = 0, i = 1, 2, \ldots, m\}$. *Assume that* $|M_0(x^*)| = N$, *the matrices* A_i, $i \in M_0(x^*)$, *are linearly independent and* $\|y_i^*\| < 1$ *for* $i \in M_0(x^*)$. *Then the whole sequence* $\{z^k\}$ *converges to* z^* *quadratically.*

6. Numerical experiments. Algorithm 4.1 was implemented in MATLAB and was run on a DEC Alpha Server 8200 for the following examples, where Examples 1(a)–5 and 8 are taken from [25] and Examples 6 and 7 from [35]. Throughout the computational experiments, unless otherwise stated, we used the following parameters:

$$\delta = 0.5,\ \sigma = 0.0005,\ \bar{t} = 0.002,\ y^0 = 0,\ \text{ and }\ \gamma = 0.5.$$

We terminated our iteration when one of the following conditions was satisfied:

(1) $k > 50$;

(2) $\text{relgap}(x^k, y^k) \le$ 1e-8, $\|Ay\| \le$ 1e-12, and $\max\limits_{1\le i\le m} \|y_i\| \le$ 1+1e-8;

(3) $ls > 20$,

where ls was the number of line search at each step and

$$\text{relgap}(x, y) = \frac{\left|\sum_{i=1}^m \|b_i - A_i^T x\| - b^T y\right|}{\sum_{i=1}^m \|b_i - A_i^T x\| + 1}.$$

The numerical results which we obtained are summarized in Table 1. In this table, n, d, and m specify the problem dimensions, Iter denotes the number of iterations, which is also equal to the number of Jacobian evaluations for the function

TABLE 1
Numerical results for Algorithm 4.1.

Example	n	d	m	Iter	NH	N0	$f(x^k)$	relgap	$\|Ay\|$	$\max_{1\le i\le m}\|y_i\|$
1(a)	2	2	3	7	12	1	2.828427	0	1.11e-16	1.00
1(b)	2	2	3	6	12	1	2.828427	4.60e-12	1.19e-13	1.00
1(c)	2	2	3	6	12	1	2.828427	4.60e-12	1.19e-13	1.00
1(d)	2	2	3	6	12	1	2.828427	4.58e-12	1.18e-13	1.00
2	2	2	3	7	14	0	2.732051	0	0	1.00
3	2	2	3	7	12	0	2.828427	1.16e-16	0	1.00
4	2	2	3	7	12	1	2.828427	1.74e-15	1.11e-16	1.00
5	10	2	55	12	27	2	226.2084	2.84e-14	6.26e-13	1.00
6	16	2	17	9	20	4	25.35607	5.80e-15	2.40e-15	1.00
7	4	2	5	4	5	1	400.0200	3.83e-15	6.75e-15	1.00
8	3	3	100	11	44	0	558.6450	8.13e-16	3.83e-14	1.00

TABLE 2
Output of Algorithm 4.1 *for Example* 5.

k	relgap	$\|Ay^k\|$	$\max_{1\le i\le m}\|y_i^k\|$	t^k	N0	δ^{j_k}
1	4.91e-01	1.47e-03	3.35e+00	1.50e-03	0	5.0e-01
2	4.72e-01	1.72e-03	3.28e+00	1.48e-03	0	3.1e-02
3	4.70e-01	1.75e-03	3.27e+00	1.48e-03	0	3.9e-03
4	1.04e-01	1.08e-02	2.84e+00	1.00e-03	0	1.0e+00
5	1.08e-03	1.07e-02	3.80e+00	1.00e-03	0	1.0e+00
6	4.27e-03	9.21e-03	1.56e+00	1.00e-03	0	1.0e+00
7	4.00e-04	3.74e-03	1.10e+00	4.07e-04	0	1.0e+00
8	7.82e-05	3.20e-04	1.03e+00	3.44e-05	0	1.0e+00
9	4.40e-06	7.91e-06	1.02e+00	9.00e-07	0	1.0e+00
10	1.66e-07	3.79e-06	1.00e+00	4.13e-07	2	1.0e+00
11	1.08e-09	1.30e-10	1.00e+00	1.44e-11	2	1.0e+00
12	2.84e-14	6.26e-13	1.00e+00	6.82e-14	2	1.0e+00

H, NH denotes the number of function evaluations for the function H, N0 indicates the number of norms that are zero at the optimal solution, more precisely, which is interpreted as being zero if it is less than the tolerance 10^{-10}, $f(x^k)$ denotes the value of $f(x)$ at the final iteration, and relgap denotes the relative duality gap. The results reported in Table 1 show that this method is extremely promising. The algorithm was able to solve all examples in less than 15 iterations. Tables 2 and 3 give more detailed results for Examples 5 and 6, which show the quadratic convergence of this method. For Examples 6 and 7, the number of iterations required by our algorithm is fewer than that required by the algorithm proposed in [35].

The first few examples are of the following form:

$$\begin{aligned} &n = 2, \quad d = 2, \quad m = 3, \\ &A_1 = I, \quad A_2 = \omega I, \quad A_3 = I, \\ &b_1 = [-1, 0]^T, \quad b_2 = [0, \omega]^T, \quad b_3 = [1, 0]^T. \end{aligned} \tag{6.1}$$

Example 1(a). This is given by (6.1) with $\omega = 2$ and solution $x^* = [0.0, 1.0]^T$. The starting point $x^0 = [3.0, 2.0]^T$.

Example 1(b). Same as Example 1(a), except $x^0 = [1.0, 1.0 \times 10^{-6}]^T$.

Example 1(c). Same as Example 1(a), except $x^0 = [1.000001, -1.0 \times 10^{-6}]^T$.

Example 1(d). Same as Example 1(a), except $x^0 = [1.001, -1.0 \times 10^{-3}]^T$.

Example 2. This is given by (6.1) with $\omega = 1$ and solution $x^* = [0.0, 0.577350]^T$. The starting point $x^0 = [3.0, 2.0]^T$.

TABLE 3
Output of Algorithm 4.1 *for Example* 6.

k	relgap	$\|Ay^k\|$	$\max_{1\le i\le m} \|y_i^k\|$	t^k	N0	δ^{jk}
1	7.52e-01	1.97e-03	1.65e+00	1.75e-03	0	2.5e-01
2	5.69e-01	1.45e-02	1.62e+00	1.66e-03	0	1.2e-01
3	2.25e-01	2.60e-02	1.44e+00	1.49e-03	0	2.5e-01
4	1.77e-01	2.37e-02	1.42e+00	1.43e-03	0	1.2e-01
5	4.39e-02	1.70e-02	1.17e+00	1.00e-03	0	1.0e+00
6	5.22e-03	4.05e-03	1.03e+00	2.26e-04	0	1.0e+00
7	1.01e-04	6.83e-05	1.00e+00	3.56e-06	0	1.0e+00
8	4.10e-08	1.62e-08	1.00e+00	8.36e-10	2	1.0e+00
9	5.80e-15	2.40e-15	1.00e+00	1.34e-16	4	1.0e+00

TABLE 4
Weights: New to new and new to existing.

	New					Existing								
New	1	2	3	4	5	1	2	3	4	5	6	7	8	9
1		1	1	1	1	2	2	1	1	1	1	1	1	1
2			1	10^{-2}	10^{-1}	1	1	2	2	1	1	1	1	1
3				10^{-2}	10^{-1}	1	1	1	1	2	2	1	1	1
4					10^{-1}	1	1	1	1	1	1	2	2	1
5						1	1	1	1	1	1	1	1	2

TABLE 5
Existing facility locations.

	1	2	3	4	5	6	7	8	9
Component 1	0	2	6	6	8	7	0	0	0
Component 2	0	4	2	10	8	7	1	2	3

Example 3. This is given by (6.1) with $\omega = 1.414$ and solution $x^* = [0.0, 0.999698]^T$. The starting point $x^0 = [3.0, 2.0]^T$.

Example 4. This is given by (6.1) with $\omega = 1.415$ and solution $x^* = [0.0, 1.0]^T$. The starting point $x^0 = [3.0, 2.0]^T$.

Example 5. This is a multifacility location problem. The objective is to choose five new facilities in the plane (i.e., vectors in R^2) to minimize a weighted sum of distances between each pair of new facilities plus a weighted sum of distances between each of the new facilities and each of the existing facilities (i.e., given vectors in R^2). Tables 4 and 5 complete the description of the problem. The solution is

$$x^* = [2.03865, 3.65117; 2.24659, 3.75886; 2.24659, 3.75886; 1.45825, 2.96083; 2.03865, 3.65117]^T.$$

The starting point $x^0 = [1, 1; 1, 1; 1, 1; 1, 1; 1, 1]^T$.

Example 6. This is an SMT problem. This example contains 10 regular points. The coordinates of the 10 regular points are given in Table 6. The tree topology is given in Table 6 where for each edge, indices of its two vertices are shown next to the index of the edge. This topology is the best topology obtained by a branch-and-bound algorithm. Therefore, the shortest network under this topology is actually the SMT problem for the given 10 regular points. The starting point $x^0 = [1, 1; 1, 1; 1, 1; 1, 1; 1, 1; 1, 1; 1, 1; 1, 1; 1, 1]^T$.

Example 7. This is an SMT problem. This example contains four regular points. The coordinates of the four regular points and the tree topology are given in Table 7.

TABLE 6
The topology and the coordinates of the ten regular point in Example 6.

Point-index	x-coord	y-coord	Point-index	x-coord	y-coord
9	2.309469	9.208211	14	7.598152	0.615836
10	0.577367	6.480938	15	8.568129	3.079179
11	0.808314	3.519062	16	4.757506	3.753666
12	1.685912	1.231672	17	3.926097	7.008798
13	4.110855	0.821114	18	7.436490	7.683284
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			

TABLE 7
The topology and the coordinates of the four regular point in Example 7.

Point-index	x-coord	y-coord	Point-index	x-coord	y-coord
3	−100.0	1.0	5	−100.0	−1.0
4	100.0	1.0	6	100.0	−1.0
Edge-index	ea-index	eb-index	Edge-index	ea-index	eb-index
1	3	1	4	6	2
2	4	1	5	1	2
3	5	2			

The starting point $x^0 = [1, 1; 1, 1]^T$.

Example 8.

$$n = 3, \quad d = 3, \quad m = 100.$$

$$A_i = I, \ i = 1, 2, \ldots, m, \ \text{except } A_i = 100I \text{ if } i \bmod 10 = 1.$$

The elements of b_i, $i = 1, 2, \ldots, m$, are generated randomly. We use the following pseudorandom sequence:

$$\psi_0 = 7, \psi_{i+1} = (445\psi_i + 1) \bmod \quad 4096, i = 1, 2, \ldots,$$

$$\bar{\psi_i} = \frac{\psi_i}{4096}, \qquad i = 1, 2, \ldots.$$

The elements of b_i, $i = 1, 2, \ldots, m$, are successively set to be $\bar{\psi}_1, \bar{\psi}_2, \ldots$ in the order $(b_1)_1, \ldots, (b_1)_d, (b_2)_1, \ldots, (b_m)_d$, except that the appropriate random number is multiplied by 100 to given $(b_i)_j$ if $i \bmod 10 = 1$.

The solution $x^* = [0.586845, 0.480333, 0.509340]^T$. The initial point x^0 is set to b_m.

7. Conclusions. In this paper we presented a smoothing Newton method for the problem of minimizing a sum of Euclidean norms by applying the smoothing Newton method proposed by Qi, Sun, and Zhou [28] directly to a system of strongly semismooth equations derived from primal and dual feasibility and a complementarity

condition, and proved that this method was globally and quadratically convergent. It is deserved to point out that in our method we can control the smoothing parameter t in such a way that it converges to zero neither too quickly nor too slowly by using a particularly designed Newton equation and a line search model; see (4.11) and (4.12). Numerical results indicated that our algorithm was extremely promising. It will be an interesting work to compare this method with some existing methods, e.g., the primal-dual interior-point method proposed in [3]. However, we have been unable to do this because no code is available.

Consider the problem of minimizing a sum of Euclidean norms subject to linear equality constraints:

$$\min \left\{ \sum_{i=1}^{m} \|b_i - A_i^T x\|, \quad E^T x = b_e, \ x \in R^n \right\}, \tag{7.1}$$

where $E \in R^{n\times d}$ is an $n \times d$ matrix with full column rank and $b_e \in R^d$. In [2], Andersen and Christiansen transformed the problem (7.1) to the problem (1.1) based on the l_1 penalty function approach. So we can also apply the algorithm proposed in section 4 to solve problem (7.1).

Acknowledgments. The authors would like to thank Steve Wright, the two referees, Defeng Sun, Houduo Qi, and Song Xu for their helpful comments. Thanks also go to K. D. Andersen and E. Christiansen for providing the authors with some references.

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, *Minimizing a sum of norms subject to linear equality constraints*, Comput. Optim. Appl., 11 (1998), pp. 65–79.

[3] K. D. Andersen, 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., 22 (2000), pp. 243–262.

[4] P. H. Calamai and A. R. Conn, *A stable algorithm for solving the multifacility location problem involving Euclidean distances*, SIAM J. Sci. Statist. 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] R. Chandrasekaran and A. Tamir, *Open questions concerning Weiszfeld's algorithm for the Fermat-Weber location problem*, Math. Programming, 44 (1989), pp. 293–295.

[7] R. Chandrasekaran and A. Tamir, *Algebraic optimization: The Fermat-Weber location problem*, Math. Programming, 46 (1990), pp. 219–224.

[8] B. Chen and X. Chen, *A global and local superlinear continuation-smoothing method for $P_0 + R_0$ NCP or monotone NCP*, SIAM J. Optim., 9 (1999), pp. 624–645.

[9] 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.

[10] 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.

[11] X. Chen and Y. Ye, *On homotopy-smoothing methods for box-constrained variational inequalities*, SIAM J. Control Optim., 37 (1999), pp. 589–616.

[12] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.

[13] J. W. Eyster, J. A. White, and W. W. Wierwille, *On solving multifacility location problems using a hyperboloid approximation procedure*, AIIE Trans., 5 (1973), pp. 1–6.

[14] 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.

[15] F. Facchinei and C. Kanzow, *A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems*, Math. Programming, 76 (1997), pp. 493–512.
[16] A. Fischer, *Solution of monotone complementarity problems with locally Lipschitzian functions*, Math. Programming, 76 (1997), pp. 513–532.
[17] E. N. Gilbert and H. O. Pollak, *Steiner minimal trees*, SIAM J. Appl. Math., 16 (1968), pp. 1–29.
[18] 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.
[19] H. Jiang, *Smoothed Fischer-Burmeister Equation Methods for the Complementarity Problem*, Department of Mathematics, The University of Melbourne, Victoria, Australia, Preprint, 1997.
[20] K. Hotta and A. Yoshise, *Global convergence of a class of non-interior-point algorithms using Chen-Harker-Kanzow functions for nonlinear complementarity problems*, Math. Programming, 86 (1999), pp. 105–133.
[21] H. Jiang and L. Qi, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Control Optim., 35 (1997), pp. 178–193.
[22] H. W. Kuhn, *A note on Fermat's problem*, Math. Programming, 4 (1973), pp. 98–107.
[23] R. F. Love, J. G. Morris, and G. O. Wesolowsky, *Facility Location: Models and Methods*, North–Holland, Amsterdam, 1988.
[24] L. M. M. Ostresh, *The multifacility location problem: Applications and decent theorems*, Journal of Regional Science, 17 (1977), pp. 409–419.
[25] M. L. Overton, *A quadratically convergent method for minimizing a sum of Euclidean norms*, Math. Programming, 27 (1983), pp. 34–63.
[26] H.-D. Qi, *A regularized smoothing Newton method for box constrained variational inequality problems with P_0-functions*, SIAM J. Optim., 10 (2000), pp. 315–330.
[27] L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res., 18 (1993), pp. 227–244.
[28] 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, 87 (2000), pp. 1–35.
[29] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.
[30] 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.
[31] 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.
[32] D. Sun, *A regularization Newton method for solving nonlinear complementarity problems*, Appl. Math. Optim., 40 (1999), pp. 315–339.
[33] C. Y. Wang et al, *On the convergence and rate of convergence of an iterative algorithm for the plant location problem*, Qufu Shifan Xuebao, 2 (1975), pp. 14–25 (in Chinese).
[34] E. Weiszfeld, *Sur le point par lequel la somme des distances de n points donnes est minimum*, Tohoku Math. J., 43 (1937), pp. 355–386.
[35] 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.
[36] G. Xue and Y. Ye, *An efficient algorithm for minimizing a sum of p-norms*, SIAM J. Optim., 10 (2000), pp. 551–579.
[37] N. Yamashita and M. Fukushima, *Modified Newton methods for solving semismooth reformulations of monotone complementarity problems*, Math. Programming, 76 (1997), pp. 273–284.
[38] 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, Appl. Optim. 22, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999, pp. 421–441.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 411–425

APPROXIMATE SOLUTIONS OF ANALYTIC INEQUALITY SYSTEMS*

JEAN-PIERRE DEDIEU†

Abstract. We consider the stability problem for a system of equalities and inequalities. We prove that when such a system is satisfied approximately at a certain point, it is also satisfied exactly on a nearby point.

Key words. system of equalities and inequalities, stability, analytic, alpha-theory

AMS subject classifications. 47J99, 90C31

PII. S105262349935606X

1. Introduction.

In this paper we consider a system of inequalities

$$f_i(x) \geq 0,\ 1 \leq i \leq m,$$

where $f_i : \mathbb{R}^n \to \mathbb{R}$ is a polynomial or, more generally, an analytic function. As in Cucker and Dedieu [8], we are motivated by approximate computations and finite precision arithmetic. To know whether, on input $x \in \mathbb{R}^n$, this system of inequalities is satisfied, we first compute the quantities $f_i(x),\ 1 \leq i \leq m$, and then we check their signs. If we use finite precision arithmetic, as is often used in scientific computing, due to round-off errors, instead of $f_i(x)$ we have computed a nearby quantity $f_i(x) - \epsilon_i$. Here ϵ_i is a "small" real number which depends on the program computing $f_i(x)$ and also on the characteristics of the arithmetic. Thus, instead of exact inequalities we check the approximate system

$$f_i(x) \geq \epsilon_i,\ 1 \leq i \leq m.$$

What can we deduce from this set of inequalities?

To a vector $a \in \mathbb{R}^m$ we associate its positive part a^+ and its negative part a^-. The ith coordinate of a^+ (resp., a^-) is a_i (resp., 0) when a_i is nonnegative and 0 (resp., $-a_i$) otherwise so that, as for real numbers, $a = a^+ - a^-$. In our context we use $\|f(x)^-\|$ to measure the deviation of the vector $f(x)$ from positivity.

To begin we show that when $\|f(x)^-\|$ is small enough, there exists a certain $y \in \mathbb{R}^n$ close to x such that the system of inequalities is satisfied exactly at y:

$$f_i(y) \geq 0,\ 1 \leq i \leq m.$$

We also give an estimate for the distance of y from x in terms of $\|f(x)^-\|$ (see Theorem 1). To be as general as possible, we also study the following case of strict inequalities:

$$f_i(x) \geq 0,\ 1 \leq i \leq p, \text{ and } f_i(x) > 0,\ p+1 \leq i \leq m,$$

and we obtain in this context a similar result: when $\|f(x)^-\|$ is small enough, then the corresponding exact system is satisfied at a nearby y (see Theorem 2).

*Received by the editors May 10, 1999; accepted for publication (in revised form) June 7, 2000; published electronically October 18, 2000.

http://www.siam.org/journals/siopt/11-2/35606.html

†MIP, Département de Mathématique, Université Paul Sabátier, 31062 Toulouse Cedex 4, France (dedieu@mip.ups-tlse.fr).

Systems of inequalities are frequently found in mathematical programming. The sets of constraints of mathematical programs are often described by such systems. The Lagrange multiplier rule is another example and, more generally, so is the complementarity problem.

Since these systems involve equalities and inequalities, we also study the following case of such mixed systems:

$$f_i(x) \geq 0,\ 1 \leq i \leq p,\ f_i(x) > 0,\ p+1 \leq i \leq m,\ \text{and } g_j(x) = 0,\ 1 \leq j \leq q.$$

We consider perturbations of both equalities and inequalities and we prove the existence of a nearby y satisfying exactly this system (see Theorem 3).

Next we study the exact feasibility of a system of inequalities. To the system $f_i(x) \geq 0,\ 1 \leq i \leq m$ we associate the partition $\mathbb{R}^n = f^{\geq 0} \cup \left(\mathbb{R}^n \setminus f^{\geq 0}\right)$, where $f^{\geq 0}$ is the set of vectors $x \in \mathbb{R}^n$ satisfying the system $f_i(x) \geq 0$. Its boundary is denoted by

$$\Sigma = \partial f^{\geq 0}.$$

The inverse of the distance of $x \in \mathbb{R}^n$ from Σ is denoted by

$$\mu(x) = \mathrm{Dist}(x, \Sigma)^{-1}.$$

In Corollary 2 we deduce the exact feasibility of the system at x from its approximate feasibility: if $\|f(x)^-\|$ is small enough and if $\mu(x)$ is small enough, then

$$f_i(x) \geq 0,\ 1 \leq i \leq m.$$

But let us be more precise.

2. Main results. We denote by σ the sum of the following series:

$$\sigma = \sum_{k=0}^{\infty} \left(\frac{1}{2}\right)^{2^k - 1} = 1.63284\ldots.$$

To an analytic map $f : \mathbb{R}^n \to \mathbb{R}^m$ and to $x \in \mathbb{R}^n$ we associate the two following numbers:

$$\Gamma(f, x) = \sup_{k \geq 2} \left\| \frac{D^k f(x)}{k!} \right\|^{\frac{1}{k-1}}$$

and

$$\delta(f, x) = \left\| \left(Df(x) Df(x)^* + 4\mathrm{Diag}(f(x)^+) \right)^{-1} \right\|^{\frac{1}{2}}.$$

We also let $\delta(f, x) = \infty$ when the matrix $Df(x)Df(x)^* + 4\mathrm{Diag}(f(x)^+)$ is singular. Here A^* denotes the adjoint of the matrix A, $\|\ \|$ is the operator norm associated with the usual Euclidean norms in R^n and R^m, and $\mathrm{Diag}(d)$ is the diagonal matrix with diagonal entries $d_1, d_2, \ldots$. Comments about $\delta(f, x)$ are given in Remark 1 below.

$\Gamma(f, x)$ is always finite because f is analytic. If we denote by $R(f, x)$ the radius of convergence of the Taylor series of f at x, then $R(f, x) \geq 1/\Gamma(f, x)$: inside the ball about x with radius $1/\Gamma(f, x)$ the Taylor series converges.

Our first result is the following.

THEOREM 1. *Let $x \in \mathbb{R}^n$ be such that*

$$\|f(x)^-\| \leq \frac{1}{8(1+\Gamma(f,x))\delta(f,x)\max(1,\delta(f,x))}.$$

Then the set $f^{\geq 0}$ is nonempty and

$$\mathrm{Dist}(x, f^{\geq 0}) \leq \sigma\delta(f,x)\|f(x)^-\|.$$

When f is a polynomial system with degree $f_i \leq 2$ this result holds when

$$\|f(x)^-\| \leq \frac{1}{4(2+\|D^2 f(x)\|)\delta(f,x)^2}.$$

Remark 1. (1) When $\delta(f,x) = \infty$ this theorem is "empty": $f(x)^-$ has to be equal to 0, i.e., $f(x) \geq 0$.

(2) The condition $\delta(f,x) = \infty$ is not a "geometric" condition. By geometric we mean a condition which depends only on the feasible set $f^{\geq 0}$. For example, the two systems

$$f_1(x_1,x_2) = x_1^2 - x_2^2 \text{ and } f_2(x_1,x_2) = x_1$$

and

$$g_1(x_1,x_2) = x_1 + x_2 \text{ and } g_2(x_1,x_2) = x_1 - x_2$$

define the same feasible set, but $\delta(f,(0,0)) = \infty$ and $\delta(g,(0,0)) < \infty$.

(3) The condition $\delta(f,x) = \infty$ may be satisfied in the interior of the feasible set $f^{\geq 0}$. It is the case at $x = (0,0)$ for the system

$$f_1(x_1,x_2) = x_1^2 + x_2^2 \text{ and } f_2(x_1,x_2) = 1 - x_1.$$

(4) The proof of Theorem 1 (given in section 6) involves the introduction of squared slack variables. We will consider $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ defined by

$$F(x,t) = (f_i(x) - t_i^2)_{1 \leq i \leq m}.$$

It will be shown that $DF(x,t)$ has rank m if and only if

$$DF(x,t)DF(x,t)^* = Df(x)Df(x)^* + 4\mathrm{Diag}(t_i^2)$$

is nonsingular. The hypothesis $\delta(f,x) < \infty$ will ensure that $DF(x, \sqrt{f(x)^+})$ has rank m: this fact is crucial in our proof.

In the following we consider the case of a single inequality.

COROLLARY 1. *Let $f : \mathbb{R}^n \to \mathbb{R}$ and $x \in \mathbb{R}^n$ be given. If*

$$f(x)^- \leq \frac{\|Df(x)\|\min(1,\|Df(x)\|)}{8(1+\Gamma(f,x))},$$

then $f^{\geq 0}$ is nonempty and

$$\mathrm{Dist}(x, f^{\geq 0}) \leq \sigma\|Df(x)\|^{-1} f(x)^-.$$

When f is a polynomial with degree $f = 2$, then this inequality holds as soon as

$$f(x)^- \leq \frac{\|Df(x)\|^2}{4(2 + \|D^2 f(x)\|)}.$$

In the following we prove that $f(x) \geq 0$ as soon as $\|f(x)^-\|$ and $\mu(x)$ are small.

COROLLARY 2. *Let $x \in \mathbb{R}^n$ be such that*

$$\|f(x)^-\| \leq \frac{1}{8(1 + \Gamma(f,x))\delta(f,x)\max(1, \delta(f,x))}$$

and

$$\mu(x) < \frac{1}{\sigma\delta(f,x)\|f(x)^-\|}.$$

Then $f(x) \geq 0$.

In Theorem 1 we do not consider the case of strict inequalities. This is achieved in the following.

THEOREM 2. *Let $x \in \mathbb{R}^n$ be such that*

$$\|f(x)^-\| \leq \frac{1}{8(1 + \Gamma(f,x))\delta(f,x)\max(1, \delta(f,x))}.$$

Let us also suppose that

$$f_i(x) > (\sigma\delta(f,x)\|f(x)^-\|)^2$$

for each i in a certain set $J \subset \{1, \ldots, m\}$. Then there exists $y \in f^{\geq 0}$ such that

$$f_i(y) > 0 \text{ for each } i \in J$$

and

$$\|x - y\| \leq \sigma\delta(f,x)\|f(x)^-\|.$$

Another formulation of this theorem is given below.

COROLLARY 3. *Let $x \in \mathbb{R}^n$ and $\epsilon \in \mathbb{R}^m$ be given. Let us denote $\epsilon^{<0}$ the vector in $\mathbb{R}^m$ with the ith component equal to ϵ_i when $f_i(x) < 0$ and 0 otherwise. When the three following conditions are satisfied:*

- *$f_i(x) + \epsilon_i \geq 0$ for each $i = 1, \ldots, p$,*
- *$f_i(x) > (\sigma\delta(f,x)\|\epsilon^{<0}\|)^2$ for each $i = p+1, \ldots, m$,*
- *$\|\epsilon^{<0}\| \leq 1/(8(1 + \Gamma(f,x))\delta(f,x)\max(1, \delta(f,x)))$,*

then, there exists $y \in \mathbb{R}^n$ such that

- *$f_i(y) \geq 0$ for each $i = 1, \ldots, p$,*
- *$f_i(y) > 0$ for each $i = p+1, \ldots, m$,*
- *$\|x - y\| \leq \sigma\delta(f,x)\|\epsilon^{<0}\|$.*

Another interesting case to consider is given by a system of inequalities $f(x) \geq 0$ defined on a submanifold $\mathbb{E} \subset \mathbb{R}^n$ or on the set

$$\mathbb{E} = \{x \in \mathbb{R}^n \ : \ g_j(x) = 0, \ 1 \leq j \leq q\}$$

with $g : \mathbb{R}^n \to \mathbb{R}^q$ an analytic function. We consider here perturbations of both the system of inequalities and the equalities defining $\mathbb{E}$. We measure the deviation to nonnegativity by the quantity $\|f(x)^-\|$ and the distance to $\mathbb{E}$ by the number $\|g(x)\|$. As in Theorem 2, we are seeking to know that there exists $y \in R^n$ close to x such that $f(y) \geq 0$, $f_i(y) > 0$ for any $i \in J$ and $g(y) = 0$. This is achieved in Theorem 3. Before stating this theorem we need more notation.

Let $x \in R^n$ be given. We define $\Gamma(f, g, x)$ as the value of Γ corresponding to the map $(f, g) : \mathbb{R}^n \to \mathbb{R}^m \times \mathbb{R}^q$. We also consider the following $(m+q) \times (m+q)$ matrix:

$$G(x) = \begin{pmatrix} Df(x)Df(x)^* + \mathrm{Diag}(4f(x)^+) & Df(x)Dg(x)^* \\ Dg(x)Df(x)^* & Dg(x)Dg(x)^* \end{pmatrix}$$

and we define $\delta(f, g, x) = \|G(x)^{-1}\|^{\frac{1}{2}}$ or ∞ when $G(x)$ is singular.

THEOREM 3. *Let $x \in \mathbb{R}^n$ such that*

$$(\|f(x)^-\|^2 + \|g(x)\|^2)^{1/2} \leq \frac{1}{8(1 + \Gamma(f, g, x))\delta(f, g, x)\max(1, \delta(f, g, x))}.$$

Let us also suppose that

$$f_i(x) > (\sigma\delta(f, g, x)\|f(x)^-\|)^2$$

for each i in a certain set $J \subset \{1, \ldots, m\}$. Then there exists $y \in \mathbb{R}^n$ such that $f_i(y) \geq 0$ for any $i = 1, \ldots, m$, $f_i(y) > 0$ for any $i \in J$, $g_j(y) = 0$ for any $j = 1, \ldots, q$, and

$$\|y - x\| \leq \sigma\delta(f, g, x)(\|f(x)^-\|^2 + \|g(x)\|^2)^{1/2}.$$

In the following corollary we suppose the equality constraints are satisfied at x. Let us define $\Sigma_{\mathbb{E}}$ the boundary of $f^{\geq 0} \cap \mathbb{E}$ in $\mathbb{E}$ and, for any $x \in \mathbb{E}$, $\mu_{\mathbb{E}}(x)$ the inverse of the distance of x from $\Sigma_{\mathbb{E}}$. In this context we have the following.

COROLLARY 4. *Let $x \in \mathbb{E}$ be such that*

$$\|f(x)^-\| \leq \frac{1}{8(1 + \Gamma(f, g, x))\delta(f, g, x)\max(1, \delta(f, g, x))} \quad \textit{and} \quad \mu_{\mathbb{E}}(x) < \frac{1}{\sigma\delta(f, g, x)\|f(x)^-\|}.$$

Then we have $f(x) \geq 0$.

3. Comparison with other results. A first result involving perturbations of a system of inequalities is due to Hoffman who considers linear inequalities. Hoffman's theorem was published for the first time in 1952 [14] and reconsidered by Güler, Hoffman, and Rothblum in 1995 [12].

THEOREM 4 (Hoffman). *Let $A \in \mathbb{R}^{m \times n}$. Then there exists a scalar $K(A)$, such that for each $b \in \mathbb{R}^m$ for which the set $A^{\leq b} = \{x' \in \mathbb{R}^n \ : \ Ax' \leq b\}$ is not empty and for each $x \in \mathbb{R}^n$*

$$\mathrm{Dist}(x, A^{\leq b}) \leq K(A)\|(Ax - b)^+\|.$$

There have been a number of generalizations of Hoffman's theorem to nonlinear cases. A first class of results uses a convexity assumption and is proved via convex analysis. Robinson [22] extended Hoffman's bound to a system of convex inequalities defining a convex and bounded set with nonempty interior, Mangasarian [20] considered a closed convex set defined by a system of finitely many differentiable convex inequalities, and Auslender and Crouzeix [1] extended Mangasarian's result to convex nondifferentiable functions. A recent paper in these directions is Lewis and Pang [16].

Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ be an extended-valued closed proper convex function and $\mathcal{S}$ the closed convex set defined by $f(x) \leq 0$. We denote by $f'(x,d)$ the directional derivative of f at x along a direction d and by $\mathcal{N}(x,\mathcal{S})$ the normal cone of $\mathcal{S}$ at a vector $x \in \mathcal{S}$. With these notations Lewis and Pang prove the following.

THEOREM 5 (Lewis–Pang). *The following statements are equivalent:*

- $\mathrm{Dist}(x,\mathcal{S}) \leq \gamma f(x)^+$ *for any* $x \in \mathbb{R}^n$.
- *For any* $x \in f^{-1}(0)$ *and* $d \in \mathcal{N}(x,\mathcal{S})$

$$f'(x,d) \geq \gamma^{-1}\|d\|.$$

Another generalization of Hoffman's theorem to a nonlinear and nonconvex case may be obtained via Lojasiewicz's inequality. This result was proved for the first time by Lojasiewicz [17] for semianalytic or semialgebraic sets and functions and then extended to the subanalytic case by Hironaka [13]. A good exposé of such questions is contained in Bierstone and Milman [6].

DEFINITION 1. *A subset* X *of* $\mathbb{R}^n$ *is semialgebraic if there are real polynomials* $P_{i,j}$ *such that*

$$X = \bigcup_{i=1}^{r} \bigcap_{j=1}^{s_i} \{x \in \mathbb{R}^n \mid P_{i,j}(x)\ \epsilon_{i,j}\ 0\},$$

where $\epsilon_{i,j} \in \{<,>,=\}$. *A function* $f : X \to \mathbb{R}^m$ *is semialgebraic when its graph is itself semialgebraic.*

The class of semialgebraic sets is stable for elementary set operations (union, intersection, set difference) and also under projection (Tarsky–Seidenberg theorem).

DEFINITION 2. *A subset* X *of* $\mathbb{R}^n$ *is semianalytic if for each* $a \in \mathbb{R}^n$ *there are a neighborhood* U *of* a *and real analytic functions* $f_{i,j}$ *on* U *such that*

$$X \cap U = \bigcup_{i=1}^{r} \bigcap_{j=1}^{s_i} \{x \in U \mid f_{i,j}(x)\ \epsilon_{i,j}\ 0\},$$

where $\epsilon_{i,j} \in \{<,>,=\}$. *A function* $f : X \to \mathbb{R}^m$ *is semianalytic when its graph is itself semianalytic.*

The class of semianalytic sets is not stable under projections. For this reason Hironaka introduced the concept of subanalytic sets.

DEFINITION 3. *A subset* X *of* $\mathbb{R}^n$ *is subanalytic if each point of* $a \in \mathbb{R}^n$ *admits a neighborhood* U *such that* $X \cap U$ *is a projection of a relatively compact semianalytic set: there is a semianalytic bounded set* A *in* $\mathbb{R}^{n+p}$ *such that* $X \cap U = \Pi(A)$, *where* $\Pi : \mathbb{R}^{n+p} \to \mathbb{R}^n$ *is the projection. A function is subanalytic when its graph is itself subanalytic.*

THEOREM 6 (Lojasiewicz). *Let* K *be a compact and subanalytic set contained in* $\mathbb{R}^n$. *Let* $u, v : K \to \mathbb{R}$ *be continuous subanalytic functions. If* $u^{-1}(0) \subset v^{-1}(0)$, *there exist* $\alpha > 0$ *and an integer* $N > 0$ *such that for all* $x \in K$,

$$\alpha\ |v(x)|^N \leq |u(x)|.$$

Taking $v(x) = \mathrm{Dist}(x, f^{\geq 0})$ and $u(x) = \|f(x)^-\|$ in Lojasiewicz's inequality gives the following corollary.

COROLLARY 5. *Let K be a compact and subanalytic set contained in $\mathbb{R}^n$. Let $f : K \to \mathbb{R}^m$ be a continuous subanalytic function. There exist $\alpha > 0$ and an integer $N > 0$ such that for all $x \in K$,*

$$\alpha \ \mathrm{Dist}(x, f^{\geq 0})^N \leq \|f(x)^-\|.$$

This is a very general and powerful method. It has already been used in the context of optimization theory by Warga [25] and Dedieu [10] (penalty functions), Luo and Luo [18] (polynomial inequalities), and Luo and Pang [19] (analytic inequalities).

Hoffman's theorem has also been extended by Ioffe [15] to locally Lipschitz functions using Clarke's subgradient, and more recently by Azé, Corvellec, and Lucchetti [2] who consider the case of lower semicontinuous functions defined over Banach spaces. Ioffe's theorem may be seen as an extension of the mean value theorem.

Let us first define Clarke's subgradient [7].

DEFINITION 4. *Let $g : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz. For any x and $v \in \mathbb{R}^n$ the directional derivative of f at x in the direction v is*

$$Dg(x, v) = \limsup_{\substack{y \to x \\ \lambda \to 0_+}} \frac{g(y + \lambda v) - g(y)}{\lambda}.$$

The generalized gradient $\partial g(x)$ of g at x is the set of vectors x^* in $\mathbb{R}^n$ satisfying

$$\langle v, x^* \rangle \leq Dg(x, v)$$

for each $v \in \mathbb{R}^n$.

We now recall Rademacher's theorem. It asserts that a function which is Lipschitz on an open subset of $\mathbb{R}^n$ is differentiable almost everywhere on that subset. Based on this result, the generalized gradient has the following characterization: For any set S of measure zero, we have

$$\partial g(x) = co \left\{ \lim_{i \to \infty} \nabla g(x_i) \mid g \text{ is differentiable at } x_i,\ x_i \notin S,\ x_i \to x \right\}$$

where *co* denotes the convex hull.

THEOREM 7 (Ioffe). *Let $g : \mathbb{R}^n \to \mathbb{R}$ be locally Lipschitz, let A be the set of zeros of g, and $a \in A$. Suppose there exist ϵ and $c > 0$ such that for each $x \in \bar{B}(a, \epsilon) \setminus A$ and $x^* \in \partial g(x)$ the inequality $\|x^*\| \geq c$ holds. Then for each $x \in \bar{B}(a, \epsilon/2)$, we have*

$$c\mathrm{Dist}(x, A) \leq |g(x)|.$$

Taking $g(x) = \|f(x)^-\|$ gives, using Ioffe's theorem, the following corollary.

COROLLARY 6. *Given $f : \mathbb{R}^n \to \mathbb{R}^m$ locally Lipschitz, let $x \in \mathbb{R}^n$ with $f(x) \geq 0$ and $\epsilon > 0$. Let us define $c = \min \|y^*\|$, where the minimum is taken for $\|y - x\| \leq \epsilon$ with $y \notin f^{\geq 0}$ and $y^* \in \partial \|f^-\|(y)$. Then*

$$c\mathrm{Dist}(y, f^{\geq 0}) \leq \|f(y)^-\|$$

for any y with $\mathrm{Dist}(y, f^{\geq 0}) \leq \epsilon/2$.

Remark 2. (1) To our knowledge there is no analogue in the literature to Theorems 2 and 3 where strict inequalities are considered.

(2) In Theorems 1, 2, and 3 we prove the exact feasibility of the corresponding system. This exact feasibility is assumed in Hoffman's theorem, in Lewis–Pang's theorem, in Lojasiewicz's theorem, and in Ioffe's theorem.

(3) The proofs of Theorems 1, 2, and 3 are constructive. We are able to compute $y \in \mathbb{R}^n$ satisfying the corresponding system. See section 6.

4. Examples.

4.1. First example. Theorem 1, Lewis–Pang's theorem, and Ioffe's theorem are "empty" when 0 is a singular value of the system $f(x) = 0$, unlike Lojasiewicz's theorem. For example, with $f(x) = -x^2$ we have

$$f^{\geq 0} = \{0\}, \quad \|f(x)^-\| = x^2, \quad \text{and} \quad \operatorname{Dist}(x, f^{\geq 0}) = |x|.$$

The inequality

$$\alpha \operatorname{Dist}(x, f^{\geq 0})^N \leq \|f(x)^-\|$$

holds for any x in a neighborhood of 0 with $\alpha = 1$ and $N = 2$. We cannot have $N = 1$ as it is necessarily the case in Lewis–Pang's theorem or in Ioffe's theorem. Theorem 1 is also empty at $x = 0$ because $\delta(f, 0) = \infty$.

4.2. Second example. Theorem 1 describes a neighborhood of the set $f^{\geq 0}$ such that

$$\operatorname{Dist}(x, f^{\geq 0}) \leq \sigma\delta(f, x)\|f(x)^-\|$$

in this neighborhood. Here the quantity $\sigma\delta(f, x)$ depends on x unlike the constants appearing in Theorems 4, 5, and 6. We may replace $\sigma\delta(f, x)$ with a quantity independent on x when $\sup_x \sigma\delta(f, x) < \infty$. But this gives a weaker result and a serious limitation on f.

Let us consider the feasible set defined by the polynomial equation $f(x_1, x_2) = x_1x_2 - 1$. We have here

$$\|Df(x)\| = \|x\|, \quad \|D^2f(x)\| = 1, \quad \text{and} \quad \delta(f, x) = \|x\|^{-1}.$$

Corollary 1 becomes

$$\operatorname{Dist}(x, f^{\geq 0}) \leq \sigma\|x\|^{-1}f(x)^-$$

as soon as either $x_1x_2 - 1 \geq 0$ or $x_1x_2 - 1 < 0$ and $f(x)^- = 1 - x_1x_2 \leq (x_1^2 + x_2^2)/12$. See Figure 1. For any x in this set we have

$$0 < \sigma\delta(f, x) \leq \sigma\sqrt{\frac{7}{6}}.$$

4.3. Third example. In this example we have $\sup_x \sigma\delta(f, x) = \infty$. Let us consider $f : \mathbb{R}^2 \to \mathbb{R}^2$ defined by

$$f_1(x_1, x_2) = x_1^2 - x_2 \quad \text{and} \quad f_2(x_1, x_2) = x_2.$$

The feasible set $f^{\geq 0}$ is located in the plane between the curve $x_2 = x_1^2$ and the abscissa axis. The condition

$$\|f(x)^-\| \leq \frac{1}{4(2 + \|D^2f(x)\|)\delta(f, x)^2}$$

given in Theorem 1 defines a neighborhood of the set $f^{\geq 0}$ located between the implicit curves

$$x_2 \geq 0 \quad \text{and} \quad x_2 - x_1^2 = \frac{(4x_1^2 + 1)(4x_2 + 1) - 1}{16\left(4x_1^2 + 4x_2 + 2 + \sqrt{4(x_1^2 - x_2)^2 + 1}\right)}$$

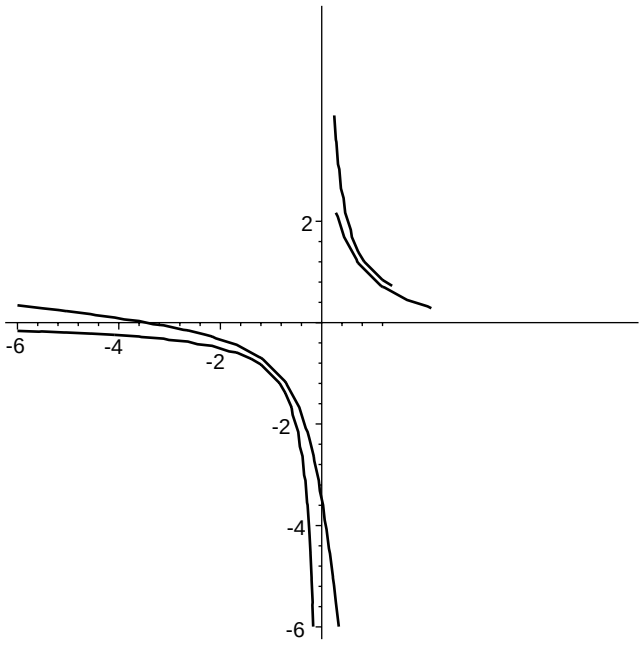


FIG. 1.

and

$$x_2 \le 0 \quad \text{and} \quad -x_2 = \frac{8x_1^2 - 4x_2}{16\left(4x_1^2 - 2x_2 + 1 + \sqrt{(4x_1^2 - 2x_2)^2 + 1}\right)}.$$

Notice that these two curves, together with $x_2 = x_1^2$ and $x_2 = 0$, have an order one contact at the origin. It is easy to see that $\delta(f, 0) = \infty$. See Figure 2. For any x in this neighborhood, by Theorem 1 we have

$$\text{Dist}(x, f^{\geq 0}) \le \sigma\delta(f, x)\|f(x)^-\|.$$

The expression of $\delta(f, x)$ is given by

$$\delta(f, x)^2 = \frac{4x_1^2 + 4x_2 + 2 + \sqrt{4(x_1^2 - x_2)^2 + 1}}{(4x_1^2 + 1)(4x_2 + 1) - 1}$$

when $x = (x_1, x_2)$ is such that $f_1(x_1, x_2) < 0$ and $f_2(x_1, x_2) \ge 0$ and by

$$\delta(f, x)^2 = \frac{4x_1^2 - 2x_2 + 1 + \sqrt{(4x_1^2 - 2x_2)^2 + 1}}{8x_1^2 - 4x_2}$$

when $f_1(x_1, x_2) \ge 0$ and $f_2(x_1, x_2) < 0$.

5. Scheme of the proofs: Alpha-theory for underdetermined systems of equations. To prove Theorems 1, 2, and 3, we use neither convex analysis, nor Lojasiewicz's inequality, nor nonsmooth analysis but another powerful argument based on Smale's alpha-theory.

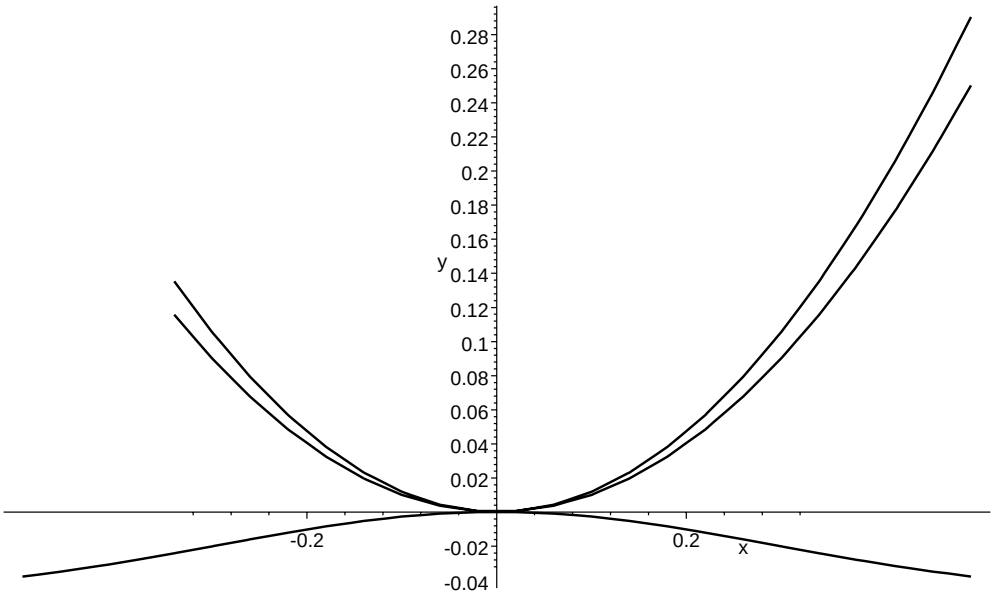


FIG. 2.

We associate to the system of inequalities $f(y) \geq 0$ the underdetermined system

$$F_i(y,t) = f_i(y) - t_i^2,\ 1 \leq i \leq m,$$

which is a system of m equations in $m+n$ unknowns: $(y,t) \in \mathbb{R}^n \times \mathbb{R}^m$. We see easily that $F(y,t) = 0$ implies $f(y) \geq 0$. To prove the existence of a zero (y,t) for F with y close to x, we show that Newton's sequence $(x_{k+1}, t_{k+1}) = N_F(x_k, t_k)$ starting at $(x_0, t_0) = (x, \sqrt{f(x)^+})$ is converging. Its limit is a zero for F. This process is interesting because it provides a very efficient way to compute y.

Such a method has already been used by Cucker and Smale in [9], where the authors study the complexity of the feasibility of a system of polynomial equalities and inequalities in n variables.

Newton's method for underdetermined systems of equations was introduced for the first time in 1966 by Ben-Israel [3]. This iteration is defined by

$$N_F(z) = z - DF(z)^\dagger F(z), \quad z_{k+1} = N_F(z_k),$$

where z_0 is given. We denote here by $DF(z)^\dagger$ the Moore–Penrose inverse of the derivative $DF(z)$. When $DF(z)$ is onto and, more generally, for a surjective linear operator L between two Euclidean spaces, its Moore–Penrose inverse is given by $L^\dagger = L^*(LL^*)^{-1}$ with L^* the adjoint of L.

To prove the convergence of the sequence $(x_{k+1}, t_{k+1}) = N_F(x_k, t_k)$, we use a theorem due to Shub and Smale [23] (Theorem 8 given below). See also Dedieu and Kim [11] for a more general context and Ben-Israel [3] for a convergence result "à la Kantorovich."

Let $F : \mathbb{E} \to \mathbb{F}$ be an analytic function between two Euclidean spaces. We suppose here that $\dim\ \mathbb{E} \geq \dim\ \mathbb{F}$. To F and a given $z \in \mathbb{E}$ we associate the three following numbers:

$$\alpha(F,z) = \beta(F,z)\gamma(F,z),$$

$$\beta(F,z) = \|DF(z)^\dagger F(z)\|,$$

$$\gamma(F,z) = \sup_{k\geq 2} \left\| DF(z)^\dagger \frac{D^k F(z)}{k!} \right\|^{\frac{1}{k-1}}.$$

We give the value ∞ to these three numbers when $DF(z)$ is not onto. When $DF(z)$ is onto, $\gamma(F,z)$ is finite because F is analytic. It can be proved (see Blum et al. [4, Chap. 8, Prop. 6]) that the radius of convergence R of the Taylor series of F at z satisfies $R \geq 1/\gamma(F,z)$. $\beta(F,z)$ is the size of Newton's correction at z. The following theorem is taken from Shub and Smale [23, Theorem C1].

THEOREM 8. *There is a universal constant α_0, approximately $1/7$, with the following property. For any $z_0 \in \mathbb{E}$ with $\alpha(F,z_0) < \alpha_0$, all the Newton iterates*

$$z_{k+1} = z_k - DF(z_k)^\dagger F(z_k),\ k \geq 0,$$

are defined, converge to $\zeta \in \mathbb{E}$ with $F(\zeta) = 0$, and for all $k \geq 0$

$$\|z_{k+1} - z_k\| \leq \left(\frac{1}{2}\right)^{2^k - 1} \|z_1 - z_0\|.$$

In particular,

$$\|\zeta - z_0\| \leq \sigma\beta(F,z_0).$$

The numbers α, β, and γ appear for the first time in a paper by Smale [24] in the context of well-determined systems of equations.

6. Proofs of Theorems 1, 2, 3, and Corollary 1. Theorem 1 is an easy consequence of Theorem 2. The proof of Theorem 2 goes as follows. When $\delta(f,x) = \infty$, we deduce from the hypothesis the equality $f(x)^- = 0$. Thus, $f(x) \geq 0$ and we are done. When $\delta(f,x) < \infty$, let $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ be defined by

$$F(y,t) = (f_i(y) - t_i^2)_{1\leq i\leq m}.$$

Its derivative is given by

$$DF(y,t) = \begin{pmatrix} Df_1(y) & -2t_1 & 0 & \dots & 0 \\ Df_2(y) & 0 & -2t_2 & \dots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ Df_m(y) & 0 & 0 & \dots & -2t_m \end{pmatrix} = \left(Df(y)\ T\right).$$

$DF(y,t)$ has rank m if and only if

$$DF(y,t)DF(y,t)^* = Df(y)Df(y)^* + T^2$$

is nonsingular. Since $\delta(f,x)$ is finite, the matrix

$$DF\left(x, \sqrt{f(x)^+}\right) DF\left(x, \sqrt{f(x)^+}\right)^* = Df(x)Df(x)^* + \mathrm{Diag}(4f(x)^+)$$

is nonsingular; thus $DF(x, \sqrt{f(x)^+})$ has rank m. Its Moore–Penrose inverse is equal to

$$DF^\dagger = DF^*(DFDF^*)^{-1}.$$

To compute the operator norm of this matrix we use the classical equality $\|B\| = \rho(B^*B)^{1/2}$ valid for any $m \times n$ matrix, where $\rho(A)$ denotes the spectral radius of the matrix A. We also have $\|A\| = \rho(A)$ when A is real symmetric. This yields

$$\|DF^\dagger\| = \|DF^*(DFDF^*)^{-1}\| = \rho((DF^*(DFDF^*)^{-1})^* DF^*(DFDF^*)^{-1})^{1/2}$$

$$= \rho((DFDF^*)^{-1} DFDF^*(DFDF^*)^{-1})^{1/2} = \rho((DFDF^*)^{-1})^{1/2}$$

$$= \|(DF(x, \sqrt{f(x)^+})DF(x, \sqrt{f(x)^+})^*)^{-1}\|^{1/2} = \delta(f, x).$$

The second derivative of F is given by

$$D^2F_i(y,t)((u,r),(v,s)) = D^2 f_i(y)(u,v) - 2r_i s_i.$$

Thus

$$\|D^2F(y,t)\| \le \|D^2 f(y)\| + 2.$$

The other derivatives satisfy $D^k F(y,t) = D^k f(y)$ for any $k \ge 3$ so that

$$\Gamma(F,(y,t)) \le 1 + \Gamma(f,y).$$

Let us now give an estimate for $\alpha(F, (x, \sqrt{f(x)^+}))$. According to the definition of this number, we have

$$\alpha\left(F, (x, \sqrt{f(x)^+})\right) = \beta\left(F, (x, \sqrt{f(x)^+})\right)\ \gamma\left(F, (x, \sqrt{f(x)^+})\right)$$

$$= \|DF(x, \sqrt{f(x)^+})^\dagger F(x, \sqrt{f(x)^+})\| \sup_{k\ge 2} \left\| DF(x, \sqrt{f(x)^+})^\dagger \frac{D^k(F, (x, \sqrt{f(x)^+}))}{k!} \right\|^{\frac{1}{k-1}}$$

$$\le \|DF(x, \sqrt{f(x)^+})^\dagger\| \|F(x, \sqrt{f(x)^+})\| \sup_{k\ge 2} \|DF(x, \sqrt{f(x)^+})^\dagger\|^{\frac{1}{k-1}} \Gamma(F, (x, \sqrt{f(x)^+}))$$

$$\le \delta(f,x)\|f(x)^-\| \max(1, \delta(f,x))(1 + \Gamma(f,x)).$$

From the hypothesis we get

$$\alpha(F, (x, \sqrt{f(x)^+})) \le \frac{1}{8} < \alpha_0;$$

thus, by Theorem 8 there exists $(y,t) \in \mathbb{R}^n \times \mathbb{R}^m$ such that $F(y,t) = 0$ and

$$\|(y,t) - (x, \sqrt{f(x)^+})\| \le \sigma\beta(F, (x, \sqrt{f(x)^+}))$$

$$\leq \sigma \|DF(x, \sqrt{f(x)^+})^\dagger\| \|F(x, \sqrt{f(x)^+})\| \leq \sigma\delta(f,x)\|f(x)^-\|.$$

For any i we have

$$f_i(y) = t_i^2 \geq 0.$$

Moreover, for any $i \in J$,

$$|t_i - \sqrt{f_i(x)^+}| \leq \|(y,t) - (x, \sqrt{f(x)^+})\| \leq \sigma\delta(f,x)\|f(x)^-\|.$$

Since, by the hypothesis, $\sigma\delta(f,x)\|f(x)^-\| < \sqrt{f_i(x)^+}$, we obtain $t_i > 0$ so that $f_i(y) = t_i^2 > 0$ for any $i \in J$. To finish the proof of Theorem 2 we notice that

$$\mathrm{Dist}(x, f^{\geq 0}) \leq \|x - y\| \leq \|(y,t) - (x, \sqrt{f(x)^+})\| \leq \sigma\delta(f,x)\|f(x)^-\|.$$

The last assertion in Theorem 1 comes from the following. When f is a polynomial function with degree $f_i \leq 2$, then

$$\gamma(F, (x, \sqrt{f(x)^+})) = \|DF(x, \sqrt{f(x)^+})^\dagger \frac{D^2F(x, \sqrt{f(x)^+})}{2}\| \leq \delta(f,x)\left(1 + \frac{\|D^2 f(x)\|}{2}\right).$$

Thus, $\alpha(F, (x, \sqrt{f(x)^+})) \leq \frac{1}{8} < \alpha_0$ as soon as

$$\|f(x)^-\| \leq \frac{1}{4(2 + \|D^2 f(x)\|)\delta(f,x)^2}. \qquad \square$$

Proof of Theorem 3. We proceed similarly to the proof of Theorem 2. When $\delta(f,g,x) = \infty$, then $g(x) = 0$ and $f(x) \geq 0$. We take $y = x$ and we are done. When $\delta(f,g,x) < \infty$, let us define $F : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m \times \mathbb{R}^q$ by $F_i(y,t) = f_i(y) - t_i^2$, $1 \leq i \leq m$, and $F_j(y,t) = g_j(y)$, $1 \leq j \leq q$. We recall that $q \leq n$. Its derivative is given by

$$DF(y,t) = \begin{pmatrix} Df(y) & \mathrm{Diag}(-2t_i) \\ Dg(y) & 0 \end{pmatrix}.$$

Since $\delta(f,g,x) < \infty$, as in the proof of Theorems 1 and 2, $DF(x, \sqrt{f(x)^+})$ has rank $m+n$. Its Moore–Penrose inverse is equal to $DF^\dagger = DF^*(DFDF^*)^{-1}$ and the norm of this inverse is equal to $\|DF^\dagger\| = \delta(f,g,x)$. Thus

$$\beta(F, (x, \sqrt{f(x)^+})) \leq \delta(f,g,x)(\|f(x)^-\|^2 + \|g(x)\|^2)^{1/2}.$$

We also have, as in the proof of Theorem 2,

$$\Gamma(F, (x, \sqrt{f(x)^+})) \leq 1 + \Gamma(f,g,x).$$

This yields

$$\alpha(F, (x, \sqrt{f(x)^+})) \leq \delta(f,g,x)(\|f(x)^-\|^2 + \|g(x)\|^2)^{1/2} \max(1, \delta(f,g,x))(1 + \Gamma(f,g,x))$$

so that, from the hypothesis, we get $\alpha(F, (x, \sqrt{f(x)^+})) < 1/8 < \alpha_0$. Theorem 8 ensures the existence of a zero (y,t) for F satisfying

$$\|(y,t) - (x, \sqrt{f(x)^+})\| \leq \sigma\delta(f,g,x)(\|f(x)^-\|^2 + \|g(x)\|^2)^{1/2}.$$

The rest of the proof is similar to the proof of Theorem 2. □

Proof of Corollary 1. Under the hypothesis, $\delta(f,x) = (\|Df(x)\|^2 + 4f(x)^+)^{-\frac{1}{2}}$ so that, by Theorem 1, the inequality

$$f(x)^- \leq \frac{(\|Df(x)\|^2 + 4f(x)^+)^{\frac{1}{2}} \min(1, (\|Df(x)\|^2 + 4f(x)^+)^{\frac{1}{2}})}{8(1+\Gamma(f,x))}$$

implies

$$\mathrm{Dist}(x, f(x)^{\geq 0}) \leq \sigma(\|Df(x)\|^2 + 4f(x)^+)^{-\frac{1}{2}} f(x)^-.$$

When $f(x) \leq 0$, this hypothesis is equivalent to

$$f(x)^- \leq \frac{\|Df(x)\| \min(1, \|Df(x)\|)}{8(1+\Gamma(f,x))}$$

and the conclusion becomes

$$\mathrm{Dist}(x, f(x)^{\geq 0}) \leq \sigma \|Df(x)\|^{-1} f(x)^-.$$

In the case $f(x) \geq 0$ this corollary is obvious. □

Proofs of Corollaries 2 *and* 4. The proof of Corollary 4 is similar to the proof of Corollary 2. The proof of Corollary 2 goes as follow. By Theorem 1 there exists a certain y with $f(y) \geq 0$ and $\|x-y\| \leq \sigma\delta(f,x)\|f(x)^-\|$. If $f(x) \ngeq 0$, then the interval $[x,y]$ contains necessarily a point $z \in \Sigma$, the boundary of $f^{\geq 0}$. Thus $\mathrm{Dist}(x,\Sigma) \leq \|x-y\| \leq \sigma\delta(f,x)\|f(x)^-\|$ which contradicts the hypothesis

$$\mathrm{Dist}(x,\Sigma)^{-1} = \mu(x) < \frac{1}{\sigma\delta(f,g,x)\|f(x)^-\|}. \qquad \square$$

Proof of Corollary 3. For any $i = 1, \ldots, m$ one has either $f_i(x) \geq 0$ so that $f_i(x)^- = 0$ or $f_i(x) < 0$. In this last case $\epsilon_i \geq -f_i(x) = f_i(x)^- > 0$. Thus

$$\|f(x)^-\| \leq \|\epsilon^{<0}\| \leq \frac{1}{8(1+\Gamma(f,x))\delta(f,x)\max(1,\delta(f,x))}.$$

As in the proof of Theorem 2, there exists $(y,t) \in \mathbb{R}^n \times \mathbb{R}^m$ with $f(y) = t^2$ and

$$\|(y,t) - (x, \sqrt{f^+(x)})\| \leq \sigma\delta(f,x)\|f(x)^-\|.$$

This gives $f(y) \geq 0$, and

$$\|y-x\| \leq \sigma\delta(f,x)\|f(x)^-\| \leq \sigma\delta(f,x)\|\epsilon^{<0}\|.$$

Moreover,

$$\left| t_i - \sqrt{f_i^+(x)} \right| \leq \sigma\delta(f,x)\|f(x)^-\| \leq \sigma\delta(f,x)\|\epsilon^{<0}\|$$

so that

$$t_i \geq \sqrt{f_i^+(x)} - \sigma\delta(f,x)\|\epsilon^{<0}\|.$$

When $i = p+1, \ldots, m$, this last quantity is positive and, consequently, $f_i(y) = t_i^2 > 0$. □

Acknowledgments. The author is very grateful to the referees for their careful reading and for their constructive remarks which improved the paper significantly.

REFERENCES

[1] A. Auslender and J.-P. Crouzeix, *Global regularity theorems*, Math. Oper. Res., 13 (1988), pp. 243–253.

[2] D. Azé, J.-N. Corvellec, and R. F. Lucchetti, *Variational Pairs and Applications to Stability in Nonsmooth Analysis*, Tech. report, Université Paul Sabátier, Toulous, France, 1999.

[3] A. Ben-Israel, *A Newton-Raphson method for the solution of systems of equations*, J. Math. Anal. Appl., 15 (1966), pp. 243–252.

[4] L. Blum, F. Cucker, M. Shub, and S. Smale, *Complexity and Real Computation*, Springer-Verlag, Berlin, New York, 1998.

[5] L. Blum, M. Shub, and S. Smale, *On a theory of computation and complexity over the real numbers; NP completeness, recursive functions and universal machines*, Bull. Amer. Math. Soc. (New Series), 21 (1989), pp. 1–46.

[6] E. Bierstone and P. Milman, *Semianalytic and subanalytic sets*, Inst. Hautes Études Sci. Publ. Math., 67 (1988), pp. 5–42.

[7] F. Clarke, *Optimization and Nonsmooth Analysis*, J. Wiley and Sons, New York, 1983.

[8] F. Cucker and J.-P. Dedieu, *Decision Problems and Round-Off Machines*, preprint, 1998.

[9] F. Cucker and S. Smale, *Complexity estimates depending on condition and round-off error*, J. ACM, 46 (1999), pp. 113–184.

[10] J.-P. Dedieu, *Penalty functions in subanalytic optimization*, Optimization, 26 (1992), pp. 27–32.

[11] J.-P. Dedieu and M.-H. Kim, *Newton's Method for Analytic Systems of Equations with Constant Rank Derivatives*, preprint, 1999.

[12] 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.

[13] H. Hironaka, *Introduction to Real-Analytic Sets and Real-Analytic Maps*, mimeographed notes, University of Pisa, Pisa, 1973.

[14] A. Hoffman, *On approximate solutions of systems of linear inequalities*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 263–265.

[15] A. Ioffe, *Regular points of Lipschitz functions*, Trans. AMS, 251 (1979), pp. 61–69.

[16] A. Lewis and J.-S. Pang, *Error bounds for convex inequality systems*, in Generalized Convexity, Generalized Monotonicity, J.-P. Crouzeix, J.-E. Martinez-Legaz, and M. Volle, eds., Kluwer, Dordrecht, 1998, pp. 75–110.

[17] S. Lojasiewicz, *Ensembles Semi-Analytiques*, mimeographed notes, Institut des Hautes Études Scientifiques Bures nur Yvette, France, 1964.

[18] X.-D. Luo and Z.-Q. Luo, *Extension of Hoffman's error bound to polynomial systems*, SIAM J. Optim., 4 (1994), pp. 383–392.

[19] X. Luo and J. Pang, *Error bounds for analytic systems and their applications*, Math. Programming, 67 (1995), pp. 1–28.

[20] O. Mangasarian, *A condition number for differentiable convex inequalities*, Math. Oper. Res., 10 (1985), pp. 175–179.

[21] J. Renegar, *Is it possible to know a problem instance ill-posed?*, J. Complexity, 10 (1994), pp. 1–56.

[22] S.M. Robinson, *An application of error bounds for convex pogramming in a linear space*, SIAM J. Control Optim., 13 (1975), pp. 271–273.

[23] M. Shub and S. Smale, *Complexity of Bezout's theorem* IV*: Probability of success; extensions*, SIAM J. Numer. Anal., 33 (1996), pp. 128–148.

[24] S. Smale, *Newton's method estimates from data at one point*, in The Merging of Disciplines: New Directions in Pure, Applied, and Computational Mathematics, R. Ewing, K. Gross, and C. Martin, eds., Springer-Verlag, Berlin, New York, 1986, pp. 185–196.

[25] J. Warga, *A necessary and sufficient condition for a constrained minimum*, SIAM J. Optim., 2 (1992), pp. 665–667.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 426–443

OPTIMUM PROBLEMS WITH MEASURABLE SET-VALUED CONSTRAINTS*

ZSOLT PÁLES† AND VERA ZEIDAN‡

Abstract. In this paper, we provide a complete analysis of second-order admissible variations to inequality-type constraints, which are given in terms of measurable set-valued functions whose images are closed convex sets with nonempty interior. As an application, we consider optimization problems where such constraints are present, and we deduce second-order necessary conditions for optimality.

Key words. measurable set-valued maps, convex sets in $L^\infty(\Omega, \mathbb{R}^m)$, support functional, tangent and normal cones, second-order admissible variations, second-order optimality conditions

AMS subject classifications. Primary, 90C48, 90C34; Secondary, 58C06, 47H04

PII. S1052623499350943

1. Introduction.

Consider the following optimization problem:

$$(\mathcal{P}) \qquad \text{Minimize} \quad F_0(z) \quad \text{subject to} \quad E(z) = 0,\ F(z) \leq 0,\ G(z) \in \mathbf{Q},$$

where $F_0 : \mathcal{D} \to \mathbb{R}$, $E : \mathcal{D} \to Y$, $F : \mathcal{D} \to \mathbb{R}^p$, $G : \mathcal{D} \to X$, and X, Y, Z are Banach spaces, $\mathcal{D} \subset Z$ is nonempty and open, and $\mathbf{Q} \subset X$ is a closed convex set with nonempty interior.

The prototype of such problems arises, for instance, in optimal control theory with control constraints in the inclusion form $x(t) \in Q(t)$ (for all $t \in \Omega$), where Q is a measurable set-valued map with closed convex nonempty interior images on the complete finite measure space $(\Omega, \mathcal{A}, \mu)$.

Better understanding of optimality conditions is an ongoing research program for several researchers. This question is of great value in theory and in applications. Usually, such conditions must be given in terms of the original data of the problem and, in the context of necessity, are expected to be as strong as they can be.

In 1988, Kawasaki [11], [12] discovered, for the problem $(\mathcal{P})$, where $\mathbf{Q}$ is a cone, second-order necessary conditions that contain an extra term manifesting the presence of infinitely many inequalities in the constraint $G(z) \in \mathbf{Q}$. This phenomenon is known as the "envelope-like effect." Such result was generalized by Cominetti in [4]. Both results assumed a Mangasarian–Fromovitz-type condition.

In [18] the authors have generalized the previous results in [11], [12], and [4] to the nondifferentiable case without assuming a Mangasarian–Fromowitz condition. The second-order admissible variation set used therein (defined first by Dubovitskii and Milyutin in [7], [8]) is described in the following definition.

DEFINITION. *Let X be a normed space, $\mathbf{Q} \subset X$, $x \in \mathbf{Q}$, and $d \in X$. A vector $v \in X$ is called a* second-order admissible variation of $\mathbf{Q}$ at x in the direction d *if*

*Received by the editors January 25, 1999; accepted for publication (in revised form) May 11, 2000; published electronically October 18, 2000.
http://www.siam.org/journals/siopt/11-2/35094.html

†Institute of Mathematics and Informatics, Debrecen University, H-4010 Debrecen, Pf. 12, Hungary (pales@math.klte.hu). This author's research was supported by Hungarian National Foundation for Scientific Research (OTKA) grant T-030082 and by Hungarian Higher Education Research and Development Fund (FKFP) grant 0310/1997.

‡Department of Mathematics, Michigan State University, East Lansing, MI 48824 (zeidan@math.msu.edu). This author's research was supported partially by the Department of Mathematics at Michigan State University.

there exists $\overline{\varepsilon} > 0$ *such that*

$$x + \varepsilon d + \varepsilon^2(v + w) \in \mathbf{Q} \qquad \text{for all } 0 < \varepsilon < \overline{\varepsilon},\ \|w\| < \overline{\varepsilon},\ w \in X.$$

The set of all such variations is denoted by $V(x, d|\mathbf{Q})$. It follows directly from the definition that $V(x, d|\mathbf{Q})$ is an open set. If $\mathbf{Q}$ is also convex, then $V(x, d|\mathbf{Q})$ is convex as well. Results related to those in [18] were obtained by Maruyama [14, Theorem 3.2], where Neustadt's derivative was used to handle the nonsmoothness of data.

In order to derive meaningful second-order optimality conditions, it is imperative to choose directions d that guarantee the nonemptiness of $V(x, d|\mathbf{Q})$. Such directions $d \in X$ are labeled as the *critical directions of* $\mathbf{Q}$ *at* x and form a set called *critical direction cone to* $\mathbf{Q}$ *at* x. Throughout this paper, this cone will be denoted by $C(x|\mathbf{Q})$. It can be easily seen that $C(x|\mathbf{Q})$ is a convex set such that

$$\operatorname{cone}(Q - x) \subset C(x|Q) \subset \overline{\operatorname{cone}}(Q - x).$$

In order to recall the first- and second-order necessary conditions for $(\mathcal{P})$, obtained in [18, Corollary 2], we need to introduce the following notation and notions.

- A point $\hat{z} \in \mathcal{D}$ is called an *admissible point* for $(\mathcal{P})$ if $E(\hat{z}) = 0$, $F(\hat{z}) \leq 0$, and $G(\hat{z}) \in \mathbf{Q}$ hold. A point $\hat{z} \in \mathcal{D}$ is a *solution (local minimum)* of the problem if it is admissible and there exists a neighborhood U of $\hat{z}$ such that $F_0(z) \geq F_0(\hat{z})$ for all admissible points $z \in U$.
- A point $\hat{z} \in \mathcal{D}$ is called a *regular point* for $(\mathcal{P})$ if
  - (R_1) $F_0, F = (F_1, \ldots, F_p)$ are locally Lipschitz at $\hat{z}$;
  - (R_2) G is strictly Fréchet differentiable at $\hat{z}$;
  - (R_3) E is strictly Fréchet differentiable at $\hat{z}$ and the range of the linear operator $E'(\hat{z})$ is a closed subspace of Y.

If F_i $(i = 0, \ldots, p)$ is locally Lipschitz at $\hat{z}$, then the expression

$$F_i^o(\hat{z}; y) := \limsup_{(z,\varepsilon)\to(\hat{z},0+)} \frac{F_i(z + \varepsilon y) - F_i(z)}{\varepsilon}$$

is finite and will be called *Clarke's generalized directional derivative* in the direction y. The corresponding *generalized gradient* $\partial F_i(\hat{z})$ is defined by

$$\partial F_i(\hat{z}) := \{z^* \in Z^* : \langle z^*, z\rangle \leq F_i^o(\hat{z}; z) \text{ for all } z \in Z\}.$$

For properties of these notions, see [2].

Let $\hat{z}$ be an admissible regular point for $(\mathcal{P})$ and $d \in Z$.

- A vector $y \in Z$ is called a *critical direction* at $\hat{z}$ for $(\mathcal{P})$ if
  - (C_1) $F_i^o(\hat{z}; y) \leq 0$ for all $i = 0, \ldots, p$;
  - (C_2) $G'(\hat{z})y \in C(G(\hat{z})|\mathbf{Q})$;
  - (C_3) $E'(\hat{z})y = 0$.
- A vector $y \in Z$ is called a *regular direction* at $\hat{z}$ for $(\mathcal{P})$ if
  - (R_4) for all $i = 0, \ldots, p$,

$$F_i^{o\prime}(\hat{z}, y) := \limsup_{\varepsilon\to 0+} 2\frac{F_i(\hat{z} + \varepsilon y) - F_i(\hat{z}) - \varepsilon F_i^o(\hat{z}; y)}{\varepsilon^2}$$

  is finite;
  - (R_5) the second-order directional derivative of $L := (G, E)$

$$L''(\hat{z},y) := \lim_{\varepsilon\to 0+} 2\frac{L(\hat{z}+\varepsilon y)-L(\hat{z})-\varepsilon L'(\hat{z})y}{\varepsilon^2}$$

exists.

Clearly, the zero vector is always a regular critical direction at $\hat{z}$ for $(\mathcal{P})$.

Now we are ready to state the result of [18, Corollary 2].

THEOREM 1.1. *Let $\hat{z}$ be a regular local solution of the above problem $(\mathcal{P})$. Then, for all regular critical directions y, there correspond Lagrange multipliers $\lambda_i \geq 0$ $(i = 0,\dots,p)$, $x^* \in X^*$, and $y^* \in Y^*$ which depend on y, such that at least one of them is different from zero and the following relations hold:*

$$\lambda_i F_i(\hat{z}) = 0 \qquad \text{for all } i = 1,\dots,p \quad \text{and} \quad x^* \in N(G(\hat{z})|\mathbf{Q}), \tag{1.1}$$

$$\sum_{i=0}^{p} \lambda_i F_i^o(\hat{z};z) + \langle x^*, G'(\hat{z})z\rangle + \langle y^*, E'(\hat{z})z\rangle \geq 0 \qquad \text{for } z \in Z, \tag{1.2}$$

and

$$\sum_{i=0}^{p} \lambda_i F^{o\prime}(\hat{z},y) + \langle x^*, G''(\hat{z},y)\rangle + \langle y^*, E''(\hat{z},y)\rangle \geq 2\delta^*\Big(x^*\Big|V(G(\hat{z}),G'(\hat{z})y|\mathbf{Q})\Big). \tag{1.3}$$

(Here δ^* stands for the support function and $N(x|\mathbf{Q})$ denotes the adjoint cone of $T(x|\mathbf{Q})$, that is, the cone of outward normals to the set $\mathbf{Q}$ at the point x [24].)

We note that, using the Hahn–Banach theorem, the first-order condition (1.2) can also be expressed as an equality: There exist linear functionals $z_i^* \in \partial F_i(\hat{z})$ $(i = 0,\dots,p)$ such that

$$\sum_{i=0}^{p} \lambda_i z_i^* + x^* \circ G'(\hat{z}) + y^* \circ E'(\hat{z}) = 0.$$

Throughout this paper the term to the right-hand side of inequality (1.3) will be referred to as the *extra term in the second-order condition*.

Results along the line of Theorem 1.1 were obtained by Ioffe [10] and Penot [23] for the differentiable case and in the presence of a certain qualification condition.

Two important questions naturally surface from Theorem 1.1:

(i) How can we check the nonemptiness of $V(x,d|\mathbf{Q})$, that is, how can the critical cone $C(x|\mathbf{Q})$ be characterized, since otherwise the second-order optimality conditions would be satisfied trivially?

(ii) How can we evaluate the support function of $V(x,d|\mathbf{Q})$?

In order that d be in $C(x|\mathbf{Q})$, it is only necessary that $\mathbf{Q}$ have a nonempty interior and that d belong to $\overline{\text{cone}}(\mathbf{Q}-x) = T(x|\mathbf{Q})$, which is the tangent cone to $\mathbf{Q}$ at x. If $d \in \text{cone}(\mathbf{Q}-x)$, then $V(x,d|\mathbf{Q})$ is nonempty and $V(x,d|\mathbf{Q}) = \text{cone}(\text{cone}(\text{int}\,\mathbf{Q}-x)-d)$ (cf. [18, Theorem 4]). In this case the right-hand side in the second-order condition (1.3) vanishes. However, examples are provided by Kawasaki [11] in order to show that the necessary conditions with *extra term*, that is, when $d \in \overline{\text{cone}}(\mathbf{Q}-x)$, handle situations that cannot be handled with previous results where d is taken from $\text{cone}(\mathbf{Q}-x)$. Thus, one has to consider also directions $d \in T(x|\mathbf{Q}) \setminus \text{cone}(\mathbf{Q}-x)$. In this case the description of $V(x,d|\mathbf{Q})$ and the characterization of its nonemptiness are far from being trivial.

A significant setting is the case when $\mathbf{Q}$ is a subset of $C(T,\mathbb{R}^r)$ defined by

$$\mathbf{Q}=\sigma_C(Q)=:\{x\in C(T,\mathbb{R}^r)\mid x(t)\in Q(t)\text{ for all }t\in T\},\tag{1.4}$$

where Q is a lower semicontinuous set-valued map whose images are closed, convex sets with nonempty interior, and T is a compact Hausdorff space. The importance of this type of constraints stems from control problems with state constraints. This problem has been studied and a satisfactory answer to questions (i) and (ii) above have been provided in a recent paper by the authors [22].

Another case of great interest is when $\mathbf{Q}$ is a subset of $L^\infty(\Omega,\mathbb{R}^m)$ defined by

$$\mathbf{Q}=\sigma_\infty(Q):=\{x\in L^\infty(\Omega,\mathbb{R}^m)\mid x(t)\in Q(t)\text{ for almost every (a.e.) }t\in\Omega\},\tag{1.5}$$

where Q is a measurable set-valued map whose images are closed and have nonempty interior, and $(\Omega,\mathcal{A},\mu)$ is a complete finite measure space. This type of constraint is typical for control constraints in control problems. The main goal of this paper is to investigate this type of constraint and to obtain satisfactory necessary conditions for the corresponding optimization problem.

In the case when $\mathbf{Q}:=\sigma_\infty(Q)$, the two questions (i) and (ii) stated above are still open. They can now be rephrased as follows:

(*) Characterize the critical cone $C(x|\sigma_\infty(Q))$. Furthermore, evaluate the support functional of $V(x,d|\sigma_\infty(Q))$ in terms of the images $Q(t)$ and their support functionals $\delta^*(\cdot|Q(t))$.

Note that, by [20] and [21], the set $\sigma_\infty(Q)$ defined by (1.5) is *decomposable*, that is,

$$\chi_A x+\chi_{\Omega\setminus A}y\in\sigma_\infty(Q)\qquad\text{for all }x,y\in\sigma_\infty(Q),\ A\in\mathcal{A}.$$

(Here χ_A denotes the characteristic function of the set A.) Thus, $V(x,d|\sigma_\infty(Q))$ is also decomposable. Therefore, the L^1-closure of $V(x,d|\sigma_\infty(Q))$ can be identified with a measurable set-valued function $\mathcal{V}:\Omega\to 2^{\mathbb{R}^m}$ whose images are nonempty closed sets.

The aim of this paper is to answer positively the open questions in (*) when the values of Q are also convex sets. In section 2, the elements of $C(x|\sigma_\infty(Q))$ are characterized in Theorem 2.5 by a certain boundedness condition (2.15). Furthermore, the support function of $V(x,d|\sigma_\infty(Q))$ is evaluated in Theorem 2.2 and Corollaries 2.3 and 2.7 via the evaluation of the support functions associated with the pointwise sets of second-order admissible directions

$$\mathcal{V}(t):=V(x(t),d(t)|Q(t))\qquad(t\in\Omega).\tag{1.6}$$

The results of this section differ from their counterparts established in [19] for the case when $\mathbf{Q}=\sigma_C(Q)$ (defined in (1.4)). This distinction is mainly due to the continuity requirement on the selections. In fact, the nonemptiness condition for $V(x,d|\sigma_C(Q)$ in [19] was also phrased in terms of a boundedness below of a lower semicontinuous map (see [19, Theorem 3.5]). On the other hand, the pointwise sets $\mathcal{V}(t)$, defined above, play no role whatsoever in the evaluation of the support function of $V(x,d|\sigma_C(Q))$ (see [19, Theorem 3.10]).

In section 3, we apply the results of section 2 to the abstract optimization problem $(\mathcal{P}^*)$ (see section 3), where two types of parametric constraints are present, namely,

$$\begin{array}{lcll} g(t,z) & \in & Q(t) & \text{for a.e. } t\in\Omega,\\ h(t,z) & = & 0 & \text{for a.e. } t\in\Omega,\end{array}\tag{1.7}$$

where Q is as required in (1.5). The main result is given in Theorem 3.1, where the hypotheses and conditions of Theorem 1.1 are phrased in terms of t-pointwise conditions. In particular, condition (C_2) is given in terms of the pointwise tangent cone to $Q(t)$ (see condition (C_2^*)) given in Theorem 2.5. The extra term

$$\delta^*\Big(x^*\Big|V(G(\hat{z}),G'(\hat{z})y|\mathbf{Q})\Big)$$

appearing in the second-order optimality condition is phrased as an integral of a function associated with the set of pointwise second-order admissible variations of $Q(t)$. Another contribution of Theorem 3.1 lies in finding reasonably general conditions (R_2^*)–(R_4^*) which guarantee that the multipliers associated with the parametric constraints (1.7) are in fact represented via integrable functions.

2. Second-order admissible variations. Let $X = L_m^\infty := L^\infty(\Omega,\mathbb{R}^m)$, where $(\Omega,\mathcal{A},\mu)$ is a complete finite measure space, and let $Q:\Omega\to 2^{\mathbb{R}^m}$ be a measurable set-valued function whose images are closed sets with nonempty interior. Define the set $\sigma_\infty(Q)\subset L_m^\infty$ by (1.5). Let $x\in\sigma_\infty(Q)$, $d\in L_m^\infty$, and $\mathbf{V}:=V(x,d|\sigma_\infty(Q))$. In order that $\mathbf{V}$ be nonempty, it is necessary that $\operatorname{int}\sigma_\infty(Q)$ be nonempty. This latter condition is equivalent (by [21, Theorem 3]; see also [20]) to assuming that Q satisfies

$$\exists r\ge\rho>0 \text{ and, for a.e. } t\in\Omega,\ \exists x_t\in\mathbb{R}^m \text{ such that } B_\rho(x_t)\subset Q(t)\cap B_r \tag{2.1}$$

(where $B_\varepsilon(x)$ denotes the ball in $\mathbb{R}^m$ of radius ε centered at x; if $x=0$, then x may be omitted).

A preliminary characterization of $\mathbf{V}$ is given in the following result.

LEMMA 2.1. *Let $v\in L_m^\infty$. Then $v\in\mathbf{V}$ if and only if there exist $\overline{\varepsilon}>0$ and a set $A\in\mathcal{A}$ of full measure such that, for all $0<\varepsilon\le\overline{\varepsilon}$, $u\in B_{\overline{\varepsilon}}\subset\mathbb{R}^m$, and $t\in A$,*

$$x(t)+\varepsilon d(t)+\varepsilon^2(v(t)+u)\in Q(t). \tag{2.2}$$

Proof. Let $v\in\mathbf{V}$. Then, by definition, there exists an $\overline{\varepsilon}>0$ such that for all $0<\varepsilon\le\overline{\varepsilon}$, $w\in L_m^\infty$ with $\|w\|\le\overline{\varepsilon}$, there exists a set $A=A_{\varepsilon,w}$ of full measure that

$$x(t)+\varepsilon d(t)+\varepsilon^2(v(t)+w(t))\in Q(t)\qquad\text{for all } t\in A_{\varepsilon,w}. \tag{2.3}$$

Let $\{(\varepsilon_n,u_n)|n\in\mathbb{N}\}$ be a dense subset of $[0,\overline{\varepsilon}]\times B_{\overline{\varepsilon}}$. Then defining the measurable functions w_n by $w_n(t):=u_n$, we get from (2.3) that for all $n\in\mathbb{N}$,

$$x(t)+\varepsilon_n d(t)+\varepsilon_n^2(v(t)+u_n)\in Q(t)\qquad\text{for all } t\in\bigcap_{n=1}^\infty A_{\varepsilon_n,w_n}.$$

Using the fact that $\{(\varepsilon_n,u_n)\}$ is dense and that $Q(t)$ is closed, we obtain that (2.2) is valid for $\varepsilon\le\overline{\varepsilon}$, $u\in B_{\overline{\varepsilon}}$, and $t\in A:=\bigcap_{n=1}^\infty A_{\varepsilon_n,w_n}$.

Conversely, let $v\in L_m^\infty$ and assume that there exists $\overline{\varepsilon}>0$ and $A\in\mathcal{A}$ of full measure such that (2.2) is valid for all $0<\varepsilon\le\overline{\varepsilon}$, $u\in B_{\overline{\varepsilon}}$, and $t\in A$. Let $w\in L_m^\infty$ such that $\|w\|\le\overline{\varepsilon}$. Then there exists a set $A_w\in\mathcal{A}$ of full measure such that $|w(t)|\le\overline{\varepsilon}$ for all $t\in A_w$. Hence, by (2.2), we have (2.3) with $A_{\varepsilon,w}=A\cap A_w$. Therefore, v belongs to $\mathbf{V}$. □

An immediate consequence of this lemma is that if $v\in\mathbf{V}$, then

$$v(t)\in V(x(t),d(t)|Q(t))\quad\text{for a.e. } t\in\Omega. \tag{2.4}$$

This inclusion motivates the study of the relationship between $\mathbf{V}$ and the measurable set-valued map $\mathcal{V}$ defined by (1.6). Note that the measurability of $\mathcal{V}$ follows from standard arguments.

It is worth noting that (2.4) remains valid when, in $\mathbf{V}$, $\mathbf{Q} = \sigma_\infty(Q)$ is replaced by the set defined in (1.4). In this case, the relationship between $\mathbf{V}$ and $\mathcal{V}$ is not direct, as shown in [20], [21]. However, for the L^∞ setting, a direct connection will be established.

We recall now the notions of L^1-closedness and L^1-closure from [21]. A subset $\mathbf{Q}$ of L^∞_m is called L^1-*closed* if whenever $x_n \in \mathbf{Q}$ for $n \in \mathbb{N}$, $x \in L^\infty$, and

$$\lim_{n\to\infty} \|x_n - x\|_1 = 0,$$

then $x \in \mathbf{Q}$. The L^1-*closure* of a set in L^∞ is the smallest L^1-closed set containing it.

Another important type of closedness can be defined in the following way: A subset $\mathbf{Q}$ of L^∞_m is called Π-*closed (closed in with respect to the so-called Pontryagin* (Π-)*convergence)* if whenever there exists a sequence $x_n \in \mathbf{Q}$ such that

$$\sup \|x_n\|_\infty < \infty \quad \text{and} \quad \lim_{n\to\infty} \|x_n - x\|_1 = 0,$$

then x has to belong to $\mathbf{Q}$. The Π-*closure* of a set is the intersection of all Π-closed sets containing it. Obviously, the class of L^1-closed sets forms a proper subclass of the class of Π-closed sets. However these two notions coincide in the class of decomposable sets as shown by [21, Theorem 1] (see also [20]). The concept of Π-convergence can also be used to define the notion of the Pontryagin (Π-)minimum; see, e.g., [5], [6], [13], [16], [17], and the book by Milyutin and Osmolovskii [15], where necessary and sufficient conditions for this type of minimum are investigated.

Analogously, we can speak about Π-*continuity* of real-valued functions defined on a subset of L^∞_m, and also about (Π, Π)-*continuity* of maps from L^∞_m to L^∞_n.

The L^1-closed and decomposable set $\operatorname{cl}_1 \mathbf{V}$ is known (by [21, Theorem 2]) to be represented via a measurable set-valued map. As we shall see, this set-valued map is in fact $\overline{\mathcal{V}}$, that is, the set-valued map whose images are $\overline{\mathcal{V}(t)}$ (i.e., the closure of $\mathcal{V}(t)$).

THEOREM 2.2. *If* $\mathbf{V} \neq \emptyset$, *then*

$$\operatorname{cl}_1 \mathbf{V} = \sigma_\infty(\overline{\mathcal{V}}). \tag{2.5}$$

Proof. The proof of the "$\subset$" inclusion is obvious since if $v \in \mathbf{V}$, then, from (2.4), we have $v \in \sigma_\infty(\mathcal{V}) \subset \sigma_\infty(\overline{\mathcal{V}})$. Hence $\mathbf{V} \subset \sigma_\infty(\overline{\mathcal{V}})$.

The right-hand side of this inclusion is an L^1-closed set (see [20], [21]); therefore $\operatorname{cl}_1 \mathbf{V}$ is also contained in it.

To prove the reversed inclusion in (2.5), assume that $v_0 \in \sigma_\infty(\overline{\mathcal{V}})$. Then, for all $n \in \mathbb{N}$ and for a.e. $t \in \Omega$, the open ball $U_{1/n}(v_0(t))$ intersects $\mathcal{V}(t)$. Hence, by known selection theorems for measurable set-valued maps (see [3]), there is a measurable selection v_n of the measurable open set-valued map

$$t \mapsto \mathcal{V}(t) \cap U_{1/n}(v_0(t)).$$

Clearly, $v_n \in \sigma_\infty(\mathcal{V})$ and $\|v_n - v_0\|_\infty \leq 1/n$. Therefore, in order to prove that $v_0 \in \operatorname{cl}_1 \mathbf{V}$, it is sufficient to show that $v_n \in \operatorname{cl}_1 \mathbf{V}$. Thus the proof will be completed if we prove

$$\sigma_\infty(\mathcal{V}) \subset \operatorname{cl}_1 \mathbf{V}. \tag{2.6}$$

Let $v \in \sigma_\infty(\mathcal{V})$. Then there exists a set $A \in \mathcal{A}$ of full measure such that $v(t) \in \mathcal{V}(t) = V(x(t), d(t)|Q(t))$ for all $t \in A$. Then, for all $t \in A$, there exists $\bar{\varepsilon}_t > 0$ such that for all $0 < \varepsilon \leq \bar{\varepsilon}_t$, $u \in B_{\bar{\varepsilon}_t}$,

$$x(t) + \varepsilon d(t) + \varepsilon^2(v(t) + u) \in Q(t).$$

Define, for $(\varepsilon, u) \in (0, \infty) \times \mathbb{R}^m$,

$$A_{\varepsilon,u} := \{t \in A \mid x(t) + \varepsilon d(t) + \varepsilon^2(v(t) + u) \in Q(t)\}.$$

Clearly, $A_{\varepsilon,u}$ is measurable. Let $n \in \mathbb{N}$ be fixed, and let $\{(\varepsilon_i, u_i) | i \in \mathbb{N}\}$ be a dense subset of $[0, 1/n] \times B_{1/n}$. Then $\bigcap_{i=1}^\infty A_{\varepsilon_i,u_i}$ is measurable, and by the closedness of $Q(t)$ we have

$$\begin{aligned} A_n &:= \bigcap_{i=1}^{\infty} A_{\varepsilon_i,u_i} \\ &= \left\{t \in A \mid x(t) + \varepsilon d(t) + \varepsilon^2(v(t) + u) \in Q(t) : \text{for all}\, \varepsilon \in [0, 1/n], \text{for all}\, u \in B_{1/n}\right\}. \end{aligned}$$

For all $t \in A$, there exists $n \in \mathbb{N}$ such that $\varepsilon_t > 1/n$; hence $\bigcup_{n=1}^\infty A_n = A$. Thus $\mu(A_n) \to \mu(\Omega)$ as $n \to \infty$.

Let $\bar{v}$ be a fixed element of $\mathbf{V}$ (which exists by the assumption $\mathbf{V} \neq \emptyset$), and define the sequence of functions $\bar{v}_n$ by

$$\bar{v}_n(t) := \begin{cases} v(t) & \text{if } t \in A_n, \\ \bar{v}(t) & \text{if } t \notin A_n. \end{cases}$$

Since $\bar{v} \in \mathbf{V}$, and using Lemma 2.1, there exist a positive $\bar{\varepsilon}$ and a set $\bar{A} \in \mathcal{A}$ of full measure such that

$$x(t) + \varepsilon d(t) + \varepsilon^2(\bar{v}(t) + u) \in Q(t)$$

for all $\varepsilon \in [0, \bar{\varepsilon}]$, $u \in B_{\bar{\varepsilon}}$, and $t \in A_0$. Taking $\bar{\varepsilon}_n = \min(\bar{\varepsilon}, 1/n)$, we get that

$$x(t) + \varepsilon d(t) + \varepsilon^2(\bar{v}_n(t) + u) \in Q(t)$$

if $\varepsilon \in [0, \bar{\varepsilon}_n]$, $u \in B_{\bar{\varepsilon}_n}$, and $t \in A \cap \bar{A}$. It follows from Lemma 2.1 that $\bar{v}_n \in \mathbf{V}$. On the other hand, the sequence $\bar{v}_n$ converges to v in the L^1-norm (since $\mu(A_n) \to \mu(\Omega)$ as $n \to \infty$). Hence, we obtain that $v \in \operatorname{cl}_1 \mathbf{V}$, which completes the proof. □

Remark 2.1. We already know from (2.4) that

$$\mathbf{V} \subset \sigma_\infty(\mathcal{V}).$$

It is natural to investigate whether $\mathbf{V}$ and $\sigma_\infty(\mathcal{V})$ are also related through the reverse inclusion. In Theorem 2.2 we have shown that, by using the L^1-closure of $\mathbf{V}$,

$$\sigma_\infty(\overline{\mathcal{V}}) \subset \operatorname{cl}_1(\mathbf{V}).$$

However, one may ask whether another relation of this type exists by using the L_∞-closure of $\mathbf{V}$. As we shall show in the example below, (2.5) is the only possible such connection. In fact, we shall show that for this example

$$\sigma_\infty(\mathcal{V}) \not\subset \operatorname{cl}_\infty \mathbf{V},$$

and thus also $\sigma_\infty(\overline{\mathcal{V}}) \not\subset \mathrm{cl}_\infty \mathbf{V}$. Hence (2.5) fails to hold when instead of the L^1-closure we use the L^∞-closure.

Example 2.1. For $t \in \Omega := (0,1]$, let

$$Q(t) := \{(x_1, x_2) \in \mathbb{R}^2 \mid 0 \le x_2,\ x_1 \le \sqrt{t x_2}\}.$$

Then the values of Q are closed convex sets; furthermore, $\mathbf{Q} = \sigma_\infty(Q)$ has nonempty interior because, for all $t \in \Omega$,

$$\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 \le 0,\ x_2 \ge 0\} \subset Q(t).$$

It is obvious that $x \equiv 0 \in \mathbf{Q}$. Define $d \in L^\infty(\Omega)$ by $d(t) = (\sqrt{t}, 0)$. Then, $d \in T(0|\mathbf{Q})$, since with

$$d_n(t) := (\sqrt{t}, \varepsilon_n) \qquad (t \in \Omega,\ n \in \mathbb{N})$$

(where $\varepsilon_n \to 0+$), we have that

$$\|d - d_n\|_\infty = \varepsilon_n \qquad \text{and} \qquad x(t) + \varepsilon_n d_n(t) = (\varepsilon_n \sqrt{t}, \varepsilon_n^2) \in Q(t) \qquad (t \in \Omega,\ n \in \mathbb{N}).$$

Now we show that, for all $t \in \Omega$,

$$V(x(t), d(t)|Q(t)) = \{(v_1, v_2) \in \mathbb{R}^2 \mid v_2 > 1\}.$$

Let $t \in \Omega$ be fixed. Then (v_1, v_2) belongs to $V(x(t), d(t)|Q(t))$ if and only if there exists $\overline{\varepsilon} > 0$ such that, for all $0 < \varepsilon < \overline{\varepsilon}$, $|(u_1, u_2)| < \overline{\varepsilon}$,

$$\Big(\varepsilon\sqrt{t} + \varepsilon^2(v_1 + u_1), \varepsilon^2(v_2 + u_2)\Big) \in Q(t),$$

that is,

$$0 \le v_2 + u_2 \qquad \text{and} \qquad 1 + \frac{\varepsilon(v_1 + u_1)}{\sqrt{t}} \le \sqrt{v_2 + u_2}. \tag{2.7}$$

Taking the limit $\varepsilon \to 0$, it follows that $v_2 + u_2 \ge 1$ if $|u_2| < \overline{\varepsilon}$. Hence $v_2 > 1$ is a necessary condition.

Conversely, if $v_2 > 1$ and $v_1 \in \mathbb{R}$, then there exists a constant $c > 0$ such that

$$1 \le v_2 - c, \qquad 1 + c(v_1 + c) \le \sqrt{v_2 - c}.$$

Then, for $|u_i| \le c$, we get

$$1 \le \sqrt{v_2 + u_2}, \qquad 1 + c(v_1 + u_1) \le \sqrt{v_2 + u_2}.$$

Multiplying the first inequality by $1 - \varepsilon/(c\sqrt{t})$, the second by $\varepsilon/(c\sqrt{t})$, and adding the two inequalities so obtained, we get that (2.7) holds for $0 < \varepsilon \le c\sqrt{t}$, $|u_1| \le c$, $|u_2| \le c$. Hence, $(v_1, v_2) \in V(x(t), d(t)|Q(t))$.

Thus, all the constant functions $v(t) = (v_1, v_2)$, where $v_2 > 1$, belong to $\sigma_\infty(\mathcal{V})$. Now, we prove that $v \notin \mathrm{cl}_\infty \mathbf{V}$ for $v_1 > 0$.

We argue by contradiction. Assume that $v_1 > 0$, $v_2 > 1$, and $v \equiv (v_1, v_2) \in \mathrm{cl}_\infty \mathbf{V}$. Then there exists a sequence $w_n \in \mathbf{V}$ with $\|w_n - v\|_\infty \to 0$. By Lemma 2.1, $w_n \in \mathbf{V}$ means that there exist $0 < \varepsilon_n \le 1$ and $A_n \subset \Omega$ of full measure such that, for all $0 < \varepsilon \le \varepsilon_n$, $u \in B_{\varepsilon_n} \subset \mathbb{R}^2$, and $t \in A_n$,

$$0 \le w_{n,2}(t) + u_2, \qquad 1 + \frac{\varepsilon}{\sqrt{t}}(w_{n,1}(t) + u_1) \le \sqrt{w_{n,2}(t) + u_2}.$$

Hence, by taking $u_1 = u_2 = 0$ and $\varepsilon = \varepsilon_n$, we have for all $n \in \mathbb{N}$ and $t \in A := \bigcap_{n=1}^{\infty} A_n$,

$$1 + \frac{\varepsilon_n}{\sqrt{t}} w_{n,1}(t) \leq \sqrt{w_{n,2}(t)}.$$

Thus, due to the L^∞-convergence of w_n to $v \equiv (v_1, v_2)$, we have for some $n_0 \in \mathbb{N}$ and $\bar{A} \subset \Omega$ of full measure that $w_{n,1}(t) > v_1/2$ and $w_{n,2}(t) < 2v_2$ if $t \in \bar{A}$ and $n \geq n_0$. Hence,

$$1 + \frac{\varepsilon_n v_1}{2\sqrt{t}} \leq \sqrt{2v_2}, \qquad \text{that is,} \qquad \varepsilon_n \leq 2\sqrt{t}\frac{\sqrt{2v_2} - 1}{v_1}$$

for $t \in A \cap \bar{A}$ and for $n \geq n_0$. Therefore, $\varepsilon_n = 0$ for $n \geq n_0$, contradicting $\varepsilon_n > 0$.

It is worth noting that in the above example the boundary of the set $Q(t)$ has at $(0, 0)$ a curvature of order $1/t$, and hence it is not bounded above on Ω.

An essential result follows from Theorem 2.2 that shows how to express the support function of $\mathbf{V}$ in terms of that of $\mathcal{V}(t)$.

COROLLARY 2.3. *Assume that $\mathbf{V} \neq \emptyset$. Then, for $\varphi \in L^1_m$,*

$$\delta^*(\varphi|\mathbf{V}) = \int_\Omega \delta^*(\varphi(t)|\mathcal{V}(t))\, d\mu(t). \tag{2.8}$$

Proof. Applying the previous theorem and [21, Lemma 4 and Theorem 6], which employ results from [25], [26] and [9], we have that

$$\begin{aligned}\delta^*(\varphi|\mathbf{V}) &= \delta^*(\varphi|\operatorname{cl}_1 \mathbf{V}) = \delta^*(\varphi|\sigma_\infty(\overline{\mathcal{V}})) \\ &= \int_\Omega \delta^*(\varphi(t)|\overline{\mathcal{V}}(t))\, d\mu(t) = \int_\Omega \delta^*(\varphi(t)|\mathcal{V}(t))\, d\mu(t). \qquad \square\end{aligned}$$

The above results bring up the questions of studying

(i) the characterization of the nonemptiness of $\mathbf{V}$, and

(ii) the calculation of the support function of the images $\mathcal{V}(t)$.

If the images of the set-valued map Q are convex, then $\sigma_\infty(Q)$ is also convex. In this case, Theorem 2.5 below offers an important characterization for the nonemptiness of $\mathbf{V}$. This result is based on the following characterization of the elements of $\mathbf{V}$ that is more useful in this case than that given by Lemma 2.1.

To state these results concisely, we introduce the following functions. For $t \in \Omega$ and $\xi \in \mathbb{R}^m$, denote

$$a(t, \xi) := \delta^*(\xi|Q(t)) - \langle \xi, x(t) \rangle, \qquad b(t, \xi) := \langle \xi, d(t) \rangle. \tag{2.9}$$

LEMMA 2.4. *Let $x \in \sigma_\infty(Q)$ and $d \in L^\infty_m$. $v \in \mathbf{V} = V(x, d|\sigma_\infty(Q))$ if and only if*

$$d(t) \in T(x(t)|Q(t)) \tag{2.10}$$

for a.e. $t \in \Omega$, and there exists $\bar{\varepsilon} > 0$ such that, for a.e. $t \in \Omega$,

$$\langle \xi, v(t) \rangle \leq \begin{cases} -\bar{\varepsilon}|\xi| - \dfrac{[b(t,\xi)]^2}{4a(t,\xi)} & \text{if} \quad \bar{\varepsilon} b(t, \xi) > 2a(t, \xi), \\ -\bar{\varepsilon}|\xi| + \dfrac{a(t,\xi) - \bar{\varepsilon} b(t,\xi)}{\bar{\varepsilon}^2} & \text{if} \quad \bar{\varepsilon} b(t, \xi) \leq 2a(t, \xi) \end{cases} \tag{2.11}$$

for all $\xi \in \mathbb{R}^m \setminus \{0\}$.

Proof. Let $v \in \mathbf{V}$. Then $\mathbf{V}$ is nonempty; hence $d \in T(x|\mathbf{Q})$. Then, it follows from [21, Theorem 4] that (2.10) is valid for a.e. $t \in \Omega$.

To prove (2.11), note that for almost all $t \in \Omega$, in the first domain, that is, in the set

$$\{\xi \in \mathbb{R}^m \mid \overline{\varepsilon} b(t,\xi) > 2a(t,\xi)\},$$

we have $a(t,\xi) > 0$. This implies that the function defined in the right-hand side of the above inequality is well defined. Indeed, by (2.10), the equality $a(t,\xi) = \delta^*(\xi|\sigma_\infty(Q)) - \langle \xi, x(t)\rangle = 0$ yields $b(t,\xi) = 0$, contradicting $\overline{\varepsilon} b(t,\xi) > 2a(t,\xi) = 0$.

By Lemma 2.1, there exist $\overline{\varepsilon} > 0$ and a set $A \in \mathcal{A}$ of full measure such that (2.2) holds for all $0 < \varepsilon \le \overline{\varepsilon}$, $u \in B_{\overline{\varepsilon}} \subset \mathbb{R}^m$, and $t \in A$. Therefore, for all $\xi \in \mathbb{R}^m$, we have

$$\langle \xi, x(t)\rangle + \varepsilon \langle \xi, d(t)\rangle + \varepsilon^2 \langle \xi, v(t) + u\rangle \le \delta^*(\xi|Q(t)), \tag{2.12}$$

that is,

$$\varepsilon b(t,\xi) + \varepsilon^2 \langle \xi, v(t) + u\rangle \le a(t,\xi).$$

Putting $u = \overline{\varepsilon}\xi/|\xi|$, we deduce

$$\varepsilon b(t,\xi) + \varepsilon^2(\langle \xi, v(t)\rangle + \overline{\varepsilon}|\xi|) \le a(t,\xi) \tag{2.13}$$

for all $0 < \varepsilon \le \overline{\varepsilon}$, $t \in A$, and $\xi \in \mathbb{R}^m \setminus \{0\}$. Hence,

$$\langle \xi, v(t)\rangle \le -\overline{\varepsilon}|\xi| + \inf_{0<\varepsilon\le\overline{\varepsilon}} \frac{a(t,\xi) - \varepsilon b(t,\xi)}{\varepsilon^2}. \tag{2.14}$$

Computing the infimum on the right-hand side, we get that (2.11) is valid for all $t \in A$ and $\xi \in \mathbb{R}^m \setminus \{0\}$.

Conversely, if $v \in L^\infty_m$, (2.10) is valid, and there exists $\overline{\varepsilon} > 0$ and a set $A \in \mathcal{A}$ of full measure such that v satisfies (2.11) for all $\xi \in (\mathbb{R}^m \setminus \{0\})$, then (2.14) and (2.13) are also valid. Thus (2.12) holds for all $u \in B_{\overline{\varepsilon}}$. The set $Q(t)$ being convex, this latter inequality implies (2.2). Hence, by Lemma 2.1 again, v belongs to $\mathbf{V}$.

Thus we have proved Lemma 2.4. □

Remark 2.2. An interesting consequence of Lemma 2.4 is that if (2.10) is valid and (2.11) also holds on the domain indicated, then $d \in T(x|\sigma_\infty(Q))$. The condition (2.10) alone is only a necessary condition for d to be in the tangent cone of $\sigma_\infty(Q)$ at x (see [21]). If the images of Q are not convex, then (2.10) and (2.11) are only necessary for v to be in $\mathbf{V}$.

THEOREM 2.5. *Let $Q : \Omega \to 2^{\mathbb{R}^m}$ be a measurable set-valued map whose images are closed convex sets and satisfy* (2.1). *Let $x \in \sigma_\infty(Q)$ and let $d \in L^\infty_m$. Then the set of second-order admissible variations $\mathbf{V} = V(x,d|\sigma_\infty(Q))$ is nonempty if and only if there exists a constant $M > 0$ such that, for a.e. $t \in \Omega$, the following condition is valid:*

$$[b(t,\xi)]^2 \le M|\xi|a(t,\xi) \qquad \textit{whenever} \quad \xi \in \mathbb{R}^m \ \textit{and} \ b(t,\xi) > 0 \tag{2.15}$$

(*where the functions a and b are defined in* (2.9)).

Remark 2.3. From Theorem 2.5 it readily follows that, for a.e. t, $\mathcal{V}(t) = V(x(t),d(t)|Q(t))$ is nonempty (that is, $d(t) \in C(x(t)|Q(t))$) if and only if (2.15)

holds for some $M_t > 0$ on the domain indicated. Therefore, Theorem 2.5 can be rephrased as

$$d \in C(x|\sigma_\infty(Q)) \Longleftrightarrow \left\{ \begin{array}{l} d(t) \in C(x(t)|Q(t)) \text{ for a.e. } t \in \Omega \text{ uniformly, i.e.,} \\ \exists M_t,\, \exists M > 0 : M_t \text{ satisfies (2.15)}, M_t \leq M \text{ for a.e. } t \in \Omega. \end{array} \right.$$

Proof. Assume that $v \in \mathbf{V}$. Then, by Lemma 2.4, there exist $\overline{\varepsilon} > 0$, a set $A \in \mathcal{A}$ of full measure such that (2.11) is true for all $t \in A$ and $\xi \in \mathbb{R}^m \setminus \{0\}$.

Let $t \in A$ and $\xi \in \mathbb{R}^m$ such that $b(t,\xi) > 0$. Then, as we have seen in the proof of Lemma 2.4, $a(t,\xi) > 0$. We distinguish two cases: If $\overline{\varepsilon} b(t,\xi) > 2a(t,\xi) > 0$, then, by the first inequality in (2.11), we have

$$\frac{[b(t,\xi)]^2}{a(t,\xi)} \leq -4\left\langle \xi, v(t)\right\rangle - 4\overline{\varepsilon}|\xi| \leq 4|\xi|(\|v\| - \overline{\varepsilon}).$$

In the other case, i.e., if $\overline{\varepsilon} b(t,\xi) \leq 2a(t,\xi)$ is valid, we have

$$\frac{[b(t,\xi)]^2}{a(t,\xi)} \leq \frac{2b(t,\xi)}{\overline{\varepsilon}} = \frac{2\left\langle \xi, d(t)\right\rangle}{\overline{\varepsilon}} \leq \frac{2|\xi|\|d\|}{\overline{\varepsilon}}.$$

Hence with the constant M defined as

$$M := \max\left(4(\|v\| - \overline{\varepsilon}), \frac{2\|d\|}{\overline{\varepsilon}}\right)$$

we get that (2.15) holds on the indicated domain.

Conversely, assume that (2.15) is valid. Then (2.10) holds, because, if $d(t) \notin T(x(t)|Q(t))$, then there exists $\xi \in N(x(t)|Q(t))$ such that $\left\langle \xi, d(t)\right\rangle > 0$. This yields that $a(t,\xi) = 0$ and, by (2.15), $b(t,\xi) = \left\langle \xi, d(t)\right\rangle = 0$, leading to a contradiction.

Now we show that there exist $w \in \operatorname{int}\sigma_\infty(Q)$ and $\overline{\varepsilon} > 0$ such that $v \in L^\infty_m$ defined by

$$v = \frac{w - x - \overline{\varepsilon} d}{\overline{\varepsilon}^2} \tag{2.16}$$

belongs to $\mathbf{V}$.

As we have noted above, by [21, Theorem 3], the condition in (2.1) yields that $\sigma_\infty(Q)$ has nonempty interior. Moreover the centers x_t in (2.1) can be chosen in a measurable way. Define the measurable function w by $w(t) = x_t$. Then, from (2.1) it results that there exists a set A of full measure such that, for all $t \in A$, we have that $w(t) + B_\rho \subset Q(t)$. Hence,

$$\left\langle \xi, w(t)\right\rangle + \rho|\xi| \leq \delta^*(\xi|Q(t)) \qquad \text{for all } t \in A, \xi \in \mathbb{R}^m.$$

Let M be the constant that validates (2.15). Choose $\overline{\varepsilon} > 0$ so that $\overline{\varepsilon}^2(\overline{\varepsilon} + M/4) \leq \rho$. Thus, we have

$$\left\langle \xi, w(t)\right\rangle + \overline{\varepsilon}^2(\overline{\varepsilon} + M/4)|\xi| \leq \delta^*(\xi|Q(t)) \qquad \text{for all } t \in A, \xi \in \mathbb{R}^m.$$

To complete the proof of the theorem, we need to show that the function v defined in (2.16) satisfies (2.11) with this $\overline{\varepsilon}$.

Substituting $v = (w - x - \overline{\varepsilon} d)/\overline{\varepsilon}^2$ into this condition, it remains to prove that

$$\left\langle \xi, w(t)\right\rangle \leq \left\{ \begin{array}{ll} -\overline{\varepsilon}^3|\xi| - \overline{\varepsilon}^2\dfrac{[b(t,\xi)]^2}{4a(t,\xi)} + \left\langle \xi, x(t)\right\rangle + \overline{\varepsilon} b(t,\xi) & \text{if } \overline{\varepsilon} b(t,\xi) > 2a(t,\xi), \\ -\overline{\varepsilon}^3|\xi| + \delta^*(\xi|Q(t)) & \text{if } \overline{\varepsilon} b(t,\xi) \leq 2a(t,\xi) \end{array} \right.$$

for all $t \in A$, $\xi \in (\mathbb{R}^m \setminus \{0\})$.

By the choice of $\overline{\varepsilon}$, we have that $\overline{\varepsilon}^3 \le \rho$, and hence

$$\langle \xi, w(t)\rangle + \overline{\varepsilon}^3|\xi| \le \delta^*(\xi|Q(t)) \qquad \text{for all } t \in A, \xi \in \mathbb{R}^m,$$

that is, the second inequality above holds.

It remains to show that the first inequality holds on its domain.

If (t,ξ) belongs to this domain, then

$$\overline{\varepsilon} b(t,\xi) > 2a(t,\xi) > a(t,\xi) > 0. \tag{2.17}$$

Thus, by our assumption, there exists a positive constant M such that (2.15) is valid. Combining these inequalities, we obtain

$$\begin{aligned}\langle \xi, w(t)\rangle &\le \delta^*(\xi|Q(t)) - \overline{\varepsilon}^2(\overline{\varepsilon} + M/4)|\xi| \\ &= a(t,\xi) + \langle \xi, x(t)\rangle - \overline{\varepsilon}^2(\overline{\varepsilon} + M/4)|\xi| \\ &< \overline{\varepsilon} b(t,\xi) + \langle \xi, x(t)\rangle - \overline{\varepsilon}^3|\xi| - \overline{\varepsilon}^2 M|\xi|/4 \\ &\le \overline{\varepsilon} b(t,\xi) + \langle \xi, x(t)\rangle - \overline{\varepsilon}^3|\xi| - \overline{\varepsilon}^2 \frac{[b(t,\xi)]^2}{4a(t,\xi)}\end{aligned}$$

for all (t,ξ) satisfying $\overline{\varepsilon} b(t,\xi) > 2a(t,\xi)$, that is, the needed first inequality above is proved, and hence v is in $\mathbf{V}$. Thus the nonemptiness of $\mathbf{V}$ is proved and the proof of Theorem 2.5 is complete. □

Remark 2.4. It follows from Theorem 2.5 that if there exists a constant M such that (2.15) is satisfied on the domain indicated, then $d \in T(x|\sigma_\infty(Q))$.

The rest of this section is devoted to answering the question pertaining the calculation of the support function of the images of $\mathcal{V}(t)$ in terms of $x(t)$, $d(t)$, and $Q(t)$. Thus, for fixed t, we need to calculate $\delta^*(\xi|V(x(t),d(t)|Q(t)))$. For this reason, we recall a special case of the result derived in [22] that describes the set $V(x,d|Q)$, for $x \in \mathbb{R}^m$, $d \in T(x|Q)$, and Q a convex set in $\mathbb{R}^m$ with nonempty interior, in terms of its support functional.

Denote

$$d^\perp := \{\xi \in \mathbb{R}^m \mid \langle \xi, d\rangle = 0\}, \qquad d^> := \{\xi \in \mathbb{R}^m \mid \langle \xi, d\rangle > 0\},$$

and define from $\mathbb{R}^m$ to the extended reals the function

$$\sigma(x,d|Q)(\xi) := \begin{cases} \liminf\limits_{\substack{\zeta \to \xi \\ \zeta \in d^>}} \dfrac{[\langle \zeta, d\rangle]^2}{4[\langle \zeta, x\rangle - \delta^*(\zeta|Q)]} & \text{if } \xi \in N(x|Q) \cap d^\perp, \\ +\infty & \text{otherwise.} \end{cases} \tag{2.18}$$

One can see that $\sigma(x,d|Q)(\cdot)$ is a positively homogeneous function and also lower semicontinuous on $\mathbb{R}^m \setminus \{0\}$.

Define the convex regularization $\overline{\text{co}}\,\sigma(x,d|Q)(\cdot)$ to be the largest lower semicontinuous convex function below $\sigma(x,d|Q)$, that is,

$$\begin{aligned}\overline{\text{co}}\,\sigma(x,d|Q)(\xi) = \sup\{\varphi(\xi) \mid \varphi : \mathbb{R}^m \to [-\infty,\infty] &\text{ is convex and lower semicontinuous,} \\ &\varphi(\zeta) \le \sigma(x,d|Q)(\zeta) \text{ for all } \zeta \in \mathbb{R}^m \setminus \{0\}\}.\end{aligned}$$

It results that $\overline{\text{co}}\,\sigma(x,d|Q)(\cdot)$ is also sublinear.

THEOREM 2.6. *Let $Q \subset \mathbb{R}^m$ be closed convex with nonempty interior, let $x \in Q$, $d \in C(x|Q)$. Then a vector $v \in \mathbb{R}^m$ belongs to $V(x,d|Q)$ if and only if*

$$\langle \xi, v\rangle < \overline{\mathrm{co}}\,\sigma(x,d|Q)(\xi) \qquad \text{for all } \xi \in \mathbb{R}^m \setminus \{0\}.$$

Furthermore, for all $\xi \in \mathbb{R}^m$,

$$\delta^*(\xi|V(x,d|Q)) = \overline{\mathrm{co}}\,\sigma(x,d|Q)(\xi).$$

The following result offers an evaluation of the support function of the set $\mathbf{V} = V(x,d|\sigma_\infty(Q))$ at linear functionals that can be represented in terms of integrable functions.

COROLLARY 2.7. *Let Q be a measurable set-valued map on Ω whose images are closed convex sets that satisfy condition* (2.1)*, let $x \in \sigma_\infty(Q)$ and $d \in C(x,d|\sigma_\infty(Q))$, and let $\varphi \in L^1(\Omega,\mathbb{R}^m)$. Then*

$$\delta^*(\varphi|\mathbf{V}) = \int_\Omega \overline{\mathrm{co}}\,\sigma\Big(x(t),d(t)|Q(t)\Big)(\varphi(t))\,d\mu(t). \tag{2.19}$$

3. Applications to optimization theory. In this section we make a specification of the optimization problem $(\mathcal{P})$ and Theorem 1.1. Let Y, Z be Banach spaces, $\mathcal{D} \subset Z$ nonempty and open, $F_i : \mathcal{D} \to \mathbb{R}$ $(i = 0,\dots,p)$, and $K : \mathcal{D} \to Y$ be given. Let $(\Omega,\mathcal{A},\mu)$ be a finite complete measure space, $g : \Omega \times \mathcal{D} \to \mathbb{R}^m$, $h : \Omega \times \mathcal{D} \to \mathbb{R}^n$, and $Q : \Omega \to 2^{\mathbb{R}^m}$ be a measurable set-valued map whose values are closed convex sets and the condition (2.1) is satisfied. Then, as stated in the previous section, $\sigma_\infty(Q) \subset L^\infty_m$ has nonempty interior.

We consider the following optimization problem:

$$(\mathcal{P}^*) \qquad \text{Minimize} \quad F_0(z) \quad \text{subject to} \quad \begin{cases} F(z) \le 0, \\ g(t,z) \in Q(t) \quad \text{for a.e. } t \in \Omega, \\ h(t,z) = 0 \quad \text{for a.e. } t \in \Omega, \\ K(z) = 0. \end{cases}$$

Introduce the functions $H : \mathcal{D} \to L^\infty_n$ and $G : \mathcal{D} \to L^\infty_m$ by

$$H(z)(t) = h(t,z) \quad \text{and} \quad G(z)(t) = g(t,z). \tag{3.1}$$

Then, with $E := (H,K)$ and $F := (F_1,\dots,F_p)$, the problem $(\mathcal{P}^*)$ reduces to $(\mathcal{P})$ described in the introduction.

The main focus of this section is to apply Theorem 1.1 to the problem $(\mathcal{P}^*)$ in such a way that all of the hypotheses assumed and all the results obtained will be phrased explicitly in terms of the data F_0, F, g, Q, h, and K.

Now we define the notions of a solution, admissible and regular points, and critical and regular directions.

- A point $\hat z \in \mathcal{D}$ is *admissible* for $(\mathcal{P}^*)$ if $F(\hat z) \le 0$, $g(t,\hat z) \in Q(t)$, $h(t,\hat z) = 0$ for a.e. $t \in \Omega$ and $K(\hat z) = 0$. A point $\hat z \in \mathcal{D}$ is a *solution (local minimum)* for this problem if there exists a neighborhood U of $\hat z$ such that $F_0(z) \ge F_0(\hat z)$ for all admissible point $z \in U$.
- The *regularity* of an admissible solution $\hat z$ means that the assumption (R_1) is valid and, in addition, we have (R_1^*)–(R_4^*) below.

  (R_1^*) The map $l(t,\cdot) := (g,h)(t,\cdot)$ is L^∞-uniformly strictly Fréchet differentiable at $\hat z$ for a.e. $t \in \Omega$, that is,

$$\lim_{z_1,z_2 \to \hat z} \frac{|l(t,z_1) - l(t,z_2) - l'(t,\hat z)(z_1 - z_2)|}{\|z_1 - z_2\|} = 0$$

holds L^∞-uniformly for $t \in \Omega$. (Then the maps G, H defined by (3.1) are strictly Fréchet differentiable at $\hat{z}$.)

(R_2^*) There exist a mapping $A : L_m^\infty \to Z$ and a bounded linear operator $B : L_n^\infty \to Z$ such that $A(0) = 0$,

$$H'(\hat{z})B = I, \qquad H'(\hat{z}) \circ A = 0,$$

and

$$G'(\hat{z})\Big(A(w)\Big) - w \in T\Big(G(\hat{z})|\sigma_\infty(Q)\Big) \qquad \text{for all} \quad w \in L_m^\infty.$$

Moreover, the operator $G'(\hat{z}) \circ B$ is a (Π, Π)-continuous map at zero from L_n^∞ to L_m^∞.

(R_3^*) F_0 and F are locally Lipschitz at $\hat{z}$ and the functions $F_i^\circ(\hat{z}, A(\cdot))$ and $F_i^\circ(\hat{z}, B(\cdot))$ are Π-continuous at zero for all $i = 0, \dots, p$.

(R_4^*) K is strictly Fréchet differentiable at $\hat{z}$, $K'(\hat{z}) \circ A$ and $K'(\hat{z}) \circ B$ are weakly Π-continuous, and $K'(\hat{z}) \circ (I_Z - B \circ H'(\hat{z})) : Z \to Y$ has a closed range (where I_Z is the identity on Z).

- A direction $y \in Z$ is *critical* for $(\mathcal{P}^*)$ at an admissible regular point $\hat{z}$ if (C_1) is valid, and

(C_2^*) $g'(t, \hat{z})(y) \in C(g(t, \hat{z})|Q(t))$ for a.e. $t \in \Omega$ uniformly, that is, there exists a constant M such that for a.e. $t \in \Omega$,

$$[\langle \xi, g'(t, \hat{z})(y)\rangle]^2 \le M|\xi|\Big(\delta^*(\xi|Q(t)) - \langle \xi, g(t, \hat{z})\rangle\Big)$$

for all $\xi \in \mathbb{R}^m$ such that $\langle \xi, g'(t, \hat{z})(y)\rangle > 0$.

(C_3^*) $h'(t, \hat{z})(y) = 0$ for a.e. $t \in \Omega$ and $K'(\hat{z})(y) = 0$.

- The vector y is a *regular* direction if (R_4) and the following hold:

(R_5^*) The following second-order directional derivative exists for the function $l := (g, h)$ for a.e. $t \in \Omega$,

$$l''(t, \hat{z}, y) := \lim_{\varepsilon \to 0+} 2\frac{l(t, \hat{z} + \varepsilon y) - l(t, \hat{z}) - \varepsilon l'(t, \hat{z})y}{\varepsilon^2},$$

and the limit is L^∞-uniform in t; furthermore, K satisfies the same assumption as the function E in (R5).

The main result of the section is the following theorem. Its proof employs the results derived in section 2 and Theorem 1.1.

THEOREM 3.1. *Let $\hat{z}$ be a regular solution of the above problem $(\mathcal{P}^*)$. Then, for all regular critical directions y, there correspond Lagrange multipliers $\lambda_0, \lambda_1, \dots, \lambda_p \ge 0$, $\varphi \in L_m^1$, $\psi \in L_n^1$, and $y^* \in Y^*$ (depending on y) that do not vanish simultaneously, and the following relations hold:*

$$\lambda_i F_i(\hat{z}) = 0 \ (i = 1, \dots, p), \qquad \lambda_i F_i^o(\hat{z}; y) = 0 \ (i = 0, \dots, p) \tag{3.2}$$

for a.e. $t \in \Omega$,

$$\varphi(t) \in N(g(t, \hat{z})|Q(t)), \qquad \langle \varphi(t), g'(t, \hat{z})(y)\rangle = 0, \tag{3.3}$$

$$\sum_{i=0}^{p} \lambda_i F_i^o(\hat{z}; z) + \langle y^*, K'(\hat{z})(z)\rangle + \int_\Omega \Big[\langle \varphi(t), g'(t, \hat{z})(z)\rangle + \langle \psi(t), h'(t, \hat{z})(z)\rangle\Big] d\mu(t) \ge 0 \qquad \text{for } z \in Z, \tag{3.4}$$

and

$$\sum_{i=0}^{p} \lambda_i F_i^{o\prime}(\hat{z}; y) + \langle y^*, K''(\hat{z})(y)\rangle + \int_\Omega \Big[\langle \varphi(t), g''(t,\hat{z})(y)\rangle + \langle \psi(t), h''(t,\hat{z})(y)\rangle\Big] d\mu(t), \geq 2\int_\Omega \gamma(t,\varphi(t))\, d\mu(t), \tag{3.5}$$

where $\gamma(t,\xi) := \overline{\mathrm{co}}\,\sigma\big(g(t,\hat{z}), g'(t,\hat{z})y | Q(t)\big)(\xi)$.

Observe that, using the Hahn–Banach theorem, the first-order condition (3.4) can be written as an equality: There exist linear functionals $z_i^* \in \partial F_i(\hat{z})$ $(i = 0, \ldots, p)$ such that

$$\sum_{i=0}^{p} \lambda_i \langle z_i^*, z\rangle + \langle y^*, K'(\hat{z})(z)\rangle + \int_\Omega \Big[\langle \varphi(t), g'(t,\hat{z})(z)\rangle + \langle \psi(t), h'(t,\hat{z})(z)\rangle\Big] d\mu(t) = 0 \qquad \text{for } z \in Z.$$

Proof. We intend to apply Theorem 1.1 to our problem $(\mathcal{P}^*)$. First we verify that all the hypotheses of Theorem 1.1 concerning $\hat{z}$ and the critical direction y are satisfied.

From (R_1^*), it follows that G and H are strictly Fréchet differentiable at $\hat{z}$. Thus, (R_1)–(R_3) will be satisfied if we show that, for $E := (H, K)$, the operator $E'(\hat{z})$ has a closed range in $L_n^\infty \times Y$.

From (R_2^*) we have the surjectivity of $H'(\hat{z})$. Then, by [1, Lemma 2.1.6], the result follows if we show that $K'(\hat{z})\big(\operatorname{Ker} H'(\hat{z})\big)$ is a closed subspace. By (R_2^*), B is the right inverse of $H'(\hat{z})$. Then the image of $I_Z - B \circ H'(\hat{z})$ is $\operatorname{Ker} H'(\hat{z})$ and hence (R_4^*) yields the closedness of the image of $E'(\hat{z})$.

The criticality condition (C_2) follows from (C_2^*) and Theorem 2.5.

Conditions (R_4) and (R_5) are immediate. Hence, Theorem 1.1 applied to $(\mathcal{P})^*$ yields the existence of nontrivial multipliers $\lambda_i \geq 0$ $(i = 0, \ldots, p)$, $w^* \in (L_m^\infty)^*$, $v^* \in (L_n^\infty)^*$, and $y^* \in Y^*$ such that the first equation of (3.2) holds and

$$w^* \in N\Big(G(\hat{z}) | \sigma_\infty(Q)\Big), \tag{3.6}$$

$$\sum_{i=0}^{p} \lambda_i F_i^o(\hat{z}; z) + \langle y^*, K'(\hat{z})z\rangle + \langle w^*, G'(\hat{z})z\rangle + \langle v^*, H'(\hat{z})z\rangle \geq 0 \qquad \text{for } z \in Z \tag{3.7}$$

and

$$\sum_{i=0}^{p} \lambda_i F_i^{o\prime}(\hat{z}; y) + \langle y^*, K''(\hat{z})(y)\rangle + \langle w^*, G''(\hat{z})(y)\rangle + \langle v^*, H''(\hat{z})(y)\rangle \geq 2\delta^*\Big(w^*\Big| V(G(\hat{z}), G'(\hat{z})y | \sigma_\infty(Q))\Big). \tag{3.8}$$

First we shall show that w^* and v^* are in fact represented in terms of integrable functions.

Let $(v, w) \in L_n^\infty \times L_m^\infty$. Set

$$z = z(v,w) := A(w - G'(\hat{z})Bv) + Bv,$$

where A is given in (R_2^*), which implies

$$G'(\hat{z})\Big(A(w - G'(\hat{z})Bv)\Big) - \Big(w - G'(\hat{z})Bv\Big) \in T\Big(G(\hat{z})|\sigma_\infty(Q)\Big).$$

Hence, by (3.6),

$$\Big\langle w^*, G'(\hat{z})\Big(A(w - G'(\hat{z})Bv)\Big) - \Big(w - G'(\hat{z})Bv\Big)\Big\rangle \le 0.$$

Due to this inequality and (R_2^*), we obtain

$$\begin{aligned}
&\langle w^*, G'(\hat{z})z\rangle + \langle v^*, H'(\hat{z})z\rangle \\
&\qquad = \Big\langle w^*, G'(\hat{z})\Big(A(w - G'(\hat{z})Bv)\Big) + \Big(G'(\hat{z})Bv - w\Big) + w\Big\rangle \\
&\qquad\qquad + \Big\langle v^*, H'(\hat{z})\Big(A(w - G'(\hat{z})Bv)\Big) + H'(\hat{z})(Bv)\Big\rangle \\
&\qquad \le \langle w^*, w\rangle + \langle v^*, v\rangle .
\end{aligned}$$

Substituting $z = z(w, 0)$ and $z = z(0, v)$ into (3.7), respectively, we get that

$$\sum_{i=0}^{p} \lambda_i F_i^o(\hat{z}; A(w)) + \langle y^*, K'(\hat{z})A(w)\rangle + \langle w^*, w\rangle \ge 0 \tag{3.9}$$

for $w \in L_m^\infty$, and

$$\sum_{i=0}^{p} \lambda_i F_i^o\Big(\hat{z}; A(-G'(\hat{z})Bv) + Bv\Big) + \Big\langle y^*, K'(\hat{z})\Big(A(-G'(\hat{z})Bv) + Bv\Big)\Big\rangle + \langle v^*, v\rangle \ge 0 \tag{3.10}$$

for $v \in L_n^\infty$.

Replacing w and v by $(-w)$ and $(-v)$, respectively, in the above inequalities, we also get lower estimates for the linear functionals w^* and v^*. Using the Π-continuity assumptions of (R_2^*), (R_3^*), and (R_4^*), we obtain that v^* and w^* are Π-continuous at the origin. Then, by the Hewitt–Yosida decomposition theorem [27], there exist $\varphi \in L_m^1$ and $\psi \in L_n^1$ such that

$$\langle w^*, w\rangle = \int_\Omega \langle \varphi(t), w(t)\rangle \, d\mu(t) \quad \text{and} \quad \langle v^*, v\rangle = \int_\Omega \langle \psi(t), v(t)\rangle \, d\mu(t) \tag{3.11}$$

for all $w \in L_m^\infty$ and $v \in L_n^\infty$, respectively. Clearly, these equations reduce (3.7) to (3.4).

Using Corollary 2.7 and (3.11), the second-order necessary condition (3.8) now reduces to (3.5).

Since by (3.11) the functional w^* is represented by the L_m^1-function φ, then [21, Theorem 9] and (3.6) yield that the first equation of (3.3) holds true. Furthermore, by replacing $z = y$ in (3.7) and by using the criticality of y, we obtain the second equations of (3.2) and (3.3). Therefore, the proof of the theorem is completed. ∎

Acknowledgment. The authors wish to thank the anonymous referee for the constructive comments and for calling to their attention extra references.

REFERENCES

[1] V. M. Alekseev, S. V. Fomin, and V. M. Tihomirov, *Optimal Control*, Nauka, Moscow, 1979 (in Russian).
[2] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Canadian Math. Soc. Ser. Monogr. Adv. Texts, John Wiley, New York, Chichester, Brisbane, Toronto, Singapore, 1983.
[3] C. Castaing and M. Valadier, *Convex-analysis and measurable multifunctions*, Lecture Notes in Math. 580, A. Dold and B. Eckmann, eds., Springer-Verlag, Berlin, Heidelberg, New York, 1977.
[4] R. Cominetti, *Metric regularity, tangent sets and second-order optimality conditions*, Appl. Math. Optim., 21 (1990), pp. 265–287.
[5] A. V. Dmitruk, *Quadratic conditions for a Pontryagin minimum in an optimal control problem, linear in the control, with a constraint on the control*, Soviet Math. Dokl., 28 (1983), pp. 364–368.
[6] A. V. Dmitruk, *Quadratic order conditions of a local minimum for singular extremals in a general optimal control problem*, in Differential Geometry and Control, Proc. Sympos. Pure Math. 64, G. Ferreyra et al., eds., AMS, Providence, RI, 1998, pp. 163–198.
[7] A. Ya. Dubovitskii and A. A. Milyutin, *Extremum problems with constraints*, Dokl. Akad. Nauk SSSR, 149 (1963), pp. 759–762 (in Russian). English version available in Soviet Math. Dokl., 4 (1963), pp. 452–455.
[8] A. Ya. Dubovitskii and A. A. Milyutin, *Extremal problems in the presence of constraints*, USSR Comput. Math. and Math. Physics, 5 (1965), pp. 395–453.
[9] F. Hiai and H. Umegaki, *Integrals, conditional expectations, and martingales of multivalued functions*, J. Multivariate Anal., 7 (1977), pp. 149–182.
[10] A. D. Ioffe, *On some recent developments in the theory of second-order optimality conditions*, in Optimization, Lecture Notes in Math. 1405, S. Dolecki, ed., Springer-Verlag, New York, Berlin, 1989, pp. 55–68.
[11] H. Kawasaki, *An envelope-like effect of infinitely many inequality constraints on second-order necessary conditions for minimization problems*, Math. Programming, 41 (1988), pp. 73–96.
[12] H. Kawasaki, *Second order necessary optimality for minimizing a sup-type function*, Math. Programming, 49 (1991), pp, 213–229.
[13] E. S. Levitin, A. A. Milyutin, and N. P. Osmolovskii, *Conditions of higher-order for a local minimum in extremal problems with constraints*, Russian Math. Surveys, 33 (1978), pp. 97–168.
[14] Y. Maruyama, *Second-order necessary conditions for nonlinear optimization problems in Banach spaces by the use of Neustadt derivative*, Math. Japon., 40 (1994), pp. 509–522.
[15] A. A. Milyutin and N. P. Osmolovskii, *Calculus of Variations and Optimal Control*, AMS, Providence, RI, 1998.
[16] N. P. Osmolovskii, *Necessary and sufficient conditions of a high order for a Pontryagin and a bounded-strong minima in an optimal control problem*, Soviet Phys. Dokl., 33 (1988), pp. 883–885.
[17] N. P. Osmolovskii, *Quadratic conditions for nonsingular extremals in optimal control (a theory)*, Russian J. Math. Phys., 2 (1994), pp. 487–516.
[18] Zs. Páles and V. M. Zeidan, *Nonsmooth optimum problems with constraints*, SIAM J. Control Optim., 32 (1994), pp. 1476–1502.
[19] Zs. Páles and V. M. Zeidan, *Characterization of closed C-convex sets in $C(T, \mathbb{R}^n)$*, Acta Sci. Math. (Szeged), 65 (1999), pp. 339–357.
[20] Zs. Páles and V. M. Zeidan, *On L^1-closed decomposable sets in L^∞*, in Systems Modelling and Optimization (Detroit, MI, 1997), Chapman & Hall/CRC Res. Notes Math. 396, Boca Raton, FL, 1999, pp. 198–206,
[21] Zs. Páles and V. M. Zeidan, *Characterization of L^1-closed decomposable sets in L^∞*, J. Math. Anal. Appl., 238 (1999), pp. 491–515.
[22] Zs. Páles and V. Zeidan, *Optimum problems with certain lower semicontinuous set-valued constraints*, SIAM J. Control Optim., 8 (1998), pp. 707–727.
[23] J.-P. Penot, *Optimality conditions in mathematical programming and composite optimization*, Math. Programming, 67 (1994), pp. 225–245.
[24] R. T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1970.

[25] R. T. Rockafellar, *Integrals which are convex functionals*, Pacific J. Math., 24 (1968), pp. 525–539.
[26] R. T. Rockafellar, *Integrals which are convex functionals,* II, Pacific J. Math., 39 (1971), pp. 439–469.
[27] K. Yosida and E. Hewitt, *Finitely additive measures*, Trans. Amer. Math. Soc., 72 (1952), pp. 46–66.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 444–463

ON THE SQAP-POLYTOPE*

MICHAEL JÜNGER† AND VOLKER KAIBEL‡

Abstract. The SQAP-polytope was associated to quadratic assignment problems with a certain symmetric objective function structure by Rijal (1995) and Padberg and Rijal (1996). We derive a technique for investigating the SQAP-polytope that is based on projecting the (low-dimensional) polytope into a lower dimensional vector-space, where the vertices have a "more convenient" coordinate structure. We exploit this technique in order to prove conjectures by Padberg and Rijal on the dimension of the SQAP-polytope as well as on its trivial facets.

Key words. quadratic assignment problem, symmetric model, polyhedral combinatorics

AMS subject classifications. 90C09, 90C10, 90C27

PII. S1052623496310576

1. Introduction.

For many classical $\mathcal{NP}$-hard combinatorial optimization problems like, e.g., the *traveling salesman problem (TSP)*, the *max cut problem*, or the *stable set problem*, the methods of polyhedral combinatorics have yielded a lot of structural insight that led to big improvements in practical problem solving via cutting-plane-based methods like branch&cut. However, the *quadratic assignment problem (QAP)*—where the task is to find a permutation π that minimizes $\sum_i \sum_k a_{ik} b_{\pi(i)\pi(k)} + \sum_i c_{i\pi(i)}$ for some matrices $A = (a_{ik})$, $B = (b_{jl})$, and $C = (c_{ij})$—was merely considered from a polyhedral point of view until the work of [24, 21] and [14] (which is a preliminary version of [16]). These papers defined the QAP-polytope via a well-known mixed integer programming (MIP) formulation of the QAP and proved some basic properties of that polytope, in particular its dimension (which was also proved in [5]).

There might be two reasons why the QAP-polytope had not been considered before. One is the fact that this polytope looks in some sense "nasty," which can be overcome by mapping it in a certain way into a different space (cf. [16]). The other reason is computational. The MIP-formulation on which the QAP-polytope is based has a lot of variables such that (at least) in former times it might have seemed impractical to solve the arising linear programs (LPs), for instance, within a branch&cut algorithm. However, the LP-solvers have improved a lot during the last few years, especially due to the success of interior point methods. Now, it seems promising to attack QAP-instances of size about 20 or 25 (and maybe even larger) by cutting-plane-based algorithms that use structural insight into the QAP-polytope. When considering these orders of magnitudes, one has to note that existing branch&bound algorithms (mostly using the *Gilmore–Lawler bound*) need a large amount of (parallel) computer power to solve instances of size about 20, since they produce branch&bound trees with a lot of nodes (cf. [8]). In the meantime during the submission of this paper and the preparation of the final version, powerful branch&bound codes have been developed that rely on more elaborate lower bounding procedures [12, 6]. Nevertheless, these algorithms also produce large branch&bound trees. Due to this fact, it

*Received by the editors October 9, 1996; accepted for publication (in revised form) May 22, 2000; published electronically October 18, 2000. This work was partially supported by DFG Ju 204/4-2 and EU ESPRIT Long Term Research Project 20244 (ALCOM-IT).
http://www.siam.org/journals/siopt/11-2/31057.html

†Institut für Informatik, Universität zu Köln, Pohligstr. 1, 50969 Köln, Germany (mjuenger@informatik.uni-koeln.de).

‡MA 7-1, TU Berlin, Straße des 17. Juni 136, 10623 Berlin, Germany (kaibel@math.TU-Berlin.de).

sounds attractive to try to reduce this "tendency to implicit enumeration" by exploiting structural information about the problem that results from the polyhedral investigations.

Actually, the kind of QAP we defined above is a so-called *Koopmans–Beckmann problem (KB-QAP)*. It was introduced in [19] in order to model the situation of a set of n facilities that have certain amounts of "flow" between them and a set of n locations having certain distances, and the requirement is to assign the facilities to the locations in such a way that the sum of the products of flows and the respective distances is minimized. The c_{ij} model fixed costs that arise when placing facility i to location j independently from the assignment of the other facilities. One calls matrix A the *flow matrix*, matrix B the *distance matrix*, and matrix C the matrix of the *linear costs*. Clearly, this problem is $\mathcal{NP}$-hard, since it has many $\mathcal{NP}$-hard optimization problems as special cases, e.g., the TSP.

We call *symmetric* such instances with the property that assigning object i to location j and object k to location l always causes the same costs as assigning i to l and k to j. For example, all instances having a symmetric distance or flow matrix are symmetric in that sense. It turns out (first observed by [24, 21]) that for such symmetric instances one can drop nearly 50% of the variables in the MIP-formulation underlying the polyhedral approach. (Doing this, the quality of the LP-relaxation decreases slightly, as we will show in section 6.) This yields a different polytope, the symmetric QAP-polytope (SQAP-polytope). In [24] and [21] a set of valid equations for that polytope is derived and the dimension of the SQAP-polytope is conjectured.

In this paper, we present some basic properties of the SQAP-polytope including a proof of that conjecture. The main tool we use is a transformation that is similar to the one that allowed us to derive basic results about the QAP-polytope in a (relatively) simple way [16]. In section 2 we explain a formulation of the QAP as a minimization problem in a certain graph. Using that terminology, we give the MIP-formulations for QAP and SQAP that underlie the polyhedral approaches. In section 3 we give definitions of both the QAP- and SQAP-polytopes and describe connections between them. Then we map these polytopes isomorphically to other spaces, where they "look much nicer." (When saying a certain polytope P is *isomorphic* to a polytope P', we always mean that there is an affine transformation from $\operatorname{aff}(P)$ to $\operatorname{aff}(P')$ mapping P to P'. In particular, this implies that the two polytopes are combinatorially isomorphic, i.e., they have isomorphic face lattices.) In section 4 the dimension of the SQAP-polytope as well as the fact that the nonnegativity constraints on the variables define facets of it are established. In section 5 we present a first class of nontrivial facets of the SQAP-polytope. Some computational results concerning a lower bound obtained by exploiting these first results about the SQAP-polytope are reported in section 6. Finally we give some conclusions in section 7.

2. Problem definition. We will define the QAP as the problem of finding among certain cliques in a special graph one of minimal node and edge weight. The SQAP will be defined as a similar problem in a closely related hypergraph. We use the symbol $\binom{M}{k}$ for the set of all subsets of cardinality k of a set M.

Let the graph $\mathcal{G}_n = (\mathcal{V}_n, \mathcal{E}_n)$ have nodes

$$\mathcal{V}_n := \big\{(i,j) \;\big|\; i,j \in \{1,\dots,n\}\big\}$$

and edges

$$\mathcal{E}_n := \Big\{\{(i,j),(k,l)\} \in \tbinom{\mathcal{V}_n}{2} \;\Big|\; i \neq k, j \neq l\Big\}.$$

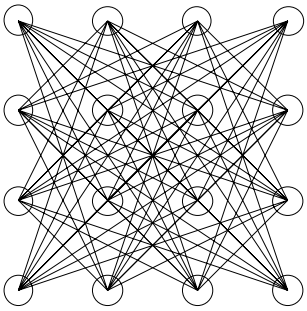


FIG. 2.1. *The graph $\mathcal{G}_n$ has all possible edges except the "horizontal" and the "vertical" ones.*

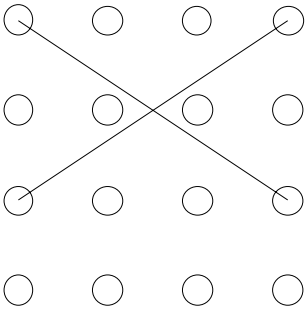


FIG. 2.2. *A pair of edges that can be identified in the symmetric case.*

We define $[i,j,k,l] := \{(i,j),(k,l)\}$ for all edges $\{(i,j),(k,l)\} \in \mathcal{E}_n$. This implies $[i,j,k,l] = [k,l,i,j]$. We usually draw $\mathcal{G}_n$ as shown in Figure 2.1.

The graph $\mathcal{G}_n$ has clique-number $\omega(\mathcal{G}_n) = n$, and the n-cliques of $\mathcal{G}_n$ correspond to the $(n \times n)$-permutation matrices. We denote the set of (node sets of) k-cliques of $\mathcal{G}_n$ by

$$\mathcal{CLQ}_k^n := \{C \subseteq \mathcal{V}_n \mid C \text{ } k\text{-clique of } \mathcal{G}_n\}.$$

For any $S \subseteq \mathcal{V}_n$, we denote by $\mathcal{E}_n(S) := \{\{v,w\} \in \mathcal{E}_n \mid v,w \in S\}$ the set of edges having both endpoints in S. As usual, for a subset $N \subseteq M$ of a finite set M and a vector $a \in \mathbb{R}^M$, we define $a(N) := \sum_{e \in N} a_e$.

The QAP is to solve

$$(\text{QAP}_{g,h}) \qquad \begin{array}{rl} \min & g(C) + h(\mathcal{E}_n(C)) \\ \text{subject to} & C \in \mathcal{CLQ}_n^n \end{array}$$

for given node weights $g \in \mathbb{R}^{\mathcal{V}_n}$ and edge weights $h \in \mathbb{R}^{\mathcal{E}_n}$. (If we have a KB-QAP defined by the matrices $A = (a_{ik})$, $B = (b_{jl})$, and $C = (c_{ij})$, we choose $g_{(i,j)} = c_{ij} + a_{ii}b_{jj}$ and $h_{[i,j,k,l]} = a_{ik}b_{jl} + a_{ki}b_{lj}$.)

The nodes and edges of $\mathcal{G}_n$ will correspond to variables in the polyhedral approach. If the instance (g,h) is *symmetric* in the sense that $h_{[i,j,k,l]} = h_{[i,l,k,j]}$ for all pairs of edges $\{[i,j,k,l],[i,l,k,j]\}$ (cf. Figure 2.2), then we can identify these two edges in our formulation, and hence reduce the number of variables by nearly 50%.

This observation (first made by [24, 21]) gives the motivation to study also a specific formulation for the special case of symmetric instances of the QAP, the *SQAP*.

In order to derive an appropriate formulation for SQAP, we model the described identification of edges by passing from the graph $\mathcal{G}_n$ having nodes $\mathcal{V}_n$ and edges $\mathcal{E}_n$ to the hypergraph $\mathcal{H}_n$ having the same nodes $\mathcal{V}_n$, but hyperedges

$$\mathcal{F}_n := \big\{\{(i,j),(k,l),(i,l),(k,j)\} \in \tbinom{\mathcal{V}_n}{4} \big| i \neq k, j \neq l\big\}.$$

There will be no hypergraph theory involved; we simply use the notions of "hypergraph" and "hyperedges." For $i \neq k$ and $j \neq l$, we write $\langle i,j,k,l\rangle := \{(i,j),(k,l),(i,l),(k,j)\}$. This implies $\langle i,j,k,l\rangle = \langle k,l,i,j\rangle = \langle i,l,k,j\rangle = \langle k,j,i,l\rangle$ for all $i \neq k$ and $j \neq l$. For an edge $[i,j,k,l] \in \mathcal{E}_n$ we call the edge $\tau([i,j,k,l]) := [i,l,k,j]$ the *mate* of $[i,j,k,l]$. Then we can assign to every edge $e \in \mathcal{E}_n$ a hyperedge $\mathrm{HYP}(e) := e \cup \tau(e) \in \mathcal{F}_n$. For a subset $R \subseteq \mathcal{E}_n$, we denote $\mathrm{HYP}(R) := \{\mathrm{HYP}(e) \mid e \in R\}$. For a subset $S \subseteq \mathcal{V}_n$, we define the set $\mathcal{F}_n(S) := \mathrm{HYP}(\mathcal{E}_n(S))$. We refer to a subset $C \subset \mathcal{V}_n$ as a *clique of* $\mathcal{H}_n$ if and only if C is a clique of the graph $\mathcal{G}_n$.

Because we need to express relationships between the asymmetric and the symmetric versions of the problem, we introduce the map

$$\mathrm{sym}_n \ : \ \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n} \longrightarrow \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$$

by defining $\mathrm{sym}_n(x,y) = (x,z)$ via $z_{e\cup\tau(e)} := y_e + y_{\tau(e)}$ for all $e \in \mathcal{E}_n$.

If $(g,h) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ and h is symmetric, then $(\mathrm{QAP}_{g,h})$ is equivalent to solving SQAP

$$(\mathrm{SQAP}_{g,\widehat{h}}) \qquad \begin{array}{rl} \min & g(C) + \widehat{h}(\mathcal{F}_n(C)) \\ \text{subject to} & C \in \mathcal{CLQ}_n^n \end{array}$$

with $\widehat{h}_{\mathrm{HYP}(e)} := h_e$ for all $e \in \mathcal{E}_n$.

In the rest of this section, we will develop MIP-formulations for QAP and SQAP. These formulations are the starting points for the polyhedral approach. The MIP-formulation for QAP was introduced by [13] and [1] (using a general linearization technique due to [2]). It is similar to a formulation by [9], which, however, was demonstrated by [13] and [1] to give a weaker LP-relaxation. The one for SQAP is due to [24] and [21]. Nevertheless, we will give short proofs of the respective theorems in our notational setting.

We need the notion of a *characteristic vector* $\chi^N \in \{0,1\}^M$ for a subset $N \subseteq M$ of a (finite) set M, defined by setting $\chi^N_p := 1$ for $p \in M$ if and only if $p \in N$. We will denote characteristic vectors of subsets of

$$\begin{array}{lll} \mathcal{V}_n & \text{by} & x^{(\cdots)}, \\ \mathcal{E}_n & \text{by} & y^{(\cdots)}, \text{ and} \\ \mathcal{F}_n & \text{by} & z^{(\cdots)}. \end{array}$$

Define $\mathrm{VERT}_n := \{(x^C, y^{\mathcal{E}_n(C)} \mid C \in \mathcal{CLQ}_n^n\}$ and $\mathrm{SVERT}_n := \{(x^C, z^{\mathcal{F}_n(C)} \mid C \in \mathcal{CLQ}_n^n\}$, i.e., VERT_n and SVERT_n are the characteristic vectors of feasible solutions to QAP and SQAP, respectively.

We denote by $\mathrm{row}_i^{(n)} := \{(i,j) \in \mathcal{V}_n \mid j = 1,\ldots,n\}$ the ith row and by $\mathrm{col}_j^{(n)} := \{(i,j) \in \mathcal{V}_n \mid i = 1,\ldots,n\}$ the jth column of the nodes $\mathcal{V}_n$. The next two theorems provide the desired MIP-formulations for QAP and SQAP, respectively. As usual, for any two disjoint subsets $S,T \subseteq \mathcal{V}_n$, $(S:T)$ is the set of all edges in $\mathcal{E}_n$ having one endpoint in S and the other one in T. For a singleton $\{v\}$, in this as well as in some other contexts, we often omit the brackets and simply write v.

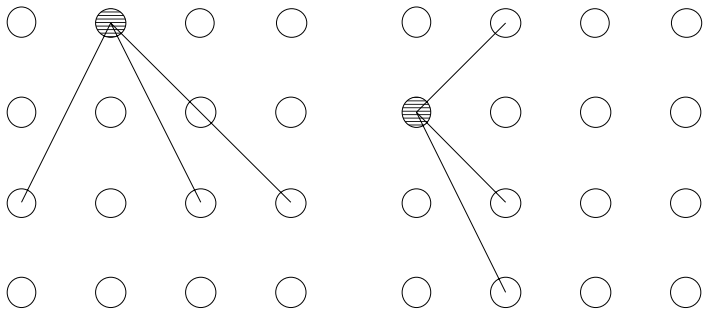


FIG. 2.3. *Equations* (2.3) *and* (2.4).

Figures 2.3 and 2.4 illustrate the used equations. We draw a hyperedge from $\mathcal{F}_n$ simply by drawing both mates from $\mathcal{E}_n$ belonging to that hyperedge. In all our figures, dashed nodes or (hyper)edges indicate coefficients -1, solid ones stand for $+1$.

THEOREM 2.1. *A vector* $(x, y) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ *is a member of* VERT_n *if and only if it satisfies the following conditions:*

$$x(\mathrm{row}_i^{(n)}) = 1 \qquad (i = 1, \ldots, n), \tag{2.1}$$

$$x(\mathrm{col}_j^{(n)}) = 1 \qquad (j = 1, \ldots, n), \tag{2.2}$$

$$-x_{(i,j)} + y((i,j) : \mathrm{row}_k^{(n)}) = 0 \qquad (i, j, k = 1, \ldots, n,\ i \neq k), \tag{2.3}$$

$$-x_{(i,j)} + y((i,j) : \mathrm{col}_l^{(n)}) = 0 \qquad (i, j, l = 1, \ldots, n,\ j \neq l), \tag{2.4}$$

$$y_e \geq 0 \qquad (e \in \mathcal{E}_n), \tag{2.5}$$

$$x_v \in \{0, 1\} \qquad (v \in \mathcal{V}_n). \tag{2.6}$$

We make one more notational convention in order to increase the readability of the following equations. For any pair $v, w \in \mathcal{V}_n$ of nodes belonging to the same row or column of $\mathcal{V}_n$, we denote by $\Delta_v^w := \{f \in \mathcal{F}_n \mid v, w \in f\}$ the set of all hyperedges in $\mathcal{F}_n$ containing both v and w (cf. Figure 2.4).

THEOREM 2.2. *A vector* $(x, z) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ *is a member of* SVERT_n *if and only if it satisfies the following conditions:*

$$x(\mathrm{row}_i^{(n)}) = 1 \qquad (i = 1, \ldots, n), \tag{2.7}$$

$$x(\mathrm{col}_j^{(n)}) = 1 \qquad (j = 1, \ldots, n), \tag{2.8}$$

$$-x_{(i,j)} - x_{(k,j)} + z(\Delta_{(i,j)}^{(k,j)}), = 0 \qquad (i, j, k = 1, \ldots, n, i < k), \tag{2.9}$$

$$-x_{(i,j)} - x_{(i,l)} + z(\Delta_{(i,j)}^{(i,l)}), = 0 \qquad (i, j, l = 1, \ldots, n, j < l), \tag{2.10}$$

$$z_e \geq 0 \qquad (e \in \mathcal{F}_n), \tag{2.11}$$

$$x_v \in \{0, 1\} \qquad (v \in \mathcal{V}_n). \tag{2.12}$$

Proof of Theorem 2.1. The "only if" part is clear. To see the other direction, let $(x, y) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ satisfy conditions (2.1)–(2.6). Obviously, x is the characteristic vector of an n-clique of $\mathcal{G}_n$, and one deduces (e.g., using two equations from (2.3) and the nonnegativity of y) that $y_{[i,j,k,l]} > 0$ implies $x_{(i,j)} = x_{(k,l)} = 1$. These two facts imply that it is impossible for two components of y belonging to mates to be both nonzero. Observing that $\mathrm{sym}_n(x, y)$ satisfies the conditions of Theorem 2.2, one obtains Theorem 2.1 from Theorem 2.2. ∎

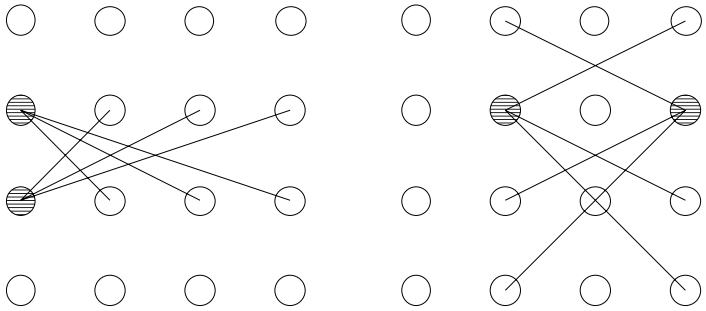


FIG. 2.4. *Equations* (2.9) *and* (2.10).

Proof of Theorem 2.2. Again, the "only if" part is obvious. Let $(x, z) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ satisfy conditions (2.7)–(2.12); hence x is the characteristic vector of an n-clique $C \in \mathcal{CLQ}_n^n$. Considering four appropriate equations from (2.9) and (2.10) (and noting the nonnegativity of z), one gets that $z_{\langle i,j,k,l\rangle} > 0$ implies $x_{(i,j)} = x_{(k,l)} = 1$ or $x_{(i,l)} = x_{(k,j)} = 1$. But then, in each of (2.9) and (2.10), there is at most one hyperedge involved corresponding to a nonzero component of z. This leads to the fact that $z_{\langle i,j,k,l\rangle} > 0$ implies $z_{\langle i,j,k,l\rangle} = 1$, and that $x_{(i,j)} = x_{(k,l)} = 1$ implies $z_{\langle i,j,k,l\rangle} = 1$. Hence, z must be the characteristic vector of $\mathcal{F}_n(C)$. □

3. The SQAP-polytope and some relatives. Theorems 2.1 and 2.2 give us the starting points for deriving and exploiting further structural information on the problems QAP and SQAP. As with many other combinatorial optimization problems, the hope is to achieve this by investigating the convex hulls of the sets of feasible solutions to the respective MIPs.

We shall define the *quadratic assignment polytope* as

$$\mathcal{QAP}_n := \operatorname{conv}\big(\{(x^C, y^{\mathcal{E}_n(C)}) \mid C \in \mathcal{CLQ}_n^n\}\big)$$

and the *symmetric quadratic assignment polytope* as

$$\mathcal{SQAP}_n := \operatorname{conv}\big(\{(x^C, z^{\mathcal{F}_n(C)}) \mid C \in \mathcal{CLQ}_n^n\}\big).$$

Before starting to consider the connection between these two polytopes, we want to mention the following facts.

OBSERVATION 1. *The two polytopes $\mathcal{QAP}_n$ and $\mathcal{SQAP}_n$ are invariant under permutations of the rows, permutations of the columns, and "transposition" of the node set $\mathcal{V}_n$. In particular, for each of the two polytopes, all the cones induced at the vertices are isomorphic.*

For the first one, the QAP-polytope, investigations were started by [24, 21, 14, 16]. There is not much known about the second one, the SQAP-polytope. Basically, there is only a conjecture of [24] and [21] concerning the dimension of $\mathcal{SQAP}_n$, which we will prove to be valid in Theorem 4.2.

This paper is concerned with the SQAP-polytope. However, it turns out that $\mathcal{SQAP}_n$ and $\mathcal{QAP}_n$ are closely related—although they are not isomorphic (e.g., we will see that they have different dimensions). The situation is quite similar to the relationship between the *asymmetric* and the *symmetric traveling salesman polytope*. While it is difficult to carry over results from the symmetric to the asymmetric case, this is (sometimes) possible for the opposite direction.

Next, we want to explain the relationship between the QAP- and the SQAP-polytope. Formally, the two polytopes are connected by

$$\mathcal{SQAP}_n = \mathrm{sym}_n(\mathcal{QAP}_n).$$

(Just consider the vertices to see this.)

We define an inequality (equation) $(u,v)^T(x,y) \leq (=)\omega$ with $(u,v) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ and $\omega \in \mathbb{R}$ to be *symmetric* if and only if components of v that belong to mates are equal, i.e., $v_e = v_{\tau(e)}$ for all $e \in \mathcal{E}_n$. A face of $\mathcal{QAP}_n$ is called *symmetric* if there is a symmetric inequality defining that face. Even if a face of $\mathcal{QAP}_n$ is defined by a nonsymmetric inequality, it may be symmetric. This is because in general a face is defined by many different inequalities (even in the case of a facet, due to the low-dimensionality of $\mathcal{QAP}_n$), but in order to be symmetric it is required that there exists only one among these inequalities which is symmetric.

Let $(u,v)^T(x,y) \leq (=)\omega$ be a symmetric valid inequality (equation) for the polytope $\mathcal{QAP}_n$. It induces a valid inequality (equation) $(u,w)^T(x,z) \leq (=)\omega$ for $\mathcal{SQAP}_n$ with $w_{\mathrm{HYP}(e)} := v_e$ for all $e \in \mathcal{E}_n$. Conversely, every valid inequality (equation) for $\mathcal{SQAP}_n$ induces a symmetric valid inequality (equation) for $\mathcal{QAP}_n$. From this, we obtain the following.

Observation 2. *There is a one-to-one correspondence between the symmetric faces of $\mathcal{QAP}_n$ and the faces of $\mathcal{SQAP}_n$. If we identify the faces of $\mathcal{QAP}_n$ and $\mathcal{SQAP}_n$ with the node sets of the cliques corresponding to their vertices, then that correspondence is inclusion-preserving.*

This observation translates into the relationship between the face lattices of the QAP- and the SQAP-polytopes.

Theorem 3.1. *The face lattice of $\mathcal{SQAP}_n$ arises by restricting the face lattice of $\mathcal{QAP}_n$ to the symmetric faces. (Note that $\emptyset$ and $\mathcal{QAP}_n$ itself are symmetric faces of $\mathcal{QAP}_n$.)*

Corollary 3.2. *A symmetric proper face of $\mathcal{QAP}_n$ induces a facet of $\mathcal{SQAP}_n$ if and only if there are only nonsymmetric faces strictly between itself and $\mathcal{QAP}_n$ in the face lattice of $\mathcal{QAP}_n$.*

In general, it will be difficult to show that strictly between a certain symmetric face and the whole polytope there are only nonsymmetric faces of $\mathcal{QAP}_n$, because it is hard to prove that a set of faces is the complete set of faces containing a given face. However, in the special case that the face under consideration is a *ridge* of $\mathcal{QAP}_n$ (i.e., a face of two dimensions less than the whole polytope), the chances are better since it is a well-known fact that any ridge is the unique intersection of two facets.

Corollary 3.3. *If a symmetric ridge of $\mathcal{QAP}_n$ is the intersection of two nonsymmetric facets of $\mathcal{QAP}_n$, then it induces a facet of $\mathcal{SQAP}_n$.*

When investigating more closely the structure of a polytope defined as the convex hull of some points, one is very soon confronted with tasks such as computing the rank of a subset of these points or showing that such a subset spans a certain subspace. In both cases, one has to deal with linear combinations of the points, which one hopes to be sparse and to look somehow nice. Working with $\mathcal{QAP}_n$ and $\mathcal{SQAP}_n$, it turns out that such nice combinations are hard to obtain. This is mainly due to the facts that the coordinate vectors of the vertices look all the same up to certain permutations of the coordinates, and that there are no pairs among them having only slightly differing supports. On the other hand, for both of the polytopes a lot of equations are holding, indicating some redundancy in the problem definition. This motivated us to try to map the polytopes isomorphically into other spaces (of lower dimensions) in such a way that the coordinate vectors of the resulting vertices have nicer structures.

Let $\mathcal{A} \subset \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ be the affine subspace of $\mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ defined by (2.1)–(2.4), i.e., $\mathcal{A} \subseteq \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n}$ is the set of solutions to the equation system

$$\begin{aligned} x(\mathrm{row}_i^{(n)}) &= 1 \quad (i = 1, \ldots, n), \\ x(\mathrm{col}_j^{(n)}) &= 1 \quad (j = 1, \ldots, n), \\ -x_{(i,j)} + y((i,j) : \mathrm{row}_k^{(n)}) &= 0 \quad (i, j, k = 1, \ldots, n, i \neq k), \\ -x_{(i,j)} + y((i,j) : \mathrm{col}_l^{(n)}) &= 0 \quad (i, j, l = 1, \ldots, n, j \neq l), \end{aligned}$$

and let $\widehat{\mathcal{A}} \subset \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ be the affine subspace of $\mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ defined by (2.7)–(2.10), i.e., $\widehat{\mathcal{A}} \subseteq \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ is the set of solutions to the system

$$\begin{aligned} x(\mathrm{row}_i^{(n)}) &= 1 \quad (i = 1, \ldots, n), \\ x(\mathrm{col}_j^{(n)}) &= 1 \quad (j = 1, \ldots, n), \\ -x_{(i,j)} - x_{(k,j)} + z(\Delta_{(i,j)}^{(k,j)}) &= 0 \quad (i, j, k = 1, \ldots, n, i < k), \\ -x_{(i,j)} - x_{(i,l)} + z(\Delta_{(i,j)}^{(i,l)}) &= 0 \quad (i, j, l = 1, \ldots, n, j < l). \end{aligned}$$

We will show that in both cases for the affine subspaces defined above all variables corresponding to vertices and edges, respectively, hyperedges involving the nth row or the nth column (the same holds for any row and any column) are redundant in the sense that the projections onto the linear subspaces of the original spaces obtained by setting all these variables to zero produce isomorphic images of these two affine subspaces. Since the two polytopes under consideration are contained in the respective affine subspaces, this implies that these projections yield isomorphic images of the polytopes.

Let $W := \mathrm{row}_n^{(n)} \cup \mathrm{col}_n^{(n)}$, $E := \{e \in \mathcal{E}_n \mid e \cap W \neq \emptyset\}$, and $F := \{f \in \mathcal{F}_n \mid f \cap W \neq \emptyset\}$. Define $\mathcal{U} := \{(x, y) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n} \mid x_W = 0, y_E = 0\}$ and $\widehat{\mathcal{U}} := \{(x, z) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n} \mid x_W = 0, z_F = 0\}$. Let $\pi : \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{E}_n} \longrightarrow \mathcal{U}$ be the orthogonal projection onto $\mathcal{U}$, and $\widehat{\pi} : \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n} \longrightarrow \widehat{\mathcal{U}}$ be the orthogonal projection onto $\widehat{\mathcal{U}}$.

PROPOSITION 3.4. *$\pi(\mathcal{A})$ is affinely isomorphic to $\mathcal{A}$ and $\widehat{\pi}(\widehat{\mathcal{A}})$ is affinely isomorphic to $\widehat{\mathcal{A}}$.*

Proof. We prove only the symmetric part of the proposition. The nonsymmetric part can be shown quite similarly [16].

First, we show that there is a way to express the components of points in $\widehat{\mathcal{A}}$ belonging to elements in W and F linearly by the components belonging to elements in $\mathcal{V}_n \setminus W$ and $\mathcal{F}_n \setminus F$.

The first observation is that this is possible for the elements in W using equations of the type $x(\mathrm{row}_i^{(n)}) = 1$ and $x(\mathrm{col}_j^{(n)}) = 1$. Now, we consider F. Here, it suffices to consider three possibilities for a hyperedge $\langle i, j, k, l\rangle \in F$. The first two are $i, j, k < n,\ l = n$ and $i, j, l < n,\ k = n$. Using $-x_{(i,j)} - x_{(k,j)} + z(\Delta_{(i,j)}^{(k,j)}) = 0$, respectively, $-x_{(i,j)} - x_{(i,l)} + z(\Delta_{(i,j)}^{(i,l)}) = 0$, the first two possibilities are done. The possibility remains that $i, j < n,\ k = n,\ l = n$. Here, we consider (e.g.) $-x_{(i,j)} - x_{(i,n)} + z(\Delta_{(i,j)}^{(i,n)}) = 0$, which allows to express $z_{\langle i,j,n,n\rangle}$ since we can already express $z_{\langle i,j,k,n\rangle}$ for $k < n$.

Up to now, we have shown that there is a linear function $\widehat{\psi} : \mathbb{R}^{\mathcal{V}_n \setminus W} \times \mathbb{R}^{\mathcal{F}_n \setminus F} \longrightarrow \mathbb{R}^W \times \mathbb{R}^F$ such that for all $(x, z) \in \widehat{\mathcal{A}}$ we have $(x_W, z_F) = \widehat{\psi}(x_{\mathcal{V}_n \setminus W}, z_{\mathcal{F}_n \setminus F})$. Hence

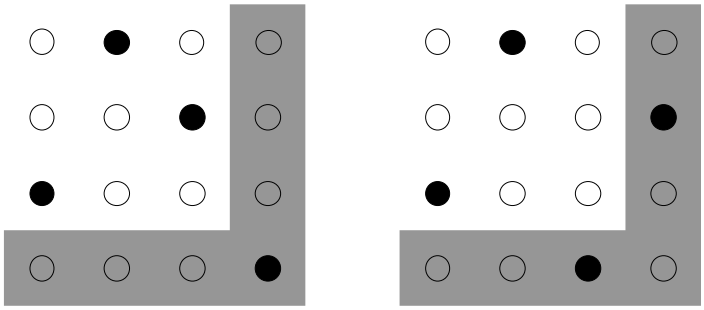


FIG. 3.1. *The effect of the projection.*

$\widehat{\phi} : \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n} \longrightarrow \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ defined via $\widehat{\phi}(x,z) = (x',z')$ with

$$\begin{aligned}(x'_W, z'_F) &:= (x_W, z_F) - \widehat{\psi}(x_{\mathcal{V}_n \setminus W}, z_{\mathcal{F}_n \setminus F}),\\ (x'_{\mathcal{V}_n \setminus W}, z'_{\mathcal{F}_n \setminus F}) &:= (x_{\mathcal{V}_n \setminus W}, z_{\mathcal{F}_n \setminus F})\end{aligned}$$

is an affine transformation (note that the corresponding matrix is an upper triangular one having 1's everywhere on the main diagonal) of $\mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ that induces on $\widehat{\mathcal{A}}$ the orthogonal projection onto $\widehat{\mathcal{U}}$. □

We identify the linear spaces $\mathcal{U}$ and $\widehat{\mathcal{U}}$ with the spaces $\mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{E}_{n-1}}$ and $\mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{F}_{n-1}}$, respectively. Hence,

$$\mathcal{QAP}^{\star}_{n-1} := \pi(\mathcal{QAP}_n) \subset \mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{E}_{n-1}}$$

is a polytope in $\mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{E}_{n-1}}$ that is isomorphic to $\mathcal{QAP}_n$, and

$$\mathcal{SQAP}^{\star}_{n-1} := \widehat{\pi}(\mathcal{SQAP}_n) \subset \mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{F}_{n-1}}$$

is a polytope in $\mathbb{R}^{\mathcal{V}_{n-1}} \times \mathbb{R}^{\mathcal{F}_{n-1}}$ that is isomorphic to $\mathcal{SQAP}_n$.

Since the vertices of these two polytopes arise as the projections of the vertices of the two original polytopes, one obtains that they are the respective characteristic vectors of the $(n-1)$- and the $(n-2)$-cliques of $\mathcal{G}_{n-1}$ (cf. Figure 3.1).

We want to make the isomorphism between $\mathcal{QAP}_n$ and $\mathcal{QAP}^{\star}_{n-1}$ as well as the one between $\mathcal{SQAP}_n$ and $\mathcal{SQAP}^{\star}_{n-1}$ a little more explicit. We denote by $\kappa : \mathcal{CLQ}^n_n \longrightarrow \mathcal{CLQ}^{n-1}_{n-1} \cup \mathcal{CLQ}^{n-1}_{n-2}$ the map defined by removing from a given n-clique in $\mathcal{G}_n$ the node(s) in the nth row and in the nth column. Notice that κ is one to one.

REMARK 1. *If two faces of $\mathcal{QAP}_n$ and $\mathcal{QAP}^{\star}_{n-1}$, respectively, $\mathcal{SQAP}_n$ and $\mathcal{SQAP}^{\star}_{n-1}$, correspond to each other with respect to the isomorphism induced by π, respectively, $\widehat{\pi}$, then their vertices (identified with cliques) correspond to each other by the bijection κ.*

This remark describes the relationship between the faces from the "inner view," i.e., in terms of the vertices. Next, we want to describe the "outer relationship," i.e., the relationship between inequalities defining corresponding faces.

REMARK 2.

(i) *If a face of $\mathcal{QAP}_n$, respectively, $\mathcal{SQAP}_n$, is defined by an inequality that has zero coefficients for all elements in $W \cup E$, respectively, $W \cup F$, then an inequality defining the corresponding face of $\mathcal{QAP}^{\star}_{n-1}$, respectively, $\mathcal{SQAP}^{\star}_{n-1}$, is obtained by projecting the coefficient vector of that inequality via π, respectively, $\widehat{\pi}$. (Note that for every face of $\mathcal{QAP}_n$, respectively, $\mathcal{SQAP}_n$, there is a*

defining inequality having zero coefficients at W and E, respectively F. This is due to the fact that the columns of the equation system defining the affine subspace $\mathcal{A}$, respectively, $\widehat{\mathcal{A}}$, corresponding to $W \cup E$, respectively, $W \cup F$, are linearly independent, as shown in the proof of Proposition 3.4.)

(ii) *From every inequality defining a face of $\mathcal{QAP}^\star_{n-1}$, respectively, $\mathcal{SQAP}^\star_{n-1}$, one obtains an inequality defining the corresponding face of $\mathcal{QAP}_n$, respectively, $\mathcal{SQAP}_n$, by zero-lifting.*

The following "star-analogons" to some facts observed for $\mathcal{QAP}_n$ and $\mathcal{SQAP}_n$ hold. First, also the "star-polytopes" are invariant under permutations of rows, permutations of columns, or "transposition" of the node set $\mathcal{V}_n$. Second, as in the relationship between $\mathcal{QAP}_n$ and $\mathcal{SQAP}_n$, by identifying mates any symmetric inequality (equation) for $\mathcal{QAP}^\star_n$ gives rise to an inequality (equation) for $\mathcal{SQAP}^\star_n$, and any inequality (equation) for $\mathcal{SQAP}^\star_n$ gives rise to a symmetric inequality (equation) for $\mathcal{QAP}^\star_n$.

THEOREM 3.5. *The face lattice of $\mathcal{SQAP}^\star_n$ arises by restricting the face lattice of $\mathcal{QAP}^\star_n$ to the symmetric faces.*

COROLLARY 3.6. *A symmetric proper face of $\mathcal{QAP}^\star_n$ induces a facet of $\mathcal{SQAP}^\star_n$ if and only if there are only nonsymmetric faces strictly between itself and $\mathcal{QAP}^\star_n$ in the face lattice of $\mathcal{QAP}^\star_n$.*

COROLLARY 3.7. *If a symmetric ridge of $\mathcal{QAP}^\star_n$ is the intersection of two nonsymmetric facets of $\mathcal{QAP}^\star_n$, then it induces a facet of $\mathcal{SQAP}^\star_n$.*

We close this section by the following "inductive construction" of $\mathcal{SQAP}_{n+1}$. It establishes a kind of "self-similarity" that shows another symmetry of the SQAP-polytope. The proof of the theorem can be found in [17].

THEOREM 3.8. *For $n \geq 1$ there are $n+1$ affine maps $\iota_\alpha : \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n} \longrightarrow \mathbb{R}^{\mathcal{V}_{n+1}} \times \mathbb{R}^{\mathcal{F}_{n+1}}$ $(\alpha = 0, \ldots, n)$ such that for the $n+1$ images $\mathcal{Q}_\alpha := \iota_\alpha(\mathcal{SQAP}_n)$ $(\alpha = 0, \ldots, n)$ of $\mathcal{SQAP}_n$ the following hold:*

(i) *Every $\mathcal{Q}_\alpha$ is isomorphic to $\mathcal{SQAP}_n$.*
(ii) *Each $\mathcal{Q}_\alpha$ is a face of $\mathcal{SQAP}_{n+1}$.*
(iii) *The $\mathcal{Q}_\alpha$ have pairwise empty intersection.*
(iv) $\mathcal{SQAP}_{n+1} = \operatorname{conv}\left(\bigcup_{\alpha=0}^{n} \mathcal{Q}_\alpha\right)$.

4. Dimension and trivial facets of $\mathcal{SQAP}_n$. In this section, we will present some basic results concerning the facial structure of the SQAP-polytope. First, we examine two sets of equations that will turn out to describe the affine hulls of $\mathcal{QAP}^\star_n$, respectively, $\mathcal{SQAP}^\star_n$. For this, we make another notational convention. For two disjoint subsets $S, T \subset \mathcal{V}_n$, $S \cap T = \emptyset$, we define $\langle S : T \rangle := \{\{v, w\} \cup \tau(\{v, w\}) \mid \{v, w\} \in (S : T)\}$. Remembering that the vertices of both $\mathcal{QAP}^\star_n$ and $\mathcal{SQAP}^\star_n$ correspond to the n- and $(n-1)$-cliques of $\mathcal{G}_n$, one verifies that

$$x(\mathrm{row}_i^{(n)}) + x(\mathrm{row}_k^{(n)}) - y(\mathrm{row}_i^{(n)} : \mathrm{row}_k^{(n)}) = 1 \qquad (i < k) \tag{4.1}$$

and

$$x(\mathrm{col}_j^{(n)}) + x(\mathrm{col}_l^{(n)}) - y(\mathrm{col}_j^{(n)} : \mathrm{col}_l^{(n)}) = 1 \qquad (j < l) \tag{4.2}$$

are valid for $\mathcal{QAP}^\star_n$, and

$$x(\mathrm{row}_i^{(n)}) + x(\mathrm{row}_k^{(n)}) - z(\langle \mathrm{row}_i^{(n)} : \mathrm{row}_k^{(n)} \rangle) = 1 \qquad (i < k) \tag{4.3}$$

and

$$x(\mathrm{col}_j^{(n)}) + x(\mathrm{col}_l^{(n)}) - z(\langle \mathrm{col}_j^{(n)} : \mathrm{col}_l^{(n)} \rangle) = 1 \qquad (j < l) \tag{4.4}$$

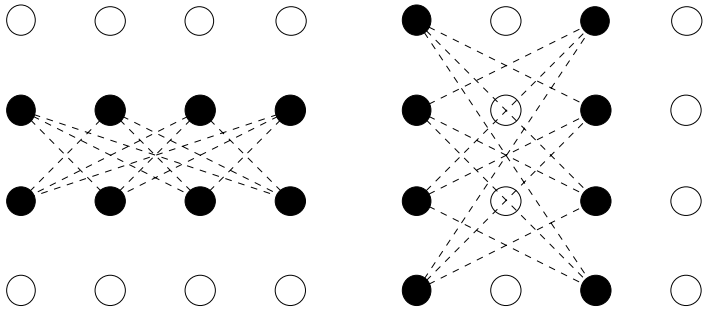


FIG. 4.1. *The equations* (4.1), (4.3) *and* (4.2), (4.4).

hold for $\mathcal{SQAP}_n^\star$ (cf. Figure 4.1).

We denote the system (4.1), (4.2) by $D(x,y) = d$ and the system (4.3), (4.4) by $\widehat{D}(x,z) = \widehat{d}$.

By saying that $\langle i,j,k,l\rangle$ $(i<k, j<l)$ is smaller than $\langle i',j',k',l'\rangle$ $(i'<k', j'<l')$ if and only if (i,k,j,l) is lexicographically smaller than (i',k',j',l'), we introduce an ordering on the hyperedges $\mathcal{F}_n$. After permutation of the columns with respect to this order the "z-part" of the matrix $\widehat{D}$ has the following shape ($n=3$):

$$\begin{pmatrix} 1 & 1 & 1 & & & & & & \\ & & & 1 & 1 & 1 & & & \\ & & & & & & 1 & 1 & 1 \\ 1 & & & 1 & & & 1 & & \\ & 1 & & & 1 & & & 1 & \\ & & 1 & & & 1 & & & 1 \end{pmatrix}.$$

But this is the node-edge-incidence matrix of the complete bipartite graph on $\frac{n(n-1)}{2} + \frac{n(n-1)}{2}$ nodes, where the left shore corresponds to the (unordered) pairs of rows, and the right shore corresponds to the (unordered) pairs of columns of $\mathcal{V}_n$. The bases of the node-edge-incidence matrix of the complete bipartite graph on $m+m$ nodes are well known to correspond to the spanning trees of that graph [4]. Using the observation that $\widehat{D}$ does not have full row-rank (since, e.g., the sum of all equations in (4.3) equals the sum of all equations in (4.4)), this leads to the following characterization of all bases of $\widehat{D}$ that do not intersect the "x-part" of $\widehat{D}$.

PROPOSITION 4.1.

(i) *Precisely one (arbitrary) equation in* $\widehat{D}(x,z) = \widehat{d}$ *is redundant, in particular* $\operatorname{rank}(\widehat{D}) = n(n-1)-1$.

(ii) *A subset* $B \subseteq \mathcal{F}_n$ *of hyperedges corresponds to a basis of* $\widehat{D}$ *if and only if*
   (a) $|B| = n(n-1)-1$;
   (b) *there is no sequence* $(f_0, f_0', f_1, f_1', \ldots, f_{k-1}, f_{k-1}')$ *(*$k \geq 2$*) of hyperedges in* B *such that* f_α *and* f_α' *connect the same rows of* $\mathcal{V}_n$ *and* f_α' *and* $f_{(\alpha+1) \bmod k}$ *connect the same columns of* $\mathcal{V}_n$ *for all* $\alpha = 0, \ldots, k-1$.

In [16] we showed that $D(x,y) = d$ is a complete equation system for $\mathcal{QAP}_n^\star$. But the system $D(x,y) = d$ consists only of symmetric equations. Hence, we can deduce that $\widehat{D}(x,z) = \widehat{d}$ must be a complete system of equations for $\mathcal{SQAP}_n^\star$, since the equations for $\mathcal{SQAP}_n^\star$ correspond precisely to the symmetric equations for $\mathcal{QAP}_n^\star$. (In fact, one can deduce the "completeness" of $\widehat{D}(x,z) = \widehat{d}$ also from the proof of Theorem 4.4.)

Consequently, the dimension of $\mathcal{SQAP}_n^\star$ is $n^2 + \frac{n^2(n-1)^2}{4} - (n(n-1)-1)$. By the isomorphism between $\mathcal{SQAP}_n$ and $\mathcal{SQAP}_{n-1}^\star$, one obtains the following theorem.

THEOREM 4.2.

$$\dim(\mathcal{SQAP}_n) = (n-1)^2 + \frac{(n-1)^2(n-2)^2}{4} - ((n-1)(n-2)-1).$$

[24] and [21] proved that the rank of the system (2.7)–(2.10) equals $(n-1)^2 + \frac{n^2(n-3)^2}{4}$ (which is equal to $(n-1)^2 + \frac{(n-1)^2(n-2)^2}{4} - ((n-1)(n-2)-1)$) and conjectured that this might be the dimension of $\mathcal{SQAP}_n$. Theorem 4.2 proves this conjecture. Moreover, knowing that the rank of this system equals $\dim(\mathcal{SQAP}_n)$, one can even conclude that the system (2.7)–(2.10) describes the affine hull of $\mathcal{SQAP}_n$. In addition, we want to give another simple proof that does not compute the rank of the system explicitly.

THEOREM 4.3.

$$\operatorname{aff}(\mathcal{SQAP}_n) = \{(x,z) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n} \mid (x,z) \textit{ satisfies } (2.7), \ldots, (2.10)\}.$$

Proof. It suffices to show that one can linearly combine the zero-liftings of (4.3) and (4.4) (for $n-1$) from (2.7)–(2.10) (for n), since then it is clear that the solution space of (2.7)–(2.10) for n—which is $\widehat{\mathcal{A}}$ (containing $\mathcal{SQAP}_n$)—is mapped isomorphically (cf. Proposition 3.4) by the projection $\widehat{\pi}$ into the solution space of (4.3), (4.4) for $n-1$, which we know from our considerations to have the same dimension as $\mathcal{SQAP}_n$.

Hence, by symmetry arguments, it suffices to exhibit a linear combination of (2.7)–(2.10) that yields

$$x(\mathrm{row}_1^{(n)} \setminus \{(1,n)\}) + x(\mathrm{row}_2^{(n)} \setminus \{(2,n)\}) - z(\langle \mathrm{row}_1^{(n)} \setminus \{(1,n)\} : \mathrm{row}_2^{(n)} \setminus \{(2,n)\}\rangle) = 1.$$

But this is obtained by adding $x(\mathrm{row}_1^{(n)}) = 1$, $x(\mathrm{row}_2^{(n)}) = 1$, $x_{(1,j)} + x_{(2,j)} - z(\Delta_{(1,j)}^{(2,j)}) = 0$ for all $1 \le j \le n-1$, and $-x_{(1,n)} - x_{(2,n)} + z(\Delta_{(1,n)}^{(2,n)}) = 0$, and finally dividing the resulting equation by 2. □

We just mention that the system (2.1)–(2.4) describes $\operatorname{aff}(\mathcal{QAP}_n)$ [24, 21, 16].

There is another nice gain when changing to the "star-polytopes." We pointed out in Corollary 3.2 that it is of interest to know that certain faces of the QAP-polytope are nonsymmetric. As mentioned above, this might not be directly seen, since a symmetric face of $\mathcal{QAP}_n$ can be defined by a nonsymmetric inequality. However, this is much easier for $\mathcal{QAP}_n^\star$.

OBSERVATION 3. *Due to the fact that all equations holding for $\mathcal{QAP}_n^\star$ are symmetric, in order to show that a given face of $\mathcal{QAP}_n^\star$ is nonsymmetric, it suffices to exhibit any nonsymmetric inequality defining it.*

For the nonsymmetric QAP-polytope, the nonnegativity constraints on y define facets, while $0 \le x \le 1$ and $y \le 1$ are already implied by $D(x,y) = d$ and $y \ge 0$ [24, 21, 16]. For the SQAP-polytope, the situation is a little bit different, as the following theorem shows.

THEOREM 4.4. *Let $n \ge 3$.*

(i) *The nonnegativity constraints $x \ge 0$ and $z \ge 0$ define facets of $\mathcal{SQAP}_n$.*

(ii) *The upper bounds $x \le 1$ and $z \le 1$ are implied by (2.7)–(2.10) and $x \ge 0$, $z \ge 0$.*

Proof. Part (ii) follows from the observation that (2.7) and (2.8) together with the nonnegativity of x imply $x \leq 1$. Furthermore, (2.7) and (2.8) even imply that the sum of any two x-variables that belong to the same row or column must be less than or equal to 1. Thus, from (2.9), (2.10), and the nonnegativity of z one obtains $z \leq 1$ as well.

To show part (i), it suffices to prove that $x \geq 0$ and $z \geq 0$ define facets of $\mathcal{SQAP}_n^\star$ (for all $n \geq 3$). We will show this only for $n \geq 5$, since this simplifies the proof. However, the claim is also true for $n = 3, 4$, as one may check by computer, for instance.

At this point, we introduce some techniques which we will also refer to in later proofs. Our usual way of proving that some inequality defines a facet of $\mathcal{SQAP}_n^\star$ is an indirect one. We denote by $L \subseteq \mathcal{CLQ}_n^n \cup \mathcal{CLQ}_{n-1}^n$ the set of cliques corresponding to the vertices of the considered face and by $\mathcal{L} := \{(x^C, z^{\mathcal{F}_n(C)}) - (x^{C'}, z^{\mathcal{F}_n(C')}) \mid C, C' \in L\}$ the set of all difference vectors of vertices of that face, i.e., $\text{lin}(\mathcal{L})$ is the subvectorspace belonging to the affine hull of the face. We choose a subset $B \subset \mathcal{F}_n$ that corresponds to a basis of the equation system $\widehat{D}(x, z) = \widehat{d}$ as well as one extra element $v_0 \in \mathcal{V}_n$ or $f_0 \in \mathcal{F}_n \setminus B$. Setting $\mathcal{B} := \{x^{v_0}\} \cup \{z^f \mid f \in B\}$, respectively, $\mathcal{B} := \{z^{f_0}\} \cup \{z^f \mid f \in B\}$, and providing that the face is a proper one, it remains to show that $\text{lin}(\mathcal{L} \cup \mathcal{B}) = \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$, since this implies that the dimension of $\text{lin}(\mathcal{L})$, which equals the dimension of the face, is at least $\dim(\mathcal{SQAP}_n^\star) - 1$. We show $\text{lin}(\mathcal{L} \cup \mathcal{B}) = \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ by successively combining the canonical unit vectors of $\mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ from elements in $\mathcal{L} \cup \mathcal{B}$.

For constructing the necessary linear combinations, the following two lemmas are useful. For a subset $S \subseteq \mathcal{V}_n$ we denote by $\mathcal{H}_n/S = (\mathcal{V}_n/S, \mathcal{F}_n/S)$ the hypergraph obtained from $\mathcal{H}_n$ by deleting all nodes lying in a common row or column with a node in S and all hyperedges involving such nodes. Note that if S intersects the same number of rows as of columns, $\mathcal{H}_n/S$ is isomorphic to an $\mathcal{H}_k$ for some $k \leq n$.

LEMMA 4.5. *Let $C \in \mathcal{CLQ}_n^n$ be an n-clique and $v \in C$ a node in C such that $C, C \setminus \{v\} \in L$. Then we have*

$$x^v + z^{\langle v:C\setminus\{v\}\rangle} \in \text{lin}(\mathcal{L}).$$

Proof. This is due to $x^v + z^{\langle v:C\setminus\{v\}\rangle} = (x^C, z^{\mathcal{F}_n(C)}) - (x^{C\setminus\{v\}}, z^{\mathcal{F}_n(C\setminus\{v\})}) \in \text{lin}(\mathcal{L})$. □

LEMMA 4.6. *Let $1 \leq r, r_1, r_2 \leq n$ be pairwise distinct, and let $1 \leq c, c_1, c_2 \leq n$ be pairwise distinct. If there is an $(n-3)$-clique C in $\mathcal{H}_n/\{(r_1, c_1), (r, c), (r_2, c_2)\}$ such that*

$$\{(r_1, c_1), (r, c), (r_2, c_2)\} \cup C, \quad \{(r_1, c_2), (r, c), (r_2, c_1)\} \cup C, \quad \{(r_1, c_1), (r_2, c_2)\} \cup C, \quad \{(r_1, c_2), (r_2, c_1)\} \cup C \in L \tag{4.5}$$

or

$$\{(r_1, c), (r, c_2)\} \cup C, \quad \{(r, c_2), (r_2, c)\} \cup C, \quad \{(r_2, c), (r, c_1)\} \cup C, \quad \{(r, c_1), (r_1, c)\} \cup C \in L, \tag{4.6}$$

then

$$z^{\langle r_1,c_1,r,c\rangle} + z^{\langle r,c,r_2,c_2\rangle} - z^{\langle r_1,c_2,r,c\rangle} - z^{\langle r,c,r_2,c_1\rangle} \in \text{lin}(\mathcal{L})$$

(cf. Figure 4.2).

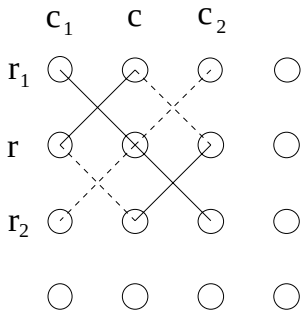


FIG. 4.2. *Notations of Lemma* 4.6.

Proof. In the first case, observe that

$$\begin{aligned} & z^{\langle r_1,c_1,r,c\rangle} + z^{\langle r,c,r_2,c_2\rangle} - z^{\langle r_1,c_2,r,c\rangle} - z^{\langle r,c,r_2,c_1\rangle} \\ = {} & z^{\{(r_1,c_1),(r,c),(r_2,c_2)\}\cup C} - z^{\{(r_1,c_1),(r_2,c_2)\}\cup C} \\ - {} & z^{\{(r_1,c_2),(r,c),(r_2,c_1)\}\cup C} + z^{\{(r_1,c_2),(r_2,c_1)\}\cup C} \in \operatorname{lin}(\mathcal{L}). \end{aligned}$$

For the second case, we have

$$\begin{aligned} & z^{\langle r_1,c_1,r,c\rangle} + z^{\langle r,c,r_2,c_2\rangle} - z^{\langle r_1,c_2,r,c\rangle} - z^{\langle r,c,r_2,c_1\rangle} \\ = {} & - z^{\{(r_1,c),(r,c_2)\}\cup C} + z^{\{(r,c_2),(r_2,c)\}\cup C} \\ & - z^{\{(r_2,c),(r,c_1)\}\cup C} + z^{\{(r,c_1),(r_1,c)\}\cup C} \in \operatorname{lin}(\mathcal{L}). \qquad \square \end{aligned}$$

Now, we proceed with the proof of Theorem 4.4. First, note that all trivial inequalities define proper faces of $\mathcal{SQAP}_n^\star$. To show that the nonnegativity constraints on x define facets of $\mathcal{SQAP}_n^\star$, it suffices to show this for $x_{(n,n)} \geq 0$. Hence, L consists of all n- and $(n-1)$-cliques of $\mathcal{H}_n$ that do not contain (n,n). We choose $B := \langle \operatorname{row}_1^{(n)} : \operatorname{row}_2^{(n)}\rangle \cup \langle \operatorname{col}_1^{(n)} : \operatorname{col}_2^{(n)}\rangle$ (cf. Proposition 4.1) and the extra element as $v_0 := (n,n)$.

Since in $\mathcal{H}_k$ there is always a k-clique not involving a prescribed node as long as $k \geq 2$, we can apply Lemma 4.6 for every choice of r, r_1, r_2, c, c_1, c_2. (Recall that we assume $n \geq 5$.) We combine all canonical unit vectors in $\mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ successively in five steps that are illustrated in Figure 4.3. For a number $a \in \{1,2\}$, we denote by $\overline{a}$ the number with $\{\overline{a}\} = \{1,2\} \setminus \{a\}$.

Step 1. $z^{\langle i,j,k,l\rangle} \in \operatorname{lin}(\mathcal{L} \cup \mathcal{B})$ for $i,j \in \{1,2\}$.

The case $k \in \{1,2\}$ or $l \in \{1,2\}$ is already clear by the choice of B. Hence, assume $k,l \notin \{1,2\}$. Choosing $r := i$, $r_1 := \overline{i}$, $r_2 := k$, $c := j$, $c_1 := \overline{j}$, and $c_2 := l$ Lemma 4.6 yields $z^{\langle \overline{i},\overline{j},i,j\rangle} + z^{\langle i,j,k,l\rangle} - z^{\langle \overline{i},l,i,j\rangle} - z^{\langle i,j,k,\overline{j}\rangle} \in \operatorname{lin}(\mathcal{L})$. Since all involved unit vectors but $z^{\langle i,j,k,l\rangle}$ are in $\mathcal{B}$, we are done.

Step 2. $z^{\langle i,j,k,l\rangle} \in \operatorname{lin}(\mathcal{L} \cup \mathcal{B})$ for $i \in \{1,2\}, j,k,l \geq 3$.

With $r := i$, $r_1 := \overline{i}$, $r_2 := k$, $c := j$, $c_1 := 1$, $c_2 := l$ one obtains from Lemma 4.6 that $z^{\langle \overline{i},1,i,j\rangle} + z^{\langle i,j,k,l\rangle} - z^{\langle \overline{i},l,i,j\rangle} - z^{\langle i,j,k,1\rangle} \in \operatorname{lin}(\mathcal{L})$. All involved unit vectors but $z^{\langle i,j,k,l\rangle}$ are either in $\mathcal{B}$ or already shown to be in $\operatorname{lin}(\mathcal{L} \cup \mathcal{B})$ in Step 1.

Step 3. $z^{\langle i,j,k,l\rangle} \in \operatorname{lin}(\mathcal{L} \cup \mathcal{B})$ for $j \in \{1,2\}, i,k,l \geq 3$.

This is done analogously to Step 2.

Step 4. $z^{\langle i,j,k,l\rangle} \in \operatorname{lin}(\mathcal{L} \cup \mathcal{B})$ for $i,j,k,l \geq 3$.

This time, we choose $r := i$, $r_1 := 1$, $r_2 := k$, $c := j$, $c_1 := 1$, and $c_2 := l$. Lemma 4.6 gives $z^{\langle 1,1,i,j\rangle} + z^{\langle i,j,k,l\rangle} - z^{\langle 1,l,i,j\rangle} - z^{\langle i,j,k,1\rangle} \in \operatorname{lin}(\mathcal{L})$, which proves the

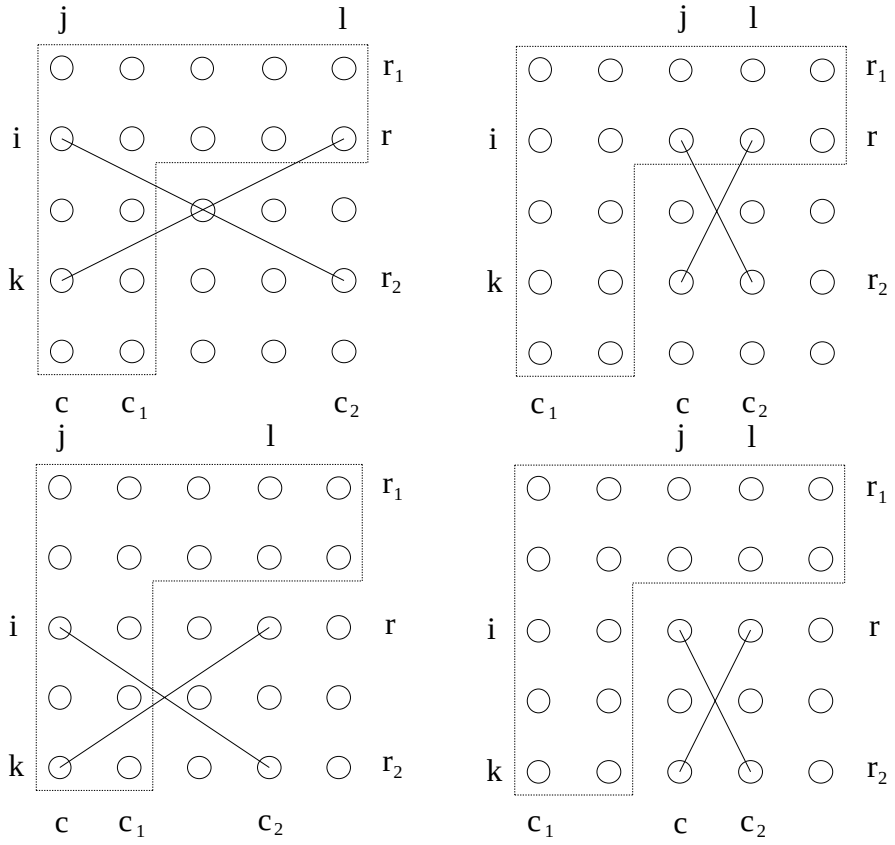


FIG. 4.3. *Examples for the hyperedges considered in Steps* 1–4 *of the proof of Theorem* 4.4. *The hyperedges inside the "angled box" are those forming the set* B.

claim, since all involved unit vectors but $z^{\langle i,j,k,l\rangle}$ are already shown to be in $\text{lin}(\mathcal{L}\cup\mathcal{B})$ in Steps 1, 2, or 3.

Step 5. $x^v \in \text{lin}(\mathcal{L}\cup\mathcal{B})$ for all $v \in \mathcal{V}_n$.

If $v = (n,n)$, we are done since $x^{(n,n)} \in \mathcal{B}$. So assume, $v \neq (n,n)$. Let $C \in \mathcal{CLQ}_n^n$ be any n-clique involving v but not (n,n). Using Lemma 4.5, we can combine x^v, since all unit vectors corresponding to hyperedges are already known to be in $\text{lin}(\mathcal{L}\cup\mathcal{B})$.

It remains to show that $z \geq 0$ define facets of $\mathcal{SQAP}_n^\star$. It suffices to show this for $z_{\langle n,n-1,n-1,n\rangle} \geq 0$. Now, L is the set of all n- and $(n-1)$-cliques of $\mathcal{H}_n$ that contain at most one node from $\{(n,n-1),(n-1,n),(n-1,n-1),(n,n)\}$. Note that it is always possible to find a k-clique in $\mathcal{H}_k$ that intersects $\{(k,k-1),(k-1,k),(k-1,k-1),(k,k)\}$ in at most one node as long as $k \geq 3$.

We choose B as above, and as the extra element, we take the hyperedge $\langle n, n-1, n-1, n\rangle$. Then, Steps 1, 2, and 3 work analogously. The only case in which Step 4 does not work is the case of the hyperedge $\langle n, n-1, n-1, n\rangle$, but this time this one is covered by the extra element. In Step 5, now we do not need an extra element anymore, since we can extend every node (also one from $\{(n,n-1),(n-1,n),(n-1,n-1),(n,n)\}$) to an n-clique not containing more than one node from $\{(n,n-1),(n-1,n),(n-1,n-1),(n,n)\}$. □

There is an alternative way of proving that the nonnegativity constraints $z \geq 0$

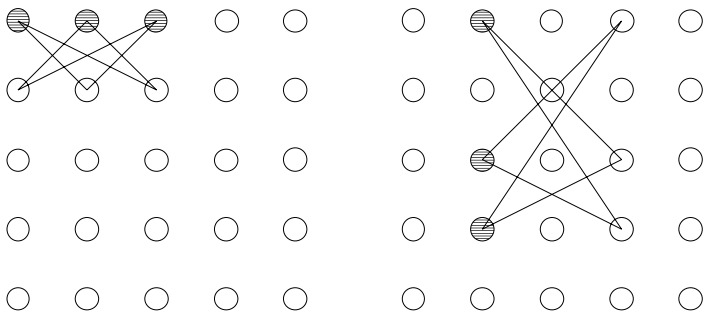


FIG. 5.1. *The curtain inequalities.*

define facets of $\mathcal{SQAP}_n^\star$. In [16] we showed that $y \geq 0$ define facets of $\mathcal{QAP}_n^\star$. By a slight modification of that proof, one can show that $y_e + y_{\tau(e)} \geq 0$ defines a ridge of $\mathcal{QAP}_n^\star$ for any edge $e \in \mathcal{E}_n$. Since that symmetric ridge is the intersection of the two nonsymmetric (cf. Observation 3) facets defined by $y_e \geq 0$ and $y_{\tau(e)} \geq 0$, the claim follows from Corollary 3.7.

5. The curtain facets. For any subset $S \subseteq \{1, \ldots, n\}$, we define for $i \in \{1, \ldots, n\}$ the *restriction of* $\mathrm{row}_i^{(n)}$ *to* S as $\mathrm{row}_i^{(n)}|_S := \{(i,j) \in \mathrm{row}_i^{(n)} \mid j \in S\}$, and for $j \in \{1, \ldots, n\}$, we define $\mathrm{col}_j^{(n)}|_S := \{(i,j) \in \mathrm{col}_j^{(n)} \mid i \in S\}$ to be the *restriction of* $\mathrm{col}_j^{(n)}$ *to* S.

One immediately verifies that the *row curtain inequalities*

$$-x(\mathrm{row}_i^{(n)}|_S) + z(\langle \mathrm{row}_i^{(n)}|_S : \mathrm{row}_k^{(n)}|_S \rangle) \leq 0 \qquad (i \neq k, S \subseteq \{1, \ldots, n\}) \tag{5.1}$$

and the *column curtain inequalities*

$$-x(\mathrm{col}_j^{(n)}|_S) + z(\langle \mathrm{col}_j^{(n)}|_S : \mathrm{col}_l^{(n)}|_S \rangle) \leq 0 \qquad (j \neq l, S \subseteq \{1, \ldots, n\}) \tag{5.2}$$

are valid for $\mathcal{SQAP}_n$ (cf. Figure 5.1).

These inequalities dominate the inequalities

$$-x(\mathrm{row}_i^{(n)}|_S) + z(\langle (i,j) : \mathrm{row}_k^{(n)}|_S \rangle) \leq 0 \qquad (i \neq k, S \subseteq \{1, \ldots, n\}, j \in S) \tag{5.3}$$

and

$$-x(\mathrm{col}_j^{(n)}|_S) + z(\langle (i,j) : \mathrm{col}_l^{(n)}|_S \rangle) \leq 0 \qquad (j \neq l, S \subseteq \{1, \ldots, n\}, i \in S) \tag{5.4}$$

proposed by [24] and [21].

The proof of the following theorem (which again uses the isomorphism between $\mathcal{SQAP}_n$ and $\mathcal{SQAP}_{n-1}^\star$) can be found in [17].

THEOREM 5.1. *All curtain inequalities with* $3 \leq |S| \leq n-3$ *define facets of* $\mathcal{SQAP}_n$.

We conclude this section with a consideration of the separation problem associated with the class of curtain inequalities. For this, let a (fractional) point $(\tilde{x}, \tilde{z}) \in \mathbb{R}^{\mathcal{V}_n} \times \mathbb{R}^{\mathcal{F}_n}$ be given. We want to find, e.g., a row curtain inequality using rows 1 and 2 (ordered) that "cuts off" the point $(\tilde{x}, \tilde{z})$. Hence, we want to find a subset $S \subseteq \{1, \ldots, n\}$ such that $-\tilde{x}(\mathrm{row}_1^{(n)}|_S) + \tilde{z}(\langle \mathrm{row}_1^{(n)}|_S : \mathrm{row}_2^{(n)}|_S \rangle) > 0$. But this is exactly

the task to find a characteristic vector ξ of $\{1,\dots,n\}$ that solves the *(unconstrained) Boolean quadratic* 0/1 *problem (BQP)*

$$\begin{aligned} \max \quad & \sum_{j=1}^{n}\sum_{l=j+1}^{n} \alpha_{jl}\xi_j\xi_l + \sum_{j=1}^{n}\beta_j\xi_j \\ \text{subject to} \quad & \xi \in \{0,1\}^n \end{aligned}$$

with $\alpha_{jl} := \tilde{z}_{\langle 1,j,2,l\rangle}$ and $\beta_j := -\tilde{x}_{(1,j)}$.

Hence, for each (ordered) pair of rows, respectively, columns, a BQP has to be solved. Although this is known to be $\mathcal{NP}$-hard in general, the special case of our separation problem, where all coefficients of quadratic terms are nonnegative, can be solved in polynomial time by computing a (directed) s-t minimum cut in a suitably defined graph (with nonnegative edge weights). This was first discovered by [23] (who formulated an algorithm in terms of flows) and further considered by [3] and other authors.

6. Lower bounds. For any instance of the QAP, the minimum the objective function achieves over the intersection of $\mathrm{aff}(\mathcal{QAP}_n)$ and the nonnegative orthant is a lower bound for the optimal value of the respective QAP, called the *equation bound (EQB)*. This bound can be computed by solving the linear program arising from (2.1)–(2.4) and the nonnegativity constraints on the y-variables. Similarly, if the instance is symmetric, the minimum over the intersection of $\mathrm{aff}(\mathcal{SQAP}_n)$ and the nonnegative orthant gives a lower bound, called the *symmetric equation bound (SEQB)*. This may be computed by solving the linear program defined by (2.7)–(2.10) and the nonnegativity constraints on x and z.

Let $(x,y) \in \mathrm{aff}(\mathcal{QAP}_n) \cap (\mathbb{R}^{\mathcal{V}_n}_{\geq 0} \times \mathbb{R}^{\mathcal{E}_n}_{\geq 0})$ have value θ with respect to a symmetric objective function. Then $\mathrm{sym}_n(x,y) \in \mathrm{aff}(\mathcal{SQAP}_n) \cap (\mathbb{R}^{\mathcal{V}_n}_{\geq 0} \times \mathbb{R}^{\mathcal{F}_n}_{\geq 0})$ is a vector that also has value θ (with respect to the corresponding objective function for the symmetric formulation). Hence, SEQB can never be tighter than EQB.

It is possible to strengthen SEQB by the curtain inequalities. However, again one cannot obtain a lower bound that is tighter than EQB, since the curtain inequalities induce symmetric inequalities for the nonsymmetric problem that are already implied by the equations defining $\mathrm{aff}(\mathcal{QAP}_n)$ and by the nonnegativity of the y-variables.

Hence, do the curtain inequalities have any computational value at all? Potentially, they do. By changing (in case of a symmetric instance) from the nonsymmetric problem formulation to the symmetric one, the number of variables is approximately divided by two. This leads to easier linear programs on the one hand, but to a potentially weaker bound SEQB on the other hand. So the question is, Can the curtain inequalities improve (empirically) the bound SEQB significantly toward EQB without losing too much of the efficiency gain made by the transition?

We want to mention at this point that EQB has turned out to be a very good lower bound for the QAP. The theoretical basis for this is a result due to [13] and [1] (extending work of [9]) which shows that EQB is always at least as good as the classical *Gilmore–Lawler Bound*, proposed independently by [10] and [20]. The practical indication for the quality of EQB was given most extensively in a computational study by [22]. They solved the linear programs that give the EQB for all instances in the quadratic assignment problem library (QAPLIB) [7] of size not exceeding $n = 30$ and found that EQB turned out to be the best-known lower bound in most cases.

Besides the more or less negligible weakening of the bound, there is one more important drawback when dealing with the symmetric instead of the nonsymmetric

model. It has been observed and exploited by different people [1, 11] that the LP that has to be solved in order to compute EQB has a nice structure which allows one to design efficient heuristics to solve its dual, and thus to compute bounds that are nearly as good as EQB much faster than by evoking an LP-solver. Unfortunately, this nice structure is lost when changing to the symmetric model. However, once one starts to strengthen the bound by adding cutting planes (i.e., by exploiting polyhedral knowledge—see the remarks at the end of section 7), this structure is lost immediately, and the nonsymmetric model loses its advantage.

In order to investigate empirically the relative behavior of EQB, SEQB, and bounds obtained by adding some curtain inequalities to the (symmetric) formulation, we implemented a rudimentary cutting plane procedure for symmetric QAPs. This procedure initially solves the linear program that yields SEQB and afterward performs up to five cutting plane iterations with curtain inequalities. At each cutting plane iteration, we try to separate the current (fractional) solution by solving heuristically (i.e., repeating 100 times to guess a solution and improving it by a 2-opt procedure) a BQP for each ordered pair of rows/columns. If such a BQP ends with value greater than zero then we add the corresponding curtain inequality to the current linear program. This way, up to $2n(n-1)$ curtain inequalities may be added per iteration. It turned out that this naive (and fast) separation heuristic typically found many different violated inequalities per iteration.

The results show that in most cases, SEQB is not significantly worse than EQB. In fact, over all instances from the QAPLIB of sizes at most $n = 20$ the average ratio of SEQB and EQB is .986 (and the average ratio between the EQBs and the optimal solutions, known from the literature for all tested instances, is .859). Consequently, the curtain inequalities cannot improve SEQB very much. Usually, after five iterations the gap between SEQB and EQB is closed by about 30–40%. Regarding the quite small gaps between SEQB and EQB, the curtain inequalities do not seem to be computationally attractive. Therefore, we have not tried to improve the bounds by implementing an exact separation procedure for the curtain inequalities by the methods mentioned at the end of section 5.

The CPU times that are needed to compute SEQB are about three to four times smaller than the corresponding ones for EQB. They range from about 30 seconds for small ($n = 12$) instances up to about one hour for the hardest large ($n = 20$) instances. For more details on these experiments, we refer to [17].

7. Conclusion. We briefly discuss the context in which the work presented in this paper is located, in our opinion. Clearly, what we are finally concerned with is the exact (or at least provably good) solution of QAPs. The hope is that deeper polytopal knowledge of the problem will yield the necessary very good lower bounding procedures. Important steps that had already been performed were

- the evidence that EQB is empirically and theoretically a good lower bound,
- the basic polyhedral results on the QAP-polytope, and
- the definition of the SQAP-polytope.

The steps for the (quite natural) symmetric QAP that are done by the present paper are, from our point of view, the following.

- Our computational results indicate that changing the LP giving EQB in case of a symmetric instance in the natural way to a "symmetric LP" yielding SEQB does not decrease the quality of the lower bound significantly while accelerating the computations by a factor between three and four.
- It is useless to search for additional equations in order to improve the quality

of SEQB, since the used equation system is already complete.
- The curtain inequalities (strengthening inequalities proposed by [24] and [21]) seem to be computationally not very attractive (and they cannot be strengthened, since they already define facets).
- The methods presented in this paper, in particular the star-polytopes, provide possibilities for further investigations of the facial structure of the SQAP-polytope.

In particular, the last point in this list seems to be important. In fact, in the time between the submission of the first version and the preparation of the revised version of this paper, we have identified (using the techniques presented in this paper) a large class of facet-defining inequalities for the SQAP-polytope, the *box-inequalities*. They have turned out to be quite useful within cutting plane procedures for (symmetric) QAPs. Indeed, using these inequalities, it was for the first time possible to solve several instances from the QAPLIB to optimality by pure cutting plane algorithms, including three instances of size $n = 32$. We refer to [17, 15, 18] for details.

Acknowledgment. We thank the two referees for some valuable comments.

REFERENCES

[1] W. P. Adams and T. A. Johnson, *Improved linear programming-based lower bounds for the quadratic assignment problem*, in Quadratic Assignment and Related Problems, P. M. Pardalos and H. Wolkowicz, eds., DIMACS Ser. Discrete Math. Theoret. Comput. Sci. 16, 1994, pp. 43–75.

[2] W. P. Adams and H. D. Sherali, *A tight linearization and an algorithm for zero-one quadratic programming problems*, Management Sci., 32 (1986), pp. 1274–1290.

[3] M. Balinski, *On a selection problem*, Management Sci., 17 (1970), pp. 230–231.

[4] M. L. Balinski and A. Russakoff, *On the assignment polytope*, SIAM Rev., 16 (1974), pp. 516–525.

[5] A. I. Barvinok, *Combinatorial complexity of orbits in representations of the symmetric group*, Adv. Soviet Math., 9 (1992), pp. 161–182.

[6] N. W. Brixius and K. M. Anstreicher, *Solving Quadratic Assignment Problems Using Convex Quadratic Programming Relaxations*, tech. report, Dept. of Computer Science, University of Iowa, Iowa City, IA, 2000.

[7] R. Burkard, S. Karisch, and F. Rendl, *QAPLIB—A quadratic assignment problem library*, J. Global Optim., 10 (1997), pp. 391–403.

[8] J. Clausen and M. Perregaard, *Solving large quadratic assignment problems in parallel*, Comput. Optim. Appl., 8 (1997), pp. 111–127.

[9] A. M. Frieze and J. Yadegar, *On the quadratic assignment problem*, Discrete Appl. Math., 5 (1983), pp. 89–98.

[10] P. Gilmore, *Optimal and suboptimal algorithms for the quadratic assignment problem*, J. Soc. Indust. Appl. Math., 10 (1962), pp. 305–313.

[11] P. Hahn and T. Grant, *Lower bounds for the quadratic assignment problem based upon a dual formulation*, Oper. Res., 46 (1998), pp. 912–922.

[12] P. Hahn, T. Grant, and N. Hall, *A branch-and-bound algorithm for the quadratic assignment problem based on the Hungarian method*, Eur. J. Oper. Res., 108 (1998), pp. 629–640.

[13] T. Johnson, *New Linear-Programming Based Solution Procedures for the Quadratic Assignment Problem*, Ph.D. dissertation, Clemson University, Clemson, SC, 1992.

[14] M. Jünger and V. Kaibel, *A Basic Study of the QAP-Polytope*, tech. report 96.215, Angewandte Mathematik und Informatik, Universität zu Köln, Köln, Germany, 1996.

[15] M. Jünger and V. Kaibel, *Box-Inequalities for Quadratic Assignment Polytopes*, tech. report 97.285, Angewandte Mathematik und Informatik, Universität zu Köln, Köln, Germany, 1997. Math. Program., submitted.

[16] M. Jünger and V. Kaibel, *The QAP-Polytope and the Star-Transformation*, tech. report 97.284, Angewandte Mathematik und Informatik, Universität zu Köln, Köln, Germany, 1997. *Discrete Appl. Math.*, to appear.

[17] V. Kaibel, *Polyhedral Combinatorics of the Quadratic Assignment Polytope*, Ph.D. thesis, Universität zu Köln, Köln, Germany, 1997.

[18] V. Kaibel, *Polyhedral combinatorics of QAPs with less objects than locations*, R. E. Bixby, E. A. Boyd, and R. Z. Ríos-Mercado, eds., Lecture Notes in Comput. Sci. 1412, Springer-Verlag, Berlin, 1998, pp. 409–422.

[19] T. Koopmans and M. Beckmann, *Assignment problems and the location of economic activities*, Econometrica, 25 (1957), pp. 53–76.

[20] E. Lawler, *The quadratic assignment problem*, Management Sci., 9 (1963), pp. 586–599.

[21] M. Padberg and M. Rijal, *Location, Scheduling, Design and Integer Programming*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.

[22] M. Resende, K. Ramakrishnan, and Z. Drezner, *Computing lower bounds for the quadratic assignment problem with an interior point algorithm for linear programming*, Oper. Res., 43 (1995), pp. 781–791.

[23] J. Rhys, *A selection problem of shared fixed costs and networks*, Management Sci., 17 (1970), pp. 207–207.

[24] M. Rijal, *Scheduling, Design and Assignment Problems with Quadratic Costs*, Ph.D. dissertation, New York University, New York, 1995.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 464–494

MONOTONIC OPTIMIZATION: PROBLEMS AND SOLUTION APPROACHES*

HOANG TUY†

Abstract. Problems of maximizing or minimizing monotonic functions of n variables under monotonic constraints are discussed. A general framework for monotonic optimization is presented in which a key role is given to a property analogous to the separation property of convex sets. The approach is applicable to a wide class of optimization problems, including optimization problems dealing with functions representable as differences of increasing functions (d.i. functions).

Key words. monotonicity, global optimization, increasing functions, normal sets, polyblock approximation algorithms, difference of increasing functions, multiplicative programming, polynomial programming, nonconvex quadratic programming, distance geometry

AMS subject classifications. 90C26, 65K05, 90C20, 90C30, 90C56, 78M50

PII. S1052623499359828

1. Introduction.

One of the most active current research areas in global optimization is concerned with solution methods for specially structured nonconvex problems arising from applications. Despite the inherent difficulty of most of these problems, significant progress has been achieved in recent years in the development of special solution strategies adapted to special mathematical structures.

By their very nature, many functions encountered in mathematical modeling of real world systems in a broad range of activities, including economics and engineering, exhibit monotonicity with respect to some variables (partial monotonicity) or to all variables (total monotonicity). The analysis of monotonicity properties has led to the formulation of a number of "monotonicity principles" that in several cases have enabled the authors to reduce originally difficult problems to a form amenable to effective solution methods. Quite a few works (e.g., [6], [20], [15], [19], [21], [36], [2], [3], [8], [9]) have demonstrated the usefulness of monotonicity principles in the approach to optimal design problems.

Close scrutiny also reveals hidden monotonicity in the structure of many nonconvex global optimization problems. Attempts to exploit mathematical structure for numerical purposes have been particularly successful when monotonicity is coupled with convexity or complementary convexity as happens in the class of so-called *low rank nonconvex problems* [14]. In fact, parametric methods and other duality-based decomposition approaches developed in recent years (see [12], [14], [27], [32], [30], [28]) have proved to be quite efficient tools in the study of these problems.

There is, however, a wide class of important problems with a monotonic structure not necessarily coupled with convexity or complementary convexity. These are problems described by means of functions which are monotone nondecreasing on every half line $\{a + \lambda u |\ \lambda \geq 0\}$ with $a, u \in R^n_+$. (Following some authors, e.g., [1], [23], [24], these functions will be called *increasing* in what follows.) To understand the potential advantage offered by such a monotonic structure one should bear in mind that, since local information is generally insufficient for verifying the global optimality of a given feasible solution, the search for the global optimum has to be carried

*Received by the editors August 4, 1999; accepted for publication (in revised form) April 18, 2000; published electronically October 18, 2000.
http://www.siam.org/journals/siopt/11-2/35982.html
†Institute of Mathematics, P.O. Box 631, Bo Ho, Hanoi, Vietnam (htuy@hn.vnn.vn).

out, generally speaking, on the entire feasible set. If, however, the objective function is increasing, then, once a feasible point z is known one can ignore the whole cone $z + R^n_+$ because no better feasible solution can be found in this set (for minimization problems). Analogously, if the function $g(x)$ in a constraint like $g(x) \leq 1$ is increasing, then once a point z is known to be infeasible to this constraint, the whole cone $z + R^n_+$ can be discarded from further consideration. Such information is very useful and may sometimes help to simplify the problem drastically by limiting the global search to a much restricted region of the feasible domain. The difficulty that remains, of course, is how to implement this idea computationally, i.e., how to incorporate this idea in suitable solution procedures.

The aim of the present paper is to develop a mathematical framework for solving optimization problems dealing with increasing functions, and more generally, functions representable as differences of increasing functions (d.i. functions). It turns out that to implement the above idea one can exploit a property of level sets of increasing functions which is analogous to, but not quite the same as, the classical separation property of convex sets. Specifically, if a function $g(x)$ is quasi-convex on R^n_+, then it is well known that its level set $G := \{x \in R^n_+ |\ g(x) \leq 1\}$ is convex and any point z outside the closure of G can be strictly separated from it by a half-space. As a result, G can be approximated, as closely as desired, by a nested sequence of polyhedrons. It is this property that lies in the foundation of various variants of polyhedral outer approximation methods for maximizing a quasi-convex function over a convex set. Now, if $g(x)$ is increasing, then any point z outside the closure of G can be separated from G not by a half-space but by a cone which is a translate of the nonnegative orthant. Interestingly enough, from this it follows that a normal set G (i.e., roughly speaking, the level set of an increasing function) can be approximated by a nested sequence of sets of a particular kind, which we will call "polyblocks." Just as the approximation of convex sets by polyhedrons is the basis of polyhedral outer approximation methods for quasi-convex maximization, the approximation of level sets of increasing functions by polyblocks can be used to devise specific "polyblock approximation methods" for monotonic optimization.

An early variant of this approach was first proposed in [24] for maximizing an increasing function over the level set of another increasing function, and was subsequently improved and applied in [33] to solve convex programs with an additional monotonic constraint. Computational experiments reported in [24] and [33] have shown the efficiency of the proposed method, at least for the class of problems studied in these papers, including mathematical programs with multiplicative constraints. (For the latter it seems even to outperform most existing methods.) The general framework for monotonic optimization to be presented in what follows can be considered as a further development and extension of many basic ideas and results in [24] and [33].

As was mentioned in [24], the idea of approximating a certain set by a union of hyperrectangles was already put forward in [13], though in an implicit form. (The set considered in that paper is actually "normal" and a union of hyperrectangles is a "polyblock" in our terminology.) However, the method in [13] was confined to a special problem and no attempt was made to obtain general results. Also, it should be noticed that the separation property of normal sets has been noticed and utilized in some previous works for the development of analytical tools for monotonic analysis, in the context of abstract convex analysis ([23], [22], and references therein). The approach we will take here is more geometrical and directly oriented toward

computational applications.

The paper is organized as follows. After the introduction, section 2 will present the concepts of increasing functions and normal sets. Section 3 will describe the basic problems of monotonic optimization and give some typical examples. Next, section 4 will introduce and study the concept of polyblock and the approximation of normal sets by polyblocks. Sections 5, 6, and 7 will present the polyblock and reverse polyblock approximation methods and their extension to the most general problem of d.i. optimization. Finally, to illustrate the usefulness of the proposed approach, section 8 will discuss some possible applications to various difficult global optimization problems of current interest and section 9 will present some computational results.

2. Increasing functions and normal sets. Increasing functions and normal sets were first introduced in mathematical economics [17] for the modeling and analysis of production activities within an economic system. Recently they were discussed in [24], [33], and especially in [29] as important tools for studying monotonic optimization problems. In this section we review the basic concepts and results on increasing functions and normal sets which will be needed later. We will omit most of the proofs, which are almost straightforward.

Throughout the following, borrowing the terminology from multicriteria optimization (see, e.g., [37] and references therein), for any two vectors $x', x \in R^n$ we write $x' \geq x$ and say that x' *dominates* x if $x'_i \geq x_i \ \forall i = 1, \ldots, n$. We write $x' > x$ and say that x' *strictly dominates* x if $x'_i > x_i \ \forall i = 1, \ldots, n$. Let $R^n_+ = \{x \in R^n |\ x \geq 0\}$ and $R^n_{++} = \{x \in R^n |\ x > 0\}$. For $x \in R^n_+$ let $I(x) = \{i |\ x_i = 0\}$ and denote

$$K_x = \{x' \in R^n_+ |\ x'_i > x_i \ \forall i \notin I(x)\}, \qquad \mathrm{cl}K_x = \{x' \in R^n_+ |\ x' \geq x\}. \tag{1}$$

If $a \leq b$, we define the box (hyperrectangle) $[a, b]$ to be the set of all x such that $a \leq x \leq b$. We also write $(a, b] := \{x |\ a < x \leq b\}, [a, b) := \{x |\ a \leq x < b\}$. As usual e is the vector of all ones and e^i the ith unit vector of the space under consideration.

A function $f : R^n \to R$ is said to be *increasing* on R^n_+ if $f(x) \leq f(x')$ whenever $0 \leq x \leq x'$; it is said to be increasing on a box $[a, b] \subset R^n_+$ if $f(x) \leq f(x')$ whenever $a \leq x \leq x' \leq b$. Many functions encountered in various applications are increasing in this sense. Outstanding examples are the production functions and the utility functions in mathematical economics (under the assumption that all goods are useful), polynomials (in particular quadratic functions) with nonnegative coefficients, and posynomials

$$\sum_{j=1}^{m} c_j \prod_{i=1}^{n} (x_i)^{a_{ij}} \qquad \text{with } c_j \geq 0 \text{ and } a_{ij} \geq 0$$

(e.g., Cobb–Douglas function $f(x) = \prod_i x_i^{\alpha_i},\ \alpha_i \geq 0$).

The following obvious proposition shows that the class of increasing functions includes a large variety of functions.

PROPOSITION 1. (i) *If f_1, f_2 are increasing functions, then for any nonnegative numbers λ_1, λ_2 the function $\lambda_1 f_1 + \lambda_2 f_2$ is increasing.*

(ii) *The pointwise supremum of a bounded above family $(f_\alpha)_{\alpha \in A}$ of increasing functions and the pointwise infimum of a bounded below family $(f_\alpha)_{\alpha \in A}$ of increasing functions are increasing.*

Other nontrivial examples of increasing functions are functions of the form $f(x) = \sup_{y \in a(x)} g(y)$, where $g : R^n_+ \to R$ is an arbitrary function and $a : R^n_+ \to R^n_+$ is a set-valued mapping with bounded images such that $a(x') \supset a(x)$ for $x' \geq x$.

A set $G \subset R^n_+$ is called *normal* if for any two points $x, x' \in R^n_+$ such that $x' \leq x$, if $x \in G$, then $x' \in G$, too. The empty set, the singleton $\{0\}$, and R^n_+ are special normal sets which we will refer to as *trivial subsets of* R^n_+. If G is a normal set, then $G \cup \{x \in R^n_+ |\ x_i = 0 \text{ for some } i = 1, \ldots, n\}$ is still normal.

PROPOSITION 2. *Given any set $D \subset R^n_+$ the set $N[D] := (D - R^n_+) \cap R^n_+$ is the smallest normal set containing D. If D is compact, then so is $N[D]$.*

Clearly $N[D]$ is also the intersection of all normal sets that contain D. It is called the *normal hull* of D.

PROPOSITION 3. *The intersection and the union of a family of normal sets are normal sets.*

PROPOSITION 4. *Every normal set is connected. A normal set G has a nonempty interior if and only if it contains a point $u \in R^n_{++}$. The closure of a normal set is normal.*

The next proposition shows that a bounded normal set is essentially the lower level set of an increasing function.

PROPOSITION 5. *For any increasing function $g(x)$ on R^n_+ the set $G = \{x \in R^n_+ |\ g(x) \leq 1\}$ is normal and it is closed if $g(x)$ is lower semicontinuous. Conversely, for any closed normal set $G \subset R^n_+$ with nonempty interior there exists a lower semicontinuous increasing function $g : R^n_+ \to R_+$ such that $G = \{x \in R^n_+ |\ g(x) \leq 1\}$.*

Proof. We prove only the second assertion. Let G be a closed normal set with nonempty interior. For every $x \in R^n_+$ define $g(x) = \inf\{\lambda > 0 |\ x \in \lambda G\}$. From the assumption $\text{int}G \neq \emptyset$ there is $u > 0$ such that $[0, u] \subset G$ (Proposition 4). Then the half line $\{\alpha x |\ \alpha \geq 0\}$ intersects $(0, u] \subset G$; hence $0 \leq g(x) < +\infty$. Since for every $\lambda > 0$ the set λG is normal, if $x \leq x' \in \lambda G$, then $x \in \lambda G$, too, so $g(x') \geq g(x)$, i.e., $g(x)$ is increasing. We show that $G = \{x \in R^n_+ |\ g(x) \leq 1\}$. In fact, if $x \in G$, then obviously $g(x) \leq 1$. Conversely, if $x \notin G$, then since G is closed there exists $\alpha > 1$ such that $x \notin \alpha G$; hence, since G is normal, $x \notin \lambda G\ \forall\ \lambda \leq \alpha$, which means that $g(x) \geq \alpha > 1$. Consequently, $x \in G$ if and only if $x \in R^n_+$ and $g(x) \leq 1$, i.e., $G = \{x \in R^n_+ |\ g(x) \leq 1\}$. It remains to prove the lower semicontinuity of $g(x)$. Let $\{x^k\} \subset R^n_+$ be a sequence such that $x^k \to x^0$ and $g(x^k) \leq \alpha\ \forall k$. Then for any given $\alpha' > \alpha$ we have $\inf\{\lambda |\ x^k \in \lambda G\} < \alpha'\}\ \forall k$, i.e., $x^k \in \alpha' G\ \forall k$; hence $x^0 \in \alpha' G$ in view of the closedness of the set $\alpha' G$. This implies that $g(x^0) \leq \alpha'$ and since α' can be taken arbitrarily near to α, we must have $g(x^0) \leq \alpha$. Therefore, the set $\{x \in R^n_+ |\ g(x) \leq \alpha\}$ is closed, as was to be proved. □

A point $y \in R^n_+$ is called an *upper boundary point* of a bounded normal set G if $y \in \text{cl}G$ while $K_y \subset R^n_+ \setminus G$. The set of upper boundary points of G is called the *upper boundary* of G and is denoted by $\partial^+ G$. If G is closed, then obviously $\partial^+ G \subset G$.

PROPOSITION 6. *Let $G \subset [0, b]$ be a compact normal set with nonempty interior. For every point $z \in R^n_+ \setminus \{0\}$ the half line from 0 through z meets $\partial^+ G$ at a unique point $\pi_G(z)$, which is defined by*

$$\pi_G(z) = \lambda z, \qquad \lambda = \max\{\alpha > 0 |\ \alpha z \in G\}. \tag{2}$$

Proof. Since $\text{int}G \neq \emptyset$ there exists $\alpha > 0$ such that $\alpha z \in G$. The compactness of G then implies that the number λ defined by (2) satisfies $0 < \lambda < +\infty$ and that $y := \lambda z \in G$. Furthermore, if $x \in K_y$, i.e., $x_i > y_i\ \forall i \notin I(y)$ while $x_i \geq y_i = 0\ \forall i \in I(y)$, then, since for any $z' = \alpha z$ with $\alpha > \lambda$ one has $z'_i = \alpha z_i > y_i\ \forall i \notin I(y)$ and $z'_i = 0\ \forall i \in I(y)$, there exists $\alpha > \lambda$ such that $z' = \alpha z$ satisfies $y \leq z' \leq x$; hence, by normality of G, $z' \in G$, conflicting with the definition of λ. Therefore, $K_y \cap G = \emptyset$, and so $y \in \partial^+ G$. If $y' = \lambda' z$ with $\lambda' > \lambda$, then $y' \in K_y$, hence $y' \notin G$. This means

that if $y' = \lambda' z \in \partial^+ G$, then necessarily $\lambda' \leq \lambda$. By interchanging the roles of y and y' one must also have $\lambda \leq \lambda'$; hence $y = \lambda z$ is the unique intersection point of $\partial^+ G$ with the half line from 0 through z. □

A set $H \subset R^n_+$ is called *reverse normal* if $x' \geq x \in H$ implies $x' \in H$. It is said to be *reverse normal* in a box $[0, b]$ if $b \geq x' \geq x \geq 0,\ x \in H$, implies $x' \in H$ or, equivalently, if $x \notin H$ whenever $0 \leq x \leq x' \leq b,\ x' \notin H$. Clearly, a set H is reverse normal if and only if the set $H^\flat = R^n_+ \setminus H$ is normal.

For any set $D \subset R^n_+$ the set $D + R^n_+$ is obviously the smallest reverse normal set containing D. We call it the *reverse normal hull* of D and denote it by $rN[D]$.

The following propositions are analogous to Propositions 5 and 6 and can be proved by similar arguments:

(i) For an increasing function $h(x)$ on R^n_+ the set $H = \{x \in R^n_+ |\ h(x) \geq 1\}$ is reverse normal and it is closed if $h(x)$ is upper semicontinuous. Conversely, for any closed reverse normal set H such that $R^n_+ \setminus H$ has a nonempty interior there exists an upper semicontinuous increasing function such that $H = \{x \in R^n_+ |\ h(x) \geq 1\}$.

A point $y \in R^n_+$ is called a *lower boundary point* of a reverse normal set H if $y \in \mathrm{cl}H$ and $x \notin H\ \forall x < y$. The set of lower boundary points of H is called the *lower boundary* of H and is denoted by $\partial^- H$. If H is closed, then obviously $\partial^- H \subset H$.

(ii) Let H be a closed reverse normal set and $b \in \mathrm{int}H$. For every point $z \in [0, b] \setminus H$ the half line from b through z meets $\partial^- H$ at a unique point $\rho_H(z)$, which is defined by

$$\rho_H(z) = b + \mu(z - b), \qquad \mu = \max\{\alpha > 0 |\ b + \alpha(z - b) \in H\}. \tag{3}$$

3. Basic problems of monotonicoptimization. Many optimization problems encountered in applications can be formulated as the maximization or minimization of an increasing function over an intersection of normal and reverse normal sets. We shall refer to this class of problems as monotonic optimization problems. In this section we state two basic problems of monotonic optimization and give some typical examples.

3.1. Maximizing an increasing function. Consider the problem

$$\max\{f(x) |\quad x \in G \cap H\}, \tag{A}$$

where $G \subset [0, b] \subset R^n_+$ is a compact normal set with nonempty interior, H is a closed reverse normal set, and $f(x)$ is an increasing function on $[0, b]$.

PROPOSITION 7. *The maximum of $f(x)$ over $G \cap H$, if it exists, is attained on $\partial^+ G \cap H$.*

Proof. Since $\mathrm{int}G \neq \emptyset$, if $G \cap H \neq \emptyset$, then necessarily $(G \cap H) \setminus \{0\} \neq \emptyset$, and hence, since $f(x)$ is increasing, if $f(x)$ attains a maximum on $G \cap H$, there exists a maximizer $z \neq 0$. By Proposition 6 the half line from 0 through z meets $\partial^+ G$ at some point $y = \pi_G(z)$. Since $z \in G$ we must have $z \leq y$, and this in turn implies that $y \in H$ because $z \in H$. Therefore, $y \in \partial^+ G \cap H$. On the other hand, since $f(x)$ is increasing and $y \geq z$, it follows that $f(y) \geq f(z)$, i.e., y is a maximizer of $f(x)$ over $G \cap H$. □

Often the sets G and H are defined through increasing functions $g_i(x), h_j(x)$:

$$G = \{x \in R^n_+ |\ g_i(x) \leq 1, i = 1, \ldots, m_1\},$$
$$H = \{x \in R^n_+ |\ h_j(x) \geq 1, j = m_1 + 1, \ldots, m\}.$$

Setting $g(x) = \max\{g_1(x), \ldots, g_{m_1}(x)\},\ h(x) = \min\{h_{m_1+1}(x), \ldots, h_m(x)\}$, we can write

$$G = \{x \in R^n_+ |\ g(x) \leq 1\}, \quad H = \{x \in R^n_+ | h(x) \geq 1\}. \tag{4}$$

From Proposition 7 we easily see the following.

COROLLARY 1. *If the sets G, H are given by (4), where $g(x)$ and $h(x)$ are continuous increasing functions such that $g(0) < 1$, then problem* (A) *is equivalent to*

$$\max\{f(x)| \quad g(x) = 1,\ h(x) \geq 1,\ x \geq 0\}.$$

Heuristically $f(x)$ may be a profit, $h(x)$ some utility which has to be achieved at a certain required level, while $g(x) \leq 1$ may express limitation on some scarce resources.

PROPOSITION 8. *If D is an arbitrary compact set in R^n_+, then the problem* $\max\{f(x)|\ x \in D\}$ *(where $f(x)$ is increasing) is equivalent to* $\max\{f(x)|\ x \in N[D]\}$ *where, as was already defined, $N[D]$ is the normal hull of D.*

Proof. Let z be a maximizer of $f(x)$ on D. Then for every $x \in D:\ f(z) \geq f(x)$, hence $f(z) \geq f(x')\ \forall x' \in (x - R^n_+) \cap R^n_+$. That is, $f(z) \geq f(x)\ \forall x \in \mathrm{N}[D]$. The converse is obvious since $D \subset \mathrm{N}[D]$. □

Example 1.

$$\max\{\varphi(u(x))| \quad x \in D\}, \tag{5}$$

where $D \subset R^n$ is a nonempty compact convex set, $\varphi : R^m_+ \to R$ is an increasing function, $u(x) = (u_1(x), \ldots, u_m(x)),\ u_i : D \to R_+$ being nonnegative-valued continuous functions on D. By Proposition 8, this problem can be written as $\max\{\varphi(y)|\ y \in u(D)\} = \max\{\varphi(y)|\ y \in N[u(D)]\}$, i.e.,

$$\max\{\varphi(y)| \quad y \in G\},$$

where $G := N[u(D)] = \{y \in R^m_+|\ y \leq u(x),\ x \in D\}$. This is of course a problem (A), with $H = R^m_+$ ($u(D)$ is compact by the continuity of $u(x)$, so $N[u(D)]$ is contained in some box $[0, b]$). Furthermore, without loss of generality we can assume

$$\max_{x \in D} u_i(x) > 0 \quad \forall i = 1, \ldots, m. \tag{6}$$

Indeed, if, e.g., $u_i(x) = 0 \quad \forall x \in D$ and $\forall i > m_1$, then the problem reduces to

$$\max\{\varphi(u_1(x), \ldots, u_{m_1}(x), 0, \ldots, 0)| \quad x \in D\}$$

which is a problem of the same type as (5) but with $\varphi : R^{m_1}_+ \to R$ and such that

$$\max_{x \in D} u_i(x) > 0 \quad \forall i = 1, \ldots, m_1.$$

Under assumption (6) it is easily checked that there is a $y \in G \cap R^n_{++}$, i.e., $\mathrm{int}G \neq \emptyset$. Also for every $z \in (0, b] \setminus G$ the point $\pi_G(z) = \lambda z$ defined by (2) can be determined by computing $\lambda = \max\{\alpha | \alpha z \leq u(x), x \in D\}$, which is not difficult if $u_1(x), i = 1, \ldots, u_m(x)$ are concave or convex.

Example 2.

$$\max\{\langle c, x\rangle| \quad x \in D,\ \varphi(u(x)) \leq 1\}, \tag{7}$$

where D, φ, and $u(x)$ are as previously. Observe that the set

$$H = \{y \in R^m_+|\ u(x) \leq y \text{ for some } x \in D\}$$

is closed and reverse normal, since $H = u(D) + R^m_+ = rN[u(D)]$. Define

$$\theta(y) = \begin{cases} \sup\{\langle c, x\rangle | \quad x \in D,\ u(x) \le y\} & \text{if } y \in H \\ -M & \text{otherwise,} \end{cases} \tag{8}$$

where $M > 0$ is an arbitrary number such that $-M < \min\{\langle c, x\rangle |\ x \in D\}$. Since D is nonempty compact, clearly $-\infty < \theta(y) < +\infty\ \forall y \in R^m_+$.

PROPOSITION 9. *The function $\theta(y)$ is increasing and upper semicontiuous on R^m_+. If $u_1(x), \dots, u_m(x)$ are convex then $\theta(y)$ is concave on the convex set $H = u(D) + R^m_+$.*

Proof. If $y \le y'$ and $y \notin H$, then $\theta(y) = -M$ while $\theta(y') \ge -M = f(y)$. But if $y \le y'$ and $y \in H$, then $\emptyset \ne \{x \in D |\ u(x) \le y\} \subset \{x \in D |\ u(x) \le y'\}$, and hence $\theta(y) \le \theta(y')$. Therefore $\theta(y)$ is increasing. We now show the upper semicontinuity of $\theta(y)$. Since H is closed and $\theta(y) = -M\ \forall y \notin H$, it suffices to show the upper semicontinuity of $\theta(y)$ on H. Let $y^k \to y^0$ (where $y^k \in H$) and for each k let x^k be such that $x^k \in D,\ u(x^k) \le y^k,\ \langle c, x^k\rangle = \theta(y^k)$. Since D is compact and $u(x)$ is continuous we can assume $x^k \to x^0 \in D, u(x^0) \le y^0$. Then $\theta(y^0) \ge \langle c, x^0\rangle = \lim_k \langle c, x^k\rangle = \lim_k \theta(y^k)$, as desired. Finally, if every function $u_1, \dots, u_m$ is convex and $\theta(y^1) = \langle c, x^1\rangle,\ \theta(y^2) = \langle c, x^2\rangle$ where $x^i \in D,\ u(x^i) \le y^i,\ i = 1, 2$, then for any $\alpha \in (0, 1)$ we have $x := \alpha x^1 + (1-\alpha)x^2 \in D$ and $u(x) \le \alpha u(x^1) + (1-\alpha)u(x^2) \le y^1 + (1-\alpha)y^2 = y$; hence $\theta(\alpha y^1 + (1-\alpha)y^2) \ge \langle c, \alpha x^1 + (1-\alpha)x^2\rangle = \alpha\theta(y^1) + (1-\alpha)\theta(y^2)$, proving the concavity of $\theta(y)$ on $H = u(D) + R^m_+$. □

PROPOSITION 10. *Problem* (7) *is equivalent to*

$$\max\{\theta(y) | \quad \varphi(y) \le 1,\ y \in H\} \tag{9}$$

in the sense that if $\bar{x}$ solves (7)*, then $\bar{y} = u(\bar{x})$ solves* (9)*, and conversely, if $\bar{y}$ solves* (9) *and $\theta(\bar{y}) = \langle c, \bar{x}\rangle$ for an optimal solution $\bar{x}$ of* (8) *(where $y = \bar{y}$), then $\bar{x}$ solves* (7)*.*

Proof. Let $\bar{x}$ solve (7) and $\bar{y} = u(\bar{x})$. Then $\varphi(\bar{y}) \le 1,\ \bar{y} \in H$. Furthermore, for every $y \in H$ such that $\varphi(y) \le 1$ we have $\theta(y) = \langle c, x\rangle$ for some $x \in D$ satisfying $u(x) \le y$; hence $\varphi(u(x)) \le 1$. Therefore $\theta(y) \le \langle c, \bar{x}\rangle$, so $\bar{y}$ solves (9). Conversely, let $\bar{y}$ solve (9) and $\theta(\bar{y}) = \langle c, \bar{x}\rangle$ for an optimal solution $\bar{x}$ of (8). Then for every $x \in D$ such that $\varphi(u(x)) \le 1$ we have for $y = u(x) : \varphi(y) \le 1, y \in H$, hence, on the one hand, $\theta(y) \le \theta(\bar{y}) = \langle c, \bar{x}\rangle$, and on the other hand, $\langle c, x\rangle \le \theta(y)$, hence $\langle c, x\rangle \le \langle c, \bar{x}\rangle$, so $\bar{x}$ solves (9). □

Again (9) is a problem (A) in R^m, with $G = \{y \in R^m_+ |\ \varphi(y) \le 1\}$. Note that if $u_i(x), i = 1 \dots, m$, are convex then $\theta(y)$ is the optimal value in a convex program.

3.2. Minimizing an increasing function. Consider the problem

$$\min\{f(x) | \quad x \in G \cap H\}, \tag{B}$$

where $G \subset [0, b] \subset R^n_+$ is a compact normal set, H is a closed reverse normal set with $b \in \text{int}H$, and $f(x)$ is an increasing function on $[0, b]$. By setting $x = b - y$ and

$$\tilde{f}(y) = -f(b - y), \quad \tilde{G} = b - G, \quad \tilde{H} = b - H,$$

we convert this problem to a problem (A), namely,

$$\max\{\tilde{f}(y) | \quad y \in \tilde{G} \cap \tilde{H}\},$$

where $\tilde{H} \subset [0, b]$ is a compact normal set with nonempty interior, $\tilde{G}$ is a closed reverse normal set in $[0, b]$ and $\tilde{f}(y)$ is increasing on $[0, b]$. Therefore we have the following.

PROPOSITION 11. *The minimum of $f(x)$ over $G \cap H$, if it exists, is attained on $G \cap \partial^- H$.*

COROLLARY 2. *If the sets G, H are given by (4), where $g(x)$ and $h(x)$ are continuous increasing functions, such that $h(0) < 1$, then problem* (B) *is equivalent to*

$$\min\{f(x) | \quad g(x) \leq 1,\ h(x) = 1,\ x \geq 0\}.$$

Heuristically, $f(x)$ may be a cost, $h(x) \geq 1$ expresses a constraint on the required minimum utility level, while $g(x) \leq 1$ expresses a limitation of some scarce resource.

Analogously to Proposition 8, it is easily seen that for any set $E \subset R^n_+$

$$\min\{f(x) |\ x \in E\} = \min\{f(x) | \quad x \in rN[E]\}.$$

Example 3.

$$\min\{\varphi(u(x)) | \quad x \in D\}, \tag{10}$$

where $D, \varphi, u(x)$ are as previously. This problem can be written as

$$\min\{\varphi(y) | \quad y \in u(D)\} = \min\{\varphi(y) | \quad y \in rN[u(D)]\},$$

or, equivalently,

$$\min\{\varphi(y) | \quad y \in H\}$$

with $H := rN[u(D)] = \{y \in [0, b] |\ x \in D,\ y \geq u(x)\}$, so this is a problem (B) where $G = [0, b]$. The reverse normal set H is closed in view of the compactness of D and the continuity of $u(x)$.

Example 4.

$$\min\{\langle c, x\rangle | \quad x \in D,\ \varphi(u(x)) \geq 1\} \tag{11}$$

with D, φ, h as previously. Observe that the set

$$G = \{y \in R^m_+ |\ y \leq u(x) \text{ for some } x \in D\}$$

is normal and compact, since $G = R^m_+ \cap (u(D) - R^m_+) = N[u(D)]$. Define

$$\theta(y) = \begin{cases} \min\{\langle c, x\rangle | \quad x \in D,\ y \leq u(x)\} & \text{if } y \in G, \\ M & \text{otherwise,} \end{cases} \tag{12}$$

where $M > 0$ is an arbitrary number satisfying $M > \max\{\langle c, x\rangle |\ x \in D\}$. Since D is nonempty and compact, clearly $-\infty < \theta(y) < +\infty\ \forall y \in R^m_+$ and it can easily be verified that the function $\theta(y)$ is lower semicontinuous and increasing (proof analogous to that of Proposition 9). Also $\theta(y) < M \Leftrightarrow y \in G$.

PROPOSITION 12. *Problem* (11) *is equivalent to*

$$\min\{\theta(y) | \quad \varphi(y) \geq 1,\ y \in G\} \tag{13}$$

in the sense that if $\bar{x}$ solves (11)*, then $\bar{y} = u(\bar{x})$ solves* (13) *and conversely, if $\bar{y}$ solves* (13) *and $\theta(\bar{y}) = \langle c, \bar{x}\rangle$ for an optimal solution $\bar{x}$ of* (12) *(where $y = \bar{y}$), then $\bar{x}$ solves* (11)*.*

Proof. This proof is analogous to the proof of Proposition 10. □

Thus, (11) appears to be a problem (B) in R^m, with $H = \{y \in R^m_+ |\ \varphi(y) \geq 1\}$. If $u_i(x), i = 1 \ldots, m$, are concave, then $\theta(y)$, for $y \in G$, is the optimal value in a convex program.

4. Polyblocks and approximation of normal sets. The approach we propose for the solution of monotonic optimization problems is based on the approximation of compact normal sets by simpler normal sets, called polyblocks, which were first introduced in [29].

A set $P \subset R^n_+$ is called a *polyblock* in $[a,b] \subset R^n_+$ if it is the union of a finite number of boxes $[a,z],\ z \in T \subset [a,b]$ ($|T| < +\infty$). The set T is called the *vertex set* of the polyblock; we also say that the polyblock P is generated by T. A vertex $z \in T$ is said to be *proper* if it is not dominated by any other $z' \in T$, i.e., if $z \notin [0,z']\ \forall z' \in T \setminus \{z\}$. A point $z \in T$ which is not a proper vertex is also called an *improper* element of T. Of course a polyblock is fully determined by its proper vertices. Furthermore, an increasing function $f(x)$ achieves its maximum over a polyblock at a proper vertex.

PROPOSITION 13. *Any polyblock is closed and normal. The intersection of finitely many polyblocks is a polyblock.*

Proof. The first assertion follows from the fact that any box $[a,z] \subset R^n_+$ is closed normal and the union of a finite family of closed normal sets is closed normal. The second assertion follows from the equalities $(\cup_i A_i) \cap (\cup_j B_j) = \cup_{i,j}(A_i \cap B_j)$ and $[a,p] \cap [a,q] = [a,u]$ with $u_i = \min\{p_i, q_i\}$. □

PROPOSITION 14. *Let $G \subset [0,b]$ be a compact normal set. For any $z \in [0,b] \setminus G$ there exists a point x such that $z \in K_x$ and $K_x \subset R^n_+ \setminus G$.*

Proof. Since $0 \in G$ while $z \notin G$, by Proposition 6 the half line from 0 through z meets $\partial^+ G$ at $x = \pi_G(z)$. Then $z \in K_x$ and $K_x \cap G = \emptyset$. □

PROPOSITION 15. *If $\bar{x} \in [0,b]$ and $\bar{z} \in [0,b] \cap K_{\bar{x}}$, then $P = [0,\bar{z}] \setminus K_{\bar{x}}$ is a polyblock in $[0,b]$ with vertices*

$$z^i = \bar{z} - (\bar{z}_i - \bar{x}_i)e^i, \qquad i \notin I(\bar{x}).$$

Proof. Let $K_i = \{x \in R^n_+ |\ \bar{x}_i < x_i\}$. Since $K_{\bar{x}} = \cap_{i \notin I(\bar{x})} K_i$, we have $P = [0,\bar{z}] \setminus K_{\bar{x}} = \cup_{i \notin I(\bar{x})}([0,\bar{z}] \setminus K_i)$. But

$$[0,\bar{z}] \setminus K_i = \{x |\ 0 \le x_i \le \bar{x}_i,\ 0 \le x_j \le \bar{z}_j\ \forall j \ne i\} = [0, z^i],$$

where z^i denotes the vector such that $z^i_j = \bar{z}_j\ \forall j \ne i,\ z^i_i = \bar{x}_i$, that is, $z^i = \bar{z} - (\bar{z}_i - \bar{x}_i)e^i$. □

PROPOSITION 16. *Let G be a compact set contained in a box $[0,b] \subset R^n_+$. Then the following assertions are equivalent:*

(i) *G is normal.*

(ii) *For any point $z \in G^\flat := [0,b] \setminus G$ there exists a polyblock in $[0,b]$ separating z from G (i.e., containing G but not z).*

(iii) *G is the intersection of a family of polyblocks in $[0,b]$.*

Proof. (i)⇒ (ii). By Proposition 14 if $z \in G^\flat$, then there is x such that $z \in K_x$ while $K_x \cap G = \emptyset$, i.e., $P := [0,b] \setminus K_x$ contains G but not z. On the other hand, by Proposition 15, P is a polyblock.

(ii)⇒ (iii) Let E be the intersection of all polyblocks containing G. Clearly $G \subset E$. If (ii) holds, then for any $z \in E \setminus G$ there is a polyblock containing G but not z, so $E \subset G$.

(iii)⇒ (i) Obvious by Proposition 3 because any polyblock is closed and normal. □

It follows from the above that normal sets and increasing functions are analogous to convex sets and convex functions in several respects pertinent to optimization theory. Namely, a compact normal (convex, resp.) set can be approximated as closely

as desired by a polyblock (polytope). Furthermore, *the maximum of an increasing* (convex, resp.) *function over a polyblock* (polytope) *is attained at a vertex of the polyblock* (polytope). This suggests a conceptual scheme for maximizing an increasing function over a compact set similar to the standard outer approximation scheme for convex maximization [30]. Specifically, to solve the problem

$$\max\{f(x)|\ x \in \Omega\},$$

where Ω is a compact set in R^n_+ and $f(x)$ is an increasing function, we attempt to construct a nested sequence of polyblocks outer approximating $\Omega : P_1 \supset P_2 \supset \cdots \supset \Omega$ in such a way that

$$\max\{f(x)|\ x \in P_k\} \searrow \max\{f(x)|\ x \in \Omega\}. \tag{14}$$

At iteration k, the point

$$z^k \in \operatorname{argmax}\{f(z)|\ z \in T_k\},$$

where T_k is the proper vertex set of P_k, is a maximizer of $f(x)$ over P_k. If $z^k \in \Omega$, z^k solves the problem. Otherwise, we find a polyblock P_{k+1} contained in $P_k \setminus \{z^k\}$ but still containing Ω, and we continue the procedure.

PROPOSITION 17. *Let $x^k = \pi(z^k)$. Then the polyblock P_{k+1} obtained from P_k by replacing $[0, z^k]$ with $[0, z^k] \setminus K_{x^k}$, i.e., $P_{k+1} = ([0, z^k] \setminus K_{x^k}) \cup_{z \in T_k \setminus \{z^k\}} [0, z]$, satisfies $\Omega \subset P_{k+1} \subset P_k \setminus \{z^k\}$.*

Proof. Since z^k is a proper vertex of P_k, it is not contained in any box $[0, z], z \in T_k \setminus \{z^k\}$, hence $z^k \notin P_{k+1}$. □

Note that $I(x^k) = I(z^k)$, so by Proposition 15 the above defined polyblock P_{k+1} is generated by the set

$$V_{k+1} = (T_k \setminus \{z^k\}) \cup \{z^{k,1}, \ldots, z^{k,n}\}, \qquad z^{k,i} = z^k - (z^k_i - x^k_i)e^i. \tag{15}$$

PROPOSITION 18. *The proper vertex set of $P_{k+1} = ([0, z^k] \setminus K_{x^k}) \cup_{z \in T_k \setminus \{z^k\}} [0, z]$ is obtained from V_{k+1} by removing the improper members according to the following rule:*

For every $z \in T_k \setminus \{z^k\}$: If $z \geq x^k$ while $z_i < z^k_i$ for exactly one $i \in \{1, \ldots, n\}$ (i.e., $z_i < z^k_i$ and $z_j \geq z^k_j\ \forall j \neq i$), then remove $z^{k,i}$.

Proof. Clearly any $z^{k,i}$ removed by this rule is improper. Conversely, since $T_k \setminus \{z^k\}$ has no improper members, and $z^{k,i} \leq z^k\ \forall i$, any improper member of V_{k+1} must be one $z^{k,i}$ for which there is a $z \in T_k \setminus \{z^k\}$ such that $z \geq z^{k,i} \geq x^k$, and $z_j \geq z^{k,i}_j = z^k_j\ \forall j \neq i$ and $z_i < z^k_i$ (because one cannot have $z_j \geq z^k_j\ \forall j$). □

In practice, it may be more efficient to take as P_{k+1} the polyblock $P_k \setminus K_{x^k}$ obtained from P_k by replacing every box $[0, z],\ z \in T_k \cap K_{x^k}$, with the polyblock $[0, z^k] \setminus K_{x^k}$. The proper vertex set of this polyblock can easily be computed using Propositions 17 and 18.

5. Polyblock outer approximation algorithm. From the above development we now derive solution methods for the basic problems of monotonic optimization formulated in section 3. First consider the problem (A):

$$\max\{f(x)|\ x \in G \cap H\}, \tag{A}$$

where $G \subset [0,b]$ is a compact normal set, H is a closed reverse normal set while $f(x)$ is increasing on $[0,b]$ and upper semicontinuous on H. We will assume, additionally, that

$$G \cap H \subset R^n_{++} \tag{16}$$

which implies, in particular, that $\text{int}G \neq \emptyset$ (see Proposition 4). In view of the compactness of the set $G \cap H$ the above assumption amounts, in fact, to requiring the existence of a vector a such that $0 < a < b$ and

$$0 < a \leq x \qquad \forall x \in G \cap H. \tag{17}$$

Also it is easily seen that (17) in turn is equivalent to a seemingly stronger condition, namely,

$$0 < a \leq x \qquad \forall x \in H.$$

Indeed, if condition (17) holds, then the problem does not change by replacing H with $H_a := \{x \in H |\ x \geq a\}$.

Let $\varepsilon \geq 0$ be a given tolerance. A feasible solution $\bar{x}$ such that $f(\bar{x}) \geq f(x) - \varepsilon\ \forall x \in G \cap H$ is called an *ε-optimal solution*. According to the outer approximation approach outlined in the preceding section, for finding an ε-optimal solution we proceed as follows.

First of all, if $b \notin H$ then since H is reverse normal, $[0,b] \cap H = \emptyset$; hence $G \cap H = \emptyset$, and the problem is infeasible. Therefore we may assume $b \in H$. As initial polyblock we take $P_1 = [0,b]$ with vertex set $T_1 = \{b\}$.

At iteration k we have a polyblock P_k with proper vertex set T_k. Observe that every vertex $z \in T_k \setminus H_a$ can be deleted since for these z we have $[0,z] \cap (H_a) = \emptyset$, and hence, by (17), $[0,z] \cap (G \cap H) = \emptyset$. Furthermore, if $\bar{x}^k$ is the best feasible solution known so far, and $CBV = f(\bar{x}^k)$ (current best value), then every $z \in T_k$ such that $f(z) \leq CBV + \varepsilon$ can also be deleted because for all these z we have $f(x) \leq f(\bar{x}^k) + \varepsilon\ \forall x \in [0,z]$.

Let $\tilde{T}_k$ be the set that remains from T_k after removing all $z \in T_k \setminus H_a$ and all $z \in T_k$ such that $f(z) \leq CBV + \varepsilon$. Since $P_1 = [0,b] \supset G \cap H$ it follows from the construction of $P_2, \ldots, P_k$ that if any solution x exists such that $f(x) > CBV + \varepsilon$, it can be found only in the polyblock $\tilde{P}_k$ with vertex set $\tilde{T}_k$. Therefore, if $\tilde{T}_k = \emptyset$, this means that no feasible solution x exists with $f(x) > CBV + \varepsilon$, so the procedure can be stopped: $\bar{x}^k$ is an ε-optimal solution if $CBV > -\infty$, or the problem is infeasible otherwise.

If $\tilde{T}_k \neq \emptyset$, let

$$z^k \in \text{argmax}\{f(z) |\ z \in \tilde{T}_k\} = \text{argmax}\{f(x) |\ x \in \tilde{P}_k\}.$$

Since $z^k \in H$, if $z^k \in G$, then z^k is feasible and hence it solves the problem because $f(x) \leq CBV + \varepsilon < f(z^k)\ \forall x \in (G \cap H) \setminus \tilde{P}_k$. On the other hand, if $z^k \notin G$, let $x^k = \pi_G(z^k)$ be the point computed according to (2). (Note that $z^k \in (a,b] \setminus G$.) Then, by Proposition 17 we can define the polyblock P_{k+1} with vertex set $V_{k+1} = (\tilde{T}_k \setminus \{z^k\}) \cup \{z^k - (z^k_i - x^k_i)e^i,\ i = 1, \ldots, n\}$ (see (15)) and compute its proper vertex set T_{k+1} according to Proposition 18.

We can thus state the following algorithm which will be referred to as the *Polyblock Outer Approximation Algorithm* for problem (A).

ALGORITHM 1.

Initialization. Select $\varepsilon \geq 0$ (tolerance). If $a \notin G$ the problem is infeasible (because then $G \cap H = \emptyset$). Otherwise, let $T_1 = \{b\}$. Let $\bar{x}^1$ be the best feasible solution available, $CBV = f(\bar{x}^1)$. If no feasible solution is available, let $CBV = -\infty$. Set $k = 1$.

Step 1. From T_k remove all $z \in T_k$ such that $z \notin H_a$ and all z such that $f(z) \leq CBV + \varepsilon$. Let $\tilde{T}_k$ be the set of remaining elements of T_k.

Step 2. If $\tilde{T}_k = \emptyset$, terminate: if $CBV = -\infty$, the problem is infeasible; if $CBV > -\infty$, the current best feasible solution $\bar{x}^k$ is accepted as an ε-optimal solution of (A).

Step 3. If $\tilde{T}_k \neq \emptyset$, select $z^k \in \operatorname{argmax}\{f(z) | \quad z \in \tilde{T}_k\}$.

Compute $x^k = \pi_G(z^k)$ (last point of G on the ray from 0 through z^k). If $x^k = z^k$, i.e., $z^k \in G$, then z^k is an optimal solution. Otherwise, determine the new current best value CBV and the new current best feasible solution $\bar{x}^{k+1}$.

Step 4. From $V_{k+1} = (\tilde{T}_k \setminus \{z^k\}) \cup \{z^k - (z_i^k - x_i^k)e^i, \ i = 1, \ldots, n\}$ remove the improper elements (using the rule indicated in Proposition 18) and let T_{k+1} be the resulting set.

Step 5. Set $k \leftarrow k + 1$ and return to Step 1.

THEOREM 1. *If Algorithm* 1 *is infinite, each of the generated sequences* $\{z^k\}, \{x^k\}$ *contains a subsequence converging to an optimal solution.*

Proof. Let us agree to call any vector $z^{k,i}$ obtained from z^k by formula (15) a child of z^k. If Algorithm 1 is infinite, it generates at least one infinite sequence $z^{l_1}, z^{l_2}, \ldots, z^{l_h}, \ldots$ contained in H_a such that $z^{l_{h+1}}$ is a child of z^{l_h}, i.e.,

$$z^{l_{h+1}} = z^{l_h} - (z_{i_h}^{l_h} - x_{i_h}^{l_h})e^{i_h}, \tag{18}$$

where $x^{l_h} = \pi_G(z^{l_h})$ and $i_h \in \{1, \ldots, n\}$. (Such a sequence can be obtained by taking $z^{l_1} = b$, and for $h \geq 1$, selecting $z^{l_{h+1}}$ among the children of z^{l_h} that have infinitely many descendants.)

We show that any sequence satisfying (18) converges to an optimal solution. Clearly $z^{l_1} \geq z^{l_2} \geq \cdots \geq z^{l_k} \geq \cdots \geq 0$. Let $\tilde{z}$ be an accumulation point of this bounded sequence ($z^k \in [0, b] \ \forall k$). Since $\|z^{l_1} - \tilde{z}\| \geq \|z^{l_2} - \tilde{z}\| \geq \cdots$ and a subsequence of this decreasing sequence tends to zero, it follows that $\lim_{h \to +\infty} \|z^{l_h} - \tilde{z}\| = 0$, i.e., $\tilde{z} = \lim_{h \to +\infty} z^{l_h}$. This implies $z^{l_h} - z^{l_{h+1}} \to 0$, and hence, since $z_{i_h}^{l_{h+1}} = x_{i_h}^{l_h}$,

$$z_{i_h}^{l_h} - x_{i_h}^{l_h} = |z_{i_h}^{l_h} - z_{i_h}^{l_{h+1}}| \leq \|z^{l_h} - z^{l_{h+1}}\| \to 0 \quad (h \to +\infty).$$

But $z^{l_h} - x^{l_h} = \lambda_h z^{l_h}$, so $z_{i_h}^{l_h} - x_{i_h}^{l_h} = \lambda_h z_{i_h}^{l_h} \to 0$. On the other hand, in view of the fact $z^{l_h} \in H_a$, we have $z_i^{l_h} \geq \min_{j=1,\ldots,n} a_j > 0 \ \forall i$; hence $\lambda_h = (z_{i_h}^{l_h} - x_{i_h}^{l_h})/z_{i_h}^{l_h} \to 0$. Therefore, $z^{l_h} - x^{l_h} \to 0$, and consequently, $\tilde{z} = \lim_{h \to +\infty} x^{l_h} = \lim_{h \to +\infty} z^{l_h}$. Since $x^{l_h} \in G \ \forall h$ while $z^{l_h} \in H \ \forall h$, we can conclude $\tilde{z} \in G \cap H$. Furthermore, since $f(z^{l_h}) \geq f(x) \ \forall x \in G \cap H$ and $\{z^{l_h}\} \subset H$, it follows from the upper semicontinuity of $f(x)$ on H that $f(\tilde{z}) \geq f(x) \ \forall x \in G \cap H$; i.e., $\tilde{z}$ is a global maximizer of $f(x)$ over $G \cap H$. □

Implementation issues. 1. If the set G is *robust* (i.e., any point of G is the limit of a sequence of points in $\operatorname{int}(G)$), then by replacing, if necessary, the set H by $H' = \{x \in H | \ x \geq \eta e\}$, where $\eta > 0$ is small enough, one obtains a problem satisfying condition (17), while differing only slightly from the original problem. In the general case, condition (17) can always be made to hold by simple manipulations. In fact, let

η be any positive number and define

$$\tilde{f}(x) = \begin{cases} f(x-\eta e) & \text{if } x \geq \eta e, \\ -M & \text{otherwise,} \end{cases} \tag{19}$$

$$\tilde{G} = R^n_+ \cap (G + \eta e - R^n_+), \quad \tilde{H} = H + \eta e,$$

where $M > 0$ is a sufficiently large number. Then $\tilde{f}(x)$ is increasing, $\tilde{G}$ (normal hull of $G + \eta e$) is normal, $\tilde{H}$ is reverse normal, and the original problem is equivalent to

$$(\tilde{A}) \qquad \max\{\tilde{f}(x) |\ x \in \tilde{H} \cap \tilde{G}\}.$$

Clearly $\tilde{H}$ satisfies condition (17) for $a = \eta e$.

2. To avoid storage problems in connection with the growth of the set T_k as the algorithm proceeds, and also to preclude possible jams, it may be useful to restart the algorithm whenever $|T_k| > L$, where L is a user supplied fixed number. Specifically, Step 5 of Algorithm 1 should be modified as follows. Let $\tilde{z}$ be the point where we would like to restart the algorithm (usually, $\tilde{z} = x^k$ or current best solution).

Step 5. If $|T_{k+1}| \leq L$, then set $k \leftarrow k+1$ and return to Step 1. Otherwise go to Step 6.

Step 6. Redefine $x^k = \pi(\tilde{z})$, $T_{k+1} = \{b - (b_i - x^k_i)e^i,\ i = 1, \ldots, n\}$, (i.e., $P_{k+1} = [0, b] \setminus (x^k, b]$), then set $k \leftarrow k+1$ and return to Step 1.

With this modification, an occurrence of Step 6 means a restart, i.e., the beginning of a new cycle of iterations.

3. In many cases (for instance when $H = a + R^n_+$ with $0 < a < b$), x^k is feasible for large values of k. Then, since $z^{l_h} - x^{kl_h} \to 0$, for any given $\varepsilon > 0$ an ε-optimal solution can be obtained in a finite number of iterations. In the general case, this may not be possible by Algorithm 1, but the fact $z^{k_h} - x^{k_h} \to 0$ implies that for any given $\delta > 0$ we must have $\|z^k - x^k\| \leq \delta$ for sufficiently large k. Then z^k is a maximizer of $f(x)$ over $(G \cap H) + \delta e$ and so it can be accepted as a *δ-approximate optimal solution* of the problem. Thus, to make the above algorithm finite the stopping criterion $\|z^k - x^k\| \leq \delta$ should be added in Step 2. (Since $x^k = \lambda_k z^k$ and $\|z^k\| \leq \|b\|$, this occurs if $(1 - \lambda_k)\|b\| \leq \delta$.)

4. Step 4 amounts to replacing the box $[0, z^k]$ with the polyblock $[0, z^k] \setminus K_{x^k}$. As was mentioned at the end of the previous section, to delete a larger part of P_k we can do the same for every $z \in \tilde{T}_k \cap K_{x^k}$. However, the advantage of having a smaller polyblock P_{k+1} may be offset by the disadvantage of having too numerous a vertex set.

5. In Step 2, instead of taking $x^k = \pi_G(z^k)$, where $\pi_G(z)$ is defined by formula (2), one can also take $x^k = \pi^v_G(x^k)$, where v is a fixed vector in R^n_{++}, e.g., $v = b$, and $\pi^v_G(z)$ is the first point of G on the half line from z^k in the direction $-v$, i.e.,

$$\pi^v_G(z) = z - \lambda v \qquad \text{with} \qquad \lambda = \min\{\alpha |\ z - \alpha v \in G\}. \tag{20}$$

With $x^k = \pi^v_G(z^k)$ defined by (20) the convergence of Algorithm 1 will still be guaranteed and may sometimes be even better. The convergence of this variant of Algorithm 1 can be established by an argument which is only a slight modification of the proof of Proposition 1 in the last part. Specifically, based on the relation $z^k - x^k = \lambda_k v \geq \lambda_k(\min_{i=1,\ldots,n} v_i)$, one can prove that $z^{l_h} - x^{l_h} \to 0$, hence z^{l_h} and x^{l_h} tend to a common limit $\tilde{z}$. Since $z^{l_h} \in H\ \forall h$ one must have $\tilde{z} \in H$. On the

other hand, for x^k defined from (20) we have $x^k \in G$ whenever $x^k > 0$. Noting that $x^{l_h} \to \tilde{z} \in H$, we deduce from the assumption (17) that, for all sufficiently large h, $x^{l_h} > 0$, and hence $x^{l_h} \in G$. This implies that $\tilde{z} \in G$; hence $\tilde{z} \in G \cap H$.

6. Algorithm 1 can also be viewed as a branch and bound procedure, in which the root of the associated tree is represented by the vertex b of the initial box $P_1 = [0, b]$ containing all feasible solutions. At iteration k the pending nodes of this tree correspond to the proper vertices of the polyblock P_k. Each vertex z represents the set of feasible solutions contained in the box $[0, z]$ and an upper bound for the values of $f(x)$ over the feasible solutions contained in this box is $f(x)$ if $z \in H_a$, or $-\infty$ if $z \notin H_a$ (since in the latter case there is no feasible solution in the box $[0, z]$). After "fathoming" the pending nodes with upper bound inferior to the current best value (Step 1), if no pending node remains for consideration ($\tilde{T}_k = \emptyset$), the algorithm terminates (Step 2). Otherwise, a node $z^k \in \tilde{T}_k$ with maximal upper bound is split into n new nodes (its "children") $z^{k,i} = z^k - (z^k_i - x^k_i)e^i,\ i = 1, \ldots, n$ (Steps 3 and 4), and the process is repeated. It should be noted, however, that in each iteration the boxes in the current polyblock determine a covering rather than a partition of the feasible set as in usual branch and bound procedures.

7. In Step 3 it is not necessary to always select z^k according to the criterion

$$z^k \in \operatorname{argmax}\{f(z) |\ z \in \tilde{T}_k\}. \tag{21}$$

In fact the proof of Theorem 1 shows that the bounding used in the above branch and bound interpretation of the algorithm is *consistent*. Therefore, according to the general theory of branch and bound procedures for global optimization [11, Theorem IV.3], to ensure convergence of the algorithm, it suffices that the selection of z^k be "*bound improving*," i.e., that the criterion (21) be used each time after a finite number of iterations. This flexibility in the selection of z^k may sometimes help to speed up the convergence.

Another fact worth noticing is that, given any (hyper)rectangle $[p, q] \subset [0, b]$, a bound for $\max\{f(x) |\ x \in G \cap H,\ p \leq x \leq q]$ can be obtained at cheap cost by applying just one or a few iterations of Algorithm 1 to the problem $\max\{f(p+y) |\ y \in (G - p) \cap (H - p), 0 \leq y \leq q - p\}$. By incorporating this bounding method in a rectangular partitioning procedure, we obtain a branch and bound algorithm in a more usual sense.

6. Reverse polyblock approximation algorithm. We now turn to problem B:

$$\min\{f(x) |\ x \in G \cap H\}, \tag{B}$$

where $G \subset [0, b]$ is a compact normal set and H is a closed reverse normal set while $f(x)$ is increasing on $[0, b]$ and lower semicontinuous on G. We will assume, additionally, that there exists a vector c such that $0 < c < b$ and

$$0 \leq x \leq c \quad \forall x \in G \cap H. \tag{22}$$

(The latter assumption is innocuous since it can always be satisfied by replacing b with a vector $b' > b$.)

As we saw in subsection 3.2, problem (B) can be converted to problem (A) by the transformation $x = b - y$. It may be useful, however, to see how this problem can be treated directly.

A set $P \subset R^n_+$ is called a reverse polyblock in $[0, b]$ if it is the union of a finite family of boxes $[z, b], z \in T$, where $T \subset [0, b], |T| < +\infty$. The set T is called the vertex set of P. A point $z \in T$ is called a *proper vertex* of P (or a proper element of T) if it does not dominate any other $z' \in T$. Of course a reverse polyblock is fully determined by its proper vertices and an increasing function achieves its minimum over a reverse polyblock at a proper vertex.

To solve problem (B) we construct a nested sequence of reverse polyblocks outer approximating the set $G \cap H$, or rather, outer approximating a subset of $G \cap H$ containing at least an optimal solution. As initial reverse polyblock we take $P_1 = [0, b]$ with vertex set $T_1 = \{0\}$. At iteration k, we have a reverse polyblock P_k with proper vertex set T_k. Since $z \notin (G \cap [0, c])$ implies $[z, b] \cap (G \cap H) = \emptyset$, every $z \in T_k \setminus (G \cap [0, c]$ can be deleted. Furthermore, if $\bar{x}^k$ is the best feasible solution known so far, and $CBV = f(\bar{x}^k)$ then, any $z \in T_k$ such that $f(z) \geq f(\bar{x}^k) - \varepsilon$ can also be deleted.

Let $\tilde{T}_k$ be the set that remains from T_k after removing all $z \in T_k \setminus (G \cap [0, c])$ and all $z \in T_k$ such that $f(z) \geq CBV - \varepsilon$. From the construction of P_1 and P_h for $1 \leq h \leq k$ it easily follows that every feasible solution x such that $f(x) < CBV - \varepsilon$ must be found in the reverse polyblock $\tilde{P}_k$ with vertex set $\tilde{T}_k$. Therefore, if $\tilde{T}_k = \emptyset$, this means no such feasible solution exists and the procedure terminates (with the conclusion that the best feasible solution so far obtained is an ε-optimal solution).

If $\tilde{T}_k \neq \emptyset$, let $z^k \in \operatorname{argmin}\{f(z) |\ z \in \tilde{T}_k\} = \operatorname{argmin}\{f(x) |\ x \in \tilde{P}_k\}$. Since $z^k \in G$, if $z^k \in H$, then $z^k \in G \cap H$ and z^k is an optimal solution because $f(x) \geq CBV - \varepsilon > f(z^k)\ \forall x \in (G \cap H) \setminus \tilde{P}_k$. On the other hand, if $z^k \notin H$, then $x^k > z^k$ for $x^k = \rho_H(z^k)$ (last point of H on the half line from b through z^k; see (3)), so $[z^k, b] \setminus [0, x^k)$ is a reverse polyblock with vertex set $\{z^{k,i} = z^k + (x^k_i - z^k_i)e^i,\ i = 1, \ldots, n\}$. Then we go to the next iteration, with T_{k+1} defined as the proper vertex set of the reverse polyblock P_{k+1} generated by $V_{k+1} = (\tilde{T}_k \setminus \{z^k\}) \cup \{z^{k,1}, \ldots, z^{k,n}\}$, i.e., the set obtained from V_{k+1} by removing all improper members according to the rule that for every $z \in T_k \setminus \{z^k\}$, if $z \leq x^k$ while $z_i > z^k_i$ for exactly one $i \in \{1, \ldots, n\}$, then remove $z^{k,i}$.

We can thus state the *Reverse Polyblock Approximation Algorithm* for problem (B).

ALGORITHM 2.

Initialization. Select $\varepsilon \geq 0$ (tolerance). Let $T_1 = \{0\}$. Let $\bar{x}^1$ be the best feasible solution available (the current best feasible solution), $CBV = f(\bar{x}^1)$. If no feasible solution is available, set $CBV = +\infty$. Set $k = 1$.

Step 1. From T_k remove all $z \in T_k \setminus (G \cap [0, c])$ and all $z \in T_k$ such that $f(z) \geq CBV - \varepsilon$. Let $\tilde{T}_k$ be the set of remaining elements of T_k.

Step 2. If $\tilde{T}_k = \emptyset$, terminate: if $CBV = +\infty$, the problem is infeasible; if $CBV < +\infty$, $\bar{x}^k$ is an ε-optimal solution.

Step 3. If $\tilde{T}_k \neq \emptyset$, select $z^k \in \operatorname{argmin}\{f(x) |\quad x \in \tilde{T}_k\}$.

Compute $x^k = \rho_H(z^k)$ (the last point of H on the half line from b through z^k). If $x^k = z^k$, i.e., $z^k \in H$, terminate: z^k is an optimal solution. Otherwise, determine the new CBV and the new current best feasible solution $\bar{x}^{k+1}$.

Step 4. From $V_{k+1} = (\tilde{T}_k \setminus \{z^k\}) \cup \{z^k + (x^k_i - z^k_i)e^i,\ i = 1, \ldots, n\}$ remove the improper elements and let T_{k+1} be the resulting set.

Step 5. Set $k \leftarrow k + 1$ and return to Step 1.

THEOREM 2. *If Algorithm* 2 *is infinite, each of the generated sequences* $\{z^k\}, \{x^k\}$ *contains a subsequence converging to an optimal solution.*

Proof. We omit the proof, which is similar to that of Theorem 1. □

Implementation issues. 1. The efficiency of the above algorithms may depend on the choice of the initial box $[0, b]$. For example, for problem (B) if a value $\gamma \in f(G \cap H)$, (i.e. $\gamma = f(\bar{x})$ for some feasible solution $\bar{x}$) is known from the beginning one can reset $G \leftarrow G \cap \{x|\ f(x) \leq \gamma\}$. After that let

$$\beta_i = \sup\{\beta > 0|\ \beta e^i \in G\}, \quad b' = \beta_1 e^1 + \cdots + \beta_n e^n, \tag{23}$$

$$\alpha_i = \sup\{\alpha > 0|\ b' - \alpha e^i \in H\}, \quad a = b' - \sum_{i=1}^{n} \alpha_i e^i. \tag{24}$$

Since for each $i = 1, \ldots, n$, the set $\{\beta \geq 0|\ \beta e^i \in G\}$ is a segment, β_i can be computed, for instance, by the Bolzano bisection method. Analogously, α_i can be computed easily. Then clearly $G \cap H \subset [a, b']$. Further, by resetting $G \leftarrow G \cap [a, b']$, $H \leftarrow H \cap [a, b']$, and selecting $b > b'$ then shifting the origin to a, we finally have (22), but now with a box $[0, b]$ which is a tighter approximation of $G \cap H$.

To select the initial reverse polyblock P_1, observe that $[0, \bar{x}) \subset [0, b] \setminus H$ (since H is reverse normal), so we can take $P_1 = [0, b] \setminus [0, \bar{x})$ (see Proposition 15).

2. Just as with Algorithm 1, to avoid storage problems in connection with the growth of T_k, and to preclude possible jams, it is advisable to make a *restart* when T_k exceeds a critical size L. Step 5 should then be modified as follows ($\tilde{z}$ may be x^k or the current best solution).

Step 5. If $|T_{k+1}| \leq L$, then set $k \leftarrow k + 1$ and return to Step 1. Otherwise go to Step 6.

Step 6. Redefine $x^k = \rho_H(\tilde{z})$, $T_{k+1} = \{x_i^k e^i,\ i = 1, \ldots, n\}$ (i.e., $P_{k+1} = [0, b] \setminus [0, x^k)$), then set $k \leftarrow k + 1$ and return to Step 1.

3. In Step 2, instead of taking $x^k = \rho_H(z^k)$, where $\rho_H(z)$ is defined by formula (3), one can also take $x^k = \rho_H^v(z^k)$, where $v \in R_{++}^n$ is a fixed vector (e.g., $v = e$) and $\rho_H^v(z)$ is the first point of H on the half line from z^k in the direction v, i.e.,

$$\rho_H^v(z) = z + \mu v, \qquad \mu = \min\{\alpha|\ z + \alpha v \in H\}. \tag{25}$$

Sometimes, as, e.g., in the problem in subsection 8.2 below, v can be chosen so that $\rho_H^v(z) \in H$ whenever $z \in R_+^n \setminus H$. In that case, the convergence of the algorithm is ensured without requiring (22).

7. Optimization of differences of increasing functions. The above approach can easily be extended to solve a very broad class of problems dealing with (d.i. functions).

First observe that, like the class of difference of two convex (d.c. functions), the class of d.i. functions is a linear space, closed under the operations of taking the pointwise minimum and the pointwise maximum. In this linear space the set of increasing functions forms a convex cone.

PROPOSITION 19. *If $f_1(x), \ldots, f_m(x)$ are d.i. then*

(i) *for any $\alpha_i \in R$ the function $\sum_{i=1}^m \alpha_i f_i(x)$ is also d.i.;*

(ii) *the functions $(\vee_{i=1}^m f_i)(x) := \max\{f_1(x), \ldots, f_m(x)\}$ and $(\wedge_{i=1}^m f_i)(x) = \min\{f_1(x), \ldots, f_m(x)\}$ are also d.i.*

Proof. (i) is trivial. To prove (ii) let $f_i = p_i(x) - q_i(x)$ with $p_i(x), q_i(x)$ increasing on R_+^n. Since $f_i = p_i + \sum_{j \neq i} q_j - \sum_j q_j$ we have $(\vee_{i=1}^m f_i)(x) = (\vee_i [p_i + \sum_{j \neq i} q_j])(x) - \sum_j q_j(x)$, where $\vee_i [p_i + \sum_{j \neq i} q_j]$ and $\sum_j q_j$ are increasing functions (see Proposition 1). □

Thus with respect to the operations $\vee$ and $\wedge$ defined as above, the linear space of d.i. functions is a vector lattice. Note that the set of increasing functions is also a lattice with respect to these operations.

PROPOSITION 20. *Any polynomial $P(x_1, \dots, x_n)$ on R^n_+ is a difference of two increasing functions.*

Proof. By grouping separately the terms with positive coefficients and those with negative coefficients, one can write $P(x) = P_1(x) - P_2(x)$, where each P_1, P_2 is a polynomial with nonnegative coefficients, hence an increasing function. □

COROLLARY 3. *The set of continuous d.i. functions on a box $[0, b] \subset R^n_+$ is dense in the space $C[0, b]$ of continuous functions on $[0, b]$ with the norm $\|f(x)\| = \max_{0 \le x \le b} |f(x)|$.*

Proof. By the Weierstrass approximation theorem, for any continuous function $f(x)$ on $[0, b]$ and any given $\varepsilon > 0$, one can find a polynomial $P(x)$ such that $\max_{0 \le x \le b} |f(x) - P(x)| \le \varepsilon$. By Proposition 20, $P(x)$ is a difference of two increasing functions. □

Remark 1. It is well known that a function of bounded variation of a real variable t can always be decomposed into a difference of two monotone nondecreasing functions. Hence if $f(x) = \sum_{i=1}^n f_i(x_i)$, where each $f_i(t), i = 1, \dots, n$, is a function of bounded variations, then $f(x)$ is a difference of two increasing functions.

7.1. Maximization. Consider the problem

$$\max\{f(x) - g(x) |\ x \in G \cap H\}, \tag{26}$$

where G is a normal set contained in a box $[0, b] \subset R^n_+$, H is a reverse normal set in $[0, b]$, while $f(x), g(x)$ are increasing functions on $[0, b]$.

For every $x \in [0, b]$, since $g(x) \le g(b)$ we have $g(x) + t = g(b)$ for $t = g(b) - g(x) \ge 0$. Hence, we can write the problem as

$$\max\{f(x) + t - g(b) | \quad x \in G \cap H,\ t = g(b) - g(x)\}$$

and by adding the constant $g(b)$ to the objective function we obtain the problem

$$\begin{aligned} &\max\{f(x) + t |\ x \in G \cap H, 0 \le t \le g(b) - g(x)\}, \quad \text{i.e.,} \\ &\max\{f(x) + t | \quad (x, t) \in D \cap E\}, \end{aligned}$$

where

$$\begin{aligned} D &= \{(x, t) | \quad x \in G,\ t + g(x) \le g(b),\ 0 \le t \le g(b) - g(0)\}, \\ E &= \{(x, t) | \quad x \in H,\ 0 \le t \le g(b) - g(0)\}. \end{aligned}$$

(Since any optimal solution must satisfy $t = g(b) - g(x) \le g(b) - g(0)$, one can add this constraint to the problem.) Clearly, E is reverse normal in $[0, b] \times [0, g(b) - g(0)]$. Also, since $t + g(x)$ is increasing on $[0, b] \times [0, g(b) - g(0)]$ the set D is normal in this box. Furthermore,

$$F(x, t) := f(x) + t$$

is obviously increasing on $[0, b] \times [0, g(b) - g(0)]$, so the problem (26) reduces to a problem (A) in R^{n+1}.

7.2. Minimization.

$$\min\{f(x) - g(x) | \quad x \in G \cap H\}, \tag{27}$$

where G, H, and $f(x), g(x)$ are as in Problem (26). Similar transformations to the above can be applied to convert problem (27) into a problem (B). Specifically, by adding the constant $g(b)$ to the objective function, we reduce the problem to the following:

$$\min\{f(x) + t | \quad x \in G \cap H,\ 0 \le t \le g(b) - g(0),\ t \ge g(b) - g(x)\},$$

or, equivalently, by setting $F(x,t) = f(x,t)$ and $D = \{(x,t) |\ x \in G,\ 0 \le t \le g(b) - g(0)\}, E = \{(x,t) |\ x \in H,\ t + g(x) \ge g(b),\ 0 \le t \le g(b) - g(0)\}$:

$$\min\{F(x,t) | \quad (x,t) \in D \cap E\}.$$

This is a problem (B) because E is a reverse normal set in $[0,b] \times [0, g(b) - g(0)]$, and D is a normal set in the same box. (Since any optimal solution must satisfy $t = g(b) - g(x) \le g(b) - g(0)$ one can add this constraint to the problem.)

7.3. General monotonic constraints. The most general problem of monotonic optimization is

$$\text{(GMOP)} \qquad \begin{array}{lll} \min & f_1(x) - f_2(x), & \\ \text{subject to (s.t.)} & g_i(x) - h_i(x) \le 0, & i = 1, \dots, m, \\ & x \in \Omega \subset R^n_+, & \end{array}$$

where f_1, f_2, g_i, h_i are increasing on R^n_+ and Ω is a normal set contained in a box $[0,b] \subset R^n_+$. This general problem can easily be reduced to the canonical form (B). In fact, by using, if necessary, transformations similar to those described earlier in this section, and by changing the notation, one can always assume, without loss of generality, that $f_2(x) = 0$. Further, the set of m constraints $g_i(x) - h_i(x) \le 0,\ i = 1, \dots, m$, can be written as a single inequality

$$\max_{i=1,\dots,m} \{g_i(x) - h_i(x)\} \le 0. \tag{28}$$

By Proposition 19 this inequality in turn is equivalent to

$$g(x) - h(x) \le 0,$$

where $g(x) = \max_i [g_i(x) + \sum_{j \ne i} h_j(x)]$ and $h(x) = \sum_i h_i(x)$ are both increasing functions. By adding a positive constant to both $g(x), h(x)$ one can assume $g(b) > 0$. Now, since $g(x) \le g(b)$ ($g(x)$ is increasing), we have for every $x \in [0,b] : g(x) + t \le g(b), t \ge 0$. Therefore the inequality $g(x) - h(x) \le 0$ for $x \in \Omega$ can be split into two inequalities:

$$g(x) + t \le g(b), \quad h(x) + t \ge g(b)$$

for $x \in \Omega \subset R^n_+$ and $t \in R_+$. The problem (GMOP) where $f_2(x) = 0$ thus reduces to

$$\min\{f_1(x) |\ g(x) + t \le g(b),\ h(x) + t \ge g(b),\ x \in \Omega,\ 0 \le t \le g(b) - g(0)\}$$

which is a problem (B) with $G = \{(x,t) | x \in \Omega, g(x) + t \le g(b), 0 \le t \le g(b) - g(0)\}, H = \{(x,t) \in R^n_+ \times R_+ |\ h(x) + t \ge g(b)\}$, and $G \subset [0,b] \times [0, g(b) - g(0)]$.

8. Applications. Numerous global optimization problems can be reformulated as monotonic optimization problems. Let us mention in this section some of the most noticeable possible applications.

8.1. Multiplicative programming. Optimization problems involving products of several convex or concave functions in the objective or in the constraints are often termed multiplicative programming problems. Examples are given by problems (5), (7), (10), and (11) in which $\varphi(y) = \prod_{i=1}^{m} y_i$:

$$\max\left\{\prod_{i=1}^{m} u_i(x)|\ x \in D\right\}, \tag{29}$$

$$\min\left\{\prod_{i=1}^{m} u_i(x)|\ x \in D\right\}, \tag{30}$$

$$\max\left\{\langle c, x\rangle| \quad x \in D,\ \prod_{i=1}^{m} u_i(x) \le 1\right\}, \tag{31}$$

$$\min\left\{\langle c, x\rangle| \quad x \in D,\ \prod_{i=1}^{m} u_i(x) \ge 1\right\}. \tag{32}$$

It is well known that if $u_i(x)$ are positive concave functions on R^n_+, then $(\prod_{i=1}^{m} u_i(x))^{\frac{1}{m}}$ is a concave function (see, e.g., [30]), and hence $\prod_{i=1}^{m} u_i(x)$ is a quasi-concave function. Therefore, in this case problems (29) and (32) are essentially convex problems. By contrast, when $u_i(x)$ are convex, the function $\prod_{i=1}^{m} u_i(x)$ is neither convex nor concave but a d.c. function. The above problems (29) through (32) are then highly nonconvex.

In recent years much research efforts has been devoted to multiplicative programming problems ([12], [14], [13]; see also [26] and references therein). Different methods (combining the parametric, branch and bound, outer approximation, and duality approaches) have been developed which have proved to be quite practical for solving low rank problems [14], involving products of $m \le 5$ affine or convex functions. For problems with $m > 5$ these methods often encounter serious difficulties and, to the knowledge of the author, so far no computational result has been reported in the literature.

However, as we saw, all the problems (29) through (32) are typically monotonic optimization problems and can be approached by the polyblock approximation methods. As shown by numerical experiments recently reported in [33] for problems of type (31), where $u_i(x)$ are affine, the polyblock approach can solve without difficulty problem instances with m up to 10, which could hardly be handled by existing methods.

Many multiplicative programs can be converted into one of the forms (29) through (32). Consider, for instance, the problem [25], [12]

$$\min[f_0(x) + f_1(x)f_2(x)] \quad \text{s.t. } x \in D, \tag{33}$$

where f_0, f_1, f_2 are affine functions and D is a polytope in R^n_+.

Define

$$\alpha_i = \min\{f_i(x)|\ x \in D\}, \qquad i = 1, 2.$$

Then

$$f_0(x) + f_1(x)f_2(x) = f_0(x) + \alpha_1 f_2(x) + \alpha_2 f_1(x) + (f_1(x) - \alpha_1)(f_2(x) - \alpha_2) + \alpha_1\alpha_2$$

and setting

$$\begin{aligned} u_i(x) &= f_i(x) - \alpha_i, \qquad i = 1, 2, \\ u_0(x) &= f_0(x) + \alpha_1 f_2(x) + \alpha_2 f_1(x) - \alpha_0, \end{aligned}$$

where

$$\alpha_0 = \min\{f_0(x) + \alpha_1 f_2(x) + \alpha_2 f_1(x) |\ x \in D\},$$

we have

$$f_0(x) + f_1(x)f_2(x) = u_0(x) + u_1(x)u_2(x) + \alpha_0 + \alpha_1\alpha_2.$$

Therefore the problem can be rewritten as

$$\min\{u_0(x) + u_1(x)u_2(x) |\ x \in D\},$$

where the functions $u_i(x), i = 0, 1, 2$ are affine and nonnegative on D.

Clearly the function $\varphi(y) = y_0 + y_1 y_2$ is increasing on R^3_+, so this is a problem of the same type as (10) (Example 3).

8.2. Indefinite quadratic programming. Consider the problem

$$\max\{f(x) |\ x \in D\}, \tag{34}$$

where $f(x)$ is an indefinite quadratic function and D is a compact normal set in R^n_+ such that $D \subset [a, b] \subset R^n_{++}$. As was already noticed, by grouping separately the terms with positive and the terms with negative coefficients, we can write $f(x) = f_1(x) - f_2(x)$ where f_1, f_2 are quadratic functions with nonnegative coefficients, i.e., increasing on R^n_+. Taking a constant $\gamma > f_2(b)$, we have $f_2(x) \le f_2(b) < \gamma$, i.e., $f_2(x) + t = \gamma$ for $0 < t = \gamma - f_2(x) \le \gamma - f_2(a)$; hence, by adding γ to the objective function, we can rewrite (34) as

$$\max\{f_1(x) + t |\ x \in D,\ f_2(x) + t \le \gamma,\ 0 < t \le \gamma - f_2(a)\}.$$

For any optimal solution (x, t) we have $t = \gamma - f_2(x) \ge \gamma - f_2(b) > 0$. Consequently, by setting $z = (x, t)$, $\tilde{f}(z) = f_1(x) + t$,

$$\begin{aligned} G &= \{z = (x, t) |\ x \in D,\ f_2(x) + t \le \gamma,\ 0 \le t \le \gamma - f_2(a)\}, \\ H &= \{z = (x, t) |\ x \ge a,\ t \ge \gamma - f_2(b)\}, \end{aligned}$$

the problem becomes a problem (A):

$$\max\{\tilde{f}(z) |\ z \in G \cap H\},$$

where $\tilde{f}(z)$ is an increasing function on R^{n+1}_+, G is a compact normal set, and H is a reverse normal set satisfying $(x, t) \ge (a, \gamma - f_2(b)) > (0, 0)\ \forall\ (x, t) \in H$ (i.e., condition (17)). For any given $z = (x, t) \in H \setminus G$, the point $\pi_G(z)$ is defined by $\pi_G(z) = \lambda z$ with $\lambda = \max\{\alpha |\ \alpha z \in G\}$ (see (2)). Clearly, $0 < \lambda < 1$ (because $z \notin G$) and $\lambda = \min\{\lambda_1, \lambda_2\}$, where $\lambda_1 = \max\{\alpha |\ \alpha x \in D\}, \lambda_2 = \max\{\alpha |\ \ f_2(\alpha x) + \alpha t \le \gamma\}$.

Since the function $P_z(\alpha) := f_2(\alpha x) + \alpha t$ is quadratic, monotone increasing for $\alpha > 0$, and $P_z(0) < \gamma$, we have $\lambda_2 = 1$ if $P_z(1) \le \gamma$ and λ_2 is the unique root of the quadratic equation $P_z(\lambda) = \gamma$ if $P_z(1) > \gamma$.

A special case of problem (34) is the *maximum clique problem* in the Motzkin–Strauss formulation (see [18], [7]):

$$\max\left\{\frac{1}{2}x^TQx|\ e^Tx = 1,\ x \ge 0\right\},$$

where Q is the adjacency matrix of a given graph Γ. As is known [18], the global optimal value of this problem equals $\frac{1}{2}(1 - 1/\omega(\Gamma))$ where $\omega(\Gamma)$ is the clique number of Γ. Since the matrix Q has only nonnegative entries, the function $f(x) := \frac{1}{2}x^TQx$ is increasing. By setting $x = y - a$, where $a > 0$, and defining a function $f(y)$ equal to $\frac{1}{2}(y-a)^TQ(y-a)$ if $y \ge a$ and equal to 0 otherwise, we convert this problem into a problem (A), namely,

$$\max\{f(y)|\ y \in G \cap H\},$$

where $f(y)$ is increasing on the orthant $y \ge 0$ and $G = \{y|\ e^T(y-a) \le 1,\ y \ge 0\}$ is normal, $H = \{y|\ y \ge a\}$ is reverse normal, and $y \ge a > 0\ \forall y \in G \cap H$ (condition (17)).

8.3. Polynomial programming. A polynomial programming problem is a problem of the general form

$$\min\{P_0(x)|\ P_i(x) \le 0\ (i = 1, \dots, m),\ x \in [0, b] \subset R^n_+\},$$

where $P_0, P_1, \dots, P_m$ are polynomials. By expressing each polynomial as a difference of two polynomials with nonnegative coefficients, this problem becomes

$$\text{(PLP)}\qquad \begin{array}{ll} \min & P_{0,1}(x) - P_{0,2}(x), \\ \text{s.t.} & P_{i,1}(x) - P_{i,2}(x) \le 0, \qquad i = 1, \dots, m, \\ & x \in [0, b] \subset R^n_+, \end{array}$$

where $P_{0,1}, P_{0,2}$, and $P_{i,1}, P_{i,2}$ $(i = 1, \dots, m)$ are polynomials with nonnegative coefficients. Adding $P_{0,2}(b)$ to the objective function, we rewrite the problem as

$$\begin{array}{ll} \min & P_{0,1}(x) + x_{n+1}, \\ & P_{0,2}(x) + x_{n+1} \ge P_{0,2}(b), \\ & P_{i,1}(x) - P_{i,2}(x) \le 0, \qquad i = 1, \dots, m, \\ & 0 \le x \le b, \qquad 0 \le x_{n+1} \le P_{0,2}(b) - P_{0,2}(0). \end{array}$$

Changing the notations, we thus have a problem of the form

$$\min\{f(x)|\ g_i(x) - h_i(x) \le 0\ (i = 1, \dots, p),\ x \in [0, c] \subset R^{n+1}_+\},$$

where f, g_i, h_i are polynomials in $x \in R^{n+1}$ with nonnegative coefficients. Next, setting $g(x) = \max_i[g_i(x) + \sum_{j \ne i} h_j(x)]$, $h(x) = \sum_i h_i(x)$, we can replace the set of inequalities $g_i(x) - h_i(x) \le 0$, $i = 1, \dots, p$, by the single inequality $g(x) - h(x) \le 0$ which in turn is equivalent to the system

$$g(x) + t \le g(c), \qquad h(x) + t \ge g(c), \qquad 0 \le t \le g(c) - g(0).$$

By defining then

$$G = \{(x,t)|\ g(x) + t \le g(c),\ 0 \le x \le c,\ 0 \le t \le g(c) - g(0)\},$$
$$H = \{(x,t)|\ h(x) + t \ge g(c)\},$$

and selecting a vector $b > 0$ and a constant $\eta > 0$. $b > c,\ h(b) > g(0),\ \eta > g(c) - g(0)$, we finally reduce the problem to the following problem (B):

$$\min\{f(x)|\ (x,t) \in G \cap H, \qquad 0 \le x \le b, \qquad 0 \le t \le \eta\}.$$

It is easily verified that condition (22) is fulfilled, namely,

$$0 \le (x,t) < (b,\eta) \quad \forall (x,t) \in G \cap H.$$

For solving this problem (B) we can apply the variant of Algorithm 2 using $\rho_H^v(z)$ defined by (25) with $v = (b,\eta)$. Then for every $z = (x,t) \notin H$ we have to compute $\rho_H^v(z) = (x + \mu b, t + \mu\eta)$, where μ satisfies

$$Q_z(\mu) := h(x + \mu b) + t + \mu\eta = g(c), \qquad 0 < \mu < 1.$$

Since $Q_z(\mu)$ is a polynomial in μ with nonnegative coefficients, hence a monotone increasing function of μ, this equation is very easy to solve. Thus, solving (PLP) by Algorithm 2 reduces to solving a connected sequence of polynomial equations of one variable.

An important special case of (PLP) is the general *nonconvex quadratic programming problem*, a subject which has attracted considerable interest in the last few years (see, e.g., [31]). In this case $P_{i,1}(x), P_{i,2}(x),\ i = 1,\dots,m$ are quadratic functions, so every equation $Q_z(\mu) = \eta$ is a mere quadratic equation of one variable. In other words, Algorithm 2 reduces a nonconvex quadratic program to a connected sequence of quadratic equations of one variable.

8.4. Lipschitz optimization. A Lipschitz optimization problem over a simplex can be reduced to a monotonic optimization problem, due to the following result [22].

PROPOSITION 21 (see [22]). *Let $f(x)$ be a Lipschitz function on the simplex $S = \{x \in R_+^n|\ \sum_{i=1}^n x_i = 1\}$, with Lipschitz constant $K > 0$, s.t. $\alpha := \min_{x \in S} f(x) > 0$. Then the function*

$$g(x) = \begin{cases} f\left(\dfrac{x}{\sum_{i=1}^n x_i}\right)\left(\sum_{i=1}^n x_i\right)^p, & x \ne 0, \\ 0, & x = 0, \end{cases}$$

extends $f(x)$ to the whole of R_+^n and is increasing, provided $p \ge \max\{1, \frac{2K}{\alpha}\}$.

Proof. This proposition was proved in [22]. We give a shorter proof here . Clearly $g(x) = f(x)\ \forall x \in S$, and $g(\lambda x) = \lambda^p g(x)\ \forall \lambda > 0$. We must show that $g(z) \ge g(x)$ for any $x \in R_+^n$ and $z \ge x$. Let $\sum_{i=1}^n x_i = \theta > 0$. Without loss of generality we may assume $\theta = 1$, i.e., $x \in S$, so $\lambda := \sum_{i=1}^n z_i > 1$. Let y be the point on the ray from 0 through z such that $y \in S$. Then $z = \lambda y$, so $g(z) = \lambda^p g(y) = \lambda^p f(y)$. But by the Lipschitz property of $f(x)$

$$g(x) = f(x) \le f(y) + K\|y - x\| = f(y)\left[1 + \frac{K\|y-x\|}{f(y)}\right] \le g(y)\left[1 + \frac{K\|y-x\|}{\alpha}\right]. \tag{35}$$

On the other hand,

$$\|y - x\| \leq \|y - z\| + \|z - x\|,$$

where $\|y-z\| = (\lambda-1)\|y\| \leq (\lambda-1)$ (because $\|y\| \leq 1$) and $\|z-x\| \leq (\lambda-1)$ because $z - x$ is contained in the simplex $(\lambda - 1)S$. Hence

$$\|y - x\| \leq 2(\lambda - 1)$$

which, by virtue of (35), implies that

$$g(x) \leq g(y)\left[1 + \frac{2K(\lambda - 1)}{\alpha}\right]. \tag{36}$$

It suffices now to compare the function $s(\lambda) = \lambda^p$ and the affine function $t(\lambda) = 1 + \frac{2K(\lambda-1)}{\alpha}$. Since $s(1) = t(1) = 1$ and $s'(1) = p \geq \frac{2K}{\alpha} = t'(1)$, it follows that $s(\lambda) \geq t(\lambda)\ \forall\ \lambda > 1$. Therefore, by (36), $g(x) \leq g(y)\lambda^p = g(z)$, as was to be proved. □

Since $S = G \cap H$, where $G = \{x \in R^n_+ |\ \sum_{i=1}^n x_i \leq 1\}$ is normal and $H = \{x \in R^n_+ |\ \sum_{i=1}^n x_i \leq 1\}$ is reverse normal, it follows from Proposition 21 that maximizing (or minimizing) a Lipschitz function $f(x)$ over the simplex $S = \{x \in R^n_+ |\ \sum_{i=1}^n x_i = 1\}$ reduces to solving a problem (A) (or (B), resp.).

8.5. Optimization under network constraints. Many optimization problems over a network have the following form:

$$\text{(PTP)} \qquad \begin{array}{ll} \min & \varphi(y) + \langle c, x\rangle, \\ \text{s.t.} & y \in Y, \\ & Qx = y, \\ & Bx = d, \quad x \geq 0, \end{array}$$

where $\varphi : R^p_+ \to R$ is an increasing function, Y is a polytope in R^p_+, $c, x \in R^n$, $d \in R^m$, Q is a $p \times n$ matrix of rank p, B is an $m \times n$ matrix. For instance, if y denotes a production program to be chosen from a set Y of technologically feasible production programs, and x a distribution flow over a given network, then the problem is to determine a production program y and a distribution program x so as to meet specified requirements with a minimum total cost $\varphi(y)+\langle c, x\rangle$. Let $\theta(y)$ be the optimal value of the subproblem

$$\text{(TP}(y)\text{)} \qquad \begin{array}{ll} \min & \langle c, x\rangle, \\ \text{s.t.} & Qx = y, \\ & Bx = d, \quad x \geq 0. \end{array}$$

Clearly $\theta(y)$ is a convex increasing function and it is easily verified that (PTP) is equivalent to the problem

$$\min\{\varphi(y) + \theta(y) |\ y \in Y\}. \tag{37}$$

Often the function $\varphi(y)$ is concave (by economy of scale), so (37) is a d.c. optimization problem. However, by setting

$$G = \{y \in R^p_+ |\ y \leq Qx,\ Bx = d,\ x \geq 0\}, \quad H = \{y \in R^p_+ |\ y \geq u,\ u \in Y\},$$

we see that this is actually a problem (B):

$$\min\{\varphi(y)+\theta(y)|\ y\in G\cap H\}, \tag{38}$$

where G is a normal closed set in R^p_+, H is a reverse normal closed set, and the objective function $\varphi(y)+\theta(y)$ is increasing. Note that $\theta(y)=+\infty$ for $y\notin G$ but we can make $\theta(y)$ a finite increasing function, lower semicontinuous on G, by setting $\theta(y)=M\ \forall y\notin G$, where M is a sufficiently large positive number. Assuming $Y\subset[0,b]$ it is easily seen that for every $z\in R^p_+\setminus H$ the determination of the point $\rho_H(z)=b+\mu(z-b)$ with $\mu=\sup\{\alpha>0|\ \ b+\alpha(z-b)\in Y\}$ is straightforward. For instance, if $Y=\{y\in R^p_+|\ \sum_{i=1}^p y_i=1,\ y\geq 0\}$ (so that $b=e$), then $\rho_H(z)=e+\mu(z-e)$ with $\mu=1/(1-\sum_{i=1}^p z_i)$.

8.6. Fekete points problem. One of the most challenging problems of global optimization consists in determining the distribution of Fekete points on a sphere (see, e.g., [5]). This problem can be formulated as follows:

$$\min\sum_{1\leq i<j\leq N}\frac{1}{\|x^i-x^j\|}\qquad\text{s.t.}\quad \|x^i\|=1,\quad i=1,\dots,N.$$

By rewriting it as

$$\min\left\{\sum_{1\leq i<j\leq N}y_{ij}|\quad y_{ij}\geq\frac{1}{\|x^i-x^j\|}\ (1\leq i<j\leq N),\ \|x^i\|=1\ (i=1,\dots,N)\right\}$$

we see that this problem has the form of a monotonic optimization problem, namely,

$$\min\left\{\sum_{1\leq i<j\leq N}y_{ij}\ |\quad y=(y_{ij})\in H\right\}\qquad\text{with} \tag{39}$$

$$H=\left\{y=(y_{ij})\ |\quad y_{ij}\geq\frac{1}{\|x^i-x^j\|}\ (1\leq i<j\leq N),\quad \|x^i\|=1\ (i=1,\dots,N)\right\}.$$

Here the objective function is obviously increasing for $y=(y_{ij})\geq 0$, while H is a reverse normal set because $y'\geq y\geq 0$ and $y\in H$ imply $y'\in H$. Let $M>0$ be the sum of the inverse of the mutual distances of any N chosen distinct points on the unit sphere. Since the distance between any two points on the unit sphere is at most 2, we have, for any optimal solution y of the problem and any fixed (i,j), $M\geq\frac{1}{2}[N(N-1)/2-1]+y_{ij}$; hence

$$y_{ij}\leq\eta:=M-\frac{1}{2}[N(N-1)/2-1].$$

Therefore, if we define the compact normal set $G=\{y=(y_{ij})|\ 0\leq y_{ij}\leq\eta\}$, then the problem (39) is the same as

$$\min\left\{\sum_{1\leq i<j\leq N}y_{ij}\ |\quad y=(y_{ij})\in G\cap H\right\}$$

which is exactly a problem (B). In solving this problem by the version of Algorithm 2 using $\rho_H^v(z)$ (with $v = \eta e$), the determination of $\rho_H^v(z)$ for each given $z = (z_{ij}) \in G \backslash H$ amounts to solving nonconvex quadratic programs of the form

$$\max\{\alpha \mid \quad (z_{ij} + \alpha\eta)\|x^i - x^j\| \geq 1 \ (1 \leq i < j \leq N), \quad \|x^i\| = 1 \ (i = 1, \ldots, N)\}.$$

These programs in turn can be solved by the method discussed in the two preceding subsections.

8.7. Lennard–Jones potential energy function. An important problem in computer simulations of molecular conformation and protein folding consists in finding the global minimum of the Lennard–Jones potential energy associated with a cluster of N particles. Its mathematical formulation is (see, e.g., [35])

$$\min\left\{\sum_{i<j}\left[\frac{1}{r_{ij}^{12}} - \frac{2}{r_{ij}^{6}}\right] \mid \quad r_{ij} = \|x^i - x^j\|, \ x^i \in R^3, \quad i, j = 1, \ldots, N\right\}. \tag{40}$$

One way to solve this problem is to convert it into a d.c. program [34]. However, in view of the large size of this d.c. program the global minimum is very difficult to compute exactly. Therefore the following two-stage strategy is proposed for finding an approximate global minimum.

Stage I. Compute a good lower bound for the global minimum

It has been proved in [35] that a global optimal solution must satisfy $r_{ij} = \|x^i - x^j\| \geq 0.5 \ \forall (i < j)$, so we can add this constraint to the problem. Setting $s_{ij} = \frac{1}{r_{ij}^6}$ we can then rewrite the problem as

$$\min\left\{f(s) - g(s) \mid 0 \leq s_{ij} \leq 2^6, \ s_{ij} = \frac{1}{\|x^i - x^j\|^6}, \ x^i \in R^3, \ (i, j = 1, \ldots, N)\right\}, \tag{41}$$

where

$$f(s) = \sum_{i<j} s_{ij}^2, \quad g(s) = 2\sum_{i<j} s_{ij}.$$

A lower bound for the optimal value of (40) can thus be computed by solving the relaxed problem obtained from (41) by replacing the constraint $s_{ij} = 1/\|x^i - x^j\|^6$ with $s_{ij} \leq 1/\|x^i - x^j\|^6$:

$$\theta := \min\left\{f(s) - g(s) \mid \quad 0 \leq s_{ij} \leq 2^6, \ s_{ij} \leq \frac{1}{\|x^i - x^j\|^6}, \ (1 \leq i < j \leq N)\right\}. \tag{42}$$

Let $b = (b_{ij})$ with $b_{ij} = 2^6, 1 \leq i < j \leq N$. Proceeding as in subsection 7.2 (see (27)), with γ being a positive constant s.t. $\gamma > g(b) = [N(N-1)]2^6$, we rewrite (42) as

$$\begin{aligned}\min\{f(s) + t \mid \quad & g(s) + t \geq \gamma, \ 0 \leq t \leq \gamma, \\ & 0 \leq s_{ij} \leq 2^6, \ s_{ij} \leq \frac{1}{\|x^i - x^j\|^6}, \ x^i \in R^3 \ (1 \leq i < j \leq N)\}.\end{aligned}$$

This is a problem (B):

$$\min\{f(s) + t \mid \quad (s, t) \in G \cap H\} \tag{43}$$

with

$$G = \left\{ (s,t) \mid \quad 0 \le t \le \gamma,\ 0 \le s_{ij} \le 2^6,\ s_{ij} \le \frac{1}{\|x^i - x^j\|^6},\ x^i \in R^3 \right\}, \tag{44}$$

$$H = \{(s,t) \mid \quad g(s) + t \ge \gamma,\ x^i \in R^3\}. \tag{45}$$

Therefore, Algorithm 2 can be applied to solve this problem. For every $z = (s,t) \notin H$, the point $\rho_H^e(z) = z + \mu e$ is computed by determining μ from the linear equation

$$\sum_{i<j} (s_{ij} + \mu) + t + \mu = \gamma.$$

Another subproblem which has to be solved in each iteration of Algorithm 2 is to verify whether a given point $z = (x,t)$ belongs to G. This is done by verifying, aside from the inequalities $0 \le t \le \gamma, 0 \le s_{ij} \le 2^6, 1 \le i < j \le N$ (which are immediate), also the feasibility of the convex system of inequalities (where s_{ij} are given):

$$s_{ij}\|x^i - x^j\| \le 1, \qquad x^i \in R^3, \qquad 1 \le i < j \le N. \tag{46}$$

If instead of replacing the constraint $s_{ij} = 1/\|x^i - x^j\|^6$ with $s_{ij} \le 1/\|x^i - x^j\|^6$ we simply omit it, then the set G in (44) does not involve this nonlinear constraint. In this case, no system (46) has to be considered for verifying the inclusion $z \in G$; however, the lower bound may be worsened.

Stage II. Solve a distance geometry problem to derive a feasible solution close to the global minimum.

Let $s_{ij} = \frac{1}{\delta_{ij}^6}$ be an optimal solution of (42). Now solve the following distance geometry problem:

$$\min \left\{ \sum_{i<j} (\delta_{ij} - \|x^i - x^j\|)^2 \mid \quad x^i \in R^3,\ i = 1, \ldots, N \right\}. \tag{47}$$

By writing this problem as

$$\min \left(\sum_{i<j} \|x^i - x^j\|^2 - 2 \sum_{i<j} \delta_{ij} \|x^i - x^j\| \right) \quad \text{s.t.} \quad x^i \in R^3, \quad i = 1, \ldots, N,$$

we obtain a quadratic optimization problem which can be solved by a generic branch and bound algorithm described in [31] or by the method discussed in subsection 8.4.

9. Computational results. We have presented a theory of monotonic optimization and shown its potential wide applicability. Since the model is very general, it is unlikely that the method can be uniformly efficient for every problem of the class considered. Nevertheless, from preliminary computational experience it appears that this approach may help to better handle many problems so far resistant to known methods.

As was mentioned in the introduction, computational results on testing earlier variants of Algorithms 1 and 2 on some classes of monotonic optimization problems have been quite encouraging. Experiments are reported in [24] where instances of problems (A) of the form (8) (Example 1), of dimension around 10, which are usually considered beyond the practical capability of existing algorithms (see, e.g., [14]), were

solved fairly quickly on conventional PCs. Similar results have been obtained in [33] on solving problems of the form (9) (Example 2) and in [16] on problems (B) of the form (13) (Example 4). It should be noted that in most cases the monotonic optimization problem considered comes from some large nonconvex problem via a number of transformations and its dimension is very small compared to the total number of variables of the original problem.

As an illustration, we present in this final section two small, but nontrivial, numerical examples of problems which have been used for testing Algorithms 1 and 2 in [33] and [16], respectively.

Example 5. Consider the following linear program with an additional multiplicative constraint (Kuno–Yajima [5]):

$$\begin{array}{ll} \max & x_3, \\ \text{s.t.} & 5x_1 + 10x_2 + 5x_3 \le 28, \\ & 8x_1 + 4x_2 + 5x_3 \le 28, \\ & -130x_1 - 40x_2 + 90x_3 \le 9, \\ & x_1 \ge 0,\ x_2 \ge 0,\ x_3 \ge 0, \\ & (3x_1 - x_2 + 3)(-x_1 + 3x_2 + 4) - 18 \le 0. \end{array} \tag{48}$$

This is a problem of the form (7) with $D \subset R^3_+$ being the polytope defined by the linear constraints, $\varphi(y) = y_1 y_2$, and $u_1(x) = x_1 - \frac{1}{3}x_2 + 1,\ u_2(x) = -\frac{1}{6}x_1 + \frac{1}{2}x_2 + \frac{2}{3}$. By Proposition 10 it is equivalent to

$$\max\{\theta(y) |\ y \in H \subset R^2_+, \quad y_1 y_2 \le 1\}, \tag{49}$$

where

$$H = \{y \in R^2_+ |\ u(x) \le y,\ x \in D\},$$
$$\theta(y) = \begin{cases} \max\{x_3 |\ x \in D, u(x) \le y\} & \text{if} \quad y \in H, \\ -M \qquad \text{otherwise} \end{cases}$$

(M being a sufficiently large positive number which does not need to be specified). It can easily be verified that

$$u(D) \subset [0, b] \text{ for } b = (4.5, 2.0667).$$

Hence for initialization we take

$$z^1 = (4.5, 2.0667), \quad T_1 = \tilde{T}_1 = \{(4.5, 2.0667)\},$$
$$y^1 = \pi(z^1) = (1.4756, 0.6777), \quad \bar{y}^1 = y^1,\ CBV = \theta(\bar{y}^1) = 0.9751.$$

(To compute $y = \pi(z)$, note that $y = \mu z$ with $(\mu z_1) \times (\mu z_2) = 1$; hence $y = (\sqrt{z_1/z_2},\ \sqrt{z_2/z_1})$.)

Iteration 1. $\theta(z^1) = 2.7095,\ \theta(\bar{z}^1) = 0.9751.$

Members of T_2 (vertices of the rectangle $[y^1, z^1]$ which are adjacent to z^1); and associated values of $\theta(.)$:

vectors z	$\theta(z)$
(1.4756, 2.0667)	2.0192
(4.5000, 0.6777)	2.7095

No $z \in T_2$ can be removed, so $\tilde{T}_2 = T_2$.

Computing $z^2 = \text{argmax}\{\theta(z)|z \in \tilde{T}_2\}$ yields $z^2 = (4.5000, 0.6777)$, with $y^2 = \pi(z^2) = (2.5769, 0.3881)$. Since $\theta(y^2) = -M$, we have $\bar{y}^2 = \bar{y}^1 = y^1$, and $CBV = 0.9751$.

Iteration 2. $\theta(z^2) = 2.7095,\ \theta(\bar{z}^2) = 0.9751$.

Members of T_3 and associated values of $\theta(.)$:

vectors y	$\theta(.)$
(1.4756, 2.0667)	2.0192
(2.5769, 0.6777)	2.6643
(4.500, 0.3881)	2.7095

No $z \in T_3$ can be removed, so $\tilde{T}_3 = T_3$.

Computing $z^3 = \text{argmax}\{\theta(z)|z \in \tilde{T}_3\}$ yields $z^3 = (2.5769, 0.6777)$ with $y^3 = \pi(z^3) = (3.4053, 0.2937),\ \theta(y^3) = 2.0192 > \theta(\bar{y}^2)$, so $\bar{y}^3 = y^3$ with $CBV = \theta(\bar{y}^3) = 2.0192$.

Continuing this way, with a tolerance $\epsilon = 0.0001$ (relative error), the algorithm terminates after 52 iterations, yielding 2.400 as the optimal value, with $y_{opt} = (3.0002, 0.3333)$ as optimal solution of problem (49) and $x_{opt} = (2.000, 0.000, 2.400)$ as optimal solution of problem (48). Note that, though the number of iterations may seem a bit high, the computational time is very small because each iteration involves only very simple computations.

For more detail on computational results of testing Algorithm 1 on problems (7), we refer the interested reader to [33].

Example 6. Given a polytope $D = \{x \in R^n|\ Ax \leq b, x \geq 0\}$ and a $p \times n$-matrix U, a point $x \in D$ is said to be *weakly efficient* with respect to U if there is no $x' \in D$ satisfying $Ux' > Ux$. Let D_{we} be the set of all weakly efficient points (the weakly efficient set) and $c \in R^n$. Consider the problem of *optimization over the weakly efficient set*

$$\max\{\langle c, x\rangle|\ \ x \in D_{we}\}. \tag{50}$$

If we define

$$\varphi(y) = \max\{t|\ \ Ux - te \geq y,\ x \in D\}, \tag{51}$$

then, as was shown in [16], problem (50) can be written as

$$\max\{\langle c, x\rangle|\ \ x \in D, \varphi(Ux) \leq 0\} \tag{52}$$

or, equivalently, as

$$\max\{\theta(y)|\ \ \varphi(y) \leq 0,\ y \in G\}, \tag{53}$$

where

$$\theta(y) = \begin{cases} \max\{\langle c, x\rangle|\ x \in D, Cx \geq y\} & \text{if} \quad y \in G, \\ -M \quad \text{otherwise}, \end{cases} \tag{54}$$

$$G = \{y \in R^p_+|\ \exists x \in X\ Cx \geq y\} \tag{55}$$

(M being a sufficiently large positive number). But it is easily seen that both functions $\theta(y)$ and $\varphi(y)$ are *decreasing* on R^p_+, while G is a closed normal set in R^p_+. Furthermore, it can be shown that $\varphi(y)$ is continuous and concave [16] . Therefore, (53) is a problem

of the form (13) considered in Example 4, subsection 3.2. By Proposition 12, if $\bar{x}$ solves (50) then $\bar{y} = U\bar{x}$ solves (53) and conversely, if $\bar{y}$ solves (53) and $\theta(\bar{y}) = \langle c, \bar{x}\rangle$ for an optimal solution $\bar{x}$ of (54) (where $y = \bar{y}$), then $\bar{x}$ solves (50).

Now let us solve problem (53) or, equivalently, (50), with the following data:

Vector $c \in R^{15}$:

$$c = (\ 0.40,\ -0.97,\ -0.16,\ -0.13,\ -0.15,\ 0.98,\ 0.25,\ -0.80$$
$$-0.55,\ 0.34,\ -0.48,\ 0.55,\ 0.70,\ 0.90,\ -0.68\ ).$$

Matrix A (5×15) :

2.50	7.60	3.00	3.50	1.70	−1.40	−2.20	4.80
	9.90	0.10	8.30	3.70	−6.30	−3.20	−8.10
8.60	−9.40	8.50	1.40	8.40	0.70	6.50	3.50
	−8.00	−1.70	−4.30	−1.40	0.00	8.60	0.30
−1.20	−0.80	8.20	0.20	−7.00	−7.10	7.50	−0.10
	−3.30	7.10	3.80	−6.80	−2.50	1.10	9.30
1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
	1.00	1.00	1.00	1.00	1.00	1.00	1.00
4.00	−6.90	3.20	−5.90	−7.00	−1.50	−1.90	6.50
	5.50	0.50	4.50	3.10	4.30	−6.10	4.20

vector $b \in R^5$:

$$b = (\ 5.61,\ 4.72,\ 3.47,\ 6.90,\ -3.40\ ).$$

Matrix U (3×15) :

0.52	0.14	0.18	0.63	0.94	0.49	0.80	0.67
	0.08	0.21	0.49	0.31	0.81	0.68	0.28
0.78	0.16	0.06	0.85	0.14	0.02	0.86	0.93
	0.24	0.74	0.61	0.96	0.34	0.34	0.14
0.92	0.53	0.84	0.29	0.84	0.99	0.53	0.64
	0.84	0.27	0.85	0.80	0.84	0.58	0.23

With tolerance 0.001, Algorithm 2 found the optimal value 6.743 after 4 iterations. An optimal solution for the corresponding problem (53) is

$$y_{opt} = (0.040,\ 0.040,\ 6.829)$$

and the associated optimal solution for problem (50) is

$$x_{opt} = (0.000,\ 0.000,\ 0.000,\ 0.000,\ 0.000,\ 6.887,\ 0.000,\ 0.000,$$

$$0.013,\ 0.000,\ 0.000,\ 0.000,\ 0.000,\ 0.000,\ 0.000).$$

Details of computational results on solving problems (13) by Algorithm 2 can be found in [16].

Acknowledgment. The author is grateful to the referees for several useful remarks.

REFERENCES

[1] T.M. Abasov, *Normal sets and monotone functions and their applications*, J. Comput. Math. Math. Phys., 3 (1993), pp. 1004–1011 (in Russian).

[2] S. Azarm, *Local monotonicity in Optimal Design*, Ph.D. thesis, University of Michigan, Ann Arbor, 1984.
[3] S. Azarm and P. Papalambros, *An automated procedure for local monotonicity analysis*, ASME Journal of Mechanisms, Transmissions, and Automation in Design, 106 (1984), pp. 82–89.
[4] H.P. Benson, *Optimization over the efficiency set*, J. Optim. Theory Appl., 73 (1992), pp. 47–64.
[5] T. Erber and G.M. Hockney, *Equilibrium configurations of N equal charges on a sphere*, J. Phys. A: Math. Gen., 24 (1991), pp. L1369–L1377.
[6] A.M. Geoffrion, *Solving bi-criterion mathematical programs*, Oper. Res., 15 (1967), pp. 38–54.
[7] L.E. Gibbons, D.W. Hearn, P.M. Pardalos, and M.V. Ramana, *Continuous characterizations of the maximum clique problem*, Math. Oper. Res., 22 (1997), pp. 754–768.
[8] C.R. Hammond and G.E. Johnson, *A general approach to constrained optimal design based on symbolic mathematics*, in Advances in Design Automation—Design methods, Computer Graphics and Expert Systems, S.S. Rao, ed., ASME, Fairfield, NJ, 1987, pp. 31–40.
[9] P. Hansen, B. Jaumard, and S.H. Lu, *Some further results on monotonicity in globally optimal design*, Journal of Mechanic SMS, Transmissions and Automation in Design, 111 (1989), pp. 345–352.
[10] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econom. and Math. Systems, 187, Springer-Verlag, New York, 1981.
[11] R. Horst and H. Tuy, *Global Optimization (Deterministic Approaches)*, 3rd ed., Springer-Verlag, Berlin, New York, 1996.
[12] H. Konno and T. Kuno, *Multiplicative programming problems*, in Handbook of Global Optimization, R. Horst and P. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, Boston, London, 1995, pp. 369–405.
[13] H. Kuno, Y. Yajima, Y. Yamamoto, and H. Konno, *Convex programs with an additional constraint on the product of several convex functions*, European J. Oper. Res., 77 (1991), pp. 314–324.
[14] H. Konno, P.T. Thach, and H. Tuy, *Optimization on Low Rank Nonconvex Structures*, Kluwer Academic Publishers, Dordrecht, Boston, London, 1997.
[15] H.L. Li and P. Papalambros, *A production system for use of global optimization knowledge*, ASME Journal of Mechanisms, Transmissions, and Automation in Design, 107 (1985), pp. 277–284.
[16] L.T. Luc, *Polyblock approximation approach for optimization over the weakly efficient and efficient Sets*, Acta Math. Vietnam., to appear.
[17] V.L. Makarov and A.M. Rubinov, *Mathematical Theory of Economic Dynamic and Equilibria*, Springer-Verlag, Berlin, New York, 1977.
[18] T.S. Motzkin and E.G. Strauss, *Maxima for graphs and a new proof of a theorem of Turan*, Canad. J. Math. 17 (1965), pp. 533–540.
[19] P. Papalambros and H.L. Li, *Notes on the operational utility of monotonicity in optimization*, ASME Journal of Mechanisms, Transmissions, and Automation in Design, 105 (1983), pp. 174–180.
[20] P. Papalambros and D.J. Wilde, *Principles of Optimal Design—Modeling and Computation*, Cambridge University Press, Cambridge, UK, New York, 1986.
[21] U. Passy, *Global solutions of mathematical programs with intrinsically concave functions*, in Advances in Geometric Programming, M. Avriel, ed., Plenum Press, New York, 1980, pp. 355–373.
[22] A. Rubinov and M. Andramonov, *Lipschitz programming via increasing convex-along-rays functions*, Optim. Methods Softw., 10 (1999), pp. 763–781.
[23] A. Rubinov and B.M. Glover, *Duality for increasing positively homogeneous functions and normal sets*, RAIRO Rech. Opér., 32 (1998), pp. 105–123.
[24] A. Rubinov, H. Tuy, and H. Mays, *Algorithm for a monotonic global optimization problem*, Optimization, to appear.
[25] S. Schaible and C. Sodini, *Finite algorithm for generalized linear multiplicative programming*, J. Optim. Theory Appl., 87 (1995), pp. 441–455.
[26] M. Sniedovich, *C-programming and the minimization of pseudolinear and additive concave functions*, Oper. Res. Lett., 5 (1986), pp. 185–189.
[27] H. Tuy, *The complementary convex structure in global optimization*, J. Global Optim., 2 (1992), pp. 21–40.
[28] H. Tuy, *D.C. optimization: Theory, methods and algorithms*, Handbook on Global Optimization, R. Horst and P.M. Pardalos, eds., Kluwer Academic Publishers, Dordrecht, Boston,

London, 1995, pp. 149–216.
[29] H. Tuy, *Normal sets, polyblocks and monotonic optimization*, Vietnam J. Math., 27 (1999), pp. 277–300.
[30] H. Tuy, *Convex Analysis and Global Optimization*, Kluwer Academic Publishers, Dordrecht, Boston, London, 1998.
[31] H. Tuy, *Normal branch and bound algorithms for general nonconvex quadratic programming*, in Combinatorial and Global Optimization, P.M. Pardalos, A. Migdalas, and R. Burkard, eds., World Scientific, River Edge, NJ, to appear.
[32] H. Tuy, *The MCCNFP with a fixed number of nonlinear arc costs: Complexity and approximation*, in Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems, P.M. Pardalos, ed., Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 525–549.
[33] H. Tuy and Le Tu Luc, *A new approach to optimization under monotonic constraint*, J. Global Optim., to appear.
[34] H. Tuy, *On some recent advances and applications of d.c. optimization*, in Optimization, V.H. Nguyen, J.J. Strodiot, and P. Tossings, eds., Lecture Notes in Econom. and Math. Systems 481, Springer-Verlag, New York, 2000, pp. 473–497.
[35] G.L. Xue, *Minimum inter-particle distance at global minimizers of Lennard–Jones clusters*, J. Global Optim., 11 (1997), pp. 83–90.
[36] D.J. Wilde, *A maximum activity principle for eliminating over-constrained optimization cases*, ASME Journal of Mechanisms, Transmissions, and Automation in Design, 108 (1986), pp. 312–314.
[37] P.L. Yu, *Multicriteria Decision Making*, Plenum Press, New York, 1995.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 495–521

A COMPARISON OF A MOREAU–YOSIDA-BASED ACTIVE SET STRATEGY AND INTERIOR POINT METHODS FOR CONSTRAINED OPTIMAL CONTROL PROBLEMS*

M. BERGOUNIOUX†, M. HADDOU†, M. HINTERMÜLLER‡, AND K. KUNISCH‡

Abstract. This research is devoted to the numerical solution of constrained optimal control problems governed by elliptic partial differential equations. The main purpose is a comparison between a recently developed Moreau–Yosida-based active set strategy involving primal and dual variables and two implementations of interior point algorithms.

Key words. optimal control, interior point methods, Moreau–Yosida approximation, active sets

AMS subject classifications. 49J20, 65K, 90C20

PII. S1052623498343131

1. Introduction.

In recent years significant research efforts were focused on developing numerical techniques to solve optimal control problems governed by partial differential equations. For unconstrained problems a high level of sophistication was reached. We refer to the contributions in [AM, GT, KS] and many further references given therein. For constrained optimal control problems the level of research is less complete. Common approaches are based on applying a quasi-Newton or sequential quadratic programming (SQP) technique to the constrained, possibly nonlinear optimal control problem and to resolve the resulting quadratic subproblems by some standard methods. Frequently the constrained quadratic subproblems can be the most costly part in this approach.

In this note we focus on a comparison of two efficient methods to solve quadratic constrained optimal control problems governed by elliptic partial differential equations. One of them is based on a generalized Moreau–Yosida formulation of the constrained optimal control problem which results in an active set strategy involving primal and dual variables. The second approach is based on interior point methods. We sorted out two of the most successful versions and developed efficient codes utilizing the structure of the underlying partial differential equation.

Optimal control problems are infinite-dimensional problems which require discretization before they can be solved numerically. Once discretized they can, in principle, be treated as generic minimization problems whose numerical solution has received a significant amount of attention. Interior point methods are considered to be extremely efficient in solving such convex minimization problems and are therefore a natural choice to compare with the newly developed Moreau–Yosida-based primal-dual active set strategies.

A comparison of algorithmic methods is a delicate matter and thus a word describing the approach is in order. Both the second and the third authors wrote

*Received by the editors August 3, 1998; accepted for publication (in revised form) April 27, 2000; published electronically October 18, 2000.

http://www.siam.org/journals/siopt/11-2/34313.html

†UMR-CNRS 6628, Université d'Orléans, U.F.R. Sciences, B.P. 6759, F-45067 Orléans Cedex 2, France (maitine@labomath.univ-orleans.fr).

‡Institute for Mathematics, University of Graz, A-8010 Graz, Austria (michael.hintermueller@kfunigraz.ac.at, karl.kunisch@kfunigraz.ac.at). The work of the third and fourth authors was supported in part by the Fonds zur Förderung der wissenschaftlichen Forschung under "Spezialforschungsbereich Optimierung und Kontrolle," SFB03.

independent codes following different trends in interior point methods. Both codes are predictor-corrector central path-following codes. While Haddou's code parallels closely the theory developed in [BG, BPR, M, MTY, P] for which computational complexity estimates are available provided that the iterates stay sufficiently close to the central path, Hintermüller's code follows the concepts of [Meh, V1, Wr, ZZ, Z2] which favor minimizing the number of iterates over satisfying assumptions which guarantee complexity estimates.

Upon completion both interior point codes were compared to the primal-dual active set codes described in [BIK, BK2]. Finally the two interior point algorithms were compared against each other. This last comparison led to several improvements of the two interior point algorithms in their own right. We emphasize, however, that the focus of this contribution is the comparison of two significant representatives of interior point algorithms to the primal-dual active set strategy.

To describe the optimal control problems for which comparisons were carried out let Ω denote a bounded subset of $\mathbb{R}^2$ with boundary Γ. Further let $\alpha > 0, z_d, u_d \in L^2(\Omega)$ and $\varphi \in L^2(\Omega)$, $\psi \in L^2(\Omega)$. We consider optimal control problems with distributed control of the type

$$(\mathcal{P}) \qquad \begin{cases} \text{minimize} & J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y - z_d)^2 dx + \dfrac{\alpha}{2}\int_\Omega (u - u_d)^2 dx \\ \text{subject to} & -\Delta y = u \text{ in } \Omega,\ y = 0 \text{ on } \Gamma, \\ & (y,u) \in K_1 \times K_2, \end{cases}$$

where

$$K_1 = \left\{\ y \in L^2(\Omega) \mid y(x) \le \varphi(x) \text{ almost everywhere (a.e.) in } \Omega\ \right\} \subset L^2(\Omega)\,,$$
$$K_2 = \left\{\ u \in L^2(\Omega) \mid u(x) \le \psi(x) \text{ a.e. in } \Omega\ \right\} \subset L^2(\Omega).$$

Many generalizations of the problem concerning the cost functional, the differential equation, or the constraints are possible. But these are not the focus of this note and thus we limit ourselves from the start to problems of the type $(\mathcal{P})$. If in $(\mathcal{P})$ only the state variable y or the control variable u are constrained and $K_2 = L^2(\Omega)$ or $K_1 = L^2(\Omega)$, then we refer to $(\mathcal{P})$ as a state constrained or control constrained problem, respectively. Let us note the difference with respect to regularity of the variables y and u in $(\mathcal{P})$. While $u \in L^2(\Omega)$, the state variable y is in the Sobolev space $H_o^1(\Omega) \cap H^2(\Omega)$. Moreover it will follow from the optimality systems to be presented in section 2 that there are distinct differences with respect to the regularity properties of the dual variables for the control and state constrained problems. This distinction gets lost if we consider $(\mathcal{P})$ only in its discretized form.

In this research we consider problems with distributed control in spite of the fact that boundary control problems may be more practical. From the optimization point of view the distributed control problem involves a higher-dimensional problem and a more complicated active set structure. In this case the control space is significantly larger and the decision whether a point is active or not has to be made for a much larger set of points than in the case of boundary controls.

The remaining sections are organized as follows. In section 2 we give precise statements for the control and the state constrained cases separately. Section 3 is devoted to a description of algorithms based on an active set strategy involving primal and dual variables. The following section 4 contains presentations of the interior point algorithms which served as algorithms with respect to which the primal-dual active

set algorithm was compared. The last section 5 contains a limited number of selected numerical examples that illustrate our numerical findings and which are part of the basis for our conclusions.

2. Problem statement and optimality conditions. In this section we describe the control and the state constrained optimal control problems and the corresponding first-order necessary optimality conditions. The separate treatment of the two cases is motivated by the fact that the analytical properties of their solutions and subsequently the behavior of the numerical algorithms differ significantly. Since our main goal is the comparison of the generalized Moreau–Yosida-based algorithm to interior point algorithms we prefer to keep separate the phenomena due to other types of constraints.

We shall focus our attention on constrained optimal control problems with quadratic cost and affine constraints. Besides their independent interest they are also an essential building block in Newton and SQP algorithms applied to general nonlinear optimal control problems for partial as well as after time discretization unstationary partial differential equations; see, for instance, [GT, JS, LS]. In each iteration of such an algorithm a constrained linear-quadratic subproblem of the type considered below must be solved. It is frequently the most time-consuming part of the whole SQP or Newton algorithm.

The control constrained problem is given by

$$(\mathcal{P}^c) \qquad \begin{cases} \text{minimize} & J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y-z_d)^2 dx + \dfrac{\alpha}{2}\int_\Omega (u-u_d)^2 dx \\ \text{subject to} & -\Delta y = u \text{ in } \Omega, \quad y = 0 \text{ on } \Gamma, \\ & u \in K_2, \end{cases}$$

with $z_d, u_d, \Omega, \Gamma, \alpha, K_2$, and ψ as in section 1. This problem admits a unique solution that we denote by $(y^*, u^*) \in \big(H_o^1(\Omega) \cap H^2(\Omega)\big) \times L^2(\Omega)$. Associated to u^* we define the active set $\mathcal{A}^* = \{x : u^*(x) = \psi(x) \text{ a.e.}\}$, and the inactive set $\mathcal{I}^* = \Omega \setminus \mathcal{A}^*$. To describe the optimality condition let I_{K_2} denote the indicator function of K_2 and let ∂I_{K_2} stand for its subdifferential. Recall that for $u \in K_2$ we have $\lambda \in \partial I_{K_2}(u) \subset L^2(\Omega)$ if and only if

$$I_{K_2}(u) + (\lambda, v-u)_{L^2(\Omega)} \le I_{K_2}(v) \quad \text{for all } v \in L^2(\Omega)$$

or, equivalently,

$$(\lambda, v-u)_{L^2(\Omega)} \le 0 \quad \text{for all } v \in K_2\,,$$

where $(\cdot,\cdot)_{L^2(\Omega)}$ denotes the $L^2(\Omega)$-inner product. The optimal solution (y^*, u^*) is characterized by the existence of $(p^*, \lambda^*) \in \big(H_o^1(\Omega) \cap H^2(\Omega)\big) \times L^2(\Omega)$ such that $(y^*, u^*, p^*, \lambda^*)$ satisfies

$$(\mathcal{S}^c) \qquad \begin{cases} -\Delta y^* = u^* \text{ in } \Omega,\ y^* \in H_o^1(\Omega), \\ -\Delta p^* = -(y^* - z_d) \text{ in } \Omega,\ p^* \in H_o^1(\Omega), \\ u^* = u_d + \dfrac{1}{\alpha}(p^* - \lambda^*) \text{ in } \Omega, \\ \lambda^* \in \partial I_{K_2}(u^*). \end{cases}$$

This result is well known; see, for instance, [Ba], [BIK, Theorem 1.1]. The differential inclusion appearing as the last condition in $(\mathcal{S}^c)$ is not amenable to numerical

realization, and we therefore replace it by

$$\lambda^* = c\left(u^* + \frac{\lambda^*}{c} - \Pi_{K_2}\left(u^* + \frac{\lambda^*}{c}\right)\right) = c\max\left(0, u^* + \frac{\lambda^*}{c} - \psi\right) \tag{2.1}$$

for any $c > 0$. Here Π_{K_2} denotes the Hilbert space projection of $L^2(\Omega)$ onto K_2, and max stands for the pointwise maximum as x varies in Ω. The equivalence between $\lambda^* \in \partial I_{K_2}(u^*)$ for $u^* \in K_2$ and (2.1) can easily be verified by a direct computation [IK]. A short computation also shows that (2.1) holds for some $c > 0$ if and only if it holds for all $c > 0$. In particular (2.1) does not depend on a specific choice of $c > 0$. Let us also observe that (2.1) is equivalent to

$$u^* = \Pi_{K_2}\left(u^* + \frac{\lambda^*}{c}\right)$$

for $c > 0$, which is a form of the optimality condition familiar from the projected gradient method. Here we prefer to use (2.1) since it is more suggestive of the update strategy which will be used.

Let us next turn to the state constrained problem given by

$$(\mathcal{P}^s) \qquad \begin{cases} \text{minimize} & J(y,u) = \dfrac{1}{2}\displaystyle\int_\Omega (y - z_d)^2 dx + \dfrac{\alpha}{2}\int_\Omega (u - u_d)^2 dx \\ \text{subject to} & -\Delta y = u \text{ in } \Omega,\ y = 0 \text{ on } \Gamma, \\ & u \in L^2(\Omega),\ y \in K_1, \end{cases}$$

where z_d, u_d, Ω, Γ, α, K_1, and φ are defined as in section 1. We assume the existence of at least one $u \in L^2(\Omega)$ such that the solution $y \in H_o^1(\Omega) \cap H^2(\Omega)$ to $-\Delta y = u$ satisfies $y \in K_1$. This implies that the set of feasible pairs (y, u) satisfying the constraints in $(\mathcal{P}^s)$ is nonempty. The existence of a solution $(y^*, u^*) \in \left(H_o^1(\Omega) \cap H^2(\Omega)\right) \times L^2(\Omega)$ can then easily be proved. Using techniques from [BK1, C, BC], for example, the following optimality system can be derived: There exists a pair $(p^*, \lambda^*) \in L^2(\Omega) \times \mathcal{M}(\Omega)$ such that

$$(\mathcal{S}^s) \qquad \begin{cases} -\Delta y^* = u^* \text{ in } \Omega, \quad y^* \in H_o^1(\Omega), \\ (p^*, -\Delta v)_{L^2(\Omega)} + \langle \lambda^*, v\rangle + (y^* - z_d, v)_{L^2(\Omega)} = 0 \text{ for all } v \in H_o^1(\Omega) \cap H^2(\Omega), \\ u^* = u_d + \dfrac{1}{\alpha}\, p^* \text{ in } \Omega, \\ \langle \lambda^*, y - y^*\rangle \le 0 \text{ for all } y \in K_1. \end{cases}$$

Here $\mathcal{M}(\Omega)$ is the set of Radon measures on Ω and $\langle \cdot, \cdot \rangle$ denotes the duality product between $\mathcal{M}(\Omega)$ and the set of continuous functions on $\overline{\Omega}$.

Let us note that the regularity of the primal variables is such that $(y^*, u^*) \in H^2(\Omega) \times L^2(\Omega)$ for both $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$. In fact all admissible pairs of primal variables for $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$ satisfy $(y, u) \in H^2(\Omega) \times L^2(\Omega)$. The extra regularity of y over u explains the higher accuracy of the interior point methods for the y components with respect to the u components recorded in Tables 5.2, 5.4, 5.5, and 5.6 of section 5. On the other hand the regularity of the adjoint variables (p^*, λ^*) is very different for $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$ with $(p^*, \lambda^*) \in H^4(\Omega) \times L^2(\Omega)$ in the control constrained case and $(p^*, \lambda^*) \in L^2(\Omega) \times \mathcal{M}(\Omega)$ in the state constrained case. The roughness of the adjoint variables suggests that the numerical solutions exhibit oscillatory behavior. Besides initialization it is most likely the main reason that the Moreau–Yosida-based active

set strategy requires in general more iterations for $(\mathcal{P}^s)$ than for $(\mathcal{P}^c)$ before it stops at the exact solution of the discretized versions of $(\mathcal{S}^c)$, respectively, $(\mathcal{S}^s)$.

3. Generalized Moreau–Yosida-based algorithms. This section is devoted to the description of the Moreau–Yosida-based algorithms for control and state constrained problems. Before we commence with the description of the algorithms we acknowledge the fact that $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$ are infinite-dimensional problems whose numerical realization requires discretization.

For this purpose Ω is endowed with a uniform grid Ω_h with mesh-size h. We proceed in general terms with finite differences and finite element realizations in mind. Let z_{dh}, y_h, φ_h, u_{dh}, u_h, $\psi_h \in \mathbb{R}^N$ be finite-dimensional approximations to z_d, y, φ, u_d, u, and ψ, respectively. Further let $A_h \in \mathbb{R}^{N\times N}$ stand for a symmetric positive-definite approximation to $-\Delta$. The discretized control constrained problem is given by

$$(P_h^c) \qquad \begin{cases} \text{minimize} & J_h(y_h, u_h) \\ \text{subject to} & A_h y_h = u_h,\ u_h \le \psi_h\,, \end{cases}$$

where

$$J_h(y_h, u_h) = \frac{1}{2}(y_h - z_{dh})^t M_{1h}(y_h - z_{dh}) + \frac{\alpha}{2}(u_h - u_{dh})^t M_{2h}(u_h - u_{dh}) \tag{3.1}$$

and $M_{1h} \in \mathbb{R}^{N\times N}$ is positive-semidefinite and $M_{2h} \in \mathbb{R}^{N\times N}$ is positive-definite. The matrices M_{1h} and M_{2h} result from the numerical integration of the cost functional J. For finite difference approximation with integration based on the trapezoidal rule, M_{1h} and M_{2h} are positive-definite diagonal matrices. It is simple to derive the optimality system for (P_h^c). Note that the discretized version of (2.1) is given by

$$\lambda_h^* = c \max\left(0, u_h^* + \frac{\lambda_h^*}{c} - \psi_h\right), \quad c > 0\,, \tag{3.2}$$

where the max operator is understood componentwise. The essential ingredient of the Moreau–Yosida-based algorithm to solve (P_h^c) is a primal-dual active set strategy that is motivated by (3.2); see [BIK]. To describe the key step of this algorithm let us assume that $(u_{h,n-1}, \lambda_{h,n-1})$ is available from the previous iteration level. Then (3.2) suggests to define

$$\mathcal{A}_n = \left\{ i \ \mid \ \left(u_{h,n-1} + \frac{\lambda_{h,n-1}}{c}\right)_i > (\psi_h)_i \right\} \quad \text{and} \quad \mathcal{I}_n = \Omega_h \setminus \mathcal{A}_n\,, \tag{3.3}$$

i.e., $\mathcal{A}_n$ consists of the set of indices i such that the ith coordinate of $u_{h,n-1} + c^{-1}\lambda_{h,n-1} - \psi_h$ is positive, and $\mathcal{I}_n$ is its complement. We refer to $\mathcal{A}_n$ as the active set at the nth iteration level and to $\mathcal{I}_n$ as the inactive set. In the statement of the algorithm below, the subscript h will be dropped for the problem variables $(u_h, y_h, p_h, \lambda_h)$. We remain to indicate the discretization for the problem data Ω_h, z_{dh}, ψ_h, u_{dh}, M_{1h}, and M_{2h}.

ALGORITHM (BIK^c).

1. Initialization: choose $y_o, \lambda_o, u_o \in \mathbb{R}^N$, $c > 0$, and set $n = 1$.
2. Determine the subset of active/inactive indices according to (3.3).
3. If $n \ge 2$ and $\mathcal{A}_n = \mathcal{A}_{n-1}$, then STOP; otherwise go to step 4.

4. Find (y_n, p_n) such that

$$(A_h y_n)_i = \begin{cases} (\psi_h)_i & \text{for } i \in \mathcal{A}_n, \\ \left(u_{dh} + \frac{1}{\alpha} M_{2h}^{-1} p_n\right)_i & \text{for } i \in \mathcal{I}_n, \end{cases}$$

$$A_h p_n = -M_{1h}(y_n - z_{dh}),$$

and set

$$(u_n)_i = \begin{cases} (\psi_h)_i & \text{for } i \in \mathcal{A}_n, \\ \left(u_{dh} + \frac{1}{\alpha} M_{2h}^{-1} p_n\right)_i & \text{for } i \in \mathcal{I}_n. \end{cases}$$

5. Set $\lambda_n = p_n - \alpha M_{2h}(u_n - u_{dh})$, $n = n + 1$, and go to step 2.

Algorithm (BIK^c) was first analyzed and tested in [BIK]. Before we recapitulate some of its basic properties let us turn to the discretized state constrained problem

$$(P_h^s) \qquad \begin{cases} \text{minimize} & J_h(y_h, u_h) \\ \text{subject to} & A_h y_h = u_h,\ y_h \le \varphi_h\,, \end{cases}$$

where J_h is defined in (3.1). The optimality system for (P_h^s) is given by

$$\begin{cases} A_h y_h = u_h, \\ A_h p_h + M_{1h}(y_h - z_{dh}) + \lambda_h = 0, \\ \alpha M_{2h}(u_h - u_{dh}) = p_h, \\ \lambda_h^t(\bar{y}_h - y_h) \le 0 \quad \text{for all } \bar{y}_h \in \mathbb{R}^N, \bar{y}_h \le \varphi_h. \end{cases}$$

In analogy to (3.3) the active and inactive sets are now updated according to

$$\mathcal{A}_n = \left\{ i \;\middle|\; \left(y_{h,n-1} + \frac{\lambda_{h,n-1}}{c} \right)_i > (\varphi_h)_i \right\} \quad \text{and} \quad \mathcal{I}_n = \Omega_h \setminus \mathcal{A}_n\,. \tag{3.4}$$

The Moreau–Yosida-based algorithm for the discretized state constrained problem (P_h^s) is specified next. Again we drop the subscript h for problem variables and keep it for problem data.

Algorithm (BIK^s).

1. Initialization: choose $y_o, \lambda_o, u_o \in \mathbb{R}^N$, $c > 0$, and set $n = 1$.
2. Determine the subset of active/inactive indices according to (3.4).
3. If $n \ge 2$ and $\mathcal{A}_n = \mathcal{A}_{n-1}$, then STOP; otherwise go to step 4.
4. Find $(y_n, u_n, p_n, \lambda_n)$ as the solution to

$$\begin{aligned} &A_h y_n = u_n, \\ &A_h p_n + \lambda_n + M_{1h}(y_n - z_{dh}) = 0, \\ &p_n = \alpha M_{2h}(u_n - u_{dh}), \\ &(y_n)_i = (\varphi_h)_i \text{ for } i \in \mathcal{A}_n,\ (\lambda_n)_i = 0 \text{ for } i \in \mathcal{I}_n. \end{aligned}$$

5. Set $n = n + 1$ and go to step 2.

For finite element discretizations the relationship between $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$ to the discretized problems (P_h^c) and (P_h^s) was treated in several publications; see [Be, TT], for instance. In [Ba] general conditions on finite element discretizations are discussed which guarantee that $\lim_{h\to 0}(\|u_h^* - u^*\|_{L^2(\Omega)} + \|y_h^* - y^*\|_{H_o^1(\Omega)}) = 0$, where (y_h^*, u_h^*) denote the solutions to (P_h^c) and (P_h^s), respectively. For finite difference discretizations, convergence for the control constrained case will be discussed elsewhere.

Sufficient conditions for the convergence of (BIK^c) and (BIK^s) in finitely many steps were given in [BIK, BK2, BK3]. For the control constrained problem, convergence could also be proved for the infinite-dimensional analogue of (BIK^c); see [BIK, section 3.1]. Due to the difficulties related to the fact that in the state constrained case the Lagrange multiplier is only a measure, convergence of the infinite-dimensional version of (BIK^s) was not yet analyzed successfully.

Let us summarize some properties of (BIK^c) and (BIK^s):

- The iterates can be infeasible (both primal and dual variables).
- The algorithms do not rely on a globalization strategy.
- For $n > 1$ the algorithms are independent of c.
- If the algorithms stop in step 3, then the exact solution of the discretized problems are obtained [BIK, BK2].
- Utilizing $(y_h)_i = (\varphi_h)_i$ for $i \in \mathcal{A}_n$ the primal system in step 4 of (BIK^s) need only be solved for $i \in \mathcal{I}_n$.
- If $(\bar{y}_h, \bar{u}_h) = \operatorname{argmin}\{J_h(y_h, u_h) : A_h y_h = u_h\}$ satisfies $\bar{u}_h \le \psi_h$ and $\mathcal{A}_o = \emptyset$, then (BIK^c) computes the solution in one step. An analogous statement holds for (BIK^s).
- In our numerical tests we found that (BIK^c) or (BIK^s) typically terminate in step 3. The exceptional cases will be addressed in section 5.
- Both (BIK^c) and (BIK^s) have the property that from one iteration to the next many coordinates of the discretized control or state vector can move from $\mathcal{A}_{n-1}$ to $\mathcal{I}_n$ and vice versa, respectively. This correction process is especially efficient for control constrained problems where it may change the (discrete) interior of $\mathcal{A}_{n-1}$ or $\mathcal{I}_{n-1}$ from one iteration to the next. For state constrained problems changes from active to inactive sets occur primarily along the boundary between active and inactive sets. This is due to the fact that λ_h for (BIK^s) is the discretization of the measure $\lambda^* \in \mathcal{M}(\Omega)$ whose singular part is concentrated at the boundary of the active set [BK2].
- For the iterates u_n of (BIK^c) it is typically the case that they are all infeasible except the last one, i.e., (BIK^c) stops at the first feasible iterate (except for u_o).
- For (BIK^s), on the other hand, the iterates are mostly feasible and the active sets $\mathcal{A}_n$ typically approximate $\mathcal{A}_h^*$ from outside. This approximation is typically monotone with respect to the cardinality of the active set but nonmonotone in the setwise sense.

For several examples we observed that (BIK^c) converged in only a very few iterations—possibly in one. If the solution leaves the inequality constraints inactive and $\mathcal{A}_o = \emptyset$, then this is clear from the structure of the algorithms. But one-step convergence can be proved under more general conditions. This is the contents of the following result for which we decompose the active set $\mathcal{A}_h^* = \{i|(u_h^*)_i = (\psi_h)_i\}$ into the strongly and weakly active set, i.e.,

$$\mathcal{A}_h^{*,+} = \{i \in \mathcal{A}_h^* | (\lambda_h^*)_i > 0\} \quad \text{and} \quad \mathcal{A}_h^{*,0} = \{i \in \mathcal{A}_h^* | (\lambda_h^*)_i = 0\}.$$

By $\mathcal{I}_h^* = \{i|(u_h^*)_i < (\psi_h)_i\}$ we denote the inactive set. Note that $\mathcal{A}_h^{*,0}$ is the set where strict complementarity is violated.

THEOREM 3.1 (for BIK^c). *Assume that A_h is an M-matrix; $u_o \le \psi_h$ is chosen*

such that

$$\begin{aligned}&\left(\lambda_h^* + \alpha M_{2h}(u_h^* - u_o) + A_h^{-1}M_{1h}A_h^{-1}(u_h^* - u_o)\right)_i > c(\psi_h - u_o)_i \text{ for all } i \in \mathcal{A}_h^{*,+},\\ &\left(\alpha M_{2h}(u_h^* - u_o) + A_h^{-1}M_{1h}A_h^{-1}(u_h^* - u_o)\right)_i \le c(\psi_h - u_o)_i \text{ for all } i \in \mathcal{I}_h^* \cup \mathcal{A}_h^{*,0},\end{aligned} \tag{3.5}$$

where $c > 0$, and the remaining variables y_o, p_o, and λ_o are initialized by solving

$$A_h y_o = u_o, \tag{3.6}$$

$$A_h p_o = -M_{1h}(y_o - z_{dh}), \tag{3.7}$$

$$\lambda_o = \max\{p_o - \alpha M_{2h}(u_o - u_{dh}), 0\}\,. \tag{3.8}$$

Then (BIK^c) *is one-step convergent.*

Proof. The discrete first-order optimality conditions for (P_h^c) are

$$A_h y_h^* = u_h^*, \tag{3.9}$$

$$A_h p_h^* = -M_{1h}(y_h^* - z_{dh}), \tag{3.10}$$

$$\alpha M_{2h}(u_h^* - u_{dh}) = p_h^* - \lambda_h^*, \tag{3.11}$$

$$\lambda_h^* = \max\{u_h^* + c^{-1}\lambda_h^* - \psi_h, 0\}\,. \tag{3.12}$$

From (3.9)–(3.11) we derive

$$u_{dh} = u_h^* + \alpha^{-1}M_{2h}^{-1}\left(\lambda_h^* + A_h^{-1}M_{1h}(A_h^{-1}u_h^* - z_{dh})\right). \tag{3.13}$$

Using (3.13) and $p_o = -A_h^{-1}M_{1h}(A_h^{-1}u_o - z_{dh})$ (cf. (3.6) and (3.7)) in (3.8) yields

$$\lambda_o = \max\{\lambda_h^* + \alpha M_{2h}(u_h^* - u_o) + A_h^{-1}M_{1h}A_h^{-1}(u_h^* - u_o), 0\}\,. \tag{3.14}$$

Thus (3.5) yields

$$(\lambda_o)_i = \left(\lambda_h^* + \alpha M_{2h}(u_h^* - u_o) + A_h^{-1}M_{1h}A_h^{-1}(u_h^* - u_o)\right)_i > c(\psi_h - u_o)_i \ge 0$$

for all $i \in \mathcal{A}_h^{*,+}$, and $(\lambda_o)_i \le c(\psi_h - u_o)_i$ for all $i \in \mathcal{I}_h^* \cup \mathcal{A}_h^{*,0}$. Therefore the determination of $\mathcal{A}_1$ and $\mathcal{I}_1$ in algorithm (BIK^c) yields

$$\begin{aligned}\mathcal{A}_1 &= \{i | (u_o + c^{-1}\lambda_o)_i > (\psi_h)_i\} = \mathcal{A}_h^{*,+}\,,\\ \mathcal{I}_1 &= \{i | (u_o + c^{-1}\lambda_o)_i \le (\psi_h)_i\} = \mathcal{I}_h^* \cup \mathcal{A}_h^{*,0}\,.\end{aligned}$$

Hence, $(\lambda_1)_i = (\lambda_h^*)_i = 0$ is set for all $i \in \mathcal{I}_1 = \mathcal{I}_h^* \cup \mathcal{A}_h^{*,0}$ and $(u_1)_i = (u_h^*)_i = (\psi_h)_i$ is set for all $i \in \mathcal{A}_h^1 = \mathcal{A}_h^{*,+}$. From step 4 of (BIK^c) we obtain $y_1 = y_h^*$ and $p_1 = p_h^*$. From $u_1 = u_{dh} + \alpha^{-1}M_{2h}^{-1}(p_1 - \lambda_1)$ we get $(\lambda_1)_i = (\lambda_h^*)_i$ for all $i \in \mathcal{A}_1$ and $(u_1)_i = (u_h^*)_i$ for all $i \in \mathcal{I}_1$. ∎

COROLLARY 3.2. *Assume that A_h is an M-matrix, that the trapezoidal rule is applied for discretizing $J(y,u)$, and that the algorithm is initialized by $u_o = \psi_h \in \mathbb{R}^N$ and* (3.6)–(3.8). *Then if*

$$(\lambda_h^*)_i + \left(\alpha M_{2h}(u_h^* - \psi_h) + A_h^{-1}M_{1h}A_h^{-1}(u_h^* - \psi_h)\right)_i > 0 \quad \text{for all } i \in \mathcal{A}_h^{*,+} \tag{3.15}$$

is satisfied, (BIK^c) *is one-step convergent.*

Proof. Since A_h is M-matrix, from a standard result about the inverse of M-matrices, we obtain $A_h^{-1} \ge 0$. Applying the trapezoidal rule for discretizing J yields $M_{1h} = M_{2h} = h^2 I$. Hence, from $\psi_h = u_o$ we obtain $A_h^{-1}M_{1h}A_h^{-1}(u_h^* - u_o) \le 0$ and

$$\left(\alpha M_{2h} + A_h^{-1}M_{1h}A_h^{-1}\right)(u_h^* - u_o) \le 0 = c(\psi_h - u_o)\,.$$

Thus, (3.5) is satisfied and Theorem 3.1 yields the result. □

To interpret (3.15) note that λ_h^* is nonnegative and $u_h^* = u_o$ on $\mathcal{A}_h^{*,+}$. Thus (3.15) is a condition on the influence of the negative values of $u_h^* - \psi_h$ on $\mathcal{I}_h^*$, which are distributed and strongly damped by the action of the approximation A_h^{-1} to Δ^{-1}. Thus (3.15) represents a type of strict complementarity condition.

For the state constrained case the conditions

$$(3.16)\quad (y_o - y_h^*)_i > 0 \text{ for all } i \in \mathcal{A}_h^{*,+} \quad \text{and} \quad (y_o - y_h^*)_i \le 0 \text{ for all } i \in \mathcal{I}_h^* \cup \mathcal{A}_h^{*,0}$$

are sufficient for one-step convergence of (BIK^s) provided that the algorithm is initialized with the solution of (P_h^s) without the inequality constraint and $\lambda_o := 0$. Here $\mathcal{A}_h^{*,+}$, $\mathcal{A}_h^{*,0}$, and $\mathcal{I}_h^*$ denote the strongly, weakly, and inactive sets, respectively, analogously defined to the corresponding sets for the control constrained case. These conditions are highly unlikely to be satisfied unless $\mathcal{A}_h^{*,+} = \Omega_h$. In section 5 we discuss a test problem which nearly satisfies (3.16) and for which (BIK^s) shows one-step convergence.

4. Primal-dual path-following interior point methods. In this section, we outline two different representatives of interior point methods applied to the class of constrained optimal control problems considered in this paper. Both algorithms fall into the framework of primal-dual path-following methods, which turn out to be robust techniques; see, for instance, [VY]. Moreover, both algorithms are based on predictor-corrector strategies; see [BG, BPR, G, LMS, Meh, MTY, V2, Wr, Y, Z2] and the references therein. We shall go into some details describing the linear algebra specific for such problems involving two sets of independent variables, namely, the state variable y_h and the control variable u_h. We stress the fact that the first interior point algorithm is exemplarily derived for control constrained problems, but it can readily be accommodated to state constraints, whereas the second interior point method is exemplarily derived for state constraints, and it can readily be accommodated to control constraints as well.

As interior point methods are developed in the finite-dimensional context, the starting point of our application of primal-dual path following are the discretized, finite-dimensional versions of $(\mathcal{P}^c)$ and $(\mathcal{P}^s)$ given by (P_h^c) and (P_h^s), respectively. Concerning the matrices A_h, M_{1h}, and M_{2h} we henceforth utilize the following assumption:

$$(A)\qquad \begin{cases} A_h \text{ is sparse banded, symmetric, and positive-definite,} \\ M_{1h} \text{ is sparse banded and positive-semidefinite,} \\ M_{2h} \text{ is sparse banded and positive-definite.} \end{cases}$$

We remark that the discretization of $-\Delta$ by the well-known five-point star scheme implies that A_h is a symmetric positive-definite banded M-matrix. Applying the trapezoidal rule for approximating J implies that M_{1h} and M_{2h} are positive-definite and diagonal.

Following the convention already used in section 3 we omit h for problem variables and keep it for problem data throughout this section.

4.1. First interior point algorithm. We start by deriving a primal-dual path-following interior point algorithm for the control constrained problem (P_h^c). The subsequent algorithm is a modification of the algorithms in [Meh], [Wr, Algorithm MPC]. The difference consists essentially in adaptation to convex quadratic programming problems, including linear algebra issues taking advantage of the special problem

structure considered in this paper. We have already mentioned that the subsequent technique can readily be applied to state constrained problems. Numerical results for both types of constraints can be found in section 5.

Introducing a vector of slack variables $w \geq 0$, $w \in \mathbb{R}^N$ in the inequality constraint in (P_h^c) and splitting up the free variable y we obtain

$$u + w = \psi_h, \quad w \geq 0, \tag{4.1}$$

$$y - g + t = 0, \quad g, t \geq 0\,. \tag{4.2}$$

Let us note that splitting techniques for general purpose optimization are also revisited by Vanderbei and are incorporated in his code LOQO; see [V1] for a brief discussion on the splitting of free variables. We shall resume this discussion later in this section. The modifications (4.1) and (4.2) of (P_h^c) lead to

$$(P_h^{c,\mathrm{mod}}) \quad \begin{cases} \text{minimize} & \frac{1}{2}(y - z_{dh})^t M_{1h}(y - z_{dh}) + \frac{\alpha}{2}(u - u_{dh})^t M_{2h}(u - u_{dh}) \\ \text{subject to} & B_h y - h^2 u = 0, \\ & \psi_h - u - w = 0, \\ & y - g + t = 0, \\ & w, g, t \geq 0, \end{cases}$$

where $B_h := h^2 A_h \in \mathbb{R}^{N \times N}$. Note further that there exist feasible solutions of $(P_h^{c,\mathrm{mod}})$. Thus, the convex problem $(P_h^{c,\mathrm{mod}})$ admits an optimal solution. The corresponding first-order Karush–Kuhn–Tucker conditions are equivalent to

$$\begin{cases} B_h y - h^2 u &= 0, \\ \psi_h - u - w &= 0, \\ y - g + t &= 0, \\ w, g, t &\geq 0, \\ M_{1h}(y - z_{dh}) - B_h p - r_1 &= 0, \\ \alpha M_{2h}(u - u_{dh}) + h^2 p + \lambda &= 0, \\ r_1 + r_2 &= 0, \\ \lambda, r_1, r_2 &\geq 0, \\ W\Lambda e &= 0, \\ G R_1 e &= 0, \\ T R_2 e &= 0, \end{cases} \tag{4.3}$$

where we used the common convention that W, G, T, Λ, R_1, and R_2 are diagonal matrices with elements $(w)_i$, $(g)_i$, $(t)_i$, $(\lambda)_i$, $(r_1)_i$, $(r_2)_i$, where $i = 1, \ldots, N$, and e denotes the vector of all ones in $\mathbb{R}^N$.

Next define the orthant $\mathcal{O} = \{(y, u, \ldots, r_2) \mid w \geq 0,\, g \geq 0,\, t \geq 0,\, \lambda \geq 0,\, r_j \geq 0,\, j = 1, 2\}$ in primal-dual space, and let $\mathcal{O}^o$ denote its strict interior. The defining equations for a point $(y, \ldots, r_2) \in \mathcal{O}^o$ on the primal-dual central path (which was introduced in [Meg]) are

$$\begin{cases} B_h y & -h^2 u & & & & & & & & = 0, \\ & u & +w & & & & & & & = \psi_h, \\ y & & & -g & +t & & & & & = 0, \\ M_{1h} y & & & & & -B_h p & & -r_1 & & = M_{1h} z_{dh}, \\ & \alpha M_{2h} u & & & & +h^2 p & +\lambda & & & = \alpha M_{2h} u_{dh}, \\ & & & & & & & r_1 & +r_2 = 0, & \\ & & W\Lambda e = \mu e, & & G R_1 e = \mu e, & & T R_2 e = \mu e\,, & & & \end{cases} \tag{4.4}$$

where μ denotes a strictly positive parameter. Observe that (4.4) is obtained from (4.3) by relaxing the last three equations and neglecting nonnegativity, which is enforced by the definition of $\mathcal{O}$. Further note that (4.4) is a nonlinear system of $9N$ equations in $9N$ unknowns and has a unique solution in $\mathcal{O}^o$. (To see this consider the fact that (4.4) denotes the first-order conditions for a strictly convex barrier problem.) A motivation and comprehensive discussion of the primal-dual central path can be found, for example, in [Meg, V2, Wr, Y] and the references therein.

Suppose that we have decided on the target value for μ and that $(y, u, \ldots, r_2)$ is the actual point in $\mathcal{O}^o$. Let $(y + \Delta y, u + \Delta u, \ldots, r_2 + \Delta r_2) \in \mathcal{O}^o$ denote the point on the central path corresponding to the value of μ. Thus $(y + \Delta y, \ldots, r_2 + \Delta r_2)$ satisfies (4.4), from which, after negating and rearranging certain rows, we obtain the following symmetric system for the increments $(\Delta y, \ldots, \Delta r_2)$:

$$
\begin{bmatrix}
 & & & & & & & & I & I \\
-G^{-1}R_1 & & & & & & & & & -I \\
 & -W^{-1}\Lambda & & & & & -I & & & \\
 & & -\alpha M_{2h} & & -h^2 I & & -I & & & \\
 & & & -M_{1h} & B_h & & & & I & \\
 & & -h^2 I & B_h & & & & & & \\
 & -I & -I & & & & & & & \\
I & & & & & & & R_2^{-1}T & & \\
I & -I & & & I & & & & &
\end{bmatrix}
\begin{bmatrix} \Delta t \\ \Delta g \\ \Delta w \\ \Delta u \\ \Delta y \\ \Delta p \\ \Delta \lambda \\ \Delta r_2 \\ \Delta r_1 \end{bmatrix}
=
\begin{bmatrix} \beta_3 \\ -\gamma_2 \\ -\gamma_1 \\ -\beta_2 \\ -\beta_1 \\ \alpha_1 \\ \alpha_2 \\ \gamma_3 \\ \alpha_3 \end{bmatrix}, \tag{4.5}
$$

where

$$
\begin{aligned}
\alpha_1 &= -B_h y + h^2 u, & \beta_1 &= -M_{1h}(y - z_{dh}) + B_h p + r_1, \\
\alpha_2 &= -\psi_h + u + w, & \beta_2 &= -\alpha M_{2h}(u - u_{dh}) - h^2 p - \lambda, \\
\alpha_3 &= -y + g - t, & \beta_3 &= -r_1 - r_2,
\end{aligned}
$$

$$
\begin{aligned}
\gamma_1 &= \mu(W^{-1} - \Lambda - W^{-1}\Delta\Lambda\Delta W)e, \\
\gamma_2 &= \mu(G^{-1} - R_1 - G^{-1}\Delta R_1 \Delta G)e, \\
\gamma_3 &= \mu(R_2^{-1} - T - R_2^{-1}\Delta R_2 \Delta T)e.
\end{aligned}
$$

The predictor-corrector idea pursued in Algorithm (IP1) below consists in obtaining an estimate of a suitable target value μ from the predictor step (affine scaling direction), and then in using this target value for the corrector step, which improves centrality and compensates some of the nonlinearity. For a detailed discussion on predictor-corrector algorithms (of this type) we refer to [LMS, Meh, V1, Wr, ZZ, Z2] and the references therein.

Next we describe the primal-dual path-following predictor-corrector algorithm where we use the following notation: $z_n := (u_n, y_n)$, $v_n := (t_n, g_n, w_n)$, $\omega_n := (r_{2,n}, r_{1,n}, \lambda_n)$ with subscript n denoting the actual iteration level. In the algorithm we use $\mu_n = (v_n^t \omega_n)/(3N)$, which is a common way to obtain an estimate of the central path parameter; see, for instance, [LMS, Meh, V1, Wr, VY, Z2].

Algorithm (IP1).

1. *Initialization*: Choose $v_o > 0$, $\omega_o > 0$, z_o, p_o, a stopping tolerance $\varepsilon_s > 0$, $e_f \in \mathbb{N}_+$, and set $n := 0$.
2. *Check stopping criteria*: If

$$
\mathrm{res}^p := \frac{\|(\alpha_{1,n}, \alpha_{2,n}, \alpha_{3,n})^t\|_1}{1 + \|\psi_h\|_1} \le \varepsilon_s, \; \mathrm{res}^d := \frac{\|(\beta_{1,n}, \beta_{2,n}, \beta_{3,n})^t\|_1}{1 + \|(M_{1h} z_{dh}, \alpha M_{2h} u_{dh})^t\|_1} \le \varepsilon_s,
$$

and $f^n := \max\{-\log_{10}[(J_n - \phi_n)/(1+|J_n|)], 0\} \geq e_f$, then STOP; otherwise go to step 3.

3. *Predictor step* $\Delta z_a, \Delta v_a, \Delta p_a, \Delta \omega_a$: Solve (4.5) with right-hand side $(\beta_{3,n},\ r_{1,n},\ \lambda_n,\ -\beta_{2,n},\ -\beta_{1,n},\ \alpha_{1,n},\ \alpha_{2,n},\ -t_n,\ \alpha_{3,n})^t$. Calculate the step τ_a to the boundary along Δv_a, $\Delta \omega_a$ and the estimate μ_a by

$$\begin{aligned}
\tau_a^p &:= \operatorname{argmax}\{\tau \in [0,1] | v_n + \tau \Delta v_a \geq 0\},\\
\tau_a^d &:= \operatorname{argmax}\{\tau \in [0,1] | \omega_n + \tau \Delta \omega_a \geq 0\},\\
\tau_a &:= \min\{\tau_a^p, \tau_a^d\},\\
\mu_a &:= \frac{(v_n + \tau_a \Delta v_a)^t (\omega_n + \tau_a \Delta \omega_a)}{3N}, \quad \sigma := \left(\frac{\mu_a}{\mu_n}\right)^3.
\end{aligned}$$

4. *Corrector step* $\Delta z_c, \Delta v_c, \Delta p_c, \Delta \omega_c$: Solve (4.5) with right-hand side $r_c := r_{ce} + r_{co}$, where

$$\begin{aligned}
r_{ce} &:= (0, -\sigma\mu_n e^t G_n^{-1}, -\sigma\mu_n e^t W_n^{-1}, 0, 0, 0, 0, \sigma\mu_n e^t R_{2,n}^{-1}, 0)^t,\\
r_{co} &:= (0, e^t G_n^{-1} \Delta R_{1,a} \Delta G_a, e^t W_n^{-1} \Delta \Lambda_a \Delta W_a, 0, 0, 0, 0, -e^t R_{2,n}^{-1} \Delta T_a \Delta R_{2,a}, 0)^t.
\end{aligned}$$

5. *Search direction and step*: Put

$$(\Delta z_n, \Delta v_n, \Delta p_n, \Delta \omega_n) := (\Delta z_a, \Delta v_a, \Delta p_a, \Delta \omega_a) + (\Delta z_c, \Delta v_c, \Delta p_c, \Delta \omega_c)$$

and compute the step length τ_n by

$$\begin{aligned}
\tau_n^p &:= \operatorname{argmax}\{\tau \in [0,1] | v_n + \tau \Delta v_n \geq 0\},\\
\tau_n^d &:= \operatorname{argmax}\{\tau \in [0,1] | \omega_n + \tau \Delta \omega_n \geq 0\},\\
\tau_n^{pd} &:= \min\{\tau_n^p, \tau_n^d\}, \quad \tau_n := \min\{0.99\tau_n^{pd}, 1\}.
\end{aligned}$$

Calculate the new iterates $(v_{n+1}, z_{n+1}) := (v_n, z_n) + \tau_n(\Delta v_n, \Delta z_n)$, $(p_{n+1}, \omega_{n+1}) := (p_n, \omega_n) + \tau_n(\Delta p_n, \Delta \omega_n)$. Set $n := n+1$, and go to step 2.

A few remarks and motivations concerning Algorithm (IP1) are in order:

- The first two stopping criteria in step 2 check the smallness of the relative primal and dual residuals, i.e., of res^p and res^d, respectively. The criterion $f_n \geq e_f$ checks the number of digits of coincidence between the primal objective value, i.e., $J_n = J_h(y_n, u_n)$, and the dual objective value, i.e., $\phi_n = -J_n + p_n^t(h^2 u_{dh} - B_h z_{dh}) + \lambda^t(u_{dh} - \psi_h) - r_2^t z_{dh}$. Note that by standard duality theory the difference between the optimal primal and dual objective values vanishes.
- The computation of the centering parameter σ in step 3 follows a heuristic suggested in [Meh]; see also [Wr, Z2]. Moreover, we also impose an upper bound on σ in each iteration in order to limit σ to a value strictly smaller than one.
- At each iteration (in steps 3 and 4) an efficient solution of the primal-dual systems is of extreme importance. Note that the extra cost of the corrector direction is small since we only have to consider the same system with different right-hand sides. Moreover, due to the sparsity structure of the system matrix of (4.5) (recall also assumption (A)), we realize the so-called normal equations approach [Mes, V1, Wr], i.e., by choosing specific pivots in advance we reduce the large indefinite system (4.5) to the form

$$(4.6)\quad \left(M_{1h} + D + A_h(\alpha M_{2h} + W^{-1}\Lambda)A_h\right)\Delta y = \frac{1}{h^2} A_h(\alpha M_{2h} + W^{-1}\Lambda)\alpha_1 + \beta,$$

where $\beta = -\tilde{\beta}_1 - A_h\tilde{\beta}_2$, with $\tilde{\beta}_1$ and $\tilde{\beta}_2$ given by

$$\tilde{\beta}_1 = -\beta_1 - D(\alpha_3 - \gamma_3 + GR_1^{-1}\gamma_2 + TR_2^{-1}\beta_2)\,, \quad \tilde{\beta}_2 = -\beta_2 + \gamma_1 + W^{-1}\Lambda\alpha_2\,,$$

where $D = (TR_2^{-1} + GR_1^{-1})^{-1}$. In the special case of M_{1h} and M_{2h} being positive-semidefinite and positive-definite diagonal matrices, respectively, one can alternatively use

$$\tilde{A}(H + \tilde{D})^{-1}\tilde{A}^t\Delta p = \alpha_1 + \tilde{A}(H + \tilde{D})^{-1}(\tilde{\beta}_2, \tilde{\beta}_1)^t\,, \tag{4.7}$$

where

$$\tilde{D} := \begin{bmatrix} W^{-1}\Lambda & \\ & D \end{bmatrix}, \quad \tilde{A} := [-h^2 I \quad B_h\,], \quad \text{and} \quad H := \begin{bmatrix} \alpha M_{2h} & \\ & M_{1h} \end{bmatrix}.$$

Note that the approach (4.7) is defined only in cases where that splitting is applied. In fact, if no splitting is used, then $D = 0$, and due to the semidefiniteness of M_{1h} the inverse $(H + \tilde{D})^{-1}$ is not defined in general. On the contrary, (4.6) is always applicable.

Observe that in (4.6) the system matrix is sparse banded and positive-definite due to assumption (A). The system matrix in the alternative case is sparse banded and positive-definite since M_{1h} and M_{2h} are diagonal, respectively. Thus, in each case the system matrix can be factorized efficiently by applying sparse Cholesky techniques (see, for instance, [Z2]) or by multilevel techniques. In the case of (4.6), after having obtained Δy the remaining unknowns Δu, Δp, $\Delta\lambda$, Δt, Δr_1, Δw, Δr_2, and Δg are computed by efficient backward substitution in the order given here. In the case of (4.7) the roles of p and y in the order given before change.

In our numerical tests we prefer to use—if possible—(4.7). A partially heuristic explanation is as follows: Suppose that $\mathrm{cond}(M_{1h}) = O(1)$, where $\mathrm{cond}(M_{1h})$ denotes the condition number of M_{1h}, and that A_h comes from a five-point star finite difference approximation to $-\Delta$. Then it is well known that $\mathrm{cond}(B_h) = O(h^{-2})$. In the case of (4.6) without splitting, the condition number of the system matrix $M_{1h} + A_h(\alpha M_{2h} + W^{-1}\Lambda)A_h$ is dominated by $A_h(\alpha M_{2h} + W^{-1}\Lambda)A_h$. We have $\mathrm{cond}(A_h) \approx O(h^{-2})$ and $\mathrm{cond}(A_h^2) \approx O(h^{-4})$, which becomes even worse since $W^{-1}\Lambda$ becomes increasingly ill-conditioned in the course of the iteration. In the case of (4.7), let us first analyze $(H + \tilde{D})^{-1}$, which is diagonal and consists of the two blocks $(\alpha M_{2h} + W^{-1}\Lambda)^{-1}$ and $(M_{1h} + D)^{-1}$. Thus, the system matrix can be written as $h^4(\alpha M_{2h} + W^{-1}\Lambda)^{-1} + B_h(M_{1h} + D)^{-1}B_h$. The first term suffers from the ill-conditioning of $W^{-1}\Lambda$ only. For the second term we observe that the diagonal elements of $(M_{1h} + D)^{-1}$ are

$$(a)_i := \frac{(t)_i(r_1)_i + (g)_i(r_2)_i}{(m_{1h})_i[(t)_i(r_1)_i + (g)_i(r_2)_i] + (r_1)_i(r_2)_i} \le \frac{1}{(m_{1h})_i}\,,$$

where $M_{1h} = \mathrm{diag}(m_{1h})$ with $(m_{1h})_i > 0$ for all $i = 1, \ldots, N$. In our numerical tests we typically observe that the dual variables $(r_1)_i$ and $(r_2)_i$ approach zero faster than the corresponding primal variables $(g)_i$ and $(t)_i$, respectively; see [V1] for a similar observation in the case of general convex quadratic problems. Therefore, $(a_n)_i \to 1/(m_{1h})_i > 0$ for all i. Due to $\mathrm{cond}(M_{1h}) = O(1)$, we have that $\mathrm{cond}(B_h(M_{1h} + D)^{-1}B_h)$ approaches $O(h^{-4})$ in the course of

the iteration. In conclusion, we see that the alternative approach (4.7) usually results in better-conditioned system matrices (than for (4.6)) during the iterations.

- If the iterates prematurely get too close to the boundary of $\mathcal{O}$ (and hence some of the pairwise products $(v_{n+1})_i(\omega_{n+1})_i$ are much smaller than their average value μ_{n+1}), then only little progress can be made along the search directions computed there. Typically, the step sizes tend to be rather small in order to keep the iterates in the interior of $\mathcal{O}$. To avoid such behavior, in step 5 we consider a specific neighborhood of the central path, i.e.,

$$\mathcal{N}_{-\infty}(\gamma) := \left\{ (v,\omega) > 0 \,|\, (v)_i(\omega)_i \geq \gamma\mu, i = 1,\dots,3N, \text{ where } \mu = \frac{v^t\omega}{3N} \right\}. \tag{4.8}$$

  A reasonable value for the positive quantity γ is $\gamma := 0.001$. Whenever (v_{n+1}, w_{n+1}) fails to be in $\mathcal{N}_{-\infty}(\gamma)$ (preassuming that $(v_n, \omega_n) \in \mathcal{N}_{-\infty}(\gamma)$), the step length τ_n is reduced such that $(v_{n+1}, w_{n+1}) \in \mathcal{N}_{-\infty}(\gamma)$ is satisfied. In order to make the requirement that the iterates have to stay in $\mathcal{N}_{-\infty}(\gamma)$ less stringent during later iterations (and thus allowing fast progress), we decrease γ whenever a significant step along the actual direction is taken, i.e., $\gamma := (1-\kappa)\gamma_{\min} + \kappa\gamma$ if $\tau_n > 1 - \bar{\tau}$. Typical values are $\gamma_{\min} = 0.0001$, $0 < \kappa \leq 0.5$, and $\bar{\tau} := 0.4$. Usually we start with $\gamma = 0.01$. Note that the requirement $(v_n, \omega_n) \in \mathcal{N}_{-\infty}(\gamma)$ for all n is similar to the centering conditions in [ZZ, Z1], and is implemented (with a small but fixed γ) in LIPSOL [Z2].
- Let us note that the key assumptions of [ZZ] (see also [Z1]) for proving convergence and complexity results of Algorithm (IP1) (with a slightly more specific step-size rule for the centered corrector direction in step 4) are satisfied. In fact, there exist points satisfying (4.4) except for the last three equations. This can be seen from feasibility of $(P_h^{c,\mathrm{mod}})$ and its dual. Further, after eliminating y, u, and p, the homogeneous version of (4.4) without the last three equations becomes

$$\begin{bmatrix} I & A_h & -A_h \\ -\alpha A_h M_{2h} & M_{1h} & -M_{1h} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w \\ g \\ t \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ A_h & -I & 0 \\ 0 & I & I \end{bmatrix} \begin{bmatrix} \lambda \\ r_1 \\ r_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

  implying $w^t\lambda + g^t r_1 + t^t r_2 \geq 0$ due to assumption (A).

4.2. Second interior point algorithm. The second representative of interior point algorithms is a (large neighborhood) modification of the algorithm in [MTY]. See also [BPR], where the corresponding convergence analysis can be found. It will be exemplarily presented for state constrained problems. However, adaptation to control constraints is straightforward. At the end of this subsection we shall discuss some of the fundamental differences between Algorithm (IP1) and the algorithm presented subsequently.

Introducing a vector of slack variables denoted by $w \in \mathbb{R}^N$ in the inequality constraint in (P_h^s), we obtained the following modification:

$$(P_h^{s,\mathrm{mod}}) \qquad \begin{cases} \text{minimize} & J_h(y,u) \\ \text{subject to} & B_h y - h^2 u = 0, \\ & y + w = \varphi_h, \\ & w \geq 0\,, \end{cases}$$

with B_h as in section 4.1. By similar arguments as in section 4.1 we see that the convex problem $(P_h^{s,\mathrm{mod}})$ admits an optimal solution, which is characterized by the following system equivalent to the first-order Karush–Kuhn–Tucker conditions:

$$\left\{\begin{array}{rcl} B_h w + h^2 u - B_h \varphi_h &=& 0, \\ w &\geq& 0, \\ M_{1h} w + B_h p - \lambda - M_{1h}(\varphi_h - z_{dh}) &=& 0, \\ \alpha M_{2h}(u - u_{dh}) + h^2 p &=& 0, \\ \lambda &\geq& 0, \\ \Lambda W e &=& 0\,. \end{array}\right. \tag{4.9}$$

Note that the variable u is subject to no splitting.

The defining equations for a point on the primal-dual central path are obtained by replacing the last equation in (4.9) by $\Lambda W e = \mu e$, with μ being some positive parameter. By similar arguments as before it can be seen that the resulting $(4N \times 4N)$-system admits a unique solution. As in section 4.1 we suppose that we have decided on a target value for μ, that $(w, u, p, \lambda) \in \mathbb{R}^{4N}$ satisfies $w, \lambda > 0$, and that $(w+\Delta w, \ldots, \lambda+\Delta\lambda)$ denotes the point on the primal-dual central path corresponding to μ. Thus, we obtain the following system for the increments $(\Delta w, \ldots, \Delta\lambda)$:

$$(S^s(w,\lambda)) \qquad \begin{bmatrix} \alpha M_{2h} & & h^2 I & \\ & M_{1h} & B_h & -I \\ h^2 I & B_h & & \\ & \Lambda & & W \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta w \\ \Delta p \\ \Delta\lambda \end{bmatrix} = \begin{bmatrix} \beta_2 \\ \beta_1 \\ \alpha_1 \\ \gamma_1 \end{bmatrix},$$

with

$$\begin{array}{rclcrcl} \alpha_1 &=& B_h(\varphi_h - w) - h^2 u, & & \beta_1 &=& M_{1h}(\varphi_h - z_{dh} - w) - B_h p + \lambda, \\ \beta_2 &=& \alpha M_{2h}(u_{dh} - u) - h^2 p, & & \gamma_1 &=& \mu e - W\Lambda e - \Delta W \Delta\Lambda e\,. \end{array}$$

We introduce the (large) neighborhood of the central path:

$$\mathcal{N}(\nu) = \{(w, \lambda, \mu) \in \mathbb{R}^N \times \mathbb{R}^N \times \mathbb{R}_+ | \nu\mu e \leq \Lambda W e \leq \nu^{-1}\mu e\}\,,$$

where $0 < \nu < 1$ is a given constant. Typical values for ν are $\nu = 0.01$ and $\nu = 0.001$. In our experiments these values give essentially the same results.

Next we describe the second primal-dual path-following predictor-corrector interior point algorithm. Again, the subscript n denotes the iteration level.

Algorithm (IP2).

1. *Initialization*: Choose $0 < \nu < 1$ and $(w_o, \lambda_o, \mu_o) > 0$, with $(w_o, \lambda_o, \mu_o) \in \mathcal{N}(\nu)$, a stopping tolerance $\varepsilon_s > 0$ and $e_f \in \mathbb{N}_+$. Set $n := 0$.
2. *Check stopping criteria*: If

$$\mathrm{res}^p = \frac{\|\alpha_{1,n}\|_1}{1 + \|B_h\varphi_h\|_1} \leq \varepsilon_s, \quad \mathrm{res}^d = \frac{\|(\beta_{1,n}, \beta_{2,n})^t\|_1}{1 + \|(M_{1h}(\varphi_h - z_{dh}), \alpha M_{2h} u_{dh})^t\|_1} \leq \varepsilon_s,$$

   and $f^n = \max\{-\log_{10}[(J_n - \phi_n)/(1 + |J_n|)], 0\} \geq e_f$, then STOP; otherwise go to step 3.
3. *Corrector step* $\Delta u_c, \Delta w_c, \Delta p_c, \Delta\lambda_c$: Solve $S^s(w_n, \lambda_n)$ with right-hand side $(0, 0, 0, \mu_n e - W_n\Lambda_n e)^t$. Compute $\tau_c \in (0, 1]$ such that $((w_n, \lambda_n) + \tau_c(\Delta w_c, \Delta\lambda_c), \mu_n) \in \mathcal{N}(\nu)$, and put

$$(u_{n+\frac{1}{2}}, w_{n+\frac{1}{2}}, p_{n+\frac{1}{2}}, \lambda_{n+\frac{1}{2}}) = (u_n, w_n, p_n, \lambda_n) + \tau_c(\Delta u_c, \Delta w_c, \Delta p_c, \Delta\lambda_c)\,.$$

4. *Predictor step* $\Delta u_a, \Delta w_a, \Delta p_a, \Delta\lambda_a$: Solve $S^s(w_{n+\frac{1}{2}}, \lambda_{n+\frac{1}{2}})$ with right-hand side $(\beta_{2,n}, \beta_{1,n}, \alpha_{1,n}, -\Delta W_{n+\frac{1}{2}} \Delta\Lambda_{n+\frac{1}{2}} e)^t$. Compute τ_a, the largest value in $(0,1)$ such that $((w_{n+\frac{1}{2}}, \lambda_{n+\frac{1}{2}}) + \tau_a(\Delta w_a, \Delta\lambda_a), (1-\tau_a)\mu_n) \in \mathcal{N}(\nu)$. Put $\mu_{n+1} = (1-\tau_a)\mu_n$ and

$$(u_{n+1}, w_{n+1}, p_{n+1}, \lambda_{n+1}) = (u_{n+\frac{1}{2}}, w_{n+\frac{1}{2}}, p_{n+\frac{1}{2}}, \lambda_{n+\frac{1}{2}}) + \tau_a(\Delta u_a, \Delta w_a, \Delta p_a, \Delta\lambda_a)\,.$$

Set $n = n+1$, and go to step 2.

Let us remark that the stopping criteria in step 2 have the same meaning as the corresponding criteria in Algorithm (IP1). Moreover, the systems in step 3 and 4 are reduced in advance by choosing specific pivots; see also the discussion below. Due to (A) the key assumptions of [BPR] for proving convergence and complexity results for Algorithm (IP2) are satisfied. This follows from arguments analogous to those at the end of section 4.1.

Finally, we point out some of the fundamental differences between (IP1) and (IP2).

- Algorithm (IP2) is based on a more conservative choice of neighborhoods of the central path than (IP1). In fact, the iterates of (IP1) are not restricted to staying close to the central path. In general, smaller neighborhoods result in better complexity-estimates while larger ones in many cases yield better numerical results. For a detailed discussion of the complexity of interior point methods, we refer to [Wr]. Complexity results for (IP1) and (IP2) can be found in [ZZ] and [BPR], respectively.
- Algorithm (IP1) utilizes information of the predictor step in order to adjust the centrality parameter σ appropriately in the corrector step. Algorithm (IP2) performs intermediate updates of the variables and does not adjust the duality measure within one iteration.
- The intermediate update procedure of Algorithm (IP2) results in two matrix factorizations per iteration needed by (IP2), while for (IP1) it suffices to factorize once per iteration.
- The linear system of (IP1) is symmetrized and can be reduced to the sparse symmetric positive-definite system (4.6), or (4.7) in the alternative case. The ill-conditioning introduced by symmetrization is well understood (see [W] and the references therein) and poses no problem in solving the system. On the other hand, the linear system of (IP2) is not symmetrized, and it can be reduced to the sparse system

$$(\Lambda + WM_{1h} + \alpha W A_h M_{2h} A_h)\Delta w = r^s\,, \tag{4.10}$$

where $r^s \in \mathbb{R}^N$ denotes the appropriate right-hand side.

5. Numerical experiments. All algorithms that we presented above were tested for one- and two-dimensional domains, with $\Omega = (0,1)$ and $\Omega = (0,1)\times(0,1)$, respectively. Here we present in some detail selected results from two-dimensional test examples. Results with one-dimensional examples will be briefly discussed. For the sake of achieving an accurate comparison all final tests were performed on the same machine (DEC-alpha 500 with machine precision $\varepsilon_M \approx 1.11 \cdot 10^{-16}$) under MATLAB 5.1.

The discretization of the infinite-dimensional problems was based on a finite difference approximation with equidistant grid with mesh-length h and a five-point star

finite difference approximation to the Laplace operator. The matrices M_{1h} and M_{2h} were chosen as identity matrices of dimension $\mathbb{R}^{N\times N}$. Unless otherwise specified, the stopping tolerances ε_s and e_f of the interior point algorithms were chosen to be $\varepsilon_s = \sqrt{\varepsilon_M}$ and $e_f = 8$.

Next we describe the initialization schemes for (BIK^c), (BIK^s), (IP1), and (IP2). Unless otherwise specified (BIK^c) was initialized by setting $u_o = \psi_h$ and determining (y_o, p_o, λ_o) from (3.6)–(3.8). An alternative is given by solving (P_h^c) without the inequality constraint $u_h \le \psi_h$, i.e., to determine $(u_o, y_o, p_o, \lambda_o)$ from the optimality system

$$(\mathcal{OS}) \qquad \begin{cases} A_h y_o &= u_{dh} + \alpha^{-1} p_o, \\ A_h p_o &= z_{dh} - y_o, \\ u_o &= u_{dh} + \alpha^{-1} p_o,\ \lambda_o = 0\,. \end{cases}$$

The latter choice has the advantage that the first iteration of (BIK^c) will stop at the exact solution to (P_h^c) if $u_o \le \psi_h$. On the other hand using $(\mathcal{OS})$ requires solving a coupled system of equations, as opposed to the first initialization strategy, which depends only on two uncoupled equations. Together with the fact that the first initialization required slightly fewer iterations than the second in the cases when the inequality constraint is active, it is suggested to use the first initialization as the default strategy. – As canonical initialization for (BIK^s) we solved (P_h^s) without the inequality constraint for (y_o, u_o, p_o) and set $\lambda_o = 0$. We remark that we also tested (BIK^c) and (BIK^s) with several other initializations and observed that both algorithms are very robust with respect to different initializations. – The start-up routine for (IP1) sets all variables which initially have to be strictly positive equal to one, i.e., $v_o := e_{3N}$ and $\omega_o := e_{3N}$, with e_{3N} the vector of all ones in $\mathbb{R}^{3N}$. For the unconstrained variables the following initialization was used: $u_o := \psi_h - w_o$, $y_o := A_h^{-1} u_o$, $p_o := \frac{\alpha}{h^2} M_{2h}(u_{dh} - u_o) - \lambda_o$. The choice of v_o and ω_o is intended to provide a well-centered starting point while u_o and y_o satisfy the primal equality constraints without splitting. For this initialization scheme the residuals of the primal and dual equality constraints in our test examples are not too large, and hence it provides a good start-up configuration for fast progress toward the solution [Wr]. The seemingly natural start-up $u_o = y_o = p_o = 0$ led to no enhancement. For other reasonable choices the algorithm proved to be rather independent of the initialization. We point out that due to $r_{1,o} > 0$ and $r_{2,o} > 0$ and the constraint $r_1 + r_2 = 0$ no strictly feasible starting point for (IP1) is available. – Contrary to the last statement, there exist feasible start-up choices for (IP2). For reasons of comparison and simplicity of initialization we used (the infeasible choice) $\lambda_o = e_N$, $w_o = e_N$, and $u_o = A_h(\varphi_h - w_o)$, $p_o = \frac{\alpha}{h^2} M_{2h}(u_{dh} - u_o)$, which again yields rather small primal and dual residuals.

In what follows we denote by (y_{IPi}, u_{IPi}) , $i = 1, 2$, the state-control solution pair obtained from (IPi) and by (y_{BIK}, u_{BIK}) the solution pair obtained from either (BIK^c) or (BIK^s). We further set $d^u_{\infty,i} = \|u_{BIK} - u_{IPi}\|_\infty$, $d^y_{\infty,i} = \|y_{BIK} - y_{IPi}\|_\infty$, and $d^J_i = |J_h(y_{BIK}, u_{BIK}) - J_h(y_{IPi}, u_{IPi})|$ for $i = 1, 2$, where $\|\cdot\|_\infty$ denotes the ℓ^∞-norm.

5.1. Presentation of the examples. In this subsection we specify the examples for which numerical tests will be documented.

5.1.1. Example: Control constraints. The data are $\alpha = 0.01$, $c = 0.1$, $\psi \equiv 0$, $u_d \equiv 0$, and $z_d(x_1, x_2) = \sin(2\pi x_1)\sin(2\pi x_2)\exp(2x_1)/6$. The optimal state and control are depicted in Figure 5.1.

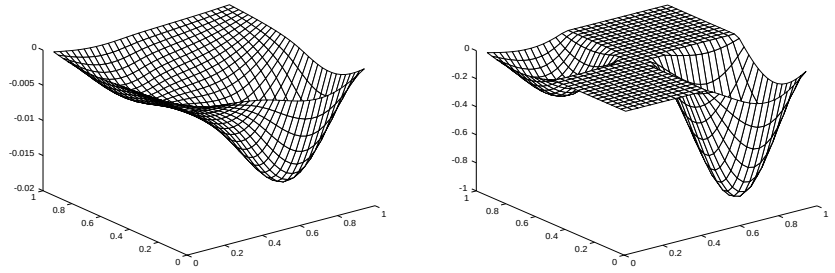


FIG. 5.1. *Optimal state (left graph) and optimal control (right graph) for Example* 5.1.1.

5.1.2. Example: Control constraints. The data are $\alpha = 0.01$, $c = 0.1$, $\psi \equiv 1$, $u_d \equiv 0$, and

$$z_d = \begin{cases} 200\ x_1 x_2\ (x_1 - 0.5)^2\ (1 - x_2) & \text{if } 0 < x_1 \leq 0.5, \\ 200\ x_2\ (x_1 - 1)(x_1 - 0.5)^2\ (1 - x_2) & \text{if } 0.5 < x_1 \leq 1. \end{cases}$$

For the optimal state and control, see Figure 5.2.

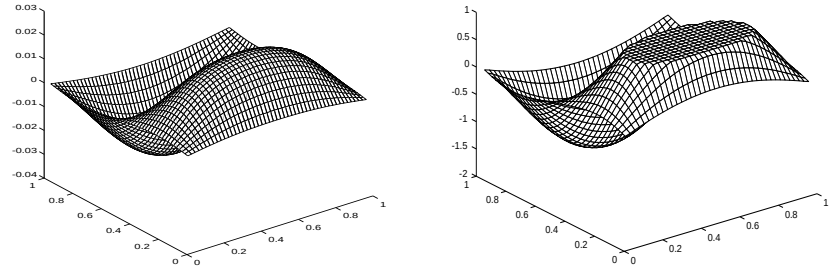


FIG. 5.2. *Optimal state (left graph) and optimal control (right graph) for Example* 5.1.2.

5.1.3. Example: State constraints. The data are $\alpha = 0.001$, $c = 0.01$, $\varphi \equiv 0.1$, $u_d \equiv 0$, and $z_d(x, y) = \sin(2\pi x\, y)$. For the optimal state and control, see Figure 5.3.

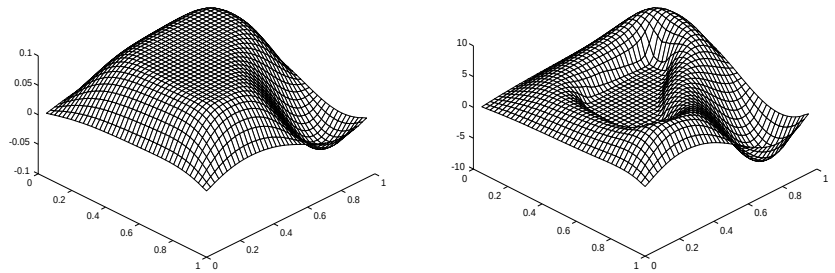


FIG. 5.3. *Optimal state (left graph) and optimal control (right graph) for Example* 5.1.3.

The following examples are depicted for discussing one-step convergence and lack of strict complementarity, respectively.

5.1.4. Example: Control constraints. This example satisfies (3.15) of Corollary 3.2. Thus, one-step convergence of (BIK^c) is expected. The data are $\alpha = 0.1$, $c = 0.5$, $z_d \equiv 1$,

$$\psi(x_1, x_2) = \begin{cases} -1 & \text{if } (x_1, x_2) \in \Omega_1, \\ -(x_1 - 0.5)^2 - (x_2 - 0.5)^2 & \text{else,} \end{cases}$$

and $u_d = u^\dagger + \alpha^{-1}(\lambda^\dagger + \Delta^{-2}u^\dagger - \Delta^{-1}z_d)$, where

$$\lambda^\dagger(x_1, x_2) = \begin{cases} 0.05 & \text{if } (x_1, x_2) \in \Omega_1, \\ 0 & \text{else,} \end{cases} \quad u^\dagger(x_1, x_2) = \begin{cases} \psi(x_1, x_2) & \text{if } (x_1, x_2) \in \Omega_1, \\ 10\,\psi(x_1, x_2) & \text{else,} \end{cases}$$

and $\Omega_1 = \{(x_1, x_2) \in \Omega : \|(x_1, x_2) - (0.5, 0.5)\|_{\ell^2} \le 0.25\}$. We have $u^* = u^\dagger$ and $\lambda^* = \lambda^\dagger$, with $\mathcal{A}^* = \Omega_1$.

5.1.5. Example: Control constraints. This example is constructed such that no strict complementarity holds at the solution. We put $z_d \equiv 0$, $u_d = u^\dagger + \alpha^{-1}(\lambda^\dagger - \Delta^{-2}u^\dagger)$,

$$\psi(x_1, x_2) = \begin{cases} (x_1 - 0.5)^8 & \text{if } (x_1, x_2) \in \Omega_1, \\ (x_1 - 0.5)^2 & \text{else,} \end{cases}$$

$$u^\dagger(x_1, x_2) = \begin{cases} \psi(x_1, x_2) & \text{if } (x_1, x_2) \in \Omega_1 \cup \Omega_2, \\ -1.01\psi(x_1, x_2) & \text{else,} \end{cases}$$

$$\lambda^\dagger(x_1, x_2) = \begin{cases} 2.25(x_1 - 0.75) \cdot 10^{-4} & \text{if } (x_1, x_2) \in \Omega_2, \\ 0 & \text{else,} \end{cases}$$

where $\Omega_1 = \{(x_1, x_2) \in \Omega : \|(x_1, x_2) - (0.5, 0.5)\|_{\ell^2} \le 0.15\}$ and $\Omega_2 = \{(x_1, x_2) \in \Omega : x_1 \ge 0.75\}$. We have $u^* = u^\dagger$ and $\lambda^* = \lambda^\dagger$. Notice that the strongly active set is $\mathcal{A}^{*,+} = \Omega_2$ and the weakly active set is $\mathcal{A}^{*,0} = \Omega_1 \cup \{(x_1, x_2) \in \Omega | x_1 = 0.5\}$, where strict complementarity is not satisfied. The optimal state and control are displayed in Figure 5.4.

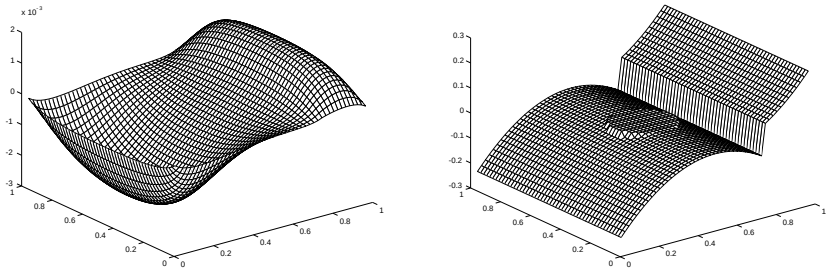


FIG. 5.4. *Optimal state (left graph) and optimal control (right graph) for Example* 5.1.5.

5.1.6. Example: State constraints. This is an example for state constraints with lack of strict complementarity at the solution. The data are $\alpha = 0.01$, $c = 0.1$, $u_d \equiv 0$, $z_d = \lambda^\dagger + y^\dagger + \alpha\Delta^2 y^\dagger$, and $\varphi = \beta[x_1 x_2(1 - x_1)(1 - x_2)]^5$, where $\beta = 1\text{e}3$,

$$\lambda^\dagger = \begin{cases} 0 & \text{if } x_1 \notin [0.5\,,\, 0.7], \\ x_1 - 0.5 & \text{if } 0.5 \le x_1 \le 0.7, \end{cases} \qquad y^\dagger = \begin{cases} \varphi(x_1, x_2) & \text{if } x_1 \le 0.7, \\ \varphi(x_1, x_2)(1.7 - x_1) & \text{if } 0.7 < x_1 \le 1. \end{cases}$$

The exact solution is $y^* = y^\dagger$ and $\lambda^* = \lambda^\dagger$. Note that strict complementarity does not hold for $x_1 \le 0.5$.

5.1.7. Example: State constraints. The example is chosen such that we numerically observe one-step convergence. The data are $\alpha = 1\text{e-}10$, $c = 0.01$, $u_d \equiv 0$, $z_d = \lambda^\dagger + y^\dagger + \alpha\Delta^2 y^\dagger$, and $\varphi = \beta\,[x_1x_2(1-x_1)(1-x_2)]^5$, where $\beta = 5\text{e}4$,

$$\lambda^\dagger = \begin{cases} 0 & \text{if } 0 < x_1 \le 0.5, \\ (x_1 - 0.5)^{10} & \text{if } 0.5 < x_1 \le 1, \end{cases} \qquad y^\dagger = \begin{cases} 0.999 \cdot \varphi(x_1, x_2) & \text{if } 0 < x_1 \le 0.5, \\ \varphi(x_1, x_2) & \text{if } 0.5 < x_1 \le 1. \end{cases}$$

We have $y^* = y^\dagger$ and $\lambda^* = \lambda^\dagger$. Note that strict complementarity holds on all of Ω, and that due to $\lambda^* = \lambda^\dagger$ a remarkable degree of degeneracy can be observed, i.e., λ^* is close to zero near the interface between the active and inactive sets and, in addition here, $y^* - \varphi$ is close to zero on a subset of the inactive set.

5.2. Dependence on the mesh-size. In Tables 5.1–5.4 we document the results for Examples 5.1.1 and 5.1.3. Control constrained problems are considered in Tables 5.1 and 5.2. Here (BIK^c) and (IP1) refer to the algorithms of sections 3 and 4.1, and (IP2) stands for an adaptation of the interior point algorithm for state constrained problems to the control constrained case. Similarly, in Tables 5.3 and 5.4, (IP1) denotes an adaptation of the algorithm in section 4.1 to the state constrained case.

The stopping rules for interior point methods included in the description of the algorithms in sections 4.1 and 4.2 represent standard criteria which terminate the algorithms at approximate solutions. The results reported in the subsequent subsections were computed according to these generic rules. However, if exact solutions (as computed by (BIK) for the examples in this subsection) are available, it is natural to consider the performance of interior point methods in dependence on the distance of the interior point solutions to the exact solutions. Therefore, for the test runs reported in Tables 5.1–5.4 the following procedure was designed: We computed the exact solution by running (BIK). Then we ran (IP2) with $\varepsilon_s = \sqrt{\varepsilon_M}$ and $e_f = 8$. Afterward (IP1) was started and stopped at the first iteration n, where

$$|||z^{BIK} - z_n^{IP1}||| \le |||z^{BIK} - z^{IP2}||| \le |||z^{BIK} - z_{n-1}^{IP1}|||\,, \tag{5.1}$$

where $z^{BIK} = (y^{BIK}, u^{BIK}, J_h(y^{BIK}, u^{BIK}))$ and $z_n^{IPi} = (y_n^{IPi}, u_n^{IPi}, J_h(y_n^{IPi}, u_n^{IPi}))$, $i = 1, 2$. Thus (IP1) was stopped when the distance of the $(n-1)$st iterate of (IP1) to the exact solution exceeded the corresponding distance of the solution of (IP2), but the nth iterate of (IP1) was closer to the exact solution than the solution of (IP2). The solution for (IP1) documented in Tables 5.1–5.4 corresponds to (y_k^{IP1}, u_k^{IP1}), $k \in \{n-1, n\}$, for which

$$\big|\,|||z^{BIK} - z^{IP2}||| - |||z^{BIK} - z_k^{IP1}|||\,\big| \tag{5.2}$$

was smaller. This additional procedure is motivated by the fact that for nondegenerate problems, interior point methods converge very rapidly during the final stages of the iteration. Combining (5.1) and (5.2), solutions of the interior point algorithms are obtained which approximate the exact solution approximately equally well.

Based on the results displayed in Tables 5.1–5.4 which are typical from within a list of test examples we can draw the following conclusions:

- The algorithms behave quite differently for control and state constrained problems. This concerns primarily the Moreau–Yosida-based algorithms and only to a lesser degree interior point methods.
- For control constrained problems (BIK^c) is significantly faster than interior point methods. Except for certain cases with lack of strict complementarity

TABLE 5.1
Comparison of number of iterations and CPU-time (s) for Example 5.1.1.

	(BIK^c)		(IP1)		(IP2)	
h^{-1}	#it	CPU (s)	#it	CPU (s)	#it	CPU (s)
20	3	0.2	13	0.8	7	0.7
30	3	0.9	15	3.2	8	3.3
40	3	2.5	15	8.2	8	8.9
50	3	5.5	15	18.6	8	20.2
60	3	10.8	15	35.1	8	40.2
70	4	23.8	15	63.7	8	73.4
80	4	39.6	16	116.0	8	123.8
90	4	62.2	16	182.8	8	197.3
100	3	75.0	16	268.7	8	296.5
110	4	133.4	16	381.9	8	413.9
120	4	186.1	16	532.1	8	575.7

TABLE 5.2
Accuracies of IP solutions for Example 5.1.1.

	(IP1)			(IP2)		
h^{-1}	$d^u_{\infty,1}$	$d^y_{\infty,1}$	d^J_1	$d^u_{\infty,2}$	$d^y_{\infty,2}$	d^J_2
20	8.24e-07	7.86e-10	1.16e-13	5.32e-07	4.99e-10	1.01e-13
30	3.50e-08	2.78e-11	2.96e-15	2.30e-07	1.78e-10	1.78e-14
40	3.78e-08	1.76e-11	1.48e-14	1.48e-07	6.87e-11	1.01e-13
50	1.91e-05	5.97e-09	3.07e-14	1.63e-05	5.13e-09	3.06e-14
60	2.40e-05	5.03e-09	1.39e-13	1.98e-05	4.28e-09	1.55e-13
70	7.12e-05	1.38e-08	1.89e-13	4.42e-05	8.17e-05	9.31e-13
80	3.99e-05	5.30e-09	5.08e-14	6.49e-05	8.80e-09	1.73e-12
90	7.08e-06	7.78e-10	5.65e-13	2.49e-05	2.98e-09	1.12e-13
100	4.32e-05	3.83e-09	9.09e-13	7.30e-05	7.20e-09	9.52e-13
110	1.87e-05	1.41e-09	8.44e-13	4.41e-05	3.85e-09	6.45e-13
120	3.32e-05	2.36e-09	1.25e-12	5.96e-05	4.80e-09	3.29e-12

or of degeneracy the algorithm stopped in step 3 at the exact solution of the discretized problem. The cases of lack of strict complementarity and degeneracy are discussed below.

- For state constrained problems (BIK^s) is slower than interior point methods. This can be due to the fact that changes from the active to the inactive sets and vice versa for (BIK^s) occur primarily along the boundary of active and inactive sets in layers of only one pixel depth. If an algorithm similar to (BIK^s) is used to solve obstacle problems, then this can be proved rigorously; see [KKT]. In all nondegenerate test examples (BIK^s) terminated at the exact solution in step 3. For the latter we refer to section 5.4.
- We observe numerically that (BIK^c) is mesh-independent for control constrained problems. This is not the case for (BIK^s) for state constrained problems.
- Interior point methods are mesh-independent for control constrained problems and significantly more mesh-independent for state constrained problems than (BIK^s).
- For the control constrained problem of Table 5.1 (IP1) and (IP2) require about the same CPU-time with (IP2) taking half as many iterations as (IP1). This behavior can be attributed to the fact that one iteration of (IP2) utilizes two linear equation solves (due to an intermediate update in step 3 of (IP2)),

TABLE 5.3
Comparison of number of iterations and CPU-time (s) for Example 5.1.3.

	(BIK^s)		(IP1)		(IP2)	
h^{-1}	#it	CPU (s)	#it	CPU (s)	#it	CPU (s)
20	11	1.2	13	1.0	8	0.8
30	17	8.1	14	3.4	9	3.3
40	21	30.3	16	9.9	11	11.1
50	26	89.3	15	21.0	11	25.3
60	32	207.4	17	42.2	11	49.9
70	36	423.9	18	79.5	12	101.2
80	41	800.9	14	102.3	12	186.9
90	46	1417.9	16	188.9	13	287.8
100	51	2388.9	16	274.7	14	472.4
110	58	3920.5	16	401.5	14	661.8

TABLE 5.4
Accuracies of IP solutions for Example 5.1.3.

	(IP1)			(IP2)		
h^{-1}	$d^u_{\infty,1}$	$d^y_{\infty,1}$	d^J_1	$d^u_{\infty,2}$	$d^y_{\infty,2}$	d^J_2
20	7.85e-05	8.17e-08	9.02e-12	1.13e-04	1.26e-07	6.76e-10
30	2.96e-06	1.43e-07	3.98e-12	3.02e-04	1.62e-07	4.51e-10
40	1.26e-06	3.53e-10	7.74e-15	1.72e-05	3.84e-08	7.56e-12
50	1.90e-06	3.35e-10	2.21e-14	5.94e-05	1.15e-08	3.51e-11
60	1.08e-05	2.08e-08	8.19e-15	2.16e-04	4.62e-07	5.60e-10
70	4.33e-05	5.24e-09	1.73e-14	1.13e-04	1.44e-07	4.79e-11
80	9.03e-04	8.09e-08	2.04e-11	8.12e-04	1.16e-07	8.61e-10
90	6.19e-06	3.39e-10	4.38e-14	5.71e-05	4.05e-09	2.12e-11
100	7.53e-04	6.85e-08	4.33e-13	6.87e-04	6.57e-08	4.47e-11
110	1.06e-03	3.59e-08	1.84e-12	1.49e-03	8.62e-08	4.55e-10

whereas (IP1) requires only one. For state constrained problems (IP1) gives better results with respect to CPU-time than (IP2).

- For Tables 5.1–5.4 the default start-up routines as described above were chosen. The algorithms behave robustly with respect to other choices of start-up values.
- Tests with related problems suggest that (BIK^s) can be sped up significantly by multilevel techniques. However, we do not want to pursue a comparison for a multilevel environment.

5.3. Sensitivity with respect to the α value. The degree of positive definiteness of the cost functional is determined by the value of $\alpha > 0$. The smaller the value of α, the more singular the optimal control problems are. It is therefore a natural question to ask whether the performance of the algorithms deteriorates as α becomes smaller. As can be seen from Table 5.5, which corresponds to Example 5.1.2, and Table 5.6, which gives the results for Example 5.1.3, this is not the case. (BIK^c) as well as (BIK^s) find exact solutions, and the interior point methods find approximate solutions of rather uniform accuracy for a wide range of α-values. We do not report the CPU-times, since they are essentially linear with respect to the number of iterations.

Let us now consider Tables 5.5 and 5.6 in some detail. For the control constrained case reported in Table 5.5 the iteration numbers increase with decreasing α. For (BIK^c) a possible explanation is given by Corollary 3.2. Let D_h be the set of indices

TABLE 5.5
Example 5.1.2 *(control constraints) for (IP*1*) and* $h^{-1} = 60$.

α	1e-01	1e-02	1e-03	1e-04	1e-05
#it (IP1/BIK^c)	9 / 2	12 / 3	13 / 5	13 / 6	16 / 8
$d^u_{\infty,1}$	7.75e-11	6.14e-04	2.15e-03	2.05-02	2.61e-05
$d^y_{\infty,1}$	1.13e-11	1.98e-07	2.56e-07	3.39e-06	8.13e-09
d^J_1	2.00e-12	1.60e-11	1.73e-11	2.21e-11	8.48e-12
α	1e-06	1e-07	1e-08	1e-09	1e-10
#it (IP1/BIK^c)	18 / 13	19 / 28	20 / 29	21 / 30	21 / 30
$d^u_{\infty,1}$	6.41e-02	4.03e-01	4.34e-02	3.07-02	1.21e-02
$d^y_{\infty,1}$	6.14e-06	3.60e-05	3.17e-06	1.95e-06	6.80e-07
d^J_1	3.54e-12	1.53e-11	2.05e-11	1.86e-11	1.98e-11

TABLE 5.6
Example 5.1.3 *(state constraints) for (IP*2*) and* $h^{-1} = 60$.

α	1e-01	1e-02	1e-03	1e-04	1e-05
#it ($IP2/BIK^s$)	4 / 1	13 / 29	11 / 32	10 / 20	9 / 13
$d^u_{\infty,2}$	3.85e-07	1.31e-06	2.16e-04	7.75-05	3.83e-03
$d^y_{\infty,2}$	1.37e-10	6.47e-10	4.62e-07	1.81e-08	4.56e-07
d^J_2	5.55e-17	1.41e-12	5.60e-10	2.41e-10	4.52e-10
α	1e-06	1e-07	1e-08	1e-09	1e-10
#it ($IP2/BIK^s$)	8 / 7	8 / 4	8 / 3	8 / 3	8 / 2
$d^u_{\infty,2}$	2.31e-02	2.00e-03	1.02e-03	6.13e-02	3.13e-03
$d^y_{\infty,2}$	2.87e-06	1.98e-07	7.90e-08	4.43e-06	2.20e-07
d^J_2	1.36e-09	2.51e-11	5.25e-12	5.82e-11	6.83e-12

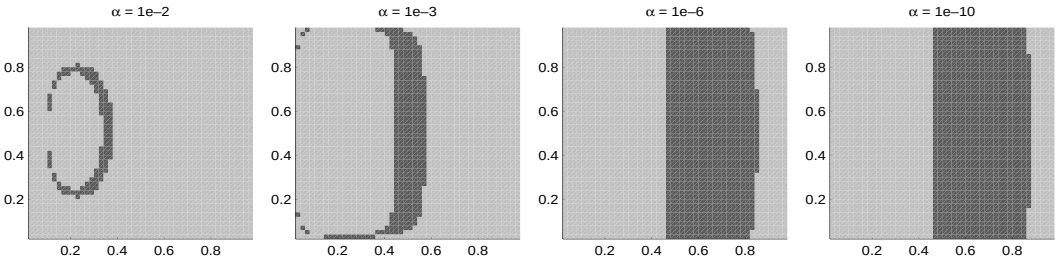


FIG. 5.5. *Evolution of* D_h *for Example* 5.1.2 *as* α *varies.*

where (3.15) is violated. Figure 5.5 shows the evolution of this set as a function of α. In our test runs we clearly see that the larger D_h is, the more iterations are required by (BIK^c) to find the optimal solution. Thus, $|D_h|$ can be seen as a degree of discrepancy to the ideal situation of $D_h = \emptyset$, for which Corollary 3.2 guarantees one-step convergence. The interior point method is slightly less affected by changes of α. This may be due to the fact that the interior methods unlike active set methods do not estimate the active set, but rather they drive the complementarity product to zero while approaching the optimal solution. Nevertheless there remains a dependence of the iteration numbers on the values of α for which, however, we cannot offer a satisfactory explanation that is supported by a wide range of test examples.

Considering Table 5.6 the influence of changes in α on the iteration numbers is reversed. The case α = 1e-1 is special, since here the unconstrained minimum

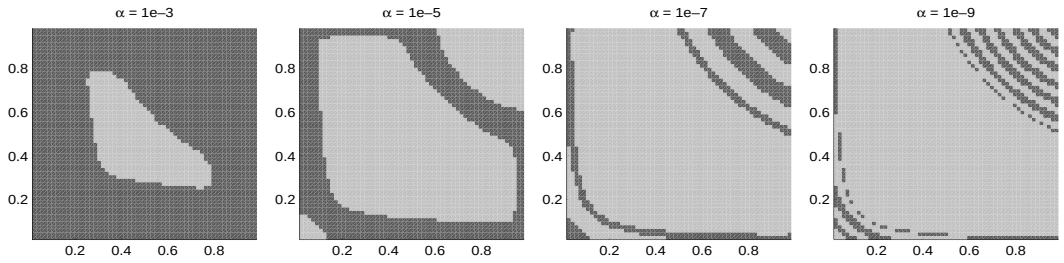


FIG. 5.6. *Evolution of D_h for Example* 5.1.3 *as α varies.*

gives the optimal solution to the constrained problem as well. Again the interior point algorithm is much less affected by changes in α than is (BIK^s). To explain the behavior of (BIK^s) an analysis analogous to Corollary 3.2 suggests the introduction of the set

$$D_h = \{i \in \mathcal{A}_h^{*,+} | (y_o)_i \leq (y_h^*)_i\} \cup \{i \in \mathcal{A}_h^{*,0} \cup \mathcal{I}_h^* | (y_o)_i > (y_h^*)_i\},$$

where we have taken into consideration that (BIK^s) is initialized by solving (P_h^s) without the inequality constraint. Again, the smaller $|D_h|$ the closer we are to one-step convergence, which holds for $D_h = \emptyset$. Figure 5.6 shows the evolution of the set D_h for Example 5.1.3 for various choices of α. The measure of D_h decreases as α decreases, and therefore the decrease of iterations of (BIK^s) comes as no surprise.

5.4. Lack of strict complementarity and degeneracy. Examples 5.1.5 and 5.1.6 are constructed such that no strict complementarity holds at the exact solution, and Example 5.1.7 is chosen such that a high degree of degeneracy can be observed. Here the degree of degeneracy is measured by the smallness of the quantities $\min\{(\lambda_h^*)_i : i \in \mathcal{A}_h^{*,+}\}$, $\min\{|(u_h^* - \psi_h)_i| : i \in \mathcal{I}_h^*\}$ in the control constrained case, respectively, $\min\{|(y_h^* - \varphi_h)_i| : i \in \mathcal{I}_h^*\}$ in the state constrained case. In fact, the smaller one of the quantities above is, the higher is the degree of degeneracy. These situations are of interest as for (BIK) numerical problems when determining the active and inactive sets at each iteration may arise: Let us demonstrate this for the control constrained case with lack of strict complementarity. For this purpose assume that the optimal control is $u_h^* \equiv 0$ with corresponding optimal multiplier $\lambda_h^* \equiv 0$. Let $\varepsilon \geq \varepsilon_M$ denote the relative accuracy in u_n and λ_n. Near the optimal numerical solution both iterates are small. In fact, for $i \in \mathcal{I}_n$ we put $(\lambda_n)_i = 0$ and obtain $(u_n)_i \approx \pm\varepsilon$ from the numerical solution of the linear system. For $i \in \mathcal{A}_n$ we set $(u_n)_i = 0$ and obtain $(\lambda_n)_i \approx \pm\varepsilon$ from the calculation. Therefore, the signs of the computed components of the iterates decide on the active and inactive sets of the next iteration. Since these signs are influenced, for instance, by round-off errors in finite precision calculations the active and inactive sets may start to chatter. The objective value J_n remains on the order ε^2 ($\ll \varepsilon_M$ typically). A similar situation occurs for degenerate problems. In order to cope with this difficulty we add a stopping criterion to step 3 of (BIK). We develop this criterion for the discretized algorithm (BIK^c). For this purpose we introduce

$$\mathcal{S}_n^c = \{i \in \mathcal{A}_n \mid (\lambda_n)_i \leq 0\} \text{ and } \mathcal{T}_n^c = \{i \in \mathcal{I}_n \mid (u_n)_i > (\psi_h)_i\},$$

which are the subsets of Ω_h that change from active to inactive sets and inactive to active sets from the nth to the $(n+1)$st iteration. Moreover, define

$$r_n^{\mathcal{S}} := \max_{i \in \mathcal{S}_n^c} \{|(\lambda_n)_i|\} \qquad \text{and} \qquad r_n^{\mathcal{T}} := \max_{i \in \mathcal{T}_n^c} \{|(\psi_h)_i - (u_n)_i|\}$$

if $\mathcal{S}_n^c \neq \emptyset$ and $\mathcal{T}_n^c \neq \emptyset$, and $r_n^{\mathcal{S}} = 0$, respectively, $r_n^{\mathcal{T}} = 0$ otherwise. If we find that $r_n^{\mathcal{S}}$ and $r_n^{\mathcal{T}}$ are of the order of the accuracy expected for the solution of the linear system in step 4 of the discretized algorithm (BIK^c), then we cannot rely on the determination of the inactive and active sets in the following iteration, and hence we stop the algorithm. This criterion can readily be carried over to the state constrained case.

We report first on two test runs of (BIK^c) and (IP1), respectively, for Example 5.1.5 with $h = 1/50$, $\varepsilon =$ 1e-10, $e_f = 10$, and $c = 0.1$. For the first run we choose $\alpha =$ 1e-5. Algorithm (BIK^c) stops after 6 iterations at the exact solution, i.e., the algorithm terminates in step 3. Algorithm (IP1) needs 16 iterations to satisfy the stopping criterion in step 2. Moreover, $d_{\infty,1}^u =$ 3.96e-3, $d_{\infty,1}^y =$ 1.11e-5, and $d_1^J =$ 9.67e-12. For the second run we fix $\alpha = 0.01$. In this case (BIK^c) approaches the optimal solution after 2 iterations but starts to chatter at iterates satisfying the first-order conditions of (P_h^c) with a residual of order 1e-13. The additional stopping rule developed above stops the algorithm at the third iteration. Algorithm (IP1) needs 15 iterations to satisfy the stopping criteria.

For Example 5.1.6 with $h = 1/60$ (BIK^s) the new stopping criterion terminates the algorithm at iteration 61. Algorithm (IP2) needs 16 iterations to satisfy its stopping criteria. In this case, we have $d_{\infty,2}^u =$ 2.31e-5, $d_{\infty,2}^y =$ 2.12e-7, and $d_2^J =$ 3.94e-13.

As for the interior point methods our tests confirm theoretical results [MW], which assert linear rate of convergence in the case of lack of strict complementarity. For problems with strict complementarity we obtained superlinear convergence rates.

In conclusion, from our test runs we may state that the convergence speed of the Moreau–Yosida-based active set strategies is not affected by degeneracy or lack of strict complementarity. In fact, the predominant factor influencing the number of iterations is the discrepancy set D_h. However, an additional stopping rule has to be implemented which guarantees that the algorithm stops at a solution which satisfies the first-order conditions to a high degree of accuracy. We also point out that there exist several test problems for degenerate examples and examples with lack of strict complementarity where the solution obtained with the modified stopping criterion satisfied $\mathcal{A}_n = \mathcal{A}_{n+1}$, and therefore the exact solution was obtained. Interior point algorithms, on the other hand, are slowed down in the case of lack of strict complementarity and only converge with a linear rate.

5.5. One-step convergence. Theorem 3.1 and Corollary 3.2 give sufficient conditions for (BIK^c) to terminate after 1 iteration at the exact solution. Example 5.1.4 satisfies (3.15) of Corollary 3.2. For (BIK^c) we indeed observe one-step convergence. Algorithm (IP1) stops with the following result (for $h = 1/60$): 11 iterations, $d_{\infty,1}^u =$ 4.32e-10, $d_{\infty,1}^y =$ 1.52e-10, and $d_1^J =$ 1.81e-10.

If we change $\lambda^\dagger$ to $\lambda^\dagger_{|\Omega_1} = 0.01$, then (3.15) is no longer satisfied and (BIK^c) needs two iterations to compute the exact solution. The result for (IP1) (again for $h = 1/60$) is 12 iterations, $d_{\infty,1}^u =$ 5.04e-9, $d_{\infty,1}^y =$ 3.23e-10, and $d_1^J =$ 3.22e-10. Thus, we can see that criterion (3.15) is quite accurate.

In the state constrained case, condition (3.16) is sufficient for one-step convergence of (BIK^s) with the initialization discussed at the beginning of this section. For

Example 5.1.7 we observe that (3.16) is satisfied numerically in the sense that the first part of condition (3.16) is violated on a layer of maximum two-pixels depth at the interface between the active and inactive sets. The order of the violation is about 1e-17. Moreover, $y_o - y_h^* \approx \lambda_h^*$. The additional stopping rule developed in section 5.4 terminates the algorithm after 1 iteration at an approximate solution $\tilde{y}_h^*$ satisfying $\|\tilde{y}_h^* - y_h^*\|_\infty \leq 5.56 \cdot$ 1e-17 (for $h = 1/50$).

5.6. One-dimensional problems. One-dimensional test examples for both control and state constrained problems with mesh-size 10^{-3} and smaller were successfully computed, and similar properties as for the two-dimensional examples were observed. Especially mesh-independence for (BIK^c) with very fine resolution can be confirmed.

6. Conclusion. Detailed numerical comparisons between generalized Moreau–Yosida-based algorithms and two independent interior point algorithms for a class of control and state constrained optimal control problems were carried out. Depending on whether the control or the state are constrained, and depending further on the value of the cost parameter α and on the mesh-size, generalized Moreau–Yosida-based algorithms or interior point algorithms can be more efficient from the point of view of CPU-time. The generalized Moreau–Yosida-based algorithms have the advantage that under certain conditions they provide exact solutions, whereas it is inherent to the interior point approach that approximate solutions are computed. Finally the generalized Moreau–Yosida-based algorithms are significantly simpler to program than the interior point methods.

Acknowledgments. The authors are indebted to the referees for numerous helpful comments and suggestions.

REFERENCES

[AM] W. Alt and K. Malanowski, *A Lagrange-Newton method for nonlinear optimal control problems*, Comput. Optim. Appl., 2 (1993), pp. 77–100.

[Ba] V. Barbu, *Analysis and Control of Nonlinear Infinite Dimensional Systems*, Academic Press, Boston, 1993.

[Be] M. Bergounioux, *Augmented Lagrangian method for distributed optimal control problems with state constraints*, J. Optim. Theory Appl., 78 (1993), pp. 493–521.

[BC] J. F. Bonnans and E. Casas, *On the choice of the function space for some state constrained control problems*, Numer. Funct. Anal. Optim., 7 (1985), pp. 333–348.

[BIK] M. Bergounioux, K. Ito, and K. Kunisch, *Primal-dual strategy for constrained optimal control problems*, SIAM J. Control Optim., 37 (1999), pp. 1176–1194.

[BK1] M. Bergounioux and K. Kunisch, *Augmented Lagrangian techniques for elliptic state constrained optimal control problems*, SIAM J. Control Optim., 35 (1997), pp. 1524–1543.

[BK2] M. Bergounioux and K. Kunisch, *Primal-Dual Strategy for State-Constrained Optimal Control Problems*, Rapport de Recherche 98-20, Université d'Orléans, Orleans, France, 1998.

[BK3] M. Bergounioux and K. Kunisch, *Active set strategy for constrained optimal control problems: The finite dimensional case*, Lecture Notes in Econom. and Math. Systems 481, V. H. Nguyen, J. J. Strodiot, and P. Tossings, eds., Springer-Verlag, New York, 1999, pp. 36–54.

[BG] J. F. Bonnans and C. C. Gonzaga, *Convergence of interior point algorithms for the monotone linear complementarity problem*, Math. Oper. Res., 21 (1996), pp. 1–25.

[BPR] J. F. Bonnans, C. Pola, and R. Rebaï, *Perturbed path following interior point algorithms*, Optim. Methods Softw., 11-12 (1999), pp. 183–210.

[C] E. Casas, *Control of an elliptic problem with pointwise state constraints*, SIAM J. Control Optim., 24 (1986), pp. 1309–1318.

[GMPS] P. E. Gill, W. Murray, D. B. Ponceléon, and M. A. Saunders, *Primal dual methods for linear programming*, Math. Programming, 70 (1995), pp. 251–277.

[GT] H. Goldberg and F. Tröltzsch, *On a Lagrangian-Newton method for a nonlinear parabolic boundary control problem*, Optim. Methods Softw., 8 (1998), pp. 225–247.

[G] J. Gondzio, *Multiple centrality corrections in a primal-dual method for linear programming*, Comput. Optim. Appl., 6 (1996), pp. 137–156.

[IK] K. Ito and K. Kunisch, *Augmented Lagrangian formulation of nonsmooth, convex optimization in Hilbert spaces*, in Control of Partial Differential Equations, Lecture Notes in Pure and Appl. Math. 174, E. Casas, ed., Marcel Dekker, New York, 1995, pp. 107–117.

[JS] H. Jäger and E. W. Sachs, *Global convergence of inexact reduced SQP methods*, Optim. Methods Softw., 7 (1997), pp. 83–110.

[KKT] T. Kärkkäinen, K. Kunisch, and P. Tarvainen, *Primal-Dual Active Set Methods for Obstacle Problems*, Report B 2/2000, Dept. of Mathematical Information Technology, University of Jyväskylä, Finland, 2000.

[KS] S. F. 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.

[LMS] 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.

[LS] F. Leibfritz and E. W. Sachs, *Inexact SQP interior point methods and large scale optimal control problems*, SIAM J. Control Optim., 38 (1999), pp. 272–293.

[Meg] N. Meggido, *Pathways to the optimal set in linear programming*, in Progress in Mathematical Programming, N. Megiddo, ed., Springer-Verlag, New York, 1989, pp. 131–158.

[Meh] S. Mehrotra, *On the implementation of a primal-dual interior point method*, SIAM J. Optim., 2 (1992), pp. 575–601.

[Mes] C. Mészáros, *The Separable and Non-Separable Formulations of Convex Quadratic Problems in Interior Methods*, Report WP 98-3, Hungarian Academy of Sciences, Budapest, Hungary, 1998.

[M] S. Mizuno, *A new polynomial time method for a linear complementarity problem*, Math. Programming, 56 (1992), pp. 31–43.

[MTY] S. Mizuno, M. Todd, and Y. Ye, *On adaptive step primal-dual interior-point algorithms for linear programming*, Math. Oper. Res., 18 (1993), pp. 964–981.

[MW] R. D. C. Monteiro and S. J. Wright, *Interior-point algorithms for degenerate linear complementarity problems*, Comput. Optim. Appl., 3 (1994), pp. 131–155.

[P] F. A. Potra, *An O(nL) infeasible interior point algorithm for LCP with quadratic convergence*, Ann. Oper. Res., 62 (1996), pp. 81–102.

[TT] D. Tiba and F. Tröltzsch, *Error estimates for the discretization of state constrained convex control problems*, Numer. Funct. Anal. Optim., 17 (1996), pp. 1005–1028.

[VY] R. J. Vanderbei and B. Yang, *On the Symmetric Formulation of Interior-Point Methods,* Technical report SOR 94-05, Princeton University, Princeton, NJ, 1994.

[V1] R. J. Vanderbei, *LOQO: An Interior Point Code for Quadratic Programming*, Technical report SOR 94-15, Princeton University, Princeton, NJ, 1994 (revised 1998).

[V2] R. J. Vanderbei, *Linear Programming: Foundations and Extensions*, Kluwer Academic Publishers, Boston, MA, 1997.

[W] M. H. Wright, *Ill-conditioning and computational error in interior methods for nonlinear programming*, SIAM J. Optim., 9 (1999), pp. 84–111.

[Wr] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.

[Y] Y. Ye, *Interior Point Algorithms: Theory and Analysis*, Wiley-Intersci. Ser. Discrete Math. Optim., John Wiley, New York, 1997.

[Z1] Y. Zhang, *On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem*, SIAM J. Optim., 4 (1994), pp. 208–227.

[ZZ] Y. Zhang and D. Zhang, *On polynomiality of the Mehrotra-type predictor-corrector interior-point algorithms*, Math. Programming, 68 (1995), pp. 303–318.

[Z2] 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. 11, No. 2, pp. 522–546

APPROXIMATING BINARY IMAGES FROM DISCRETE X-RAYS*

PETER GRITZMANN†, SVEN DE VRIES†, AND MARKUS WIEGELMANN†

Abstract. We study the problem of approximating binary images that are accessible only through few evaluations of their discrete X-ray transform, i.e., through their projections counted with multiplicity along some lines. This inverse discrete problem belongs to a class of generalized set partitioning problems and allows natural packing and covering relaxations. For these ($\mathbb{NP}$-hard) optimization problems we present various approximation algorithms and provide estimates for their worst-case performance. Further, we report on computational results for various variants of these algorithms. In particular, the corresponding integer programs are solved with only small absolute error for instances up to 250,000 binary variables.

Key words. discrete tomography, set packing, set covering, set partitioning, greedy algorithm, matching, approximation algorithm, worst-case performance, polynomial time

AMS subject classifications. 05B40, 05C35, 05C70, 68R05, 68U10, 82D25, 90C27, 92C55

PII. S105262349935726X

1. Introduction.

The present paper studies various algorithms for finding approximate solutions for an inverse discrete problem that is most prominently motivated by the demand in material science for developing a tool for the reconstruction of 3-dimensional crystalline structures that are accessible only through some images provided by high resolution transmission electron microscopy. In fact, the articles [13] and [18] describe a new technique called *QUANTITEM* for the quantitative analysis of the information provided by transmission electron microscopy that can effectively measure the number of atoms lying on each line parallel to a given set of directions.

Mathematically, this is the inverse problem of reconstructing certain discrete density functions from their discrete X-rays in certain directions. More precisely, the basic question is the following. Can a finite set of points in the integer lattice $\mathbb{Z}^3$ be (approximately) reconstructed from measurements of the number of its points lying on each line parallel to one of a small prescribed number of directions specified by nonzero vectors in $\mathbb{Z}^3$? Here small means 3, 4, or 5 since the energy that is needed to produce the images is about 200 keV so that after a few exposures the object is damaged.

Various approaches have been suggested for solving the general reconstruction problem, and various theoretical results are available; see, e.g., [9] for a survey. In the present paper we concentrate on approximative solutions. Even though most of the resulting combinatorial optimization problems are $\mathbb{NP}$-hard, we prove in section 3 that some (relatively) simple algorithms yield already very good worst-case bounds. As section 4 indicates, these algorithms perform even better in computational practice.

Let us close the introduction with a word of warning. Typically, when one is dealing with optimization problems in practice it is completely satisfactory to produce solutions that are close to optimal. For instance, a tour for a given instance of the traveling salesman problem that is off by only a few percents is for many practical purposes almost as good as an optimal tour. This is because the particular optimization

*Received by the editors May 25, 1999; accepted for publication (in revised form) May 31, 2000; published electronically November 2, 2000.

http://www.siam.org/journals/siopt/11-2/35726.html

†Zentrum Mathematik, Technische Universität München, D-80290 München, Germany (gritzman@mathematik.tu-muenchen.de, devries@mathematik.tu-muenchen.de, markus.wiegelmann@ db.com). The first and second authors were supported in part by German Federal Ministry of Education, Science, Research and Technology grant 03-GR7TM1.

is typically just part of a much more complex real world task, and the improvement over existing methods is governed by so many much harder to influence factors that a small error in the optimization step does not really matter by any practical means. This is different in the context of our prime application. The relevant measure for the quality of an approximation to a binary image would of course be the deviation from this image. Hence, in order to devise the most appropriate objective function one would have to know the underlying solution of the given inverse problem. However, the whole point is of course to find this unknown solution. Hence, one can only consider objective functions, with respect to which the approximation is evaluated, that are based on the given input data. While a good approximation in this sense is close to a solution in that its X-ray images in the given directions are close to those of the original set, the approximating set itself may be off quite substantially. In fact, the inverse discrete problem is ill-posed and it is precisely this property that causes additional difficulties. In particular, if the input data do not uniquely determine the image, even a "perfect" solution that is completely consistent with all given data may be quite different from the unknown real object.

Obviously there is more work to be done to handle the ill-posedness of the problem in practice. Hence, the results of this paper should be regarded only as a first (yet reassuring!) step in providing a computational tool that is adequate for the real world applications outlined above. In particular, our approximate algorithms can be used to provide lower bounds in branch-and-cut approaches that incorporate strategies to handle the nonuniqueness of solutions and the presence of noise in the data.

The paper is organized as follows. Section 2 provides the basic notation, states the problems and algorithmic paradigms that are most important in the context of the present paper, and gives a brief overview of our main results. Section 3 studies various polynomial-time iterative improvement strategies for inner and outer approximation. We derive performance ratios that show that in this model the optimum can be approximated up to a relative error that depends only on the number m of directions in which the X-ray data are available. The analysis is based on work of Hurkens and Schrijver [12], Goldschmidt, Hochbaum, and Yu [8], and Halldórsson [11] for set packing and set covering heuristics. Our theoretical worst-case bounds are complemented by extremely satisfactory computational results described in section 4.

2. Preliminaries and results.

2.1. Basics of discrete tomography. We use the general setting of a d-dimensional Euclidean space $\mathbb{E}^d$, with $d \geq 2$, though only the cases $d = 2, 3$ are relevant in practice. Let $\mathcal{S}_{1,d}$ be the set of all 1-dimensional subspaces in $\mathbb{E}^d$, and let $\mathcal{F}^d$ denote the family of finite subsets of $\mathbb{Z}^d$. For $F \in \mathcal{F}^d$ let $|F|$ be the cardinality of F. A vector $v \in \mathbb{Z}^d \setminus \{0\}$ is called a *lattice direction*; $\mathcal{L}_{1,d}$ denotes the subset of $\mathcal{S}_{1,d}$ spanned by a lattice direction. For $S \in \mathcal{S}_{1,d}$ let $\mathcal{A}(S)$ denote the family of all lines parallel to S. The *(discrete) 1-dimensional X-ray parallel to* S of a set $F \in \mathcal{F}^d$ is the function $X_S F : \mathcal{A}(S) \to \mathbb{N}_0 = \mathbb{N} \cup \{0\}$ defined by

$$X_S F(T) = |F \cap T| \qquad \text{for } T \in \mathcal{A}(S).$$

Since F is finite, the X-ray $X_S F$ has finite support $\mathcal{T} \subset \mathcal{A}(S)$.

In the inverse reconstruction problem, we are given *data functions* $\phi_i : \mathcal{A}(S_i) \to \mathbb{N}_0$, $i = 1, \ldots, m$, with finite support, and we want to find a set $F \subset \mathbb{Z}^d$ with corresponding X-rays. More formally, for $S_1, \ldots, S_m \in \mathcal{L}_{1,d}$ pairwise different, the most important algorithmic task in our context can be stated as follows.

RECONSTRUCTION$(S_1, \dots, S_m)$.

Given data functions $\phi_i : \mathcal{A}(S_i) \to \mathbb{N}_0$ for $i = 1, \dots, m$, find a finite set $F \subset \mathbb{Z}^d$ such that $\phi_i = X_{S_i} F$ for all $i = 1, \dots, m$ or decide that no such F exists.

Clearly, when investigating the computational complexity of the above problem in the usual *binary Turing machine model* one has to describe suitable finite data structures. We do not go into such details here but refer the reader to [5]. For the purpose of this paper, handling an input of m data functions $\phi_1, \dots, \phi_m$ with supports $\mathcal{T}_1, \dots, \mathcal{T}_m$, respectively, is facilitated with the aid of a set $G \subset \mathbb{Z}^d$ of candidate points. This set G consists of the intersection of all (finitely many) translates of $\bigcap_{i=1}^m S_i$ that arise as the intersection of m lines parallel to $S_1, \dots, S_m$ with $\mathbb{Z}^d$, respectively, whose data function value is nonzero, i.e.,

$$G = \mathbb{Z}^d \cap \bigcap_{i=1}^{m} \bigcup_{T \in \mathcal{T}_i} T.$$

To exclude trivial cases, in the following we will always assume that $G \neq \emptyset$ and that $\bigcap_{i=1}^m S_i = \{0\}$. Hence, in particular $m \geq 2$.

The incidences of G and $\mathcal{T}_i$ can be encoded by an incidence matrix A_i. To fix the notation, let G consist of, say, N points, and let $M_i = |\mathcal{T}_i|$ and $M = M_1 + \cdots + M_m$. Then the incidence matrices $A_i \in \{0,1\}^{M_i \times N}$ can be joined together to form a matrix $A \in \{0,1\}^{M \times N}$. Identifying a subset of G with its characteristic vector $x \in \{0,1\}^N$, the reconstruction problem amounts to solving the integer linear feasibility program

$$Ax = b \quad \text{s.t. } x \in \{0,1\}^N, \tag{1}$$

where $b^T = (b_1^T, \dots, b_m^T)$ contains the corresponding values of the data functions $\phi_1, \dots, \phi_m$ as the right-hand sides of $A_1, \dots, A_m$, respectively.

Let us point out here in passing that more general inverse discrete problems can be modeled in a similar way. In fact, query sets (which are lines in the present paper) could be chosen in various different and meaningful ways. (For instance, if the lines are replaced by the translates of some k-dimensional subspaces, we obtain the reconstruction problem for discrete *k-dimensional X-rays*.)

It is not difficult to see that the matrix A is totally unimodular when $m = 2$. In particular, for $m = 2$ the integer linear program (1) and its linear programming relaxation

$$Ax = b \quad \text{s.t. } x \in [0,1]^N, \tag{2}$$

where the condition $x \in \{0,1\}^N$ is replaced by the weaker constraint $x \in [0,1]^N$, are equivalent in the sense that all vertices of the polytope $\{x\colon Ax = b \wedge x \in [0,1]^N\}$ are 0-1 vectors anyway; see, e.g., [17] for an exposition of the underlying theory. Hence, for $m = 2$ the reconstruction problem is solvable in polynomial time; see [15], [1], and [7] for different proofs that do not rely on the fact that linear programming problems can be solved in polynomial time. RECONSTRUCTION$(S_1, \dots, S_m)$ becomes $\mathbb{NP}$-hard, however, when $m \geq 3$; see [5]. (For an introduction to the theory of computational complexity see [6].) This means that (unless $\mathbb{P} = \mathbb{NP}$) exact solutions of (1) require (in general) a superpolynomial amount of time. In polynomial time only approximate solutions can be expected. We will henceforth always assume that $m \geq 3$.

Let us stress the fact that while the solutions of the polynomial-time solvable LP-relaxation (2) do provide some information about (1) (see [3]), it is our goal to solve (1) rather than (2), since the objects underlying our prime application are crystalline structures forming (physical) sets of atoms rather than "fuzzy" sets; see [10] for some additional discussion of this point.

2.2. Two optimization problems. For measuring the quality of approximation methods, we introduce objective functions so as to formulate the reconstruction problem as optimization problems. Two very natural such formulations are the following problems, Best-Inner-Fit and Best-Outer-Fit.

BEST-INNER-FIT$(S_1, \ldots, S_m)$ [BIF].
Given data functions $\phi_1, \ldots, \phi_m$, find a set $F \subset G$ of maximal cardinality such that

$$X_{S_i} F(T) \leq \phi_i(T) \text{ for all } T \in \mathcal{T}_i \text{ and } i = 1, \ldots, m.$$

Equivalently, [BIF] can be formulated as the integer linear program

$$\begin{aligned} &\max \mathbb{1}^T x \text{ s.t.} \\ &Ax \leq b \text{ and } x \in \{0,1\}^N, \end{aligned} \tag{3}$$

where $\mathbb{1}$ is the all-ones vector.

The "outer counterpart" of this inner approximation is defined as follows.

BEST-OUTER-FIT$(S_1, \ldots, S_m)$ [BOF].
Given data functions $\phi_1, \ldots, \phi_m$, find a set $F \subset G$ of minimal cardinality such that

$$X_{S_i} F(T) \geq \phi_i(T) \text{ for all } T \in \mathcal{T}_i \text{ and } i = 1, \ldots, m.$$

Again, the problem is equivalent to an integer linear program, precisely to

$$\begin{aligned} &\min \mathbb{1}^T x \text{ s.t.} \\ &Ax \geq b \text{ and } x \in \{0,1\}^N. \end{aligned} \tag{4}$$

Note that while for any given instance of [BIF] $\emptyset$ is a feasible solution, [BOF] may be infeasible. In order to exclude this degeneracy, we will in the following always assume that

$$\phi_i \leq X_{S_i} G \text{ for } i = 1, \ldots, m.$$

The two problems [BIF] and [BOF] are then complementary to each other in the following sense. The complement $\bar{F} = G \setminus F$ of a solution $F \subset G$ of an instance of one problem is a solution of the instance with complementary data functions $\bar{\phi}_i$ defined by $\bar{\phi}_i(T) = |G \cap T| - \phi_i(T)$ of the other problem. This reflects the fact that reconstructing the "positive" or the "negative" of a binary picture are equivalent. However, as the direct conversion of an approximation result for [BIF] of the form $|V|/|F| \geq \alpha$ (F is an optimal solution and V is some solution) yields a bound $|\bar{V}|/|\bar{F}| \leq \alpha + (1-\alpha)|G|/|\bar{F}|$ for [BOF] that is dependent on the "density" $|F|/|G|$ of an optimal solution in the underlying candidate grid, bounds for the relative error of one problem are usually

not "identical" to bounds for the other. More importantly, our algorithms for [BOF] are actually insertion methods rather than "dual" deletion methods. Hence, we will consider [BIF] and [BOF] separately in section 3.

Let us remark in passing that one can of course consider other kinds of optimization problems related to RECONSTRUCTION$(S_1, \ldots, S_m)$. For instance, rather than measuring the approximability in terms of the points inserted into the candidate grid one may count the number of lines on which an X-ray of a solution coincides with the given value of the corresponding data function. An intractability result for this kind of approximation can be found in [10].

2.3. The basic algorithmic paradigm. In this section we describe a general algorithmic scheme for solving [BIF] and [BOF] that provides the framework for the subsequent approximation algorithms studied in sections 3 and 4. See [10] for a discussion of some other algorithmic paradigms that comprise most of the methods for solving RECONSTRUCTION$(S_1, \ldots, S_m)$ that have been suggested by various authors in the past.

In the present paper we give a theoretical and computational analysis of various iterative improvement strategies that are built on some greedy method. In the simplest classes of local search algorithms for [BIF] and [BOF] the neighborhood of a set S is defined as the collection of all supersets of S of cardinality $|S| + 1$ or of all subsets of cardinality $|S| - 1$, respectively, and the choice is based on some greedy strategy (that may or may not use weights for breaking ties).

In order to increase the performance of such iterative insertion or deletion algorithms, one can apply *r-improvements* for $r \in \mathbb{N}_0$, where an r-point $\langle(r+1)$-point$\rangle$ subset of a current feasible solution $F \subset G$ for the given instance of [BIF] $\langle$[BOF]$\rangle$ is deleted and $r + 1$ $\langle r \rangle$ points of $(G \setminus F)$ are inserted while maintaining feasibility. A feasible set $F \subset G$ is called *t-optimal* for the given instance of [BIF] $\langle$[BOF]$\rangle$ if no r-improvement is possible for any $r \leq t$. Note that 0-optimality agrees with the common greedy-optimality (no point can be inserted without destroying feasibility for [BIF] and no point can be removed without destroying feasibility for [BOF]). However, since our algorithms for [BOF] are based on greedy-type *insertions* rather than greedy-*deletions*, the greedy algorithm of section 3.3 need not produce 0-optimal solutions per se.

The following paradigm comprises a large class of iterative improvement methods for [BIF]. A similar paradigm can be formulated for [BOF]. (A symbolic formulation in the realm of commutative algebra of a general reduction process involving a set of binomials in an appropriate toric ideal is given in [20].)

PARADIGM 2.1 (iterative inner approximation).

- INPUT: *Data functions $\phi_1, \ldots, \phi_m$ for the given lines $S_1, \ldots, S_m$, respectively.*
- OUTPUT: *A feasible set $F \subset G$ for the given instance of [BIF].*
- COMPUTATION:

  Start with $F = \emptyset$ and successively apply r-improvements for $r \leq t$ for some fixed constant $t \in \mathbb{N}_0$ until no further improvement is possible.

Since it is not specified how to select the points for insertion and deletion, Paradigm 2.1 is so general and flexible that it covers a large number of algorithms that incorporate promising refinements. For example, the X-ray data can be used for back-projection-like techniques to express preferences between points to be chosen; see Algorithm 3.7. In addition, connectivity of the solution (in a sense that is justified by the physical structure of the analyzed material) can be rewarded by introducing

adjustable weights. Similarly, information from neighboring layers can be taken into account in a layerwise reconstruction of a 3-dimensional object. In fact, the positive results of section 3 will apply to the general paradigm.

2.4. Main results. The simplest algorithm for [BIF] within the framework of Paradigm 2.1 is the plain greedy algorithm which considers the positions of the grid in an arbitrary order and successively fills in points. We will refer to it as GreedyA. GreedyB and GreedyC will be variants with refined insertion order. As a first result (Theorem 3.1, with $t = 0$), we see that

$$|V|/|F| \geq 1/m,$$

where V is the set obtained by GreedyA (or GreedyB or GreedyC), and F is an optimal solution. Recall that m is the number of directions, whence the sharp and (considering that it is hard to think of any algorithm that is simpler than GreedyA) surprisingly good lower bound $1/m$ reflects the fact that the more data are given, the harder it is for a greedy strategy to satisfy them. In our experiments, it turns out that $|V|/|F|$ is typically greater than 0.9 and for large instances greater than 0.96 even for $m = 5$; see section 4.

There are two natural ways to improve this algorithm:

(a) using a better order to visit the candidate points and

(b) using 1-improvements, 2-improvements, etc.

In terms of (a) we use a strategy (GreedyB) that is motivated by a method of [16] for solving consistent [BIF]-instances exactly for $m = 2$. In our computational study GreedyB clearly outperforms GreedyA for all m considered; see Figures 9 and 10.

In GreedyC weights are assigned dynamically to the candidate points to represent the "changing importance" of a point to be included in a solution. Our computational study shows that GreedyC gives smaller relative errors than GreedyA and GreedyB; see Figure 9. In fact, even the *absolute errors* are small; the average case for GreedyC for $250,000$ positions and density 50%, i.e., solutions of cardinality $125,000$ being 21.62, 64.13, 111.88 missing atoms for 3, 4, 5 directions, respectively. The price to pay for this excellent performance is GreedyC's considerably longer running time; see section 4.

In terms of (b), Theorem 3.1 shows that, for a t-optimal solution V,

$$\frac{|V|}{|F|} \geq \frac{2}{m} - \epsilon_m(t),$$

where $\epsilon_m(t)$ is given explicitly and approaches 0 exponentially fast. Computationally, it turns out that performing 1-improvements after GreedyA, GreedyB, or GreedyC typically yields substantial improvements. In fact, in our computational study the *absolute* errors go down for ImprovementC to 1.07, 23.28, 64.58 for 3, 4, 5 directions, respectively; see Figure 10.

Theorem 3.9 provides worst-case guarantees for [BOF]. Part (a) shows that a simple greedy-type insertion algorithm yields a solution U such that

$$|U|/|F| \leq H(m),$$

where $H(m) = 1 + 1/2 + \cdots + 1/m$ is the mth harmonic number. If additional matching techniques are applied to obtain a stronger optimality condition ("*matching-optimality*"), then

$$|U|/|F| \leq H(m) - 1/6;$$

see Theorem 3.9(b).

Theorem 3.10(a) shows that the t-optimality of a solution U guarantees that

$$|U|/|F| \le m/2 + \epsilon_m(t),$$

where again $\epsilon_m(t)$ is given explicitly and tends to 0 exponentially fast. If, finally, the solution is matching-optimal and (what will be defined later) *effect*-3-t-*optimal* for $t \ge 5$, then

$$|U|/|F| \le H(m) - 1/3;$$

see Theorem 3.10(c). That is, for $m = 3, 4, 5$ the bounds are $\frac{3}{2}$, $\frac{7}{4}$, and $\frac{39}{20}$.

Note that in the case of single coverings, there is a slightly better bound for a certain semi-local search algorithm due to Duh and Fürer [2]. If their approach could be extended to [BOF] it would read $|U|/|F| \le H(m) - 1/2$. However, currently it is not known whether such an extension is possible.

Let us close this section with two remarks. First, all our results are formulated within the realm of discrete tomography due to its main objective. It goes without saying that the theoretical performance ratios apply also to more general multiple packing and multiple covering problems. Second, as already pointed out in the introduction, our analysis makes substantial use of ideas of Hurkens and Schrijver [12], Goldschmidt, Hochbaum, and Yu [8], and Halldórsson [11] for set packing and set covering heuristics. There may be a way to axiomize how to extend results for simple packings and coverings to more general settings including our discrete tomography to evoke some of their results directly. In general, however, multiple packing and multiple covering appear to be harder: general reductions to single packing or covering problems are not known and not likely to exist. For this reason (and as a service to the reader) we give a full direct analysis of each of the considered algorithms.

3. Worst-case performance guarantees for iterative improvement algorithms.

3.1. Effects. Let $V \subset G$ and $g \in G \setminus V$. The *effect* $e_V(g)$ of g with respect to V is the number of lines $g + S_i$ through g for which the X-ray bound is not yet achieved by V, i.e.,

$$e_V(g) = |\{i \in \{1, \dots, m\} \colon X_{S_i} V(g + S_i) < \phi_i(g + S_i)\}|.$$

Clearly, the effect of a point is an integer between 0 and m. The notion can easily be extended to subsets of $G \setminus V$. More precisely, let $V' \subset G \setminus V$; then the *effect* $e_V(V')$ of V' with respect to V is defined by

$$e_V(V') = \sum_{i=1}^{m} \sum_{T \in \mathcal{T}_i} e_{V,V',i}(T),$$

where

$$e_{V,V',i}(T) = \begin{cases} |V' \cap T| & \text{if } |(V \cup V') \cap T| \le \phi_i(T); \\ \phi_i(T) - |V \cap T| & \text{if } |V \cap T| < \phi_i(T) \text{ and } |(V \cup V') \cap T| \ge \phi_i(T); \\ 0 & \text{if } |V \cap T| \ge \phi_i(T). \end{cases}$$

Clearly, $e_V(g) = e_V(\{g\})$; also $e_V(V')$ lends itself to a successive evaluation. In fact, if $V' = \{g_1, \ldots, g_l\}$,

$$e_V(V') = \sum_{i=1}^{l} e_{V \cup \{g_1, \ldots, g_{i-1}\}}(g_i).$$

Furthermore,

$$e = \sum_{i=1}^{m} \sum_{T \in \mathcal{T}_i} \phi_i(T)$$

is called the *total effect* of the given instance. Clearly, if L and U are feasible for the given instance of [BIF] and [BOF], respectively, then $m|L| \leq e \leq m|U|$. In particular, if F is an exact solution of RECONSTRUCTION$(S_1, \ldots, S_m)$, then $e = m|F|$.

3.2. Inner approximation algorithms. The following result gives worst-case performance guarantees for a wide class of primal algorithms for [BIF] that fit into Paradigm 2.1. In particular, all algorithms of section 4 are covered.

THEOREM 3.1. *Let $t \in \mathbb{N}_0$, let V be t-optimal for a given instance of [BIF], and let F be an optimal solution for that instance. Then*

$$\frac{|V|}{|F|} \geq \frac{2}{m} - \epsilon_m(t),$$

where

$$\epsilon_m(t) = \begin{cases} \dfrac{m-2}{m((m-1)^{s+1}-1)} & \text{if } t = 2s; \\ \dfrac{2(m-2)}{m(m(m-1)^s - 2)} & \text{if } t = 2s-1. \end{cases}$$

Observe that $\epsilon_m(t) \to 0$ as $t \to \infty$. To give an impression of how t enters the bound on the right-hand side of Theorem 3.1, we point out that for $t = 0, \ldots, 5$ the values of $2/m - \epsilon_m(t)$ are $\frac{1}{3}$, $\frac{1}{2}$, $\frac{5}{9}$, $\frac{3}{5}$, $\frac{13}{21}$, $\frac{7}{11}$ when $m = 3$ and $\frac{1}{4}$, $\frac{2}{5}$, $\frac{7}{16}$, $\frac{8}{17}$, $\frac{25}{52}$, $\frac{26}{53}$ when $m = 4$.

For the proof of the case $t > 0$ of Theorem 3.1 we need the following combinatorial result of Hurkens and Schrijver [12, Theorem 1].

PROPOSITION 3.2 (Hurkens and Schrijver). *Let $p, q \in \mathbb{N}$, let V be a set of size q, and let $E_1, \ldots, E_p$ be subsets of V. Furthermore, let $m, t \in \mathbb{N}$ with $m \geq 3$ such that the following hold:*

(i) *Each element of V is contained in at most m of the sets $E_1, \ldots, E_p$.*
(ii) *For any $r \leq t$, any r of the sets among $E_1, \ldots, E_p$ cover at least r elements of V.*

Then

$$\frac{p}{q} \leq \begin{cases} \dfrac{m(m-1)^s - m}{2(m-1)^s - m} & \text{if } t = 2s-1; \\ \dfrac{m(m-1)^s - 2}{2(m-1)^s - 2} & \text{if } t = 2s. \end{cases}$$

It is convenient to regard V and $\mathcal{E} = \{E_1, \ldots, E_p\}$ as a hypergraph $(V, \mathcal{E})$. It is clear that under the hypothesis of (i) and (ii) there is some bound on the quotient

p/q. The bounds given in Proposition 3.2, however, are not that obvious and were proved by a quite involved induction. (In addition, [12] shows that these bounds are tight.)

Let us point out that Hurkens and Schrijver [12] apply Proposition 3.2 to derive bounds for the approximation error of certain set packing heuristics, while in [11] Halldórsson utilizes it for set covering. Our subsequent analysis is based on the ideas of these papers.

Proof of Theorem 3.1. For a direct proof of the case $t = 0$, note that the effect of V has to be at least $|F|$ since otherwise the effect of $F \setminus V$ with respect to V would be greater than $(m-1)|F|$. In this case some point of F would have effect m and could hence be added to V without violating the constraints of [BIF], in contradiction to the assumption. Since the effect of V is exactly $m|V|$, the result follows.

Turning now to the general result, we note first that it suffices to give a proof under the additional assumption that $V \cap F = \emptyset$. The general case then follows via a reduction of the data functions by the X-rays of $V \cap F$ with the aid of the inequality

$$\frac{|V|}{|F|} \geq \frac{|V| - |V \cap F|}{|F| - |V \cap F|} \quad \text{for } |V| < |F|.$$

We define a hypergraph $H = (V, \mathcal{E})$ on the vertex set V with exactly $|F|$ hyperedges (one for each element of F) that satisfies the conditions (i) and (ii) of Proposition 3.2. Let $F = \{f_1, \ldots, f_p\}$ and $V = \{v_1, \ldots, v_q\}$. The family $\mathcal{E}$ of hyperedges is defined by associating to each $k = 1, \ldots, p$ with $f_k \in F$ a set $E_k \subset V$ which encodes the conflicts which the insertion of f_k would cause with respect to $\{f_1, \ldots, f_{k-1}\}$ and V.

For each line $T \in \mathcal{T}$ define a map $\iota_T : F \cap T \mapsto (F \cup V) \cap T$. Let $F \cap T = \{f_{i_1}, f_{i_2}, \ldots, f_{i_a}\}$ and $V \cap T = \{v_{j_1}, v_{j_2}, \ldots, v_{j_b}\}$.

Let $k = |F \cap T| - |V \cap T|$. If $k \leq 0$, we set $\iota_T(f_{i_l}) = v_{j_l}$. If $k > 0$, let

$$\iota_T(f_{i_l}) = \begin{cases} f_{j_l} & : \text{for } l \leq k \text{ and} \\ v_{i_{l-k}} & : \text{otherwise.} \end{cases}$$

Now we define the improvement set E_f for a given $f \in F$ by

$$E_f = \{\iota_T(f) \colon T \ni f\} \cap V.$$

We show that the assumptions of Proposition 3.2 are satisfied for $t' = t + 1$. To verify (i) recall that a point $v \in V$ lies in a set E_f if and only if there is a line T_v with $\iota_{T_v}(f) = v$. This can happen only once for each line through v; hence v is contained in at most m different sets E_f.

Next, we show that H has property (ii) of Proposition 3.2. Assume on the contrary that there are sets $E_{k_1}, \ldots, E_{k_{r+1}}$ that cover at most r elements of V for some $r \leq t$. (Here we write E_{k_i} for $E_{f_{k_i}}$ to avoid triple indices.) By choosing r to be minimal with this property, we can assume that $E_{k_1}, \ldots, E_{k_{r+1}}$ cover exactly r elements of V.

Let us consider the set

$$S = \big(V \setminus (E_{k_1} \cup \cdots \cup E_{k_{r+1}})\big) \cup \{f_{k_1}, \ldots, f_{k_{r+1}}\}.$$

We show that the set S is feasible for the given instance of [BIF]. Let $T \in \mathcal{T}$. If $|F \cap T| \leq |V \cap T|$, we have

$$|S \cap T| \leq |V \cap T| \leq \phi(T).$$

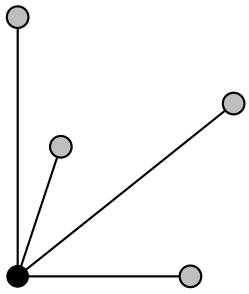


FIG. 1. *The greedy bound is tight. (Grey points belong to F; the black point constitutes V.)*

On the other hand, $|F \cap T| > |V \cap T|$ yields

$$|S \cap T| \leq |F \cap T| \leq \phi(T).$$

(The latter inequalities $|V \cap T| \leq \phi(T)$ and $|F \cap T| \leq \phi(T)$ follow from the fact that both V and F are feasible solutions.) This shows that S is indeed feasible for the given instance of [BIF].

Since S is obtained from V by deleting the r elements of $E_{k_1} \cup \cdots \cup E_{k_{r+1}}$ and inserting the $r+1$ elements $\{f_{k_1}, \ldots, f_{k_{r+1}}\}$, S facilitates an r-improvement, which is a contradiction to the assumption of t-optimality of V.

Summarizing, we have seen that (i) and (ii) of Proposition 3.2 hold for H and $t' = t+1$, and we obtain

$$\frac{p}{q} = \frac{|F|}{|V|} \leq \begin{cases} \dfrac{m(m-1)^s - m}{2(m-1)^s - m} & : \quad t+1 = 2s-1, \\ \dfrac{m(m-1)^s - 2}{2(m-1)^s - 2} & : \quad t+1 = 2s. \end{cases}$$

Hence

$$\frac{|V|}{|F|} = \frac{2}{m} - \left(\frac{2}{m} - \frac{|V|}{|F|}\right) \geq \frac{2}{m} - \epsilon_m(t)$$

which yields the assertion. □

Deterministic polynomial-time algorithms that meet the requirements of Theorem 3.1 include the greedy algorithm (for $t = 0$), or any other algorithm, according to Paradigm 2.1. In case that t-optimality is guaranteed for some $t \in \mathbb{N}_0$ when the algorithm stops, the results, however, also extend to techniques like *simulated annealing* where, with some probability, changes are allowed that replace a current feasible set by an inferior one.

The following examples show that the bounds given in Theorem 3.1 are tight in the worst case already in the most basic situations.

EXAMPLE 3.3. *Let $m \geq 3$ and let $u_1, \ldots, u_m \in \mathbb{Z}^d$ be m pairwise different lattice directions in $\mathbb{E}^d$. Let $F = \{\nu_1 u_1, \ldots, \nu_m u_m\} \subset \mathbb{Z}^d$ for some scaling factors $\nu_1, \ldots, \nu_m \in \mathbb{Z} \setminus \{0\}$. The X-rays of F in the directions $u_1, \ldots, u_m$ are taken as data functions for an instance of [BIF]. If the factors ν_i are chosen so that $G = F \cup \{0\}$, then $V = \{0\}$ is a greedy-optimal solution for [BIF]. Of course $\frac{|V|}{|F|} = \frac{1}{m}$; see Figure* 1.

FIG. 2. *The* 1*-optimality bound is tight for three directions. (Black points denote* F *in the left picture and* V *in the right picture.)*

FIG. 3. *The* 1*-optimality bound is tight for four directions. (Black points denote* F *in the left picture and* V *in the right picture.)*

FIG. 4. *The* 1*-optimality bound is tight for five directions. (Black points denote* F *in the left picture and* V *in the right picture.)*

EXAMPLE 3.4. *Let* $m = 3$ *and let* $u_1, u_2, u_3 \in \mathbb{Z}^2$ *be the directions* $(1,0), (0,1), (1,1)$. *The X-rays of* $F = \{(0,1), (1,1), (2,2), (3,2)\}$ *in the directions* u_1, u_2, u_3 *are taken as data functions for an instance of [BIF]. Then* $V = \{(1,2), (2,1)\}$ *is* 1*-optimal and* $\frac{|V|}{|F|} = \frac{1}{2} = \frac{2}{3} - \epsilon_3(1)$*; see Figure* 2.

EXAMPLE 3.5. *Let* $m = 4$ *and let* $u_1, u_2, u_3, u_4 \in \mathbb{Z}^2$ *be the directions* $(1,0), (0,1), (1,1), (1,2)$. *The X-rays of* $F = \{(0,0), (1,5), (3,4), (4,3), (5,3)\}$ *in the directions* u_1, u_2, u_3, u_4 *are taken as data functions for an instance of [BIF]. Then* $V = \{(1,0), (5,5)\}$ *is* 1*-optimal and* $\frac{|V|}{|F|} = \frac{2}{5} = \frac{2}{4} - \epsilon_4(1)$*; see Figure* 3.

EXAMPLE 3.6. *Let* $m = 5$ *and let* $u_1, \ldots, u_5 \in \mathbb{Z}^2$ *be the directions* $(1,0), (0,1), (1,1), (1,2), (2,1)$. *The X-rays of* $F = \{(0,0), (0,3), (1,3), (2,5), (4,3), (5,4)\}$ *in the directions* $u_1, \ldots, u_5$ *are taken as data functions for an instance of [BIF]. Then* $V = \{(2,4), (5,5)\}$ *is* 1*-optimal and* $\frac{|V|}{|F|} = \frac{1}{3} = \frac{2}{5} - \epsilon_5(1)$*; see Figure* 4.

Clearly, there are smarter ways to insert points into the grid than by just greedily putting one in when it fits. A more natural strategy is, for example, to apply a back-projection technique, where each candidate point gets a weight based on the X-ray values of all lines through this point. A typical example is given in Algorithm 3.7 below. In this algorithm, a specific direction S_1 is chosen, which dictates the order in which candidate points are considered for insertion into the set of points L that will eventually form V and the set of holes E (that is disjoint from V). For a fixed line T parallel to S_1, each point g on T gets a weight which depends on the number of

points still to be inserted and on the number of candidate points still available on the lines $g+S_i$ for $i \geq 2$; cf. step 2.1 of Algorithm 3.7. The corresponding ratio is a value in $[0,1]$. A value of 0 for a line $g+S_i$ indicates that the point g cannot be inserted into L and a value of 1 indicates that the point must be inserted into L. Therefore, the product over all $m-1$ other lines is a natural indicator for comparing the relative importance of the points on line T.

ALGORITHM 3.7 (weighted greedy strategy).

- INPUT: *Data functions $\phi_1, \ldots, \phi_m$ for the given lines $S_1, \ldots, S_m$.*
- OUTPUT: *A set $L \subset G$ feasible for the given instance of [BIF].*
- COMPUTATION:
  1. *Initialize $L = E = \emptyset$ and choose a specific direction, say S_1.*
  2. *For all $T \in \mathcal{T}_1$ do:*
     2.1. *For all $g \in G \cap T$ determine*

$$w_g = \prod_{i=2}^{m} \frac{\phi_i(g+S_i) - |(g+S_i) \cap L|}{|(G \setminus (L \cup E)) \cap (g+S_i)|}.$$

     2.2. *Sort $G \cap T$ according to decreasing weights w_g, $g \in G \cap T$, and add the*

$$\min\{\phi_1(g+S_1), |\{g \in G \cap T \colon w_g > 0\}|\}$$

first elements of $G \cap T$ to L and the remaining ones to E.

It is a well-known result, and already known by Lorentz [14], that a similar strategy (with a proper ordering of the lines) leads to an exact algorithm for $m = 2$ directions for consistent instances in the plane; cf. [16]. This suggests that Algorithm 3.7 might be substantially better for arbitrary m than the pure greedy algorithm, an expectation that is confirmed by the experiments stated in section 4.

Let us point out that the solutions produced by the variant of Algorithm 3.7 that is obtained by replacing the weights w_g by

$$w'_g = \prod_{i=1}^{m} \frac{\phi_i(g+S_i) - |(g+S_i) \cap L|}{|(G \setminus (L \cup E)) \cap (g+S_i)|}$$

coincide with the solutions produced by Algorithm 3.7. In fact, while w'_g usually differs from w_g, the order of points on a line in direction S_1 produced by these weights are the same.

3.3. Greedy-type insertion for outer approximation. By changing the stopping rule in Paradigm 2.1, an algorithm for solving [BIF] can be extended to an algorithm for solving [BOF]. Instead of inserting points into a set $U \subset G$ only as long as all constraints of [BIF] are satisfied, such an algorithm inserts points until the constraints of [BOF] are satisfied for the first time. As one would never insert a point into the set U that has effect 0, any such heuristic approximates [BOF] by a factor of at most m. This seems to be the dual result to Theorem 3.1 for the case $t = 0$ but it is not since the final set U is not 0-optimal in general.

ALGORITHM 3.8 (greedy insertion strategy for [BOF]).

- INPUT: *Data functions $\phi_1, \ldots, \phi_m$ for the given lines $S_1, \ldots, S_m$.*
- OUTPUT: *A set $U \subset G$ feasible for the given instance of [BOF].*
- COMPUTATION:
  1. *Initialize $U = \emptyset$ and $l = m$.*

2. *Repeat the following step until* $l = 0$*:*
   2.1. *Add points of effect* l *to* U *as long as such points exist.*
   2.2. *Decrease* l *by* 1.

In what follows it will often be necessary to regard the points of U as ordered. This underlying order will always be the point insertion order produced by Algorithm 3.8.

The performance guarantees given in the next theorem are derived by a careful analysis of the m iterations of step 2.1 in Algorithm 3.8. Further, an additional slight refinement of the algorithm is analyzed. This refinement consists of a combined treatment of points of effects 1 and 2 by means of matching techniques. More precisely, for $l = 1, \dots, m$ let $U_l \subset U$ be the set of points constructed for the parameter l in step 2.1. Then, in the modified version, $U_m, \dots, U_1$ are first constructed by step 2 of Algorithm 3.8 and, subsequently, the following computation is appended as step 3 in order to decrease $|U_1 \cup U_2|$.

3. *Repeat the following procedure until no further improvements occur:*
   3.1. *Define a graph* (V, E) *on the vertex set* $V = \cup_{i=1}^{m} \mathcal{T}_i$ *of all lines as follows: For a vertex* $v \in \mathcal{T}_i$, $1 \le i \le m$, *define the degree*

$$b_v = \max\{0, \phi_i(v) - X_{S_i}(U_3 \cup \dots \cup U_m)(v)\}.$$

The edges E *are given by means of the set* $G' = G \setminus (U_3 \cup \dots \cup U_m)$ *in the following way: For* $g \in G'$ *let* $e_g = \{v \in V : g \in v \text{ and } b_v > 0\}$. *(Note that* $|e_g| \le 2$ *since there are no points of effect at least* 3 *left in* G'*.) Now construct a minimum* b*-edge-cover* M *for* (V, E) *and set* $U_{1,2} = \{g \in G' : e_g \in M\}$ *and* $U = U_{1,2} \cup U_3 \cup \dots \cup U_m$.

Algorithm 3.8 can be implemented so as to have a polynomial running time. Using, e.g., Gabow and Tarjan's [4] weighted perfect matching algorithm to solve the capacitated b-matching problem and the b-edge cover problem, step 3.1 can also be carried out in polynomial time. A feasible set U (together with an insertion order) which does not allow any further improvements by means of the procedure in step 3.1 is called *matching-optimal* (with respect to that order). Note that the iteration of step 3 terminates in one step, since after one call upon 3.1 no further improvements are possible. This will be different after another refinement of Algorithm 3.8 is appended as step 3.2 in subsection 3.4.

THEOREM 3.9. *Let* U *be a set of points constructed by Algorithm* 3.8 *and let* F *be any feasible solution for [BOF].*

(a) *Then*

$$\frac{|U|}{|F|} \le 1 + \frac{1}{2} + \dots + \frac{1}{m} = H(m) < 1 + \log(m).$$

(b) *If* U *is matching-optimal, e.g., constructed by Algorithm* 3.8 *extended by step* 3*, then*

$$\frac{|U|}{|F|} \le \frac{5}{6} + \frac{1}{2} + \dots + \frac{1}{m} = H(m) - \frac{1}{6} < \frac{5}{6} + \log(m).$$

The bounds for $|U|/|F|$ in Theorem 3.9(a) are $\frac{11}{6}, \frac{25}{12}, \frac{137}{60}$ for $m = 3, 4, 5$, respectively. In (b) they are $\frac{5}{3}, \frac{23}{12}, \frac{127}{60}$.

Proof of Theorem 3.9. (a) Let U_l be again the set of points inserted in step 2.1 of Algorithm 3.8 for parameter l, i.e., the points which yield an effect of l upon

insertion, and let u_l be the cardinality of U_l. The effect e_l of $U_1 \cup \cdots \cup U_l$ with respect to $U_{l+1}, \ldots, U_m$ is given by $e_l = u_1 + 2u_2 + \cdots + lu_l$. On the other hand, we show that e_l is bounded from above by $l|F|$ for some l.

To this end, let e be the total effect to be attained and suppose to the contrary that $e_l > l|F|$. Consider the set $F' = F \setminus (U_{l+1} \cup \cdots \cup U_m)$. The union of F' and $U_{l+1} \cup \cdots \cup U_m$ contains F, which is feasible for the given instance of [BOF], and thus has effect e. Therefore, the effect of F' with respect to $U_{l+1} \cup \cdots \cup U_m$ is exactly e_l and hence, by our assumption, greater than $l|F|$. Since $|F'| \leq |F|$, this implies by the pigeonhole principle that there is at least one point $g \in F'$ with effect at least $l + 1$. This, however, means that the algorithm would have chosen g rather than some point in $U_1 \cup \cdots \cup U_l$ since all these points have effect at most l with respect to $U_{l+1} \cup \cdots \cup U_m$, a contradiction. Thus

$$e_l = u_1 + 2u_2 + \cdots + lu_l \leq l|F| \tag{5}$$

for $l = 1, \ldots, m$. Denoting the inequality (5) for parameter $l \in \{1, \ldots, m\}$ by I_l we consider the positive linear combination

$$\frac{1}{m} I_m + \sum_{l=1}^{m-1} \frac{1}{l(l+1)} I_l \tag{6}$$

of $I_1, \ldots, I_m$. Collecting the terms on the left and on the right of (6) we obtain

$$\sum_{i=1}^{m} u_i \leq |F| + \sum_{l=1}^{m-1} \frac{1}{l+1} |F|,$$

which is equivalent to the assertion in (a).

The proof of (b) uses the same arguments as that of (a) with the difference that appending step 3.1 to Algorithm 3.8 allows us to improve inequality I_2 to $u_1 + u_2 \leq |F|$ (instead of $u_1 + 2u_2 \leq 2|F|$).

To prove the new inequality, note that the subset $U_{1,2}$ of G' is determined in step 3.1 as a minimum b-edge-cover of (V, E). By construction it follows that $U = U_{1,2} \cup U_3 \cup \cdots \cup U_m$ is feasible for the given instance of [BOF]. Moreover, with $F' = F \setminus (U_3 \cup \cdots \cup U_m)$ the set $\{e_g \colon g \in F'\}$ is also a b-edge-cover of (V, E). Since $U_{1,2}$ is the disjoint union of (the new sets) U_1 and U_2 and is a minimum b-edge-cover, it follows that

$$|U_{1,2}| = u_1 + u_2 \leq |F'| \leq |F|. \tag{7}$$

With this inequality (instead of inequality I_2) we are led to consider a positive linear combination of type (6) with the coefficient $1/2$ of I_1 replaced by $1/3$. This reduces the contribution of I_1 to the coefficient of F on the right-hand side by $1/6$. Since the other factors remain unchanged, the bound of (b) follows. □

3.4. Outer approximation via r-improvements. The aim of this subsection is to analyze an additional refinement of Algorithm 3.8 by means of r-improvements. The first step on the way to improved bounds is to study the impact of r-improvements separately (Theorem 3.10(a)). Afterwards, the additional gain of r-improvements applied to a matching-optimal configuration is considered by appending to Algorithm 3.8 the following step 3.2 for some (fixed) $t \in \mathbb{N}_0$.

3.2. *Apply all r-improvements for $r \le t$ to $U_1 \cup U_2 \cup U_3$ that decrease U without destroying its feasibility.*

Note that the notation U_1, U_2, U_3 refers to the *updated* sets that are produced in the course of the algorithm. As in this variant r-improvements are applied only to the set $U_1 \cup U_2 \cup U_3$, the resulting algorithm is faster than a general r-improvement algorithm.

Clearly, since $t \in \mathbb{N}$ is a fixed parameter, step 3.2 can be performed in polynomial time. A trivial upper bound for the running time is $O(|G|^{2t+2})$. The geometry of discrete tomography, however, allows us to significantly reduce this bound for many values of t. The reason is that we do not need to consider all pairs of t- and $t+1$-subsets of $U_1 \cup U_2 \cup U_3$ but only those which satisfy certain compatibility conditions.

A set $U \subset G$ (together with an insertion order) is called *effect-3-t-optimal* (with respect to this order), if it cannot be decreased by the procedure of step 3.2 above, i.e., by any r-improvement, on the points of effects 1, 2, and 3.

THEOREM 3.10. *Let F be a minimum solution for a given instance of [BOF] and let $t \in \mathbb{N}_0$.*

(a) *Let U be t-optimal for that instance; then*

$$\frac{|U|}{|F|} \le \frac{m}{2} + \epsilon_m(t)\,, \text{ where } \epsilon_m(t) = \begin{cases} \dfrac{m(m-2)}{4(m-1)^{s+1} - 2m} & : \text{if } t = 2s; \\ \dfrac{(m-2)}{2(m-1)^s - 2} & : \text{if } t = 2s-1. \end{cases}$$

(b) *Let $m = 3$ and $t = 2s+1$, $s \in \mathbb{N}$. Furthermore, assume that U is matching-optimal and t-optimal (that is, effect-3-t-optimal); then*

$$\frac{|U|}{|F|} \le \frac{7}{5} + \epsilon'(t)\,, \text{ where } \epsilon'(t) = \begin{cases} \dfrac{6}{25 \cdot 2^{r+1} - 15} & : \text{if } s = 2r-1; \\ \dfrac{2}{5(5 \cdot 2^r - 1)} & : \text{if } s = 2r. \end{cases}$$

(c) *Let $t \ge 5$ and let U be matching-optimal and effect-3-t-optimal; then*

$$\frac{|U|}{|F|} \le \frac{2}{3} + \frac{1}{2} + \cdots + \frac{1}{m} < \frac{2}{3} + \log(m).$$

The values of $m/2 + \epsilon_m(t)$ in Theorem 3.10(a) for $m = 3$ and $t = 0, \ldots, 5$ are $3, 2, \frac{9}{5}, \frac{5}{3}, \frac{21}{13}, \frac{11}{7}$ and for $m = 4$ they are $4, \frac{5}{2}, \frac{16}{7}, \frac{17}{8}, \frac{52}{25}, \frac{53}{26}$. The values of $7/5 + \epsilon'(t)$ for $t = 3, 5, 7, 9, 11$ in (b) are $\frac{11}{7}, \frac{3}{2}, \frac{25}{17}, \frac{13}{9}, \frac{53}{37}$. Note that $\epsilon_m(t), \epsilon'_m(t) \to 0$ for $t \to \infty$ for all $m \ge 3$. The upper bound for $|U|/|F|$ in (c) is $\frac{3}{2}, \frac{7}{4}, \frac{39}{20}$ for $m = 3, 4, 5$, respectively.

Proof of Theorem 3.10. (a) is proved by defining a hypergraph $H = (V, \mathcal{E})$ on the vertex set $V = F$ with edges defined for each $g \in U$ that satisfies (i) and (ii) of Proposition 3.2. As in the proof of Theorem 3.1, it suffices to prove the result for $U \cap F = \emptyset$. Again, we define a map $\iota_T : U \cap T \mapsto (U \cup F) \cap T$. This time $\iota_T(u)$ encodes the information which point on T is added to compensate the deletion of u. For each line $T \in \mathcal{T}$ let $U \cap T = \{u_{i_1}, u_{i_2}, \ldots, u_{i_a}\}$ and $F \cap T = \{f_{j_1}, f_{j_2}, \ldots, f_{j_b}\}$.

If $|F \cap T| \ge |U \cap T|$ we set $\iota_T(u_{i_l}) = f_{j_l}$ for $l = 1, 2, \ldots, a$. If $|F \cap T| < |U \cap T|$ let

$$\iota_T(u_{i_l}) = \begin{cases} f_{j_l} : \text{for } l \le |F \cap T| \text{ and} \\ u_{i_l} : \text{otherwise.} \end{cases}$$

Now we define the "improvement sets" E_u for a given $u \in U$ by

$$E_u = \{\iota_T(u) \colon T \ni u\} \cap F.$$

As in the proof of Theorem 3.1, the number m of directions gives the bound in (i) and the t-optimality implies condition (ii) of Proposition 3.2 for $t' = t + 1$. Thus Proposition 3.2 can be applied, and the bound given in (a) follows.

In order to prove (b), let $U = U_1 \cup U_2 \cup U_3$ be a partition of U into subsets of points of effect 1, 2, and 3, respectively. As each point u of U_1 has effect 1 we can associate with it the line $T(u)$ to which it contributes. For $T \in \mathcal{T}$ let

$$U_T = \{u \in U_1 \cap T \colon T = T(u)\}.$$

Since $|U_T| \le \phi(T) \le |F \cap T|$ for $T \in \mathcal{T}$ we can define an injection $\kappa_T : U_T \mapsto F \cap T$. Now $U_1 = \bigcup_{T \in \mathcal{T}} U_T$, and let $\kappa : U_1 \mapsto F$ be the map induced by the injections κ_T. We show that κ is injective. In fact, if there were $u_1, u_2 \in U_1$ with $\kappa(u_1) = \kappa(u_2)$, then $T(u_1) \neq T(u_2)$, whence

$$(U \setminus \{u_1, u_2\}) \cup \{\kappa(u_1)\}$$

was feasible for the given instance of [BOF] contradicting the 1-optimality of U. It follows that

$$|U_1| = |F_1|, \tag{8}$$

where $F_1 = \kappa(U_1)$.

For the set of remaining points $F_0 = F \setminus F_1$, we use the fact that there is no r-improvement for U for any $r \le 2s + 1$ in order to show

$$|U_2| + |U_3| \le \left(\frac{3}{2} + \epsilon_3(s-1)\right) |F_0|. \tag{9}$$

To this end, let us first define the *reduced* X-ray functions

$$\gamma_i(T) = \min\{\phi_i(T), X_{S_i} F_0(T)\} \quad \text{for } T \in \mathcal{T}_i \text{ and } i = 1, 2, 3,$$

set $U_{2,3} = U_2 \cup U_3$, and note that $U_{2,3}$ is feasible for the instance $I = \{\gamma_1, \gamma_2, \gamma_3\}$ of [BOF]. Next we define a hypergraph $H = (F_0, \mathcal{E})$ with $|U_{2,3}|$ edges, again with the aid of maps ι_T for $T \in \mathcal{T}$. This time $\iota_T : U_{2,3} \cap T \mapsto (U_{2,3} \cup F_0) \cap T$, and $\iota_T(u)$ encodes the information which point on T is added to compensate for the deletion of u in the reduced problem. Let $T \in \mathcal{T}$ and $U_{2,3} \cap T = \{u_{i_1}, u_{i_2}, \ldots, u_{i_a}\}$, $F_0 \cap T = \{f_{j_1}, f_{j_2}, \ldots, f_{j_b}\}$.

If $|F_0 \cap T| \ge |U_{2,3} \cap T|$, we set $\iota_T(u_{i_l}) = f_{j_l}$ for $l = 1, 2, \ldots, a$. If $|F_0 \cap T| < |U_{2,3} \cap T|$, let

$$\iota_T(u_{i_l}) = \begin{cases} f_{j_l} : \text{for } l \le |F_0 \cap T| \text{ and} \\ u_{i_l} : \text{otherwise.} \end{cases}$$

Now we define the "improvement sets" E_u for a given $u \in U_{2,3}$ by

$$E_u = \{\iota_T(u) \colon T \ni u\} \cap F_0.$$

To obtain (9), we want to apply Proposition 3.2 to H. Clearly, condition (i) of Proposition 3.2 holds with $m = 3$. Next we show that condition (ii) holds with

parameter s. Assume on the contrary that there are $l+1$ sets $E_{u_{i_1}}, \dots, E_{u_{i_{l+1}}}$, with $l+1 \le s$, that cover only l elements $f_1, \dots, f_l \in F_0$, and let l be minimal with this property.

Let $\hat{U} = \{u_{i_1}, \dots, u_{i_{l+1}}\}$, $\hat{F} = \{f_1, \dots, f_l\}$ and set $S = (U_{2,3} \setminus \hat{U}) \cup \hat{F}$. Of course, S results from $U_{2,3}$ via an l-improvement. Let $e_{U_{2,3}\setminus\hat{U}}(\hat{U})$ $\langle e_{U_{2,3}\setminus\hat{U}}(\hat{F})\rangle$ denote the effect of $\hat{U}$ $\langle\hat{F}\rangle$, with respect to $U_{2,3} \setminus \hat{U}$ and the original data (ϕ), and let $\bar{e}$ be the corresponding effect-function for the reduced data (γ). We show that

$$e_{U_{2,3}\setminus\hat{U}}(\hat{U}) \le e_{U_{2,3}\setminus\hat{U}}(\hat{F}) + l + 3. \tag{10}$$

Of course, $e_{U_{2,3}\setminus\hat{U}}(\hat{U}) \le 3l+3$ and, since S is feasible for I, $\bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{U}) = \bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{F})$. Further, it follows from the minimality of l that $\bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{U}) \ge 2l$. In fact, if $\bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{U})$ $\le 2l-1$, then there must exist an $f \in \hat{F}$ of effect 1 with respect to $U_{2,3} \setminus \hat{U}$ and the reduced data; hence

$$(U_{2,3} \setminus (\hat{U} \setminus \{u_f\}) \cup (\hat{F} \setminus \{f\}),$$

where u_f is an element of $\hat{U}$ on the line T that carries the effect of f, would constitute an $(l-1)$-improvement. This contradiction implies that

$$e_{U_{2,3}\setminus\hat{U}}(\hat{F}) + l + 3 \ge \bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{F}) + l + 3 = \bar{e}_{U_{2,3}\setminus\hat{U}}(\hat{U}) + l + 3 \ge 3(l+1),$$

as claimed.

Next we want to lift the l-improvement for $U_2 \cup U_3$ to an r-improvement for $U_1 \cup U_2 \cup U_3$ with $r \le 2l$. From (10) we know that $e_\emptyset(((U_1 \cup U_{2,3}) \setminus \hat{U}) \cup \hat{F}) \ge e - (l+3)$. Hence, it suffices to add at most $l+3$ suitable elements $\{g_1, g_2, \dots, g_{l'}\}$ of F_1 to ensure that

$$\Big((U_1 \cup U_{2,3}) \setminus \hat{U}\Big) \cup \Big(\hat{F} \cup \{g_1, g_2, \dots, g_{l'}\}\Big)$$

is feasible. Furthermore, the points $\kappa^{-1}(g_1), \dots, \kappa^{-1}(g_l)$ can be deleted from U_1 without destroying feasibility; i.e.,

$$\Big((U_1 \cup U_{2,3}) \setminus \Big(\hat{U} \cup \{h_1, h_2, \dots, h_{l'}\}\Big)\Big) \cup \Big(\hat{F} \cup \{g_1, g_2, \dots, g_{l'}\}\Big)$$

is feasible for the (original) data (ϕ). Let $r = l + l'$; then $r \le 2l + 3 \le 2s + 1 = t$. Hence, the existence of this lifted r-improvement contradicts the t-optimality of U. So, property (ii) holds, Proposition 3.2 can be applied, and (9) follows.

In order to derive the bound of (b), inequality (8), matching-optimality (i.e., inequality (7)), and the bound $3|F|$ on the total effect of U are combined to obtain

$$3|U| = |U_1| + \underbrace{(|U_1| + |U_2|)}_{\le |F|} + \underbrace{(|U_1| + 2|U_2| + 3|U_3|)}_{\le 3|F|} \le |F_1| + 4|F|. \tag{11}$$

Furthermore, inequality (9) implies

$$|U| = |U_1| + (|U_2| + |U_3|) \le |F_1| + \left(\frac{3}{2} + \epsilon_3(s-1)\right)|F_0|. \tag{12}$$

Multiplying (11) with $\frac{1}{2}+\epsilon_3(s-1)$, adding (12), and using $|F_0|+|F_1|=|F|$ then give

$$\left(\frac{5}{2}+3\epsilon_3(s-1)\right)|U| \le \left(\frac{7}{2}+5\epsilon_3(s)\right)|F|,$$

which implies assertion (b).

Note that the proof provides a result that is slightly stronger than assertion (b). In fact, the argument does not use the assumption $m=3$ "globally" but only "locally." More precisely, let $m \ge 3$, $s \in \mathbb{N}$, $t = 2s+1$, and let U be matching-optimal and effect-3-t-optimal. Then

$$\begin{array}{rcl} |U_1| & = & |F_1| \quad \text{and} \\ |U_1|+|U_2|+|U_3| & \le & |F| + \left(\frac{1}{2}+\epsilon_3(s-1)\right)|F_0|. \end{array} \tag{13}$$

Finally, we turn to assertion (c). First, we form the positive linear combination

$$\frac{1}{m}I_m + \sum_{l=4}^{m-1} \frac{1}{l(l+1)} I_l$$

of the inequalities (5) derived in the proof of Theorem 3.9. Collecting terms for $U_1, \ldots, U_m$ yields

$$\frac{1}{4}|U_1| + \frac{2}{4}|U_2| + \frac{3}{4}|U_3| + |U_4| + \cdots + |U_m| \le \left(1+\frac{1}{5}+\cdots+\frac{1}{m}\right)|F|.$$

Thus it remains to show that

$$\frac{3}{4}|U_1| + \frac{2}{4}|U_2| + \frac{1}{4}|U_3| \le \frac{3}{4}|F|.$$

Since $5 \le t = 2s+1$, we can apply (13) for $s=2$. This yields

$$2|U_1| + |U_2| + |U_3| \le 2|F|.$$

Matching-optimality implies again

$$|U_1| + |U_2| \le |F|,$$

whence addition of these inequalities gives

$$3|U_1| + 2|U_2| + |U_3| \le 3|F|.$$

This concludes the proof of Theorem 3.10. □

4. Computational results. In this section we report on computational results for implementations of the different algorithms outlined in the previous sections.

4.1. Description of the implementations. We implemented six different algorithms for [BIF]. The first algorithm (GreedyA) is the plain greedy algorithm (see Figure 5) which considers all positions in a random order and tries to place atoms at these positions. The second algorithm (GreedyB) is a variant of the line following greedy algorithm, Algorithm 3.7 (Figure 6). The algorithm chooses a direction with

```
procedure GreedyA
Calculate a random permutation of all points
For each point in the order of this permutation do
    Check whether any line passing through this point is saturated
    If no line is saturated then
        Add the point to the solution set
        Update the sums of the lines passing through this point
```

FIG. 5. *The plain greedy solver.*

maximal support $|\mathcal{T}_i|$. Suppose—in accordance with the notation in Algorithm 3.7—that $i = 1$. The lines $T \in \mathcal{T}_1$ are then considered with respect to decreasing line-weights

$$\phi_1(T)/|G \cap T|.$$

The algorithm usually performs quite well. However, if one regards the "en block" point insertion procedure successively, i.e., as a point-by-point insertion, then the adapted line-weights change and at some point—possibly long before the last point of the block has been inserted—another line might be more profitable. This idea is pursued in a third greedy algorithm (GreedyC) which changes the weights of all lines and uninspected points after a new point is placed; see Figure 7. The initial problem is, of course, that after each insertion a complete search for the next position of maximum weight is necessary. This increases the computation times dramatically. A good data structure for keeping the points (partially) ordered according to their weights is a heap. After a point insertion, it suffices to update the weights of points on lines through the new point. While a heap can perform this quite efficiently, this procedure is still pretty time consuming since the weights of points change frequently, without the element even being close to the top of the heap. We decided therefore to use a lazy-update. For this we take the top element of the heap and recompute its weight. Then we compare its stored weight with its actual weight (they might differ due to recent insertions). If the weights are equal, this is still the top element of the heap, and we can try to insert it. If the weights differ, the candidate point gets the new weight and the heap needs to be restructured. After the restructuring we start again with the (new) top element.

The last type of algorithm is the 1-improvement algorithm according to Paradigm 2.1. We tried three different variants (ImprovementA, ImprovementB, ImprovementC) depending on the greedy algorithm (GreedyA, GreedyB, GreedyC) used first; see Figure 8 As the 1-improvement algorithm already needs 22 minutes on average for some instances and the results are very good, we did not implement higher improvement algorithms (like 2-improvement, etc.).

4.2. Performance of the implemented algorithms. In this subsection we report on different experiments we conducted with the algorithms described in the previous section. We performed several tests for problems of size 20×20 to 500×500, with 2 to 5 directions and of density between 10% and 90%. After analyzing the different experiments, we observed that the experiments with varying numbers of directions, but a fixed density of 50%, are most representative and the other series behave similarly. (For more data on the computational performance of the evaluated heuristics for other densities of $1\%, 5\%$, and 20% see de Vries [19].)

Even though our program can solve problems in three dimensions and on arbitrary crystal-lattices, we decided to present here only results for 2-dimensional problems on the square lattice, as in the physical application all directions belong to a single plane

```
procedure GreedyB
Determine a direction with maximal support
Sort the lines parallel to that direction by descending line-weights
For each of these lines (T) in this order do
    For each point on T do
        Calculate its weight (the product of the line-weights)
    Sort the points on T with respect to descending weights
    For each point in this order do
        Check whether any line passing through this point is saturated
        If no line is saturated then
            Add the point to the solution set
            Update the sums of the lines passing through this point
```

FIG. 6. *The line following greedy solver.*

```
procedure GreedyC
For each point do
    Calculate the weight of the point (the product of the relative line capacities)
        and insert it into the heap
While there are still points in the heap do
    Find the maximum weight and a corresponding point and remove it from the heap
    Check whether any line passing through this point is saturated
    If no line is saturated then
        Add the point to the solution set
        Update the sums of the lines passing through this point
```

FIG. 7. *The dynamically reordering greedy solver.*

(therefore the problem can be solved in a slice-by-slice manner); furthermore, this restriction should facilitate the comparison with other, less general codes currently under development by various research groups.

Whenever we report either running-times or performances, we report the average of 100 randomly generated instances. We decided to use random instances for two reasons. The first reason is that we still lack sufficient experimental data from the physicists. On the other hand, it is typically easy to detect and then eliminate invariant points, i.e., points that either must belong to every solution or do not belong to any solution. Since the invariant points carry much of the physical a priori knowledge, the reduced problem tends to be quite unstructured.

To obtain a random configuration of prescribed density, we generate a random permutation of the positions of the candidate grid and then place atoms in this order until the described density is reached. After calculating the lines and their sums we discard the configuration itself. Then we preprocessed the problem by calculating the incidence tables, which are necessary for all algorithms. The running-times we report were obtained on an SGI Origin 200 computer with four MIPS R10000 processors at 225MHz with 1GB of main memory and by running three test programs at the same time.

Note that all instances are consistent. There are two reasons for this. First, for inconsistent problems we need the exact solution to evaluate the performance of the heuristics. But for the relevant dimensions there are at present no algorithms available that produce exact solutions in reasonable time. The second reason is that the true nature of the error distribution for the real physical objects has not yet been experimentally determined by the physicists. So it is not clear how to perturb an exact instance to obtain inconsistent problems in a physically reasonable manner.

The performance plotted in Figure 9 is the quotient of the cardinality of the

```
procedure Improvement[ABC]
Calculate a solution U according to GreedyA, GreedyB, or GreedyC
Repeat
    For each point (p1) of the candidate grid do
        If p1 ∈ U then continue with the next point
        If no line passing through p1 is saturated then
            Add p1 to U
            Update the sums of the lines passing through p1
            Continue with the next point
        If more than one line passing through p1 is saturated then continue with the next point
        For each point (p2) of U on the saturated line do
            For each nonsaturated line (T1) through p1 do
                Calculate the line (T2) parallel to T1 passing through p2
                For each point (p3) on T2 not in U do
                    If the lines passing through p3 and
                        not containing p1 or p2 are nonsaturated and
                        the line passing through p3 and p1 (if existent)
                        has at least one point not in U then
                        Perform the improvement:
                            Remove p2 from U
                            Add p1 to U
                            Add p3 to U
                        Update the sums of all lines passing through p1, p2, or p3
                        Continue with the next point
Until no improvement was done in the last loop
```

FIG. 8. *The improvement solvers.*

approximate solution to that of an optimal solution. The closer it is to 1 the better the result. It turns out that the larger the problems, the better every algorithm performs in terms of relative errors (see Figure 9). Obviously, postprocessing the output of some greedy algorithm with an improvement algorithm cannot decrease the performance (usually it improves the performance). However, it turns out that GreedyB outperforms ImprovementA (for four and five directions) and that GreedyC performs better than ImprovementB (for five directions; for four directions they are similar and for three directions ImprovementB is better).

The running-times for the algorithms GreedyA and GreedyB are less than 4 seconds for all instances (of size up to 500×500). The application of the 1-improvements to their results increases the running-time to up to 110 seconds.

The running-times of GreedyC and ImprovementC increase much faster than those for the other algorithms. Still, they take only up to 1320 seconds. This is long, but in fact these algorithms provide very close approximations, while presently available exact algorithms seem incapable of solving 500×500 problems in less than a century. Furthermore, knowing a solution for a neighboring slice should speed up the solution of the next slice by a good amount; so there is hope of solving even $500 \times 500 \times 500$ real world problems in time that is acceptable in practice.

The better of the presented algorithms are so good that we also compare their *absolute* errors (see Figure 10). As can be seen, the absolute error for ImprovementC seems constant for three directions. (Of course, it follows from [5] that asymptotically there must be a more than constant worst-case error unless $\mathbb{P} = \mathbb{NP}$.) For four and five directions the absolute error appears to be $O(\sqrt{|G|})$.

Another (practically) important issue is that of the distribution of errors among different lines. For this we counted for 100 problems of size 500×500 how many constraints were satisfied with equality, how many needed only one more point for

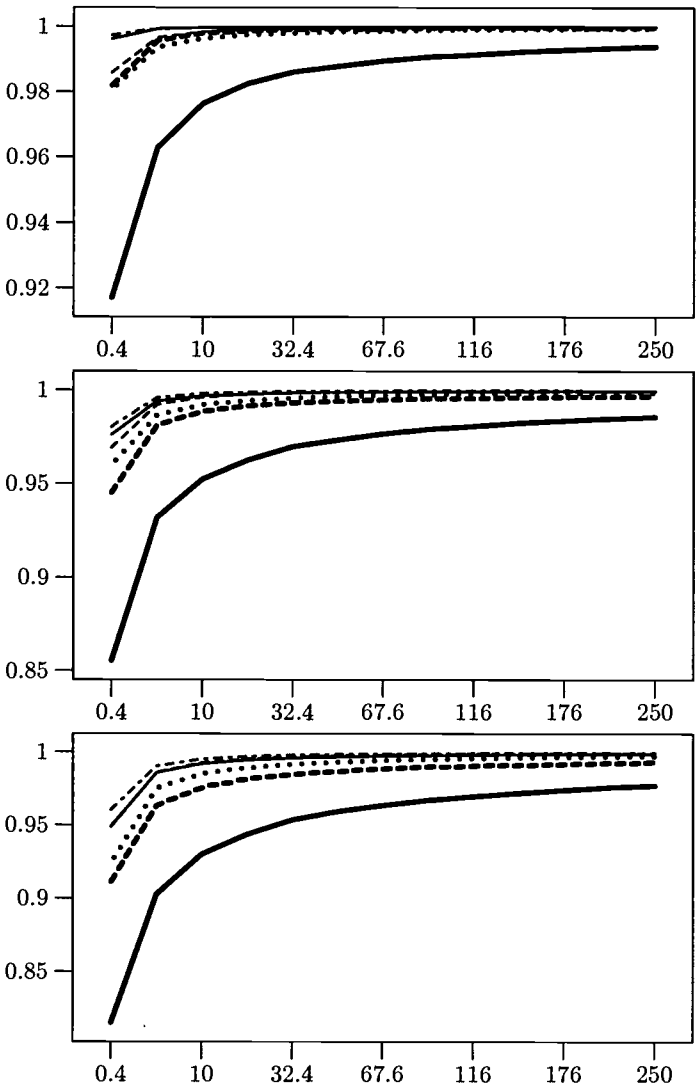


FIG. 9. *Relative performance for* 3 *(top),* 4 *(middle), and* 5 *(bottom) directions on instances of* 50% *density for GreedyA (—), ImprovementA (– –), GreedyB (· · ·), ImprovementB (— ·), GreedyC (– · –), and ImprovementC (– · · –). The abscissa depicts the number of variables in thousands at a quadratic scale and the ordinate depicts the relative performance.*

equality, and so on. Again, it turned out that the algorithms GreedyC and ImprovementC have the best error distribution. In particular, for GreedyC no line occurred with error greater than 1 for 3 and 4 directions; for 5 directions the worst case was 1 instance with a single line of error 2. For ImprovementC the worst cases were 3 instances with one line of error 2 for 3 directions, for 4 directions 1 instance with a single line of error 4, and for 5 directions 1 instance with a single line of error 5. Of course, while 1-improvements never decrease the number of points placed, the variation of errors over the single lines may increase, as it may happen that in a number of improvement steps atoms from the same line are removed.

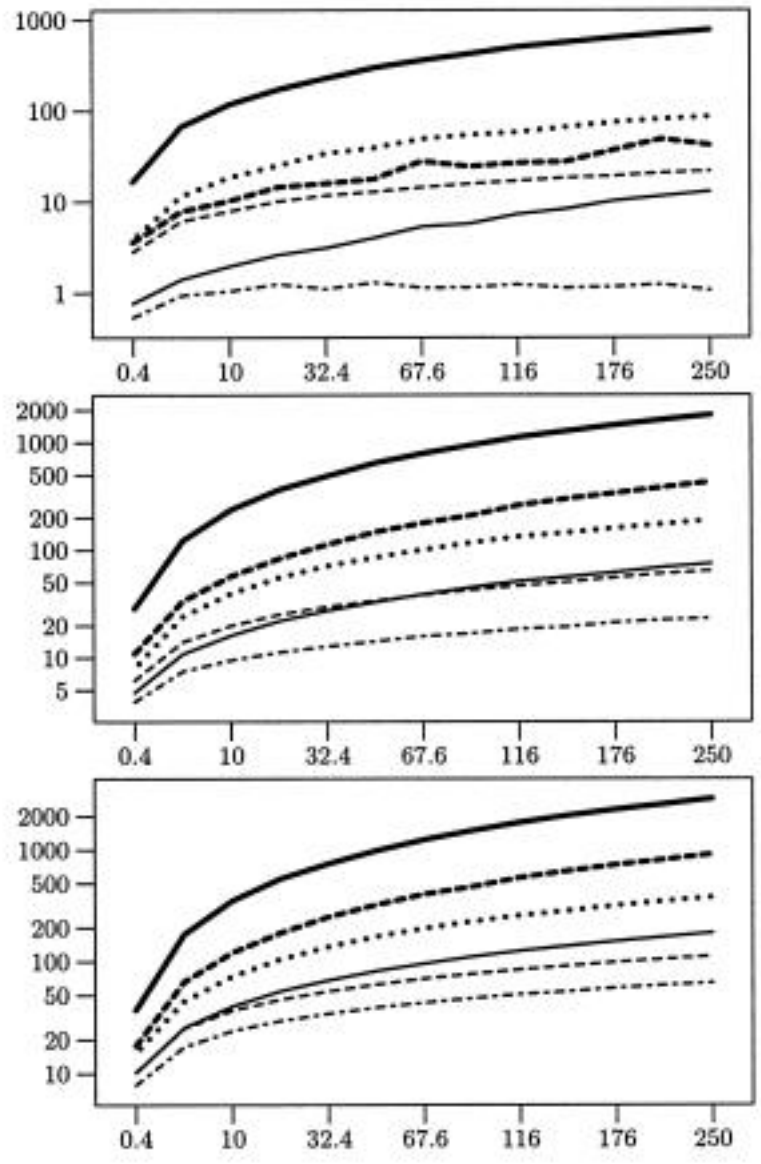


FIG. 10. *Absolute error for* 3 *(top),* 4 *(middle), and* 5 *(bottom) directions on instances of* 50% *density for GreedyA (—), ImprovementA (– –), GreedyB (· · ·), ImprovementB (— ·), GreedyC (— · ·), and ImprovementC (– – –). The abscissa depicts the number of variables in thousands at a quadratic scale and the ordinate depicts the absolute error at a logarithmic scale.*

For GreedyC only lines with error at most 2 occur, while for ImprovementC a single instance with a line of error 5 came up. In contrast, GreedyA, GreedyB, ImprovementA, and ImprovementB always have a couple of lines with a huge error (see Figure 11). For instance, for GreedyA, ImprovementA, GreedyB, and ImprovementB instances occurred with lines of error 67, 130, 109, and 66. These huge errors do seem inappropriate in the physical application since it is more likely that many lines occur with small error rather than with very large error.

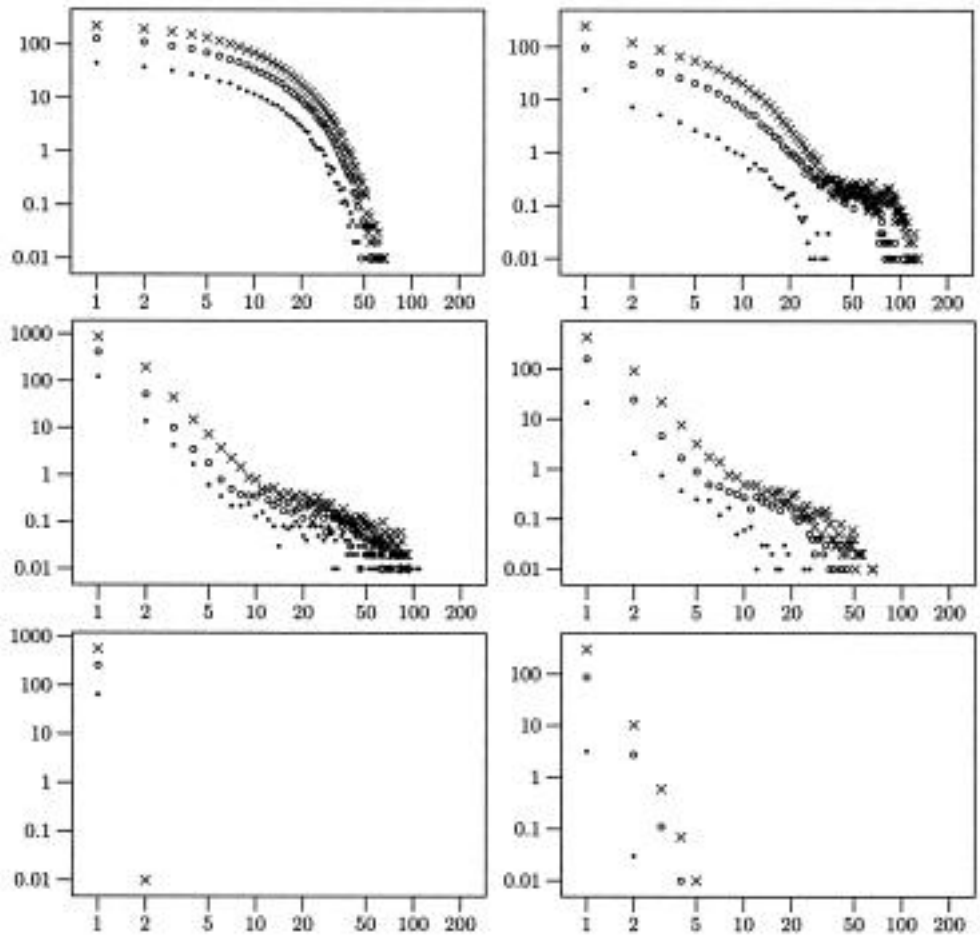


FIG. 11. *Distribution of error for* 100 *instances with* 500^2 *variables, density* 50%, *and* 3 (•), 4 (○), *and* 5 (×) *directions. Depicted are: GreedyA (top left), ImprovementA (top right), GreedyB (middle left), ImprovementB (middle right), GreedyC (bottom left), and ImprovementC (bottom right). The abscissa depicts the absolute error on a line at a logarithmic scale and the ordinate depicts the average number of lines with this error at a logarithmic scale.*

Acknowledgments. We thank Jens Zimmermann for helping with coding the data structures and algorithms. Furthermore, we thank the referees for some valuable comments.

REFERENCES

[1] S. Chang, *The reconstruction of binary patterns from their projections*, Comm. ACM, 14 (1971), pp. 21–25.

[2] R. Duh and M. Fürer, *Approximation of k-set cover by semi-local optimization*, in Proceedings, STOC '97, ACM, New York, pp. 256–264.

[3] P. Fishburn, P. Schwander, L. Shepp, and R. Vanderbei, *The discrete Radon transform and its approximate inversion via linear programming*, Discrete Appl. Math., 75 (1997), pp. 39–61.

[4] H. Gabow and R. Tarjan, *Faster scaling algorithms for general graph-matching problems*, J. ACM, 38 (1991), pp. 815–853.

[5] R. Gardner, P. Gritzmann, and D. Prangenberg, *On the computational complexity of reconstructing lattice sets from their X–rays*, Discrete Math., 202 (1999), pp. 45–71.

[6] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP–Completeness*, W.H. Freeman, San Francisco, CA, 1979.

[7] J. Gerbrands and C. Slump, *A network flow approach to reconstruction of the left ventricle from two projections*, Comput. Graphics Image Process., 18 (1982), pp. 18–36.

[8] O. Goldschmidt, D. Hochbaum, and G. Yu, *A modified greedy heuristic for the set covering problem with improved worst case bound*, Inform. Process. Lett., 48 (1993), pp. 305–310.

[9] P. Gritzmann, *On the reconstruction of finite lattice sets from their X-rays*, in Proceedings 7th International Workshop, DGCI '97, Montpellier, France, E. Ahronovitz and C. Fiorio, eds., Lecture Notes in Comput. Sci. 1347, Springer, Berlin, 1997, pp. 19–32.

[10] P. Gritzmann, D. Prangenberg, S. de Vries, and M. Wiegelmann, *Success and failure of certain reconstruction and uniqueness algorithms in discrete tomography*, Int. J. Imaging Syst. Technol., 9 (1998), pp. 101–109.

[11] M. Halldórsson, *Approximating k–set cover and complementary graph coloring*, in Proceedings 5th International Integer Programming and Combinatorial Optimization Conference, Lecture Notes in Comput. Sci., Springer-Verlag, Berlin, 1996, pp. 118–131.

[12] C. A. J. Hurkens and A. Schrijver, *On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems*, SIAM J. Discrete Math., 2 (1989), pp. 68–72.

[13] C. Kisielowski, P. Schwander, F. Baumann, M. Seibt, Y. Kim, and A. Ourmazd, *An approach to quantitative high-resolution transmission electron microscopy of crystalline materials*, Ultramicroscopy, 58 (1995), pp. 131–155.

[14] G. Lorentz, *A problem of plane measure*, Amer. J. Math., 71 (1949), pp. 417–426.

[15] H. Ryser, *Combinatorial properties of matrices of zeros and ones*, Canad. J. Math., 9 (1957), pp. 371–377.

[16] H. Ryser, *Combinatorial Mathematics*, Carus Math. Monogr. 14, Mathematical Association of America and John Wiley, New York, 1963, ch. 6, Matrices of zeros and ones, pp. 61–78.

[17] A. Schrijver, *Theory of Linear and Integer Programming*, Wiley, New York, 1986.

[18] P. Schwander, C. Kisielowski, F. Baumann, Y. Kim, and A. Ourmazd, *Mapping projected potential, interfacial roughness, and composition in general crystalline solids by quantitative transmission electron microscopy*, Phys. Rev. Lett., 71 (1993), pp. 4150–4153.

[19] S. de Vries, *Discrete Tomography, Packing and Covering, and Stable Set Problems: Polytopes and Algorithms*, Ph.D. thesis, Technische Universität München, München, Germany, 1999.

[20] M. Wiegelmann, *Gröbner Bases and Primal Algorithms in Discrete Tomography*, Ph.D. thesis, Technische Universität München, München, Germany, 1998.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 547–571

ON $\mathcal{VU}$-THEORY FOR FUNCTIONS WITH PRIMAL-DUAL GRADIENT STRUCTURE*

ROBERT MIFFLIN† AND CLAUDIA SAGASTIZÁBAL‡

Abstract. We consider a general class of convex functions having what we call primal-dual gradient structure. It includes finitely determined max-functions and maximum eigenvalue functions as well as other infinitely defined max-functions. For a function in this class, we discuss a space decomposition that allows us to identify a subspace on which the function appears to be smooth. Moreover, using the special structure of such a function, we compute smooth trajectories along which certain second-order expansions can be obtained. We also give an explicit expression for the Hessian of a related Lagrangian.

Key words. convex minimization, max-functions, second-order derivatives, $\mathcal{VU}$-decomposition, eigenvalue optimization

AMS subject classifications. Primary, 49K35, 49M27; Secondary, 65K10, 90C25

PII. S1052623499350967

1. Introduction. Motivation.

The problem of minimization of a convex max-function f arises in many applications. Well-known examples occur in Chebyshev's best polynomial approximation, decomposition approaches using Lagrangian relaxation, exact penalty methods for nonlinear programming, and shape optimization [4], [2]. Another important application that has been studied intensively in the past few years is the minimization of the maximum eigenvalue function (`mef`); see [16], [8].

One of the main difficulties with such problems is that f is not differentiable at those points, where more than one underlying function defining the max is active. Furthermore, there is no well-defined Hessian, or second-order object, and a straightforward application of a Newton-type method is not possible.

However, the `mef` has "good" structural properties: under certain regularity conditions its nonsmoothness may be dealt with by formulating a locally equivalent constrained problem that has a smooth trajectory leading to an optimal solution. This is the essential idea in [16] which is further studied in [17]. Along the same lines, a deep second-order analysis of the `mef` class was done in [21].

More recently, `mef`s have been considered in [14], this time in light of the $\mathcal{VU}$-space decomposition theory developed in [7]. In this context, $\mathcal{VU}$-theory yields a superlinearly convergent bundle-type method [15], [14] provided enough eigenvectors at each point are known. For quite general convex functions of one variable, there is a rapidly convergent $\mathcal{VU}$-algorithm [6] that requires only one subderivative at each iterate. Using the implementation of this last algorithm in [9] it would be possible to

*Received by the editors January 20, 1999; accepted for publication (in revised form) June 15, 2000; published electronically November 2, 2000.

http://www.siam.org/journals/siopt/11-2/35096.html

†Department of Pure and Applied Mathematics, Washington State University, Pullman, WA 99164-3113 (mifflin@math.wsu.edu). The research of this author was supported National Science Foundation grants DMS-9703952 and DMS-0071459 and FAPERJ (Brazil) grant E-26/171.393/97.

‡COPPE-Sistemas e Computacão, Universidade Federal do Rio de Janeiro, P.O. Box 68511, Rio de Janeiro, RJ 21945-970, Brazil (sagastiz@impa.br). Current address: INRIA, B.P. 105, 78153 Le Chesnay, France (Claudia.Sagastizabal@inria.fr). The research of this author was supported by FAPERJ (Brazil) grant E26/150.205/98.

solve the single variable eigenvalue problem in [18] which is equivalent to an n-variable equality constrained trust region subproblem.

The general idea in [7] is to decompose $\mathbb{R}^n$ into two orthogonal subspaces $\mathcal{V}$ and $\mathcal{U}$ in such a way that, near $\bar{x} \in \mathbb{R}^n$, all of f's nonsmoothness is concentrated in $\mathcal{V}$:

$$\mathcal{V} := \text{lin}(\partial f(\bar{x}) - g) \quad \text{and} \quad \mathcal{U} := \mathcal{V}^{\perp}, \tag{1.1}$$

where g is any subgradient in the subdifferential of f at $\bar{x}$ denoted by $\partial f(\bar{x})$. Here lin$Y$ denotes the linear hull of a set Y and (sub)gradients are considered to be column vectors.

As a result, f appears to be smooth on the $\mathcal{U}$-subspace and may have some kind of related Hessian. When f is structured and qualified enough such a second-order object exists; we call it a "$\mathcal{U}$-Hessian" and denote it by $\text{H}_{\mathcal{U}} f$. Moreover, it is possible to find smooth trajectories, tangent to $\mathcal{U}$, yielding a second-order expansion for f.

In Definition 2.1 below we give a precise meaning to the wording "structured enough" by introducing the concept of *primal-dual gradient* (`pdg`) structure. The class of `pdg`-structured functions is quite large and includes `mef`s as well as other convex functions such as some that are pointwise maxima of finite [10], [1] or infinite [11] collections of smooth functions. For piecewise affine functions the structure of subdifferentials is discussed in detail in [23], [12], and [13].

The `pdg` structure provides us with one or more so-called basic index sets whose associated vectors span a subspace of $\mathcal{V}$ and generate an implicit function therein from which a smooth trajectory tangent to $\mathcal{U}$ can be defined. We express our qualification conditions in the form of what we call $\mathcal{V}$*-optimality* conditions on such a trajectory, gathering together the concepts of primal feasibility, dual feasibility, and transversality. These conditions play a role similar to that of constraint qualification conditions in nonlinear programming. When they are satisfied for some basic index set, f has a second-order expansion along the associated trajectory. In particular, it is shown in section 6.1 that for the special `mef` case our transversality conditions are weaker than the strong transversality conditions used in [17], [21], and [14].

The paper is organized as follows. We introduce the `pdg`-structured class in section 2 and give some examples in section 3.1 which are used to illustrate the results in some subsequent sections. Section 3.2 shows how a general convex `mef` fits into our class. We recall the main elements of the $\mathcal{VU}$-space decomposition theory in section 4. Smooth trajectories and associated multiplier functions are studied in section 5. Finally, in section 6 we give $\mathcal{V}$-optimality conditions and first- and second-order results, including an explicit expression for $\text{H}_{\mathcal{U}} f$. The concluding section 7 gives some indication of directions for future research.

2. Function structure. Many, but not all, of the convex functions covered by our theory are maximum value functions of the form

$$f(x) := \max\{F(x,t) : t \in \mathcal{T}\} \quad \text{for } x \in \mathbb{R}^n, \tag{2.1}$$

where $\mathcal{T}$ is a closed subset of $\mathbb{R}^m$ and $F(x,t)$ and its first- and second-order partial derivatives with respect to components of x are continuous on $\mathbb{R}^n \times \mathcal{T}$. Both the finitely defined max-function ($\mathcal{T} = \{0, 1, \ldots, M\}$) and the `mef` ($\mathcal{T} = \{t \in \mathbb{R}^m : t^T t = 1\}$) are particular instances of (2.1). The former case was fully developed in [10]. As for the latter, we show in section 3.2 how it fits into our framework as a special case.

Since for a convex function f as in (2.1) $\partial f(x)$ equals the set of Clarke generalized gradients at x, if $\mathcal{T}$ is compact, the subdifferential is the following convex hull

[3, Corollary 2 to Theorem 2.8.2, p. 87]:

$$\partial f(x) = \operatorname{conv}\{\nabla_x F(x, t_a) \text{ for all } t_a \in \mathcal{T} \text{ such that } f(x) = F(x, t_a)\}. \tag{2.2}$$

Example 3.0 given below shows that the compactness of $\mathcal{T}$ is not a necessary condition for equality (2.2) to hold.

2.1. Finite max-functions. To motivate our Definition 2.1 below, we first consider convex finite max-functions as in [10], i.e., convex functions f defined by (2.1) with $\mathcal{T} = \{0, 1, \ldots, M\}$. For $x = \bar{x} \in \mathbb{R}^n$ in (2.2) we suppose that all active indices t_a are given by $t_a = 0, 1, \ldots, m_1$, where $m_1 \leq M$, so that $f(\bar{x}) > F(\bar{x}, t)$ if $t \neq t_a$. Then about $\bar{x}$ f satisfies the following conditions.

There exists a ball about $\bar{x}$, denoted by $B(\bar{x})$, $m_1 + 1$ functions

$$f_i(\cdot) := F(\cdot, i) \quad \text{for } i = 0, 1, \ldots, m_1$$

that are C^2 on $B(\bar{x})$ and a multiplier set $\Delta_1 \subset \mathbb{R}^{m_1+1}$ such that

(i) $\bar{x} \in B(\bar{x})$ and $f_i(\bar{x}) = f(\bar{x})$ for $i = 0, 1, \ldots, m_1$;

(ii) for each $x \in B(\bar{x})$, $f(x) = \max\{f_i(x) : i = 0, 1, \ldots, m_1\}$;

(iii) Δ_1 is the unit simplex in $\mathbb{R}^{m_1+1}$ given by

$$\Delta_1 := \left\{ (\alpha_0, \alpha_1, \ldots, \alpha_{m_1}) : \sum_{i=0}^{m_1} \alpha_i = 1, \alpha_i \geq 0, i = 0, 1, \ldots, m_1 \right\}; \tag{2.3}$$

(iv) for each $x \in B(\bar{x})$, $g \in \partial f(x)$ if and only if

$$g = \sum_{i=0}^{m_1} \alpha_i \nabla f_i(x),$$

where $\alpha_i = 0$ if $f_i(x) < f(x)$ and $\alpha := (\alpha_0, \alpha_1, \ldots, \alpha_{m_1}) \in \Delta_1$. □

The minimization of a max-function as above can be formulated (with an additional variable as in Remark 6.2 below) as a nonlinear programming (NLP) problem. In this context it is well known that the subspace tangent to the active constraints at the solution plays a fundamental role. This subspace is the kernel of the Jacobian of the constraints.

Suppose $\bar{x}$ is a minimizer of f. Using $\mathcal{VU}$-space decomposition, the subspace $\mathcal{V}$ from (1.1) is spanned by the active gradient differences $\{\nabla f_i(\bar{x}) - \nabla f_0(\bar{x})\}_{i=0}^{m_1}$, while its orthogonal complement $\mathcal{U}$ is the NLP tangent space mentioned above. Under certain regularity assumptions, called $\mathcal{V}$-optimality conditions, comparable to the qualification of constraints in NLP, we proved in [10] that for u near 0 there exists at least one smooth trajectory $x(u)$ in $B(\bar{x})$ which is tangent to $\mathcal{U}$ at $\bar{x}$ and has an associated smooth multiplier vector function $\alpha(u) \in \Delta_1$. These two smooth functions provide us with a constructive way to express the $\mathcal{U}$-Hessian of f in terms of the Hessians of the f_is (in fact, $\mathrm{H}_{\mathcal{U}} f$ is the $\mathcal{UU}$ block of the Hessian of the NLP Lagrangian). Along such a trajectory $x(u)$, f can be expanded up to second-order as given in (5.2) below, where the development does not depend on $\bar{x}$ being a minimizer.

2.2. `pdg` structure. We call $f(x)$ in (2.1) an infinite max-function if $\mathcal{T}$ is not a finite set. In this case the set of maximizing t_as corresponding to x in (2.2) may have a convex hull that does not have a finite number of extreme points. This can lead to $\partial f(x)$ having a continuum of extreme points.

To apply $\mathcal{VU}$-theory, it is crucial to properly describe all of the subgradients in order to identify spanning vectors for the $\mathcal{V}$-subspace. Associated with this is the fact that the multipliers forming the convex combinations in (2.2) need to satisfy certain conditions. This is the purpose of the conditions relating the finite number of functions f_i and φ_ℓ and associated multipliers α_i introduced next. In particular, the presence of the functions φ_ℓ and corresponding multipliers allows the subdifferential to have a continuum of extreme points.

DEFINITION 2.1. *We say that a convex function $f : \mathbb{R}^n \to \mathbb{R}$ has* pdg structure *about $\bar{x} \in \mathbb{R}^n$ if the following conditions hold:*

There exists a ball *about $\bar{x}$, $B(\bar{x})$, $m_1 + 1 + m_2$* primal functions

$$\{f_i(x)\}_{i=0}^{m_1} \quad \textit{and} \quad \{\varphi_\ell(x)\}_{\ell=1}^{m_2}$$

that are C^2 on $B(\bar{x})$ and a dual multiplier set *$\Delta \subset \mathbb{R}^{m_1+1+m_2}$ such that*

(i) *$\bar{x} \in \mathcal{P} := \{x \in B(\bar{x}) : \varphi_\ell(x) = 0$ for $\ell = 1, \ldots, m_2\}$ and $f_i(\bar{x}) = f(\bar{x})$ for $i = 0, 1, \ldots, m_1$;*

(ii) *for each $x \in \mathcal{P}$,*

$$f(x) = \max\{f_i(x) : i = 0, 1, \ldots, m_1\};$$

(iii) *Δ is a closed convex set such that*

(a) *if $\alpha := (\alpha_0, \ldots, \alpha_{m_1}, \alpha_{m_1+1}, \ldots, \alpha_{m_1+m_2}) \in \Delta$, then $(\alpha_0, \ldots, \alpha_{m_1})$ is an element of the unit simplex Δ_1 defined in* (2.3);

(b) *for each $i = 0, 1, \ldots, m_1$, $\mathbf{1}_{i+1} \in \Delta$, where $\mathbf{1}_j$ is the jth unit vector in $\mathbb{R}^{m_1+1+m_2}$;*

and

(c) *for each $\ell = 1, 2, \ldots, m_2$, there exists $\alpha^\ell \in \Delta$ such that $\alpha^\ell_{m_1+\ell} \neq 0$ and $\alpha^\ell_{m_1+i} = 0$ for $i \in \{1, 2, \ldots, m_2\} \backslash \{\ell\}$;*

(iv) *for each $x \in \mathcal{P}$, $g \in \partial f(x)$ if and only if*

$$g = \sum_{i=0}^{m_1} \alpha_i \nabla f_i(x) + \sum_{i=m_1+1}^{m_1+m_2} \alpha_i \nabla \varphi_{i-m_1}(x), \tag{2.4}$$

where the multipliers $\alpha_0, \alpha_1, \ldots, \alpha_{m_1+m_2}$ satisfy

Complementary slackness: *$\alpha_i = 0$ if $f_i(x) < f(x)$ and $i \leq m_1$*

and

Dual feasibility: *$\alpha = (\alpha_0, \alpha_1, \ldots, \alpha_{m_1+m_2}) \in \Delta$.* ∎

For some examples, such as a finite max-function, $m_2 = 0$, there are no φ_ℓ functions and $\Delta = \Delta_1$ as defined in (2.3). In the next section we present some examples with $m_2 \neq 0$, including the `mef`, which is developed in full detail in section 3.2. We show that in this case Δ corresponds to the set of positive semidefinite matrices having unit traces. As for the f_is and φ_ℓs, they occur quite naturally; see [21]. Essentially, they depend on some vector functions forming a basis for a subspace spanned by certain eigenvectors.

3. Some pdg-structured functions. Before studying the (rather involved) pdg structure of general maximum eigenvalue functions, we first consider simpler examples.

3.1. Initial examples. When both m_1 and m_2 are 0, then $\Delta = \{\alpha_0 = 1\}$ and $\partial f(\bar{x}) = \{\nabla f_0(\bar{x})\}$, as illustrated by the following simple function.

Example 3.0. Define the function $\mathcal{E}_0 := f$ in (2.1), with $n = 1$, $\mathcal{T} = \mathbb{R}$, $F(x,t) = tx - \frac{1}{2}t^2$. Then at any x the maximizing value of t is x,

$$\mathcal{E}_0(x) = F(x,x) = \frac{1}{2}x^2 \quad \text{and} \quad \partial\mathcal{E}_o(x) = \{x\}.$$

Therefore, for any $\bar{x} \in \mathbb{R}$, $B(\bar{x}) := \mathbb{R}$, $m_1 = m_2 := 0$, and $f_0(x) := \frac{1}{2}x^2$ satisfy Definition 2.1. Note that although $\mathcal{E}_0$ is eventually equivalent to the C^2-function f_0, its original formulation (with $\mathcal{T}$ unbounded) is neither that of a finite max-function nor of a max-eigenvalue function. Moreover, this example can be modified as follows to produce a similar C^2-function whose replacement for $\mathcal{T}$ in (2.1) depends on x as well as on a scalar parameter b:

$$f(x) := \max\left\{tx - \frac{1}{2}t^2 : t \le x + b\right\}.$$

Since, for each x, the maximizing t is $\min\{x, x+b\}$, we have $f(x) = \frac{1}{2}(x^2 - (\min(b,0))^2)$. □

In [11] we introduced two bivariate examples, $\mathcal{E}_1$ and $\mathcal{E}_2$, each of which is not the maximum of a finite number of C^2-functions and showed they both have pdg structure. For completeness, we give their definitions and main properties below.

Examples 3.1–3.2. Corresponding to the parameter $p = 1, 2$ are the following two functions of form (2.1):

$$\begin{array}{rccl} \mathcal{E}_p: & \mathbb{R}^2 & \to & \mathbb{R} \\ & (x_1,x_2) & \mapsto & \max\{t_1^2x_1^2 + 2(t_1t_2)^p x_2 : t_1^2 + t_2^2 = 1\}. \end{array}$$

When $p = 1$, the resulting function is the maximum eigenvalue of the matrix $\left[\begin{smallmatrix} x_1^2 & x_2 \\ x_2 & 0 \end{smallmatrix}\right]$, so

$$\mathcal{E}_1(x_1,x_2) = \frac{1}{2}x_1^2 + \frac{1}{2}\sqrt{x_1^4 + 4x_2^2}.$$

When $p = 2$, $\mathcal{E}_p$ is not a maximum eigenvalue function but instead has the form

$$\mathcal{E}_2(x_1,x_2) = \begin{cases} x_1^2 & \text{if } x_2 \le \dfrac{1}{2}x_1^2, \\ \dfrac{1}{2}x_2\left(1 + \dfrac{x_1^2}{2x_2}\right)^2 & \text{otherwise.} \end{cases} \tag{3.1}$$

The pdg structure of $\mathcal{E}_p$ about $\bar{x} = (0,0)^T$ has $B(\bar{x}) = \mathbb{R}^2$, $m_1 = m_2 = 1$,

$$f_0(x_1,x_2) = x_1^2, \quad f_1(x_1,x_2) = 0, \quad \varphi_1(x_1,x_2) = \frac{1}{p}x_2, \tag{3.2}$$

and $\Delta = \{(\alpha_0,\alpha_1,\alpha_2) : \alpha_0 = 1 - \alpha_1, 4(\alpha_1 - \frac{1}{2})^2 + \alpha_2^2 \le 1, (p-1)\alpha_2 \ge 0\}$. The pairs (α_1,α_2) corresponding to $(\alpha_0,\alpha_1,\alpha_2) \in \Delta$ are graphed in Figure 3.1. For the case $p = 1$ there is no nonnegativity restriction on α_2 and

$$(\alpha_0,\alpha_1,\alpha_2) \in \Delta \iff \text{the matrix} \begin{bmatrix} \alpha_0 & \frac{1}{2}\alpha_2 \\ \frac{1}{2}\alpha_2 & \alpha_1 \end{bmatrix} \begin{array}{l} \text{is positive semidefinite} \\ \text{with trace equal to 1.} \end{array} \tag{3.3}$$

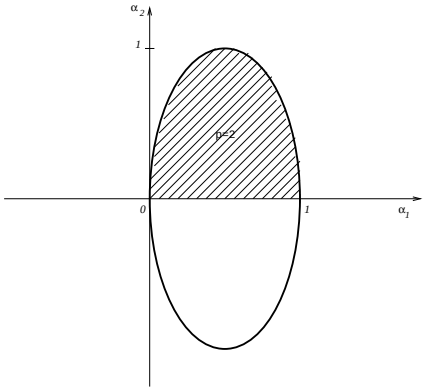


FIG. 3.1. *Multiplier sets for* $p = 1, 2$.

This correspondence for maximum eigenvalue functions is developed in general in the next section.

To illustrate our results, in subsequent sections we will employ $\mathcal{E}_p$ at $\bar{x} = (0,0)^T$, where

$$\partial \mathcal{E}_p(0,0) = \left\{ (0,\gamma)^T : \begin{array}{ll} \gamma \in [-1,1] & \text{if } p = 1, \\ \gamma \in [\ \ 0, \tfrac{1}{2}] & \text{if } p = 2 \end{array} \right\},$$

and, hence, from (1.1), for both functions at $\bar{x} = (0,0)^T$,

$$\mathcal{V} = 0 \times \mathbb{R} \quad \text{and} \quad \mathcal{U} = \mathbb{R} \times 0.$$

The subdifferential of $\mathcal{E}_p$ at $(0,0)^T$ is degenerate in the sense that it has only two extreme points, whereas the corresponding multiplier set Δ has a continuum of extreme points. □

3.2. Maximum eigenvalue functions. In this section, we produce the primal functions and dual multiplier set required by Definition 2.1 to show that rather general convex maximum eigenvalue functions have `pdg` structure about each $\bar{x} \in \mathbb{R}^n$.

We start with some notation. Suppose $A(\cdot)$ is an $m \times m$ symmetric matrix function whose elements are C^2-functions defined on $\mathbb{R}^n$. Let $f(x)$ be defined by (2.1) with

$$F(x,t) := t^T A(x) t \text{ for } x \in \mathbb{R}^n \text{ and } t \in \mathcal{T} := \{t \in \mathbb{R}^m : t^T t = 1\}. \tag{3.4}$$

Then it is well known that $f(x)$ is the maximum eigenvalue of $A(x)$. In addition, we suppose that $A(\cdot)$ is such that $f(\cdot)$ is convex on $\mathbb{R}^n$. A sufficient, although not necessary, condition for this to hold is that the off-diagonal elements of $A(\cdot)$ are affine functions while the on-diagonal elements are convex functions on $\mathbb{R}^n$. In [14] and [15], it is assumed that all element functions are affine.

We equip the space $\mathcal{S}$ of $s \times s$ symmetric matrices with the Fröbenius inner product, $\langle B, C \rangle := \operatorname{trace} BC = \sum_{j,i} b_{ij} c_{ij}$. Suppose the maximum eigenvalue of $A(x)$ has multiplicity s and that the first eigenspace of $A(x)$, $\mathcal{E}^1(x)$ has the basis matrix $E^1(x)$, whose columns are the orthonormal eigenvectors $\{e_1(x), \ldots, e_s(x)\}$. Then from [16, Theorem 3],

$$\begin{aligned} g = (g_1, \ldots, g_n)^T \in \partial f(x) \qquad & \text{if and only if} \\ g_j = \left\langle S, E^1(x)^T \frac{\partial A(x)}{\partial x_j} E^1(x) \right\rangle \quad & \text{for } j = 1, \ldots, n, \end{aligned} \tag{3.5}$$

where $S \in \Delta^s$ and Δ^s is the convex set of $s \times s$ dual feasible matrices defined by

$$\Delta^s := \{S \in \mathcal{S} : S \text{ is positive semidefinite and } \operatorname{trace} S = 1\}. \tag{3.6}$$

Note that any other orthonormal eigenvector basis matrix $Q_1(x)$ has the form $Q_1(x) = E^1(x)Q$ for some $s \times s$ orthogonal matrix Q. Furthermore, if $S \in \Delta^s$, then $D := Q^T S Q \in \Delta^s$ and, since QQ^T is an identity matrix,

$$\langle S, E^1(x)^T B E^1(x)\rangle = \langle D, Q_1(x)^T B Q_1(x)\rangle$$

for any $B \in \mathcal{S}$. Hence the expression in (3.5) can be written in the equivalent form

$$g_j = \left\langle D, Q_1(x)^T \frac{\partial A(x)}{\partial x_j} Q_1(x) \right\rangle \text{ for } j = 1, \ldots, n, \tag{3.7}$$

where $D \in \Delta^s$. Thus we see that each subgradient of f at x is a particular linear combination of n-dimensional vectors

$$\left(q_k(x)^T \frac{\partial A(x)}{\partial x_1} q_l(x), \ldots, q_k(x)^T \frac{\partial A(x)}{\partial x_n} q_l(x) \right)^T$$

for $k, l = 1, \ldots, s$, where $\{q_i(x)\}_{i=1}^s$ is any set of orthonormal eigenvectors for $\mathcal{E}^1(x)$ and the multipliers in the combination form an $s \times s$ dual feasible matrix depending on the particular subgradient.

If one wants representation (3.7) to be diagonal for a particular subgradient g, then Q can be chosen such that $D = Q^T S Q$ is a diagonal matrix, where S from (3.5) corresponds to g.

Now we begin the development to show the existence of the primal functions f_i and φ_ℓ satisfying Definition 2.1. Consider $\bar{x} \in \mathbb{R}^n$ and suppose r is the multiplicity of the maximum eigenvalue of $A(\bar{x})$. Let

$$I_1 := \{1, 2, \ldots, r\} \quad \text{and} \quad I_2 := \{(k, l) \in I_1 \times I_1 : k < l\}, \tag{3.8}$$

so that $|I_1| = r$, $|I_2| = r(r-1)/2$, and $|I_1| + |I_2| = r(r+1)/2$. From the continuity of the eigenvalues of $A(\cdot)$ [22], there exists a ball about $\bar{x}$, $B(\bar{x})$, such that for each $x \in B(\bar{x})$ the multiplicity of the maximum eigenvalue of $A(x)$ is r or less. Furthermore, from [21, pp. 557–559], $B(\bar{x})$ may be defined such that there exist r twice continuously differentiable m-dimensional vector functions $q_i : B(\bar{x}) \to \mathbb{R}^m$ for $i \in I_1$, satisfying

$$\begin{aligned} q_i(\bar{x})^T A(\bar{x}) q_i(\bar{x}) &= f(\bar{x}) \quad \text{for } i \in I_1, \\ q_k(\bar{x})^T A(\bar{x}) q_l(\bar{x}) &= 0 \qquad \text{for } (k, l) \in I_2, \end{aligned}$$

and for all $x \in B(\bar{x})$,

$$q_k(x)^T q_l(x) = \delta_{kl} \quad \text{for } (k, l) \in I_1 \times I_1, \tag{3.9}$$

where $\delta_{ii} = 1$ and $\delta_{kl} = 0$ for $k \neq l$. These functions are denoted by u_i in [21]. They are related to the columns of Q_{tot} in [14, Theorem 4.5].

Thus at $\bar{x}$, the vectors $q_i(\bar{x})$ form an orthonormal basis of eigenvectors for $\mathcal{E}^1(\bar{x})$. For $x \in B(\bar{x})$ such that $x \neq \bar{x}$, they are not necessarily eigenvectors of $A(x)$, but by construction in [21], they do form an orthonormal basis for the subspace spanned by the eigenvectors corresponding to the r largest eigenvalues of $A(x)$. Actually, there

is an infinite collection of such vector functions, depending on the choice of Q in $Q_1(\bar{x}) = E^1(\bar{x})Q$.

After a suitable set $\mathcal{P} \subseteq B(\bar{x})$ is defined in (3.11) below, the functions introduced next in (3.10) will turn out to produce the f_is and φ_ℓs satisfying Definition 2.1.

For $x \in B(\bar{x})$ and (k,l) in $I_1 \times I_1$, define the C^2-functions

$$\phi_{kl}(x) := q_k(x)^T A(x) q_l(x). \tag{3.10}$$

These functions from [21, p. 559] are related to the structure functions introduced in [12], but they additionally provide relevant second-order information.

Note that, by symmetry of $A(\cdot)$, $\phi_{kl} = \phi_{lk}$. Let

$$\mathcal{P} := \{x \in B(\bar{x}) : \phi_{kl}(x) = 0 \text{ for all } (k,l) \in I_2\}. \tag{3.11}$$

We have that $\bar{x} \in \mathcal{P}$. For any $x \in \mathcal{P}$ and $(k,l) \in I_1 \times I_1$, the definition of $\mathcal{P}$ together with (3.10) yields the equality

$$\phi_{kl}(x)\delta_{kl} = q_k(x)^T A(x) q_l(x). \tag{3.12}$$

This means that for $x \in \mathcal{P}$ the orthonormal vectors $\{q_i(x)\}_{i \in I_1}$ diagonalize $A(x)$ to give the r largest eigenvalues. Thus for such an x, $\phi_{ii}(x)$ is one of the r largest eigenvalues of $A(x)$ and $q_i(x)$ is a corresponding eigenvector. So for any $x \in \mathcal{P}$ and $i \in I_1$,

$$A(x)q_i(x) = \phi_{ii}(x)q_i(x) \quad \text{and} \quad f(x) = \max\{\phi_{ii}(x) : i \in I_1\}. \tag{3.13}$$

Let $\ell = \ell(k,l)$ be an index function enumerating elements of I_2. Setting

$$\begin{array}{rlrl} m_1 &:= r-1, & f_{i-1} &:= \phi_{ii} \quad \text{for } i \in I_1, \\ m_2 &:= r(r-1)/2, & \varphi_\ell &:= \phi_{kl} \quad \text{for } (k,l) \in I_2, \end{array} \tag{3.14}$$

we have that our `mef` f satisfies (i)–(ii) in Definition 2.1.

To satisfy (iii) in Definition 2.1, we define the vector set Δ to correspond to the convex matrix set Δ^{m_1+1} defined in (3.6) with $s = m_1 + 1$ as follows: For each matrix $S \in \Delta^{m_1+1}$ having $m_1 + 1$ diagonal elements s_{ii} and $m_2 = m_1(m_1+1)/2$ above-the-diagonal elements s_{kl}, let $\alpha = (\alpha_0, \ldots, \alpha_{m_1}, \alpha_{m_1+1}, \ldots, \alpha_{m_1+m_2})$ be a vector in Δ defined by

$$\alpha_{i-1} = s_{ii} \quad \text{for } i = 1, \ldots, m_1+1 \quad \text{and} \quad \alpha_{\ell+m_1} = 2s_{kl} \quad \text{for } \ell = 1, \ldots, m_2,$$

where $\ell = \ell(k,l)$ is an index function enumerating elements of $\{(k,l) : 1 \le k < l \le m_1\}$.

To see that Δ satisfies the requirements of Definition 2.1(iii), first note that (iii)(a) is satisfied because positive semidefinite matrices with unit trace have nonnegative diagonal elements that sum to one. To see satisfaction of (iii)(b) for each $i+1 = j = 1, 2, \ldots, m_1+1$ and of (iii)(c) for each $\ell = 1, 2, \ldots, m_2$, consider the two unit trace positive semidefinite matrices S^b and S^c, all of whose elements are zero except that

$$s^b_{jj} = 1 \quad \text{and} \quad s^c_{kk} = s^c_{ll} = s^c_{kl} = s^c_{lk} = 1/2,$$

where index ℓ corresponds to the off-diagonal index pair (k,l). Altogether, (iii) holds.

To see (iv), we need to express $\partial f(x)$ in terms of partial derivatives of the functions ϕ_{kl}. This requires the following key result.

LEMMA 3.3. *Let ϕ_{kl} and $\mathcal{P}$ be as defined in* (3.10) *and* (3.11), *respectively. If $\phi_{kk}(x) = \phi_{ll}(x)$ for some $x \in \mathcal{P}$ and some $(k,l) \in I_1 \times I_1$, then*

$$\frac{\partial \phi_{kl}(x)}{\partial x_j} = q_k(x)^T \frac{\partial A(x)}{\partial x_j} q_l(x) \quad \text{for each } j = 1, \ldots, n.$$

Proof. For $j = 1, \ldots, n$, differentiate ϕ_{kl} with respect to x_j and then use (3.13) with $i = l$ and with $i = k$ to obtain

$$\begin{aligned}\frac{\partial \phi_{kl}(x)}{\partial x_j} - q_k(x)^T \frac{\partial A(x)}{\partial x_j} q_l(x) &= \frac{\partial q_k(x)}{\partial x_j}^T A(x) q_l(x) + q_k(x)^T A(x) \frac{\partial q_l(x)}{\partial x_j} \\ &= \phi_{ll}(x) \frac{\partial q_k(x)}{\partial x_j}^T q_l(x) + \phi_{kk}(x) q_k(x)^T \frac{\partial q_l(x)}{\partial x_j}.\end{aligned}$$

Differentiating (3.9), having the constant right side δ_{kl}, gives

$$\frac{\partial q_k(x)}{\partial x_j}^T q_l(x) + q_k(x)^T \frac{\partial q_l(x)}{\partial x_j} = 0$$

and hence the desired result when $\phi_{kk}(x) = \phi_{ll}(x)$. □

Our lemma is similar to results obtained in [21] and also similar to [14, Corollary 4.6]. However, in these papers the authors consider a subset of $\mathcal{P}$ that can be proper and possibly the singleton $\{\bar{x}\}$ when their strong transversality assumptions are not satisfied.

In a way similar to (3.8), but for any $x \in \mathcal{P}$, define the sets

$$I_1(x) := \{i \in I_1 : f_{i-1}(x) = f(x)\} \quad \text{and} \quad I_2(x) := \{(k,l) \in I_1(x) \times I_1(x) : k < l\}. \tag{3.15}$$

Let $s = s(x) := |I_1(x)|$ and $Q_1(x)$ be an $m \times s$ matrix whose columns $\{q_i(x)\}_{\{i \in I_1(x)\}}$ form an orthonormal set of eigenvectors spanning the eigenspace $\mathcal{E}^1(x)$ corresponding to the maximum eigenvalue of $A(x)$ with multiplicity $s \le r = m_1 + 1$.

Coming back to (3.5), (3.6), and (3.7), we see from Lemma 3.3 that each subgradient of f at $x \in \mathcal{P}$ is a particular linear combination of n-dimensional vectors:

$$\nabla \phi_{kl}(x) = \left(\frac{\partial \phi_{kl}(x)}{\partial x_1}, \ldots, \frac{\partial \phi_{kl}(x)}{\partial x_n} \right)^T$$

for $(k,l) \in I_1(x) \times I_1(x)$, where the multipliers in the combination form an $s \times s$ matrix S in Δ^s. If $s < r = m_1 + 1$, then by appropriately appending $r - s = m_1 + 1 - s$ rows and columns of zeros to S, we can form an $(m_1+1) \times (m_1+1)$ matrix in Δ^{m_1+1} such that each $g \in \partial f(x)$ with $x \in \mathcal{P}$ is a dual feasible linear combination of all the $\nabla \phi_{kl}(x)$ for $(k,l) \in I_1 \times I_1$, with a zero multiplier on $\nabla \phi_{kl}(x)$ if $(k,l) \notin I_1(x) \times I_1(x)$.

Hence, in view of (3.14), (3.15), and the definition of Δ, we see that (iv) of Definition 2.1 holds if we let

$$\alpha_{i-1} := \begin{cases} s_{ii} & \text{for } i \in I_1(x), \\ 0 & \text{for } i \in I_1 \backslash I_1(x) \end{cases} \quad \text{and} \quad \alpha_{\ell+m_1} := \begin{cases} 2s_{kl} & \text{for } (k,l) \in I_2(x), \\ 0 & \text{for } (k,l) \in I_2 \backslash I_2(x), \end{cases}$$

where $\ell = \ell(k,l)$ is an index function enumerating elements of I_2.

4. $\mathcal{VU}$-space decomposition. In order to build up to the definitions of smooth trajectories and multipliers which allow us to express the $\mathcal{U}$-Hessian of a function with `pdg` structure, we require basis matrices for the relevant subspaces $\mathcal{U}$ and $\mathcal{V}$.

4.1. A spanning set for $\mathcal{V}$. Given $\bar{x} \in \mathbb{R}^n$, not necessarily a minimizer of a (closed) convex function f, and an arbitrary subgradient $g \in \partial f(\bar{x})$, the orthogonal subspaces from (1.1),

$$\mathcal{V} = \operatorname{lin}(\partial f(\bar{x}) - g) \quad \text{and} \quad \mathcal{U} = \mathcal{V}^{\perp},$$

define the $\mathcal{VU}$-space decomposition of [7, section 2]: $\mathbb{R}^n = \mathcal{U} \oplus \mathcal{V}$. Note that $\mathcal{V}$, depending on $\bar{x}$, is independent of the particular choice of $g \in \partial f(\bar{x})$.

Throughout the remainder of this paper, we assume that f has `pdg` structure about $\bar{x}$. In this case, it is possible to completely characterize $\mathcal{V}$ in terms of the gradients of the primal functions in Definition 2.1. To see this, define the subspace of $\mathbb{R}^n$

$$\mathcal{V}_0 := \operatorname{lin}\left(\{\nabla f_i(\bar{x}) - \nabla f_0(\bar{x})\}_{i=0}^{m_1} \cup \{\nabla \varphi_\ell(\bar{x})\}_{\ell=1}^{m_2}\right). \tag{4.1}$$

Note that by appending an appropriate multiple of $0 = \nabla f_0(\bar{x}) - \nabla f_0(\bar{x})$,

$$\mathcal{V}_0 = \left\{ \sum_{i=0}^{m_1} \beta_i \nabla f_i(\bar{x}) + \sum_{i=m_1+1}^{m_1+m_2} \beta_i \nabla \varphi_{i-m_1}(\bar{x}) : \begin{array}{l} \beta_i \in \mathbb{R} \text{ for } i = 0, \ldots, m_1 + m_2, \\ \displaystyle\sum_{i=0}^{m_1} \beta_i = 0 \end{array} \right\}. \tag{4.2}$$

If $m_1 = m_2 = 0$, then both $\mathcal{V}$ and $\mathcal{V}_0$ are $\{0\}$.

LEMMA 4.1. *Let $\mathcal{V}$ and $\mathcal{V}_0$ be as defined, respectively, in* (1.1) *and* (4.1). *Then $\mathcal{V} = \mathcal{V}_0$.*

Proof. First, we show $\mathcal{V}_0 \subseteq \mathcal{V}$. For an arbitrary $i \in \{0, 1, \ldots, m_1\}$, by Definition 2.1(iii)(b), the unit vector $\mathbf{1}_{i+1} \in \Delta$. Then (iv) of Definition 2.1 implies that

$$\nabla f_i(\bar{x}) \in \partial f(\bar{x}) \quad \text{for each } i = 0, 1, \ldots, m_1. \tag{4.3}$$

Since $\mathcal{V}$ is independent of the choice of $g \in \partial f(\bar{x})$, by taking $g = \nabla f_0(\bar{x})$ in (1.1), we have

$$\nabla f_i(\bar{x}) - \nabla f_0(\bar{x}) \in \mathcal{V} \quad \text{for each } i = 0, 1, \ldots, m_1. \tag{4.4}$$

Now for an arbitrary $\ell \in \{1, 2, \ldots, m_2\}$, take α^ℓ as in Definition 2.1(iii)(c): $\alpha^\ell := (\alpha_0^\ell, \ldots, \alpha_{m_1}^\ell, 0, \ldots, \alpha_{m_1+\ell}^\ell, \ldots, 0) \in \Delta$ with $(\alpha_0^\ell, \ldots, \alpha_{m_1}^\ell) \in \Delta_1$ and $\alpha_{m_1+\ell}^\ell \neq 0$. Then (iv) of Definition 2.1 implies that

$$g^\ell := \sum_{i=0}^{m_1} \alpha_i^\ell \nabla f_i(\bar{x}) + \alpha_{m_1+\ell}^\ell \nabla \varphi_\ell(\bar{x}) \in \partial f(\bar{x}).$$

Note that the convexity of $\partial f(\bar{x})$ together with (4.3) implies that $g = \sum_{i=0}^{m_1} \alpha_i^\ell \nabla f_i(\bar{x}) \in \partial f(\bar{x})$. Thus from (1.1), we have $\alpha_{m_1+\ell}^\ell \nabla \varphi_\ell(\bar{x}) = g^\ell - g \in \mathcal{V}$. Therefore, since $\alpha_{m_1+\ell}^\ell \neq 0$ and $\mathcal{V}$ is a subspace,

$$\nabla \varphi_\ell(\bar{x}) \in \mathcal{V} \quad \text{for } \ell = 1, \ldots, m_2. \tag{4.5}$$

Finally, by combining (4.1), (4.4), and (4.5) with the linearity of $\mathcal{V}$ the desired inclusion follows: $\mathcal{V}_0 \subseteq \mathcal{V}$.

To show that $\mathcal{V} \subseteq \mathcal{V}_0$, first take $g = \nabla f_0(\bar{x})$ in the definition of $\mathcal{V}$ and then use the fact that $\sum_{i=0}^{m_1} \alpha_i = 1$ when considering arbitrary elements of $\partial f(x)$ given by (2.4). ∎

4.2. Basic index sets. Now via Lemma 4.1 and the definition of $\mathcal{V}_0$, we have generators for $\mathcal{V}$ when f has `pdg` structure about $\bar{x}$. These are crucial for finding (smooth) trajectories tangent to $\mathcal{U}$ starting from the following definition.

DEFINITION 4.2. *An index set K of the form $K = K_f \cup K_\varphi \subseteq \{0, 1, \ldots, m_1\} \cup \{m_1 + 1, \ldots, m_1 + m_2\}$ with $0 \in K_f$ is called a* basic index set *if*

(i) *the $(n+1)$-dimensional vectors*

$$\left\{ \begin{bmatrix} \nabla f_i(\bar{x}) \\ 1 \end{bmatrix} \right\}_{i \in K_f} \bigcup \left\{ \begin{bmatrix} \nabla \varphi_{i-m_1}(\bar{x}) \\ 0 \end{bmatrix} \right\}_{i \in K_\varphi}$$

are linearly independent.

K is called a dual feasible basic index set *relative to $\bar{g} \in \partial f(\bar{x})$ if, in addition,*

(ii) *the linear system with variables α_i*

$$\begin{aligned} \sum_{i \in K_f} \alpha_i \nabla f_i(\bar{x}) + \sum_{i \in K_\varphi} \alpha_i \nabla \varphi_{i-m_1}(\bar{x}) &= \bar{g}, \\ \sum_{i \in K_f} \alpha_i &= 1 \end{aligned}$$

has a (unique) solution $\{\alpha_i = \bar{\alpha}_i\}_{i \in K}$ such that together with $\bar{\alpha}_i := 0$ for all $i \notin K$,

$$\bar{\alpha} := (\bar{\alpha}_0, \bar{\alpha}_1, \ldots, \bar{\alpha}_{m_1+m_2}) \in \Delta. \qquad \square$$

The definition assumes that, if necessary, the f_is are reindexed so that the nonempty set K_f contains $i = 0$.

The existence of such index sets can be proved for two particular structures.

LEMMA 4.3. *Suppose f in* (2.1) *is a finite max-function as in section* 2.1, *or a maximum eigenvalue function, as in section* 3.2. *Then for each $\bar{g} \in \partial f(\bar{x})$ there exists at least one dual feasible basic index set K.*

Proof. For f a finite max-function, this is [10, Lemma 3]. For f a `mef`, the result parallels the finite-max proof. By choosing D to be diagonal in representation (3.7), written with $x = \bar{x}$ and $g = \bar{g}$, the functions f_{i-1} for $i = 1, \ldots, m_1 + 1$ from (3.14) are defined such that $\bar{g}$ is an element of the convex hull $\operatorname{conv}\{\nabla f_i(\bar{x})\}_{i=0}^{m_1}$. It follows from a corollary of Carathéodory's theorem numbered 17.1.1 in [20] that $\bar{g}$ can be expressed as a convex combination of affinely independent generators, $\{\nabla f_i(\bar{x})\}_{i \in K_f}$, where K_f is some subset of $\{0, \ldots, m_1\}$. $\square$

As for general primal-dual structured functions we are content to simply assume the existence of dual feasible basic index sets rather than introducing additional structural assumptions that are satisfied by maximum eigenvalue functions and functions such as example $\mathcal{E}_2$. An alternative way to obtain such an existence is to assume strong transversality as defined in section 6 below.

In what follows, we assume that $K = K_f \cup K_\varphi$ is a basic index set, so $0 \in K_f$. Thus if K is a singleton, then K_f is the singleton $\{0\}$ and K_φ is empty. We adopt the following convention: Given a set of column vectors $\{v_1, \ldots, v_\ell\}$, we denote the corresponding matrix by $[\{v_1, \ldots, v_\ell\}]$.

LEMMA 4.4. *If $K = K_f \cup K_\varphi$ is a basic index set, then the $n \times (|K_f| - 1 + |K_\varphi|)$ matrix*

$$\bar{V} := [\{\nabla f_i(\bar{x}) - \nabla f_0(\bar{x})\}_{0 \neq i \in K_f} \cup \{\nabla \varphi_{i-m_1}(\bar{x})\}_{i \in K_\varphi}]$$

has full column rank. Moreover, the subspace of $\mathcal{V}$ *defined by*

$$\mathcal{V}_K := \operatorname{lin}\left(\{\nabla f_i(\bar{x}) - \nabla f_0(\bar{x})\}_{i\in K_f} \cup \{\nabla \varphi_{i-m_1}(\bar{x})\}_{i\in K_\varphi}\right)$$

has dimension $|K_f| - 1 + |K_\varphi|$ *and* $\bar{V}$ *is the corresponding basis matrix.*

Proof. The results follow in a straightforward manner from the linear independence condition in Definition 4.2(i). □

Note that if K is a singleton, the matrix $\bar{V}$ is vacuous and $\mathcal{V}_K$ is only the zero vector.

From here on, we assume that $\mathcal{U} \neq \{0\}$ so that $\dim\mathcal{U} \geq 1$ and $\mathcal{V} \neq \mathbb{R}^n$.

MATRIX NOTATION 4.5. *Consider the* $\mathcal{VU}$ *decomposition defined in* (1.1) *and let* $\bar{U}$ *be a basis matrix for* $\mathcal{U}$*. Corresponding to a basic index set* K *and its associated subspace* $\mathcal{V}_K \subseteq \mathcal{V}$ *from Lemma* 4.4*, let the augmented matrix* $[\bar{V}|\bar{Z}]$ *be a basis matrix for* $\mathcal{V}$*, where* $\bar{Z}$ *is vacuous if* $\mathcal{V}_K = \mathcal{V}$ *or is a basis matrix for* $\mathcal{V}\backslash\mathcal{V}_K$ *otherwise. Then* $[\bar{U}|\bar{V}|\bar{Z}]$ *is a basis matrix for* $\mathbb{R}^n$*, where* $\bar{V}$ *and* $\bar{Z}$ *depend on the particular basic index set* K *under consideration.* □

Recall from section 3.1 that for both $\mathcal{E}_1$ and $\mathcal{E}_2$ at $\bar{x} = (0,0)^T$,

$$\mathcal{V} = \operatorname{lin}\left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}\right) \quad \text{and} \quad \mathcal{U} = \operatorname{lin}\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}\right),$$

so we take $\bar{U} = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$. For both these functions, the following two dual feasible basic index sets relative to $\bar{g} = (0,0)^T \in \partial\mathcal{E}_p(0,0)$ are of interest:

$$\begin{array}{llll} K_0 := \{0\} & \text{with } K_\varphi = \phi, & \mathcal{V}_{K_0} = \{(0,0)^T\} & \text{and} \quad \bar{V} \text{ vacuous;} \\ K_{0,2} := \{0,2\} & \text{with } K_\varphi = \{2\}, & \mathcal{V}_{K_{0,2}} = \mathcal{V} & \text{and} \quad \bar{V} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \end{array}$$

5. Smooth trajectory and multiplier functions. In order to start our development to characterize smooth trajectory and multiplier functions, we first recall some additional concepts from [7] that will be used in what follows.

5.1. Definition of the $\mathcal{U}$-Lagrangian. Since $\mathbb{R}^n = \mathcal{U} \oplus \mathcal{V}$, every $x \in \mathbb{R}^n$ can be decomposed into components $x_\mathcal{U}$ and $x_\mathcal{V}$, where $x_\mathcal{U}(x_\mathcal{V})$ stands for the projection of x onto $\mathcal{U}(\mathcal{V})$. Sometimes it will be convenient to use the shorter notation $x_\mathcal{U} \oplus x_\mathcal{V}$ for the vector with components $x_\mathcal{U}$ and $x_\mathcal{V}$, where $\oplus$ is defined by

$$\mathcal{U} \times \mathcal{V} \ni (u,v) \mapsto u \oplus v := \begin{pmatrix} u \\ v \end{pmatrix} \in \mathbb{R}^n.$$

Now we can consider $\mathcal{U}$ and $\mathcal{V}$ as vector spaces, with induced scalar products satisfying

$$\langle g_\mathcal{U} \oplus g_\mathcal{V}, x_\mathcal{U} \oplus x_\mathcal{V}\rangle =: \langle g_\mathcal{U}, x_\mathcal{U}\rangle_\mathcal{U} + \langle g_\mathcal{V}, x_\mathcal{V}\rangle_\mathcal{V}.$$

We already mentioned that the apparent smoothness of f on the $\mathcal{U}$-subspace suggests that it may have some kind of Hessian. Actually, it is the *$\mathcal{U}$-Lagrangian* that contains the relevant second-order information about f near $\bar{x}$. More precisely, for a (closed) convex function f, $\bar{x} \in \mathbb{R}^n$, and $\bar{g} \in \partial f(\bar{x})$, the $\mathcal{U}$-Lagrangian of f from [7] is defined as follows:

$$\mathcal{U} \ni u \mapsto L_\mathcal{U}(u) := \inf_{v\in\mathcal{V}}\{f(\bar{x} + u \oplus v) - \langle \bar{g}_\mathcal{V}, v\rangle_\mathcal{V}\}.$$

To emphasize the dependence of $L_{\mathcal{U}}(\cdot)$ on $\bar{x} \in \mathbb{R}^n$ and the $\mathcal{V}$-component of the subgradient $\bar{g}$, we will follow [14] somewhat and adopt the complete notation $L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}})$. (Note, however, that unlike [14], we do not require $\bar{g}$ to be in the relative interior of $\partial f(\bar{x})$.)

The $\mathcal{U}$-Lagrangian is a convex function that is differentiable at $u = 0$ with $\nabla L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}}) = \bar{g}_{\mathcal{U}}$. Since from (1.1) the $\mathcal{U}$-component of any subgradient $g \in \partial f(\bar{x})$ is $\bar{g}_{\mathcal{U}}$, we call $\nabla L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}})$ the *$\mathcal{U}$-gradient* of f at $\bar{x}$. When $L_{\mathcal{U}}$ has a Hessian at 0, we say that $\nabla^2 L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}})$ is the *$\mathcal{U}$-Hessian* of f at $\bar{x}$ relative to $\bar{g}_{\mathcal{V}}$ and we denote it simply by $\mathrm{H}_{\mathcal{U}} f(\bar{x})$, so its dependence on $\bar{g}_{\mathcal{V}}$ should be kept in mind.

Smoothness of $L_{\mathcal{U}}$ depends on f through the Winning set of minimizers

$$W(u) := \mathrm{Argmin}_{v \in \mathcal{V}} \{ f(\bar{x} + u \oplus v) - \langle \bar{g}_{\mathcal{V}}, v \rangle_{\mathcal{V}} \}. \tag{5.1}$$

We could also employ the complete notation $W_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}})$, but for brevity we will often use $W(u)$. Also, sometimes we may use the abbreviated notation $\nabla L_{\mathcal{U}}(0)$ and $\nabla^2 L_{\mathcal{U}}(0)$.

Indeed, whenever $\nabla^2 L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}})$ exists, those $x(u) := \bar{x} + u \oplus w$ with $w \in W(u)$ yield the following second-order expansion of f:

$$\begin{aligned} f(x(u)) &= f(\bar{x}) + \langle \nabla L_{\mathcal{U}}(0), u \rangle_{\mathcal{U}} + \langle \bar{g}_{\mathcal{V}}, w \rangle_{\mathcal{V}} + \frac{1}{2} \left\langle \nabla^2 L_{\mathcal{U}}(0) u, u \right\rangle_{\mathcal{U}} + o(\|u\|_{\mathcal{U}}^2) \\ &= f(\bar{x}) + \langle \bar{g}, u \oplus w \rangle + \frac{1}{2} \langle \mathrm{H}_{\mathcal{U}} f(\bar{x}) u, u \rangle_{\mathcal{U}} + o(\|u \oplus w\|^2). \end{aligned} \tag{5.2}$$

Accordingly, we call such $\bar{x} + u \oplus w \in \bar{x} + u \oplus W(u)$ *smooth trajectories*. In what follows, we show how to determine smooth trajectories and a $\mathcal{U}$-Hessian for a pdg-structured function satisfying certain $\mathcal{V}$-optimality conditions. These objects depend on the primal functions f_i and φ_l and on the dual multiplier set Δ introduced in Definition 2.1.

With respect to Matrix Notation 4.5 corresponding to a basic index set K we now change notation slightly and replace $(u, v) \in \mathcal{U} \times \mathcal{V}$ by a column vector $\binom{u}{v} \in \mathbb{R}^{\dim \mathcal{U}} \times \mathbb{R}^{\dim \mathcal{V}}$ and $u \oplus v$ by $\bar{U}u + \bar{V}v_K + \bar{Z}z$, where the column vector $v = \binom{v_K}{z} \in \mathbb{R}^{\dim \mathcal{V}_K} \times \mathbb{R}^{\dim \mathcal{V} - \dim \mathcal{V}_K}$. For example, with this notation,

$$L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = \min_{(v_K, z)^T \in \mathbb{R}^{\dim \mathcal{V}}} \{ f(\bar{x} + \bar{U}u + \bar{V}v_K + \bar{Z}z) - \bar{g}^T (\bar{V}v_K + \bar{Z}z) \} \tag{5.3}$$

for $u \in \mathbb{R}^{\dim \mathcal{U}}$. So vectors in $W(u; \bar{g}_{\mathcal{V}}) \subseteq \mathcal{V}$ are of the form $\bar{V}v_K + \bar{Z}z$ with $(v_K, z)^T$ a minimizer in (5.3). The minimizer of interest in section 6 has $\bar{Z}z = 0$.

5.2. Smooth trajectories. To give formulas for the $\mathcal{U}$-Lagrangian (5.3) and its derivatives, and to obtain a second-order expansion for f as in (5.2), we need to identify smooth trajectories that are in $\bar{x} + \bar{U}u + W(u)$. We achieve this in two steps. First, based on f's primal functions corresponding to a basic index set K from Definition 4.2, we define a system of nonlinear equations whose solution generates a candidate for an element of $W(u)$ which is $o(\|u\|)$. Then in section 6, we give conditions for a basic index set to produce a successful candidate.

Accordingly, for u near $0 \in \mathbb{R}^{\dim \mathcal{U}}$, we associate a trajectory $x(u) \in B(\bar{x})$ with a basic index set K as in [10] by letting

$$x(u) := \bar{x} + \bar{U}u + \bar{V}v_K(u), \tag{5.4}$$

where the function $v_K : \mathbb{R}^{\dim \mathcal{U}} \mapsto \mathbb{R}^{\dim \mathcal{V}_K}$ is defined in our next theorem. For a differentiable vector function $c : \mathbb{R}^n \to \mathbb{R}^m$, Jc will denote the $m \times n$ Jacobian matrix, whose rows are the transposed gradients of the components of c.

THEOREM 5.1. *Let f have* pdg *structure about $\bar{x}$ and suppose $K = K_f \cup K_\varphi$ is a nonsingleton basic index set as described in Definition* 4.2(i). *For each u small enough,*

(i) *the nonlinear system with variables (u, v)*

$$\text{(5.5)} \qquad \begin{cases} f_i(\bar{x} + \bar{U}u + \bar{V}v) - f_0(\bar{x} + \bar{U}u + \bar{V}v) = 0, & 0 \neq i \in K_f, \\ \varphi_{i-m_1}(\bar{x} + \bar{U}u + \bar{V}v) = 0, & i \in K_\varphi, \end{cases}$$

has a unique solution $v = v_K(u)$ with $\bar{V}v_K(u)$ in the subspace $\mathcal{V}_K \subseteq \mathcal{V}$;

(ii) *$v_K(\cdot)$ has a continuous Jacobian*

$$Jv_K(u) = -(V(u)^T\bar{V})^{-1}V(u)^T\bar{U},$$

where

$$V(u) := [\{\nabla f_i(x(u)) - \nabla f_0(x(u))\}_{0 \neq i \in K_f} \bigcup \{\nabla \varphi_{i-m_1}(x(u))\}_{i \in K_\varphi}];$$

(iii) *the trajectory $x(\cdot)$ has a continuous Jacobian $Jx(u) = \bar{U} + \bar{V}Jv_K(u)$;*

(iv) *in particular, $v_K(0) = 0$, $x(0) = \bar{x}$, $V(0) = \bar{V}$, $Jv_K(0) = 0$, and $Jx(0) = \bar{U}$.*

Proof. Differentiating (5.5) with respect to v gives the rows of the Jacobian with respect to v of the left-hand side as

$$\begin{aligned} & \nabla f_i(\bar{x} + \bar{U}u + \bar{V}v)^T\bar{V} - \nabla f_0(\bar{x} + \bar{U}u + \bar{V}v)^T\bar{V} && \text{for all } 0 \neq i \in K_f, \\ \text{and} \quad & \nabla \varphi_{i-m_1}(\bar{x} + \bar{U}u + \bar{V}v)^T\bar{V} && \text{for all } i \in K_\varphi. \end{aligned}$$

This Jacobian at $(u, v) = (0, 0)$ is $\bar{V}^T\bar{V}$, which is nonsingular. There is also a Jacobian with respect to u, so by the implicit function theorem, there is a C^1 function $v_K(u)$ defined on a neighborhood of $u = 0$ such that $v_K(0) = 0$,

$$\begin{aligned} & f_i(\bar{x} + \bar{U}u + \bar{V}v_K(u)) - f_0(\bar{x} + \bar{U}u + \bar{V}v_K(u)) = 0 && \text{for all } 0 \neq i \in K_f, \\ \text{and} \quad & \varphi_{i-m_1}(\bar{x} + \bar{U}u + \bar{V}v_K(u)) = 0 && \text{for all } i \in K_\varphi. \end{aligned}$$

Since v_K is C^1, the Jacobians $Jv_K(u)$ and $Jx(u)$ exist and are continuous. Differentiating the system above with respect to u and using (5.4) gives

$$\begin{aligned} & (\nabla f_i(x(u)) - \nabla f_0(x(u)))^T Jx(u) = 0 \in \mathbb{R}^{1 \times \dim \mathcal{U}} && \text{for all } 0 \neq i \in K_f, \\ \text{and} \quad & \nabla \varphi_{i-m_1}(x(u))^T Jx(u) = 0 \in \mathbb{R}^{1 \times \dim \mathcal{U}} && \text{for all } i \in K_\varphi \end{aligned}$$

or, in matrix notation, $V(u)^T Jx(u) = 0 \in \mathbb{R}^{(|K|-1) \times \dim \mathcal{U}}$. Using the expression $Jx(u) = \bar{U} + \bar{V}Jv_K(u)$, we have that $V(u)^T(\bar{U} + \bar{V}Jv_K(u)) = 0$. By continuity, $V(u)^T\bar{V}$ is nonsingular; hence $Jv_K(u) = -(V(u)^T\bar{V})^{-1}V(u)^T\bar{U}$. Since $\mathcal{V}_K \perp \mathcal{U}$, $\bar{V}^T\bar{U} = 0$, so $Jv_K(0) = 0$ and $Jx(0) = \bar{U}$. □

Note that since $Jv_K(0) = 0$, $v_K(u) = o(\|u\|)$ and the trajectory $x(u)$ is tangent to $\mathcal{U}$ at $x(0) = \bar{x}$.

The results above need only the primal functions f_i and φ_ℓ to be C^1. However, for some of the subsequent results, we will need f_i and φ_ℓ to be C^2.

In addition, note that when K is a singleton, then $\bar{V}$ is vacuous. In this case, the trajectory $x(u)$ and its Jacobian are simply given by $x(u) = \bar{x} + \bar{U}u$ and $Jx(u) \equiv \bar{U}$, so in view of (5.4), it is useful to define $\bar{V}v_K(u) := 0$. This is precisely the case for $\mathcal{E}_p$ at $\bar{x} = (0, 0)^T$ when $K = K_0$. For $K = K_{0,2}$, solving (5.5) with $\bar{U} = \left[\begin{smallmatrix} 1 \\ 0 \end{smallmatrix}\right]$ and $\bar{V} = \left[\begin{smallmatrix} 0 \\ 1 \end{smallmatrix}\right]$ yields $v_{K_{0,2}}(u) = 0$, so for both index sets we obtain the same trajectory $x(u) = (u, 0)^T$ for all $u \in \mathbb{R}$.

5.3. Multiplier functions. In order to express the gradient and Hessian of $L_{\mathcal{U}}(u;\bar{g}_{\mathcal{V}})$ as combinations of the gradients and Hessians of the primal functions, it is useful to have an explicit expression for the combination coefficients. Theorem 5.2 below shows that these multipliers (denoted by $\alpha_i(u)$) are smooth functions of u depending on $x(u)$ and $\bar{g}_{\mathcal{V}}$.

From here on, we assume that $K = K_f \cup K_\varphi$ is a dual feasible basic index set relative to $\bar{g} \in \partial f(\bar{x})$.

We use the following notation. Corresponding to the solution of the linear system in the next theorem, let

- $\alpha(u)$ be the $|K|$-dimensional column vector function formed from $\{\alpha_i(u)\}_{i\in K}$, with index $0 \in K_f$ coming first, the remaining indices of K_f coming next, and the indices of K_φ coming last;
- $[e^T|0]$ be a $(|K|-1)$-dimensional row vector with $|K_f|-1$ ones followed by $|K_\varphi|$ zeros;
- $I_{|K|-1}$ be a $(|K|-1)\times(|K|-1)$ identity matrix;
- the $n\times n$ matrix function $M(u)$ be defined by

$$M(u) := \sum_{i\in K_f} \alpha_i(u)\nabla^2 f_i(x(u)) + \sum_{i\in K_\varphi} \alpha_i(u)\nabla^2 \varphi_{i-m_1}(x(u)); \tag{5.6}$$

- the $n\times|K|$ matrix function $G(u)$ be defined by

$$\begin{aligned} &\text{(a)} \quad G(u) := \left[\{\nabla f_i(x(u))\}_{i\in K_f} \cup \{\nabla\varphi_{i-m_1}(x(u))\}_{i\in K_\varphi}\right] \\ &\text{(b)} \qquad\quad = [0|V(u)] + \nabla f_0(x(u))\left[1|\left[e^T|0\right]\right]. \end{aligned} \tag{5.7}$$

  Here the columns of $[0|V(u)]$ and $G(u)$ are ordered to correspond to the ordering of elements in $\alpha(u)$. □

THEOREM 5.2. *Let f have* pdg *structure about $\bar{x}$ and suppose $K = K_f \cup K_\varphi$ is a dual feasible basic index set relative to $\bar{g} \in \partial f(\bar{x})$, with corresponding multipliers $\{\bar{\alpha}_i\}_{i\in K}$ from Definition* 4.2(ii). *For each u small enough, the following hold:*

(i) *The linear system with variables α_i*

$$\begin{cases} \bar{V}^T\left(\displaystyle\sum_{i\in K_f}\alpha_i\nabla f_i(x(u)) + \sum_{i\in K_\varphi}\alpha_i\nabla\varphi_{i-m_1}(x(u)) - \bar{g}\right) = 0 \in \mathbb{R}^{|K|-1}, \\ \displaystyle\sum_{i\in K_f}\alpha_i = 1 \end{cases}$$

has a unique solution $\alpha = \alpha(u)$, given by

$$\begin{aligned} \{\alpha_i(u)\}_{0\neq i\in K} &= -(\bar{V}^T V(u))^{-1}\bar{V}^T(\nabla f_0(x(u)) - \bar{g}), \\ \alpha_0(u) &= 1 - \sum_{0\neq i\in K_f}\alpha_i(u). \end{aligned}$$

(ii) *With the above matrix-vector notation*

$$G(u)J\alpha(u) = [0|V(u)]J\alpha(u), \tag{5.8}$$

where the Jacobian of α is given by

$$J\alpha(u) = \begin{bmatrix} [e^T|0] \\ -I_{|K|-1} \end{bmatrix}(\bar{V}^T V(u))^{-1}\bar{V}^T M(u)Jx(u) \tag{5.9}$$

with $J\alpha(u) = 0$ if K is a singleton.

In particular, for all $i \in K$ we have $\alpha_i(0) = \bar{\alpha}_i$.

Proof. These results follow from the proof of [10, Theorem 6], mutatis mutandis. More precisely, if K is a singleton, then $\bar{V}$ is vacuous and $\alpha_0(u) = 1 = \bar{\alpha}_0$ for all u gives the desired results.

Suppose K is not a singleton. Then the linear system defining $\alpha(u)$ can be written as

$$\begin{array}{rl} \bar{V}^T G(u) & \alpha(u) = \bar{V}^T \bar{g}, \\ \left[1 \middle| \left[e^T | 0\right]\right] & \alpha(u) = 1. \end{array} \tag{5.10}$$

Computing $\alpha_0(u) = 1 - \sum_{0 \neq i \in K_f} \alpha_i(u)$ from the last equation and rearranging terms in the first gives

$$\begin{aligned} &\bar{V}^T \left(\sum_{0 \neq i \in K_f} \alpha_i(u) \nabla f_i(x(u)) + \sum_{i \in K_\varphi} \alpha_i(u) \nabla \varphi_{i-m_1}(x(u)) \right) \\ &\quad = -\bar{V}^T \left(\left(1 - \sum_{0 \neq i \in K_f} \alpha_i(u) \right) \nabla f_0(x(u)) - \bar{g} \right). \end{aligned}$$

Subtracting $\bar{V}^T \sum_{0 \neq i \in K_f} \alpha_i(u) \nabla f_0(x(u))$ from both sides and using the definition of $V(u)$ in Theorem 5.1 yields

$$(\bar{V}^T V(u))\{\alpha_i(u)\}_{0 \neq i \in K} = -\bar{V}^T (\nabla f_0(x(u)) - \bar{g}),$$

and (i) follows. In particular, for $u = 0$, recall $x(0) = \bar{x}$ and Definition 4.2(ii): $\alpha_i = \bar{\alpha}_i$ for $i \in K$ therein uniquely satisfies linear system (i) above; therefore $\alpha_i(0) = \bar{\alpha}_i$.

We now prove (ii). Differentiating (5.10) with respect to u and using the definitions of $G(u)$ and $M(u)$ in (5.7)(a) and (5.6), respectively, gives

$$\begin{array}{lrll} \text{(a)} & \bar{V}^T G(u) & J\alpha(u) + \bar{V}^T M(u) J x(u) & = 0 \in \mathbb{R}^{(|K|-1) \times \dim \mathcal{U}}, \\ \text{(b)} & \left[1 \middle| \left[e^T | 0\right]\right] & J\alpha(u) & = 0 \in \mathbb{R}^{1 \times \dim \mathcal{U}}. \end{array} \tag{5.11}$$

From (5.7)(b) and (5.11)(b), we obtain (5.8). Furthermore, (5.11)(b) and (5.8) combined with (5.11)(a) imply

$$\left[\begin{array}{c|c} 1 & \left[e^T | 0\right] \\ 0 & \bar{V}^T V(u) \end{array} \right] J\alpha(u) = \left[\begin{array}{c} 0 \\ -\bar{V}^T M(u) J x(u) \end{array} \right].$$

The final desired result (5.9) is obtained by multiplying the expression on the left-hand side by the $|K| \times |K|$ inverse matrix

$$\left[\begin{array}{c|c} 1 & -\left[e^T | 0\right] [\bar{V}^T V(u)]^{-1} \\ 0 & [\bar{V}^T V(u)]^{-1} \end{array} \right]. \qquad \square$$

For our example functions $\mathcal{E}_p$, recall that $x(u) = (u, 0)^T$ for both basic index sets $K = K_0$ and $K_{0,2}$ that are dual feasible relative to $\bar{g} = (0, 0)^T \in \partial \mathcal{E}_p(0, 0) = \partial \mathcal{E}_p(\bar{x})$. Then straightforward calculations give $\alpha(u) = 1$ if $K = K_0$ and $\alpha(u) = (1, 0)^T$ if $K = K_{0,2}$ for all $u \in \mathbb{R}$. If we let $\alpha_j(u) = 0$ for $j \notin K$, then for both index sets $(\alpha_0(u), \alpha_1(u), \alpha_2(u)) = (1, 0, 0)$ for all $u \in \mathbb{R}$.

6. The $\mathcal{U}$-Lagrangian and its derivatives. When in Definition 2.1 the full index set $\{0, 1, \ldots, m_1 + m_2\}$ is a basic index set, we say that f satisfies *strong transversality* at $\bar{x}$ (in this case $\dim\mathcal{V} = m_1 + m_2$). Also, from Definitions 2.1 and 4.2, this strong assumption implies that the full index set is a dual feasible basic index set for all $\bar{g} \in \partial f(\bar{x})$. For the max-eigenvalue case, this condition is assumed by many authors to obtain second-order developments (see [17], [21], [14], [15]). In particular, in [14], the existence of a $\mathcal{U}$-Hessian is proved if strong transversality holds and $\bar{g}$ is in the relative interior of $\partial f(\bar{x})$. This latter assumption means that $\bar{g} + td \in \partial f(\bar{x})$ for each $d \in \mathcal{V}$, provided $t \in \mathbb{R}$ is small enough.

Next we consider less restrictive conditions that also ensure the existence of a $\mathcal{U}$-Hessian for a function that has pdg structure.

6.1. $\mathcal{V}$-optimality conditions. Following [10], to express the $\mathcal{U}$-Hessian of f in terms of the Hessians $\nabla^2 f_i$ and $\nabla^2\varphi_\ell$ on a smooth trajectory $x(u)$, we need some conditions in addition to pdg structure. They play a role similar to that of constraint qualification conditions in nonlinear programming. They are the assumptions of Theorem 6.3 below that imply that for some basic index set K the corresponding trajectory $x(u)$ is in $\bar{x}+\bar{U}u+W(u)$, i.e., the $\mathcal{V}$-optimality result that $\bar{V}v_K(u) \in W(u)$.

The following sufficiency conditions all correspond to a dual feasible basic index set $K = K_f \cup K_\varphi$, relative to $\bar{g} \in \partial f(\bar{x})$ and its corresponding multiplier vector $\bar{\alpha} \in \Delta$ from Definition 4.2(ii).

Strong $\mathcal{V}$-optimality conditions. We say that K satisfies

- *strong primal feasibility* if $K_\varphi = \{m_1 + 1, \ldots, m_1 + m_2\}$ and for each $j \in \{1, \ldots, m_1\}\backslash K_f$ there exists a vector $\lambda^j \in \mathbb{R}^{|K|}$ such that

$$\text{(6.1)} \qquad \begin{cases} \displaystyle\sum_{i\in K_f} \lambda_i^j \nabla f_i(\bar{x}) + \sum_{i\in K_\varphi} \lambda_i^j \nabla\varphi_{i-m_1}(\bar{x}) = \nabla f_j(\bar{x}), \\ \displaystyle\sum_{i\in K_f} \lambda_i^j = 1, \end{cases}$$

  and $\bar{U}^T M_j \bar{U}$ is positive definite, where

$$\text{(6.2)} \qquad M_j := \sum_{i\in K_f} \lambda_i^j \nabla^2 f_i(\bar{x}) + \sum_{i\in K_\varphi} \lambda_i^j \nabla^2\varphi_{i-m_1}(\bar{x}) - \nabla^2 f_j(\bar{x});$$

- *strong dual feasibility* if $\bar{\alpha}$ is in the interior of Δ relative to the hyperplane

$$\mathcal{H}_K := \left\{ \alpha \in \mathbb{R}^{m_1+1+m_2} : \sum_{i\in K_f} \alpha_i = 1, \alpha_j = 0 \text{ for all } j \notin K \right\};$$

and

- *strong transversality* if $\mathcal{V}_K = \mathcal{V}$. □

If the pdg structure of f at $\bar{x}$ is known well enough, then the validity of the above conditions can be tested. The following weaker qualification conditions depend on the trajectory $x(u)$ and multiplier function $\alpha(u)$ associated with K from sections 5.2 and 5.3, respectively, and, hence, may be difficult or impossible to verify.

Weaker $\mathcal{V}$-optimality conditions. We say that K satisfies

- *primal feasibility relative to* $x(u)$ if for $u \in \mathbb{R}^{\dim\mathcal{U}}$ small enough,

$$\begin{cases} f_j(x(u)) \le f_0(x(u)) & \text{for all } j \in \{1, \ldots, m_1\}\backslash K_f, \\ \varphi_{j-m_1}(x(u)) = 0 & \text{for all } j \in \{m_1 + 1, \ldots, m_1 + m_2\}\backslash K_\varphi; \end{cases}$$

- *dual feasibility relative to* $\alpha(u)$ if for $u \in \mathbb{R}^{\dim \mathcal{U}}$ small enough,

$$(\alpha_0(u), \alpha_1(u), \ldots, \alpha_{m_1+m_2}(u)) \in \Delta, \quad \text{where } \alpha_j(u) := 0 \text{ for } j \notin K;$$

and

- *transversality relative to* $x(u)$ *and* $\alpha(u)$ if for $u \in \mathbb{R}^{\dim \mathcal{U}}$ small enough, $(g(u) - \bar{g})$ is orthogonal to $\mathcal{V} \backslash \mathcal{V}_K$, where

$$g(u) := \sum_{i \in K_f} \alpha_i(u) \nabla f_i(x(u)) + \sum_{i \in K_\varphi} \alpha_i(u) \nabla \varphi_{i-m_1}(x(u)). \tag{6.3}$$

  In terms of Matrix Notation 4.5, this transversality condition is equivalent to $\bar{Z}^T(g(u) - \bar{g}) = 0$. Note that since $x(0) = \bar{x}$, Theorem 5.2 and Definition 4.2(ii) imply that $g(0) = \bar{g}$. □

Now we show that the strong conditions given above do indeed imply the ones said to be weaker.

THEOREM 6.1. *The above* $\mathcal{V}$*-optimality conditions satisfy the following:*

(i) *If* f *satisfies strong transversality at* $\bar{x}$*, then the full index set* $\{0, 1, \ldots, m_1 + m_2\}$ *satisfies strong primal feasibility and strong transversality.*
If, in addition, $\bar{g}$ *is in the relative interior of* $\partial f(\bar{x})$*, then the full index set also satisfies strong dual feasibility.*

(ii) *If a basic index set* K *satisfies any one of the three strong* $\mathcal{V}$*-optimality conditions, then* K *satisfies the corresponding feasibility or transversality condition relative to its* $x(u)$ *and/or* $\alpha(u)$*.*

Proof. (i) The assumption of strong transversality trivially implies that the full index set satisfies strong primal feasibility and transversality. We next show satisfaction of strong dual feasibility by showing that $\bar{\alpha}$, corresponding to $\bar{g}$, is in the interior of Δ relative to the hyperplane $\{\alpha \in \mathbb{R}^{m_1+1+m_2} : \sum_{i=0}^{m_1} \alpha_i = 1\}$.

By strong transversality, there exists a unique $\bar{\alpha} \in \Delta$ corresponding to $\bar{g} \in \mathrm{ri}\partial f(\bar{x})$, i.e., satisfying

$$\begin{aligned} \sum_{i=0}^{m_1} \bar{\alpha}_i \nabla f_i(\bar{x}) + \sum_{i=m_1+1}^{m_1+m_2} \bar{\alpha}_i \nabla \varphi_{i-m_1}(\bar{x}) &= \bar{g}, \\ \sum_{i=0}^{m_1} \bar{\alpha}_i &= 1. \end{aligned} \tag{6.4}$$

Let $\beta \in \mathbb{R}^{m_1+1+m_2}$ be any vector satisfying

$$\sum_{i=0}^{m_1} \beta_i = 0 \tag{6.5}$$

as in the expression for $\mathcal{V}_0$ in (4.2). Since $\mathcal{V} = \mathcal{V}_0$ is a linear subspace, for any $t \in \mathbb{R}$, we have $td \in \mathcal{V}$, where

$$d := \sum_{i=0}^{m_1} \beta_i \nabla f_i(\bar{x}) + \sum_{i=m_1+1}^{m_1+m_2} \beta_i \nabla \varphi_{i-m_1}(\bar{x}). \tag{6.6}$$

Since $\bar{g} \in \mathrm{ri}\partial f(\bar{x})$ and $td \in \mathcal{V}$, $\bar{g} + td \in \partial f(\bar{x})$ for all t small enough.

Now (6.4)–(6.6) imply that $\alpha_i := \bar{\alpha}_i + t\beta_i$ satisfies

$$\sum_{i=0}^{m_1} \alpha_i \nabla f_i(\bar{x}) + \sum_{i=m_1+1}^{m_1+m_2} \alpha_i \nabla \varphi_{i-m_1}(\bar{x}) = \bar{g} + td \in \partial f(\bar{x}),$$
$$\sum_{i=0}^{m_1} \alpha_i = 1$$

for all t small enough. By strong transversality, the unique solution to the system above is $\alpha := (\bar{\alpha}_0 + t\beta_0, \ldots, \bar{\alpha}_{m_1+m_2} + t\beta_{m_1+m_2})$, so from Definition 2.1, $\alpha \in \Delta$ for all t small enough.

Since β is an arbitrary vector in $\mathbb{R}^{m_1+1+m_2}$ satisfying (6.5), $\bar{\alpha}$ lies in the interior of Δ relative to the hyperplane $\{\alpha \in \mathbb{R}^{m_1+1+m_2} : \sum_{i=0}^{m_1} \alpha_i = 1\}$ and the proof of part (i) is complete.

(ii) First, suppose that K satisfies strong primal feasibility. Then since $K_\varphi = \{m_1+1, \ldots, m_1+m_2\}$, to show satisfaction of primal feasibility we need only to show, for all $u \in \mathbb{R}^{\dim \mathcal{U}}$ small enough, that $f_j(x(u)) \le f_0(x(u))$ for all $j \in \{1, \ldots, m_1\} \backslash K_f$.

We have for each $l \in \{0, 1, \ldots, m_1\}$ that

$$f_l(x(u)) = f_l(\bar{x}) + \langle \nabla f_l(\bar{x}), x(u) - \bar{x} \rangle + \frac{1}{2} \left\langle \nabla^2 f_l(\bar{x})(x(u) - \bar{x}), x(u) - \bar{x} \right\rangle + o(\|x(u) - \bar{x}\|^2) \tag{6.7}$$

and for each $l \in \{m_1 + 1, \ldots, m_1 + m_2\}$ that

$$\varphi_l(x(u)) = \varphi_l(\bar{x}) + \langle \nabla \varphi_l(\bar{x}), x(u) - \bar{x} \rangle + \frac{1}{2} \left\langle \nabla^2 \varphi_l(\bar{x})(x(u) - \bar{x}), x(u) - \bar{x} \right\rangle + o(\|x(u) - \bar{x}\|^2). \tag{6.8}$$

Multiplying (6.7) with $l = i \in K_f$ and (6.8) with $l = i - m_1 \ge 1$ by λ_i^j, summing the results and using (6.1), (6.2), and the facts that $f_i(x(u)) = f_0(x(u))$ for all $i \in K_f$ and $\varphi_{i-m_1}(x(u)) = 0$ for all $i \ge m_1 + 1$ gives

$$f_0(x(u)) = f_0(\bar{x}) + \langle \nabla f_j(\bar{x}), x(u) - \bar{x} \rangle + \frac{1}{2} \left\langle (M_j + \nabla^2 f_j(\bar{x}))(x(u) - \bar{x}), x(u) - \bar{x} \right\rangle + o(\|x(u) - \bar{x}\|^2). \tag{6.9}$$

Subtracting (6.7) with $l = j \in \{1, \ldots, m_1\} \backslash K_f$ from (6.9) and using the facts that $f_j(\bar{x}) = f(\bar{x}) = f_0(\bar{x})$ and $x(u) - \bar{x} = \bar{U}u + \bar{V}v_K(u)$ gives

$$f_0(x(u)) - f_j(x(u)) = \frac{1}{2} \left\langle M_j(\bar{U}u + \bar{V}v_K(u)), \bar{U}u + \bar{V}v_K(u) \right\rangle + o(\|\bar{U}u + \bar{V}v_K(u)\|^2).$$

Now from strong primal feasibility, $\bar{U}^T M_j \bar{U}$ is positive definite, and from Theorem 5.1, $v_K(u) = o(\|u\|)$, so $f_0(x(u)) - f_j(x(u)) > 0$ for all nonzero $u \in \mathbb{R}^{\dim \mathcal{U}}$ small enough. Thus K satisfies primal feasibility relative to $x(u)$.

Second, suppose that K satisfies strong dual feasibility. Since $\alpha_j(0) = 0$ for all $j \notin K$, Theorem 5.2 implies that

$$(\alpha_0(u), \alpha_1(u), \ldots, \alpha_{m_1+m_2}(u)) \in \mathcal{H}_K$$

and that

$$(\alpha_0(u), \alpha_1(u), \ldots, \alpha_{m_1+m_2}(u)) \to \bar{\alpha} \text{ as } u \to 0 \in \mathbb{R}^{\dim \mathcal{U}}.$$

Now since $\bar{\alpha}$ is assumed to be in the interior of Δ relative to $\mathcal{H}_K$,

$$(\alpha_0(u), \alpha_1(u), \ldots, \alpha_{m_1+m_2}(u)) \in \Delta \quad \text{for all } u \in \mathbb{R}^{\dim \mathcal{U}} \text{ small enough.}$$

Thus K satisfies dual feasibility relative to $\alpha(u)$.

Third, suppose that K satisfies strong transversality. Then $\mathcal{V} \backslash \mathcal{V}_K$ is empty, so K trivially satisfies transversality relative to its $x(u)$ and $\alpha(u)$. □

The two strong assumptions in (i) of Theorem 6.1 are sufficient for showing the existence of a desirable basic index set. The fact that neither is necessary is illustrated by [10, Examples 15 and 17], [15, Example 4.11], and the following discussion concerning examples $\mathcal{E}_1$ and $\mathcal{E}_2$. Indeed, these functions have at $\bar{x} = (0,0)^T$ two basic index sets, K_0 and $K_{0,2}$, that are dual feasible relative to $\bar{g} = (0,0)^T \in \partial \mathcal{E}_p(0,0)$, with common trajectory $x(u) = (u,0)^T$ and multipliers $(\alpha_0(u), \alpha_1(u), \alpha_2(u)) = (1,0,0)$ for all $u \in \mathbb{R}$.

$\mathcal{E}_p$ for $p = 1, 2$ does not satisfy strong transversality at $(0,0)^T$ because the vectors

$$\begin{bmatrix} \nabla f_0(0,0) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} \nabla f_1(0,0) \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \begin{bmatrix} \nabla \varphi_1(0,0) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{p} \\ 0 \end{bmatrix}$$

are linearly dependent. In particular, $\mathcal{E}_1$ is an example of a maximum eigenvalue function that does not satisfy the strong transversality assumptions of [17], [21], [14], and [15].

From the expressions for $\partial \mathcal{E}_p$ in section 3.1 we see that $\bar{g} = (0,0)^T$ is in the relative interior of $\partial \mathcal{E}_1(0,0)$, but not in that of $\partial \mathcal{E}_2(0,0)$, so $\mathcal{E}_2$ satisfies neither of the very strong conditions of Theorem 6.1(i).

Next, we show that relative to $\bar{g} = (0,0)^T \in \partial \mathcal{E}_p(0,0)$, K_0 satisfies one and $K_{0,2}$ satisfies the other two of the three strong $\mathcal{V}$-optimality conditions and that they both satisfy all three of the weaker conditions.

$K_{0,2}$ satisfies strong primal feasibility because the only index not in $K_{0,2}$ is index 1, corresponding to the zero function f_1, and in this case (6.1) and (6.2) become

$$\begin{cases} \lambda_0^1 (0,0)^T + \lambda_2^1 (0, \frac{1}{p})^T = (0,0)^T, \\ \lambda_0^1 = 1, \end{cases} \quad \text{and} \quad M_1 = 1 \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix},$$

so

$$\bar{U}^T M_1 \bar{U} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 2 > 0.$$

As for K_0, it does not satisfy strong primal feasibility because it does not contain index 2, corresponding to φ_1. However, K_0 does satisfy the given weaker primal feasibility conditions relative to $x(u)$ because for all $u \in \mathbb{R}$, $x(u) = (u,0)^T$ and thus

$$f_1(u,0) = 0 \leq u^2 = f_0(u,0) \quad \text{and} \quad \varphi_1(u,0) = \frac{1}{p} 0 = 0.$$

For consideration of dual feasibility, recall that $\bar{\alpha} = (1,0,0)$ and $\Delta = \{(\alpha_0, \alpha_1, \alpha_2) : \alpha_0 = 1 - \alpha_1, 4(\alpha_1 - \frac{1}{2})^2 + \alpha_2^2 \leq 1, (p-1)\alpha_2 \geq 0\}$ and let $B(\bar{\alpha})$ be any ball about $\bar{\alpha}$ in $\mathbb{R}^3$. K_0 satisfies strong dual feasibility because it does not contain indices 1 and 2, which implies that $\mathcal{H}_{K_0}$ is the singleton $(1,0,0) = \bar{\alpha} \in \Delta$, so

$$B(\bar{\alpha}) \cap \mathcal{H}_{K_0} = \{\bar{\alpha}\} = \Delta \cap \mathcal{H}_{K_0}$$

and thus $\bar{\alpha}$ is in the interior of Δ relative to $\mathcal{H}_{K_0}$. As for $K_{0,2}$, it does not satisfy strong dual feasibility because $\mathcal{H}_{K_{0,2}}$ is the line given by $(1,0,0)+(0,0,\alpha_2)$ for $\alpha_2 \in \mathbb{R}$ and thus

$$B(\bar{\alpha}) \cap \mathcal{H}_{K_{0,2}} \not\subset \{\bar{\alpha}\} = \Delta \cap \mathcal{H}_{K_{0,2}}.$$

However, $K_{0,2}$ does satisfy the given weaker dual feasibility condition relative to $\alpha(u)$ because for all $u \in \mathbb{R}$ $(\alpha_0(u), \alpha_1(u), \alpha_2(u)) = (1,0,0) = \bar{\alpha} \in \Delta$.

Finally, because $\mathcal{V}_{K_0} = \{(0,0)^T\} \neq \mathcal{V} = \mathcal{V}_{K_{0,2}}$, $K_{0,2}$ satisfies strong transversality and K_0 does not. However, (6.3) with $K = K_0$ gives $g(u) = 1\nabla f_0(x(u)) = (2u,0)^T$ for all $u \in \mathbb{R}$, so $g(u) - \bar{g} = (2u,0)^T - (0,0)^T = (2u,0)^T \in \mathcal{U} = \mathcal{V}^\perp$ and thus K_0 satisfies transversality relative to $x(u)$ and $\alpha(u)$.

Remark 6.2. We note in passing that $\mathcal{E}_1$ is an interesting example for studying the regularity condition of [5] for nonconvex semidefinite programming. Recall that the problem of minimizing $\mathcal{E}_1$ is equivalent to minimizing the maximum eigenvalue of the matrix $\left[\begin{smallmatrix} x_1^2 & x_2 \\ x_2 & 0 \end{smallmatrix}\right]$. In turn, this is equivalent to the following problem:

$$\begin{array}{ll} \text{minimize} & x_3 \\ \text{subject to} & x_3 \geq \mathcal{E}_1(x_1, x_2) \end{array}$$

or, equivalently,

$$\text{subject to} \quad C(x) := \begin{bmatrix} x_3 - x_1^2 & -x_2 \\ -x_2 & x_3 \end{bmatrix} \text{ being positive semidefinite.}$$

This problem is solved by $x^* = (0,0,0)^T$, where $C(x^*)$ equals the 2×2 zero matrix which has a range space basis matrix that is empty and a null space basis matrix that is the 2×2 identity. Thus the *local feasibility matrix* of [5, section 2.2], $\tilde{C}(x)$, coincides with $C(x)$. Moreover, $S(\tilde{C}, x^*)$, the matrix subspace reflecting the sparsity pattern of $\tilde{C}$, is $\mathcal{S}^2$, the whole space of symmetric matrices of order 2. As a result, the strong transversality-like regularity condition (2.5) of [5] does not hold at x^* since the three matrices $\frac{\partial C(x^*)}{\partial x_i}$ equal to

$$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \text{ if } i = 1, \quad \begin{bmatrix} 0 & -1 \\ -1 & 0 \end{bmatrix} \text{ if } i = 2, \quad \text{and} \quad \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ if } i = 3$$

do not span $\mathcal{S}^2$. □

6.2. Expressions for the $\mathcal{U}$-gradient and the $\mathcal{U}$-Hessian. The stage is now set for giving expressions for the $\mathcal{U}$-Lagrangian (5.3) and its derivatives. To do this, we first prove that the candidate $\bar{V}v_K(u)$ from Theorem 5.1 is $\mathcal{V}$-optimal in the sense that it is an element of the Winning set $W(u)$ from (5.1).

The following theorem with $f = \mathcal{E}_p$ for $p = 1, 2$ and

$\bar{x}$	$\bar{g}$	$\bar{U}$	$\bar{K}$	$x(u)$	$\alpha(u)$	$g(u)$
$(0,0)^T$	$(0,0)^T$	$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$	K_0 or $K_{0,2}$	$(u,0)^T$	$(1,0,0)^T$	$(2u,0)^T$

gives

$$L_{\mathcal{U}}(u;0) = u^2, \quad \nabla L_{\mathcal{U}}(u;0) = 2u, \quad \text{and} \quad \nabla^2 L_{\mathcal{U}}(u;0) = 2 \quad \text{for all } u \in \mathbb{R}.$$

THEOREM 6.3. *Let f have* pdg *structure about $\bar{x}$. Suppose $\bar{K} = \bar{K}_f \cup \bar{K}_\varphi$ is a dual feasible basic index set relative to $\bar{g} = \sum_{i\in\bar{K}_f} \bar{\alpha}_i \nabla f_i(\bar{x}) + \sum_{i\in\bar{K}_\varphi} \bar{\alpha}_i \nabla \varphi_{i-m_1}(\bar{x}) \in \partial f(\bar{x})$, where $\bar{\alpha} \in \Delta$. Also, suppose $\bar{K}$ satisfies the three $\mathcal{V}$-optimality conditions relative to $\alpha(u)$ from Theorem* 5.2 *with $K = \bar{K}$ and $x(u) = \bar{x} + \bar{U}u + \bar{V}v_{\bar{K}}(u)$, where $v_{\bar{K}}(u)$ is defined in Theorem* 5.1 *if $K = \bar{K}$ is not a singleton, and $\bar{V}v_{\bar{K}}(u) := 0$ otherwise.*

Then for u small enough, we have the following:

(i) *the vector $\bar{V}v_{\bar{K}}(u)$ is an element of $W_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}})$. Equivalently, the trajectory vector $x(u)$ is an element of $\bar{x} + \bar{U}u + W_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}})$.*

(ii) (a) *If $\bar{K}_\varphi$ is nonempty, $0 = \varphi_{i-m_1}(x(u))$ for each $i \in \bar{K}_\varphi$.*
(b) *The $\mathcal{U}$-Lagrangian is given by*

$$\begin{aligned} L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) &= f(x(u)) - \bar{g}^T \bar{V} v_{\bar{K}}(u) \\ &= f_i(x(u)) - \bar{g}^T \bar{V} v_{\bar{K}}(u) \quad \text{for each } i \in \bar{K}_f. \end{aligned}$$

(iii) *The gradient of $L_{\mathcal{U}}$ is given by*

$$\nabla L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = \bar{U}^T g(u),$$

where $g(u)$ is given in (6.3) *with $K = \bar{K}$ and $\bar{U} = Jx(0)$ is a basis matrix for $\mathcal{U}$.*

(iv) *The Hessian of $L_{\mathcal{U}}$ is given by*

$$\nabla^2 L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = Jx(u)^T M(u) Jx(u),$$

where $M(u)$ is given in (5.6) *with $K = \bar{K}$.*

In particular, $L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}}) = f(\bar{x})$, and the $\mathcal{U}$-gradient and $\mathcal{U}$-Hessian at $\bar{x}$ are given by

$$\nabla L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}}) = \bar{U}^T \bar{g} = \bar{U}^T g \quad \text{for all } g \in \partial f(\bar{x})$$

and

$$\nabla^2 L_{\mathcal{U}}(0; \bar{g}_{\mathcal{V}}) = \bar{U}^T \left(\sum_{i\in\bar{K}_f} \bar{\alpha}_i \nabla^2 f_i(\bar{x}) + \sum_{i\in\bar{K}_\varphi} \bar{\alpha}_i \nabla^2 \varphi_{i-m_1}(\bar{x}) \right) \bar{U}.$$

Proof. (i) From expression (5.3), the optimality condition characterizing elements in (5.1) is $w = \bar{V}v_{\bar{K}} + \bar{Z}z \in W(u)$ if there exists a subgradient $g \in \partial f(\bar{x} + \bar{U}u + \bar{V}v_{\bar{K}} + \bar{Z}z)$ such that

$$[\bar{V}|\bar{Z}]^T (g - \bar{g}) = 0. \tag{6.10}$$

To prove (i), we show that $w = \bar{V}v_{\bar{K}}(u) + \bar{Z}z$ with $\bar{Z}z = 0$ and $g = g(u)$ from (6.3) with $K = \bar{K}$ satisfy this optimality condition.

Applying Theorem 5.2(i) with $K = \bar{K}$ gives

$$\bar{V}^T (g(u) - \bar{g}) = 0. \tag{6.11}$$

Together with the transversality assumption $\bar{Z}^T(g(u) - \bar{g}) = 0$, this yields satisfaction of (6.10) when $g = g(u)$. To see that $g(u) \in \partial f(\bar{x} + \bar{U}u + \bar{V}v_{\bar{K}}(u))$, note first that since $\alpha_i(u) = 0$ for $i \notin \bar{K}$, this condition is equivalent to the condition

$$g(u) = \sum_{i=0}^{m_1} \alpha_i(u) \nabla f_i(x(u)) + \sum_{i=m_1+1}^{m_1+m_2} \alpha_i(u) \nabla \varphi_{i-m_1}(x(u)) \in \partial f(x(u)). \tag{6.12}$$

By construction of $v_{\bar{K}}(u)$ in Theorem 5.1(i), primal feasibility implies that $x(u) \in \mathcal{P}$ as well as

$$f(x(u)) = \max\{f_i(x(u)) : i = 0, 1, \ldots, m_1\} = f_i(x(u)) \quad \text{for each } i \in \bar{K}_f. \tag{6.13}$$

Complementary slackness from Definition 2.1(iv) is satisfied with $x = x(u) \in \mathcal{P}$ and $\alpha_i = \alpha_i(u)$ because if $f_i(x(u)) < f(x(u))$, then $i \notin \bar{K}_f$ and $\alpha_i(u) = 0$. Finally, (6.12) follows from (2.4) in Definition 2.1(iv) because the dual feasibility assumption says that $(\alpha_0(u), \alpha_1(u), \ldots, \alpha_{m_1+m_2}(u)) \in \Delta$.

(ii) In (i), we already used primal feasibility to conclude that $x(u) \in \mathcal{P}$, so (a) follows. Result (b) is a direct consequence of (i) and (6.13).

(iii) The proof is similar to the proof of [10, Theorem 9(i)]; i.e., use the chain rule to differentiate (ii)(a) and (ii)(b) with respect to u to obtain

$$\begin{aligned} 0 &= Jx(u)^T \nabla \varphi_{i-m_1}(x(u)) && \text{for each } i \in \bar{K}_\varphi \\ \text{and} \quad \nabla L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) &= Jx(u)^T \nabla f_i(x(u)) - Jv_{\bar{K}}(u)^T \bar{V}^T \bar{g} && \text{for each } i \in \bar{K}_f. \end{aligned}$$

Multiplying each equation by the appropriate $\alpha_i(u)$, summing the results, and using the fact that $\sum_{i \in \bar{K}_f} \alpha_i(u) = 1$ yields

$$\begin{aligned} \nabla L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) &= Jx(u)^T \left(\sum_{i \in K_f} \alpha_i(u) \nabla f_i(x(u)) + \sum_{i \in K_\varphi} \alpha_i(u) \nabla \varphi_{i-m_1}(x(u)) \right) - Jv_{\bar{K}}(u)^T \bar{V}^T \bar{g} \\ &= Jx(u)^T g(u) - Jv_{\bar{K}}(u)^T \bar{V}^T \bar{g}. \end{aligned}$$

Using the transpose of the expression for $Jx(u)$ given in Theorem 5.1(iii) with $K = \bar{K}$, we obtain

$$\nabla L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = \bar{U}^T g(u) + Jv_{\bar{K}}(u)^T \bar{V}^T (g(u) - \bar{g}),$$

which together with (6.11) yields the desired result.

(iv) Again, we follow [10], but this time we use the proof of Theorem 9(ii). From (6.3) and (5.7)(a), we have that $g(u) = G(u)\alpha(u)$, with $G(u)$ being C^1 because $x(u)$ is C^1. Differentiating (iii) and using (5.6) gives

$$\nabla^2 L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = \bar{U}^T \left(G(u) J\alpha(u) + M(u) Jx(u) \right). \tag{6.14}$$

Now combine (5.8) and (5.9) with $K = \bar{K}$ to write

$$G(u)\ J\alpha(u) = -V(u)[\bar{V}^T V(u)]^{-1} \bar{V}^T M(u) Jx(u).$$

Using the transpose of the expression for $Jv_{\bar{K}}(u)$ in Theorem 5.1(ii), we obtain

$$\bar{U}^T G(u) J\alpha(u) = Jv_{\bar{K}}(u)^T \bar{V}^T M(u) Jx(u)$$

so that from (6.14) we have $\nabla^2 L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}}) = (Jv_{\bar{K}}(u)^T \bar{V}^T + \bar{U}^T) M(u) Jx(u)$. From Theorem 5.1(iii), this is equivalent to the desired expression for $\nabla^2 L_{\mathcal{U}}(u; \bar{g}_{\mathcal{V}})$. □

7. Concluding remarks. We introduced a general class of convex functions whose structural properties allow for the construction of smooth trajectories along which the functions are twice differentiable.

This pdg-structured class includes finite max-functions, maximum eigenvalue, and other infinite max-functions such as example $\mathcal{E}_2$. It also includes the modification of Example 3.0 where $\mathcal{T}$ in (2.1) depends on x.

As suggested to us by one of the referees, a function to investigate in the future for `pdg` structure is the following one defined on the space of $s \times s$ symmetric matrices:

$$\begin{aligned} \text{PF}: \quad & \mathcal{S} \to \mathbb{R}, \\ & X \mapsto \max\{t^T X t : t \in \mathbb{R}^s_+, t^T t = 1\}. \end{aligned}$$

The *Perron–Fröbenius* function PF is proved to be the maximum eigenvalue of X whenever X has nonnegative entries [19]. However, for a general X PF, it is not a `mef`, as can be seen from its ("asymmetric") subdifferential:

$$\partial \text{PF}(X) = \{S \in \Delta^s : s_{ij} \geq 0 \text{ for all } i, j \text{ and } \operatorname{trace} SX = \text{PF}(X)\}.$$

The subdifferential of the following special case is similar to the subdifferential of the noneigenvalue example $\mathcal{E}_2$: for $n = 2$ and $X = I$, the identity matrix, $\text{PF}(X) = 1$, so

$$\partial \text{PF}(I) = \left\{ S = \begin{bmatrix} s_{11} & s_{12} \\ s_{12} & 1 - s_{11} \end{bmatrix} : S \text{ is positive semidefinite and } s_{12} \geq 0 \right\}.$$

Thus for $S \in \partial \text{PF}(I)$, $(s_{11}, 2s_{12})$ is in the shaded region in Figure 3.1, which has no center of symmetry. □

The special `pdg` structure of the subdifferential of f about $\bar{x}$ provides a set of spanning vectors for the subspace $\mathcal{V}$. A linearly independent subset of these vectors whose indices constitute the set K generate a subspace $\mathcal{V}_K \subseteq \mathcal{V}$ and an implicit function therein from which a trajectory tangent to $\mathcal{U}$ at $\bar{x}$ can be defined. If such a trajectory $x(u)$ satisfies primal feasibility, there is a second-order expansion of f along $x(u)$, depending on a subgradient $\bar{g} \in \partial f(\bar{x})$, a corresponding multiplier vector $\alpha(u)$, and an associated Hessian with respect to u. If, in addition, $\alpha(u)$ satisfies dual feasibility and, together with $x(u)$, transversality, then the Hessian is the Hessian of the $\mathcal{U}$-Lagrangian depending on the $\mathcal{V}$-component of $\bar{g}$. The above ordering of this paper's results will be important for extending some of the theory to locally Lipschitz functions.

For the special `mef` case, our structural conditions are weaker than those in [21] and [14] because we isolate the place where the implicit function theorem is applied, i.e., relative to K. Moreover, our results extend those in [14] both to matrices $A(x)$ in (3.4) with nonaffine diagonal element functions and to subgradients $\bar{g}$ not necessarily in the relative interior of $\partial f(\bar{x})$.

Acknowledgments. We thank the referees for their beneficial suggestions for improvement of this paper.

REFERENCES

[1] A. Ben-Tal and M. P. Bendsøe, *A new method for optimal truss topology design*, SIAM J. Optim., 3 (1993), pp. 322–358.

[2] M. Bendsøe, *Optimization of Structural Topology, Shape and Material*, Springer, New York, 1995.

[3] F. Clarke, *Optimization and Nonsmooth Analysis*, Classics in Appl. Math. 5, SIAM, Philadelphia, 1990.

[4] V. Dem'yanov and V. Malozemov, *Introduction to Minimax*, John Wiley, New York, 1974.

[5] A. Forsgren, *Optimality conditions for nonconvex semidefinite programming*, Math. Programming, 88 (2000), pp. 105–128.

[6] C. Lemaréchal and R. Mifflin, *Global and superlinear convergence of an algorithm for one-dimensional minimization of convex functions*, Math. Programming, 24 (1982), pp. 241–256.

[7] C. Lemaréchal, F. Oustry, and C. Sagastizábal, *The $\mathcal{U}$-Lagrangian of a convex function*, Trans. AMS, 352 (2000), pp. 711–729.
[8] A. Lewis and M. Overton, *Eigenvalue optimization*, Acta Numer., 5 (1996), pp. 149–190.
[9] R. Mifflin, *An implementation of an algorithm for univariate minimization and an application to nested optimization*, Math. Programming Stud., 31 (1987), pp. 155–166.
[10] R. Mifflin and C. Sagastizábal, *$\mathcal{VU}$-decomposition derivatives for convex max-functions*, in Ill-Posed Variational Problems and Regularization Techniques, R. Tichatschke and M. Théra, eds., Lecture Notes in Econom. and Math. Systems 477, Springer-Verlag, Berlin, Heidelberg, 1999, pp. 167–186.
[11] R. Mifflin and C. Sagastizábal, *Functions with primal-dual gradient structure and $\mathcal{U}$-Hessians*, in Nonlinear Optimization and Related Topics, G. D. Pillo and F. Giannessi, eds., Applied Optimization 36, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 219–233.
[12] M. Osborne, *Finite Algorithms in Optimization and Data Analysis*, John Wiley, New York, 1985.
[13] M. Osborne, S. Pruess, and R. Womersley, *Concise Representation of Generalized Gradients*, J. Austral. Math. Soc. Ser. B, 28 (1986), pp. 57–74.
[14] F. Oustry, *The $\mathcal{U}$-Lagrangian of the maximum eigenvalue function*, SIAM J. Optim., 9 (1999), pp. 526–549.
[15] F. Oustry, *A second-order bundle method to minimize the maximum eigenvalue function*, Rapport de Recherche 3738, INRIA Rhone-Alpes, 1999, Math. Programming, to appear.
[16] M. L. Overton, *Large-scale optimization of eigenvalues*, SIAM J. Optim., 2 (1992), pp. 88–120.
[17] M. L. Overton and R. Womersley, *Second derivatives for optimizing eigenvalues of symmetric matrices*, J. Math. Anal. Appl., 3 (1995), pp. 667–718.
[18] F. Rendl and H. Wolkowicz, *A semidefinite framework for trust region subproblems with applications to large scale minimization*, Math. Programming, 77 (1997), pp. 273–299.
[19] W. Rheinboldt and J. Vandergraft, *A simple approach to the Perron-Fröbenius theory for positive operators on general partially-ordered finite-dimensional linear spaces*, Math. Comput., 27 (1973), pp. 139–145.
[20] R. Rockafellar, *Convex Analysis*, Princeton Math. Ser. 28, Princeton University Press, Princeton, NJ, 1970.
[21] A. Shapiro and M. K. H. Fan, *On eigenvalue optimization*, SIAM J. Optim., 5 (1995), pp. 552–569.
[22] J. Wilkinson, *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford, UK, 1965.
[23] R. Womersley, *Local properties of algorithms for minimizing nonsmooth composite functions*, Math. Programming, 32 (1985), pp. 69–89.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 573–594

PATTERN SEARCH ALGORITHMS FOR MIXED VARIABLE PROGRAMMING*

CHARLES AUDET† AND J. E. DENNIS, JR.‡

Abstract. Many engineering optimization problems involve a special kind of discrete variable that *can* be represented by a number, but this representation has no significance. Such variables arise when a decision involves some situation like a choice from an unordered list of categories. This has two implications: The standard approach of solving problems with continuous relaxations of discrete variables is not available, and the notion of local optimality must be defined through a user-specified set of neighboring points. We present a class of direct search algorithms to provide limit points that satisfy some appropriate necessary conditions for local optimality for such problems. We give a more expensive version of the algorithm that guarantees additional necessary optimality conditions. A small example illustrates the differences between the two versions. A real thermal insulation system design problem illustrates the efficacy of the user controls for this class of algorithms.

Key words. pattern search algorithm, convergence analysis, bound constrained optimization, mixed variable programming, derivative-free optimization

AMS subject classifications. 49M30, 65K05, 90C11, 90C56

PII. S1052623499352024

1. Introduction.

Torczon [12] defined a class of generalized pattern search methods to minimize a function $f : \Re^n \to \Re$ without any knowledge of its derivatives. She shows that the class includes algorithms such as *coordinate search* with fixed step sizes, *evolutionary operation* using factorial design [3], the original *pattern search algorithm* [7], and the *multidirectional search algorithm* [5]. In [12] she gave general convergence results under the assumption of continuous differentiability.

The main result of [12] is that for $f \in \mathcal{C}^1$, the sequence of iterates $\{x_k\}$ of $\Re^n$ generated by any generalized pattern search (GPS) method satisfies

$$\liminf_{k\to\infty} \|\nabla f(x_k)\| = 0 \tag{1.1}$$

without ever computing or explicitly approximating derivatives. At each iteration, the function is evaluated at trial points on a discrete mesh containing the current iterate in search of one yielding any decrease in the objective function value. Lewis and Torczon [10] use positive basis theory to strengthen the result by roughly cutting in half the worst case number of trial points at each iteration without affecting the convergence result. Lewis and Torczon [9], [11] extend pattern search algorithms and the convergence theory to bound and linearly constrained minimization by adapting the exploration of the domain near the boundary of the feasible region. The optimality

*Received by the editors February 11, 1999; accepted for publication (in revised form) June 22, 2000; published electronically November 10, 2000. A preliminary version of this work appeared as Technical report TR99-02, Department of Computational and Applied Mathematics, Rice University, Houston, TX, 1999. This work was supported by DOE grant DE-FG03-95ER25257, AFOSR grant F49620-98-1-0267, The Boeing Company, Sandia National Laboratories grant LG-4253, Exxon-Mobil, and NSF CRPC grant CCR-9120008.

http://www.siam.org/journals/siopt/11-3/35202.html

†Département de Mathématiques et de Génie Industriel, École Polytechnique de Montréal, GÉRAD, C.P. 6079, Succ. Centre-Ville, Montréal, Québec, H3C 3A7, Canada (charlesa@crt.umontreal.ca). The work of this author was supported by NSERC (Natural Sciences and Engineering Research Council) fellowship PDF-207432-1998 during a postdoctoral stay at Rice University.

‡Department of Computational and Applied Mathematics, Rice University, MS 134, 6100 South Main Street, Houston, TX 77005-1892 (dennis@caam.rice.edu).

condition guaranteed by their approach is the existence of a limit point $\hat{x}$ of the sequence of iterates $\{x_k\}$ that satisfies

$$(x - \hat{x})^T \nabla f(\hat{x}) \geq 0 \qquad \text{for any feasible } x. \tag{1.2}$$

This condition reduces to (1.1) in the event that $\hat{x}$ is a strictly interior point.

Our purpose here is to further generalize the problem to be solved because many engineering optimization problems contain both continuous and discrete variables. Moreover, the discrete variables are often categorical ones; i.e., they refer to a list or set of categories and thus, the standard mixed integer approach of solving with continuous relaxations through branch and bound is not available. Of course, when branch and bound can be used, it probably should be, but that is not the issue here. Indeed, the context in which our algorithm is to be applied is that the variables are provided by the algorithm as input to a black box simulation. It would be surprising if one could run the simulation code with a continuous variable, where the simulation expects a discrete input—perhaps specifying the state of the physical medium under investigation.

We consider the problem of minimizing the function $f : \Omega \to \Re$, where the domain is partitioned into continuous and discrete variables Ω^c and Ω^d (some or all of the discrete variables may be categorical). The domain of the continuous variables is bound constrained $\Omega^c = [\ell, u]$, where $\ell, u \in \Re^{n^c} \cup \{\pm\infty\}$, $\ell < u$, and n^c is the dimension of the space. The domain of the discrete variable Ω^d has dimension n^d and may be represented by a subset of $\mathcal{Z}^{n^d}$. The continuous and discrete components of the iterates generated by the method will be denoted by $x_k = (x_k^c, x_k^d)$, where $x_k^c \in \Re^{n^c}$ and $x_k^d \in \mathcal{Z}^{n^d}$. We understand that we are abusing notation here, since we certainly mean that $x_k \in \Re^{n^c} \times \mathcal{Z}^{n^d}$. However, the purpose of notation is to explicate, not to be pedantic, and we are sure the reader will forgive us.

The function f is assumed to be continuously differentiable when the discrete variables in Ω^d are fixed. We present a general mixed variable pattern (GMVP) search method that reduces to that of Lewis and Torczon [9] when the dimension n^d is fixed to zero. Thus, like them, we deal with infeasible trial points by setting $f(x)$ to a large value.

A second objective of the paper is to slightly generalize the part of the algorithm that deals with the continuous variables and to revise and shorten the arguments developed in [12] and [10]. We first show how to obtain a limit point $\hat{x}$ of the sequence of iterates that satisfies first-order optimality conditions with respect to the continuous variables. These conditions reduce to (1.2) when there are no discrete variables. We also guarantee that the same limit point $\hat{x}$ satisfies some local optimality conditions with respect to the discrete variables. The notion of local optimality is defined through the user-specified set of neighbors $\mathcal{N}(x) \subset \Omega$ described in section 2.1. We also present a second version of the algorithm that guarantees stronger results.

The paper is structured as follows. First, we present a definition of local optimality for mixed variable programming and the optimality conditions guaranteed by our algorithm. We use a design problem for a thermal insulation system to illustrate categorical variables and our version of local optimality. Then in section 3, we formally describe a general framework for pattern search algorithms with mixed variables. In section 4, we provide the analysis to specify a subsequence of iterates whose limit points satisfy optimality conditions, including a stronger version of the algorithm that uses more function evaluations per iteration to guarantee an additional necessary optimality condition. Section 5 illustrates the difference between

the two versions of the algorithm on a small example, and it reports results for the algorithm applied to the problem in section 2.2. We use that example to illustrate some controls the user has to spend more function evaluations to gain a better local optimum.

2. Mixed variables.

2.1. Local optimality for mixed variables. In the absence of discrete variables, the definition of local optimality is straightforward: $\hat{x} \in [\ell, u]$ is a local minimizer of the bound constrained function f if there exists an $\epsilon > 0$ such that $f(\hat{x}) \leq f(v)$ for all $v \in [\ell, u]$ in a ball $B(\epsilon, \hat{x})$ of radius ϵ around $\hat{x}$.

When the optimization problem contains only discrete variables, a definition of local optimality might be the following: $f(\hat{x}) \leq f(y)$ for all y in $\mathcal{N}(\hat{x})$, where $\mathcal{N}(\hat{x})$ is a finite set of neighbors including and around the discrete variable $\hat{x}$. This specifies the quality of the solution for which one is willing to pay the necessary function values by defining the notion of "local optimality" the algorithm is to achieve with respect to the discrete variables.

An example is the quadratic assignment problem (QAP) in which n facilities must be assigned to n locations: each assignment may be represented using one of the $n!$ permutations of the vector $(1, 2, \ldots, n)$. The key point in the definition is for the user to answer the question, What property must the solution provided by the algorithm satisfy in order for it to be a satisfactory local solution? One might decide, for example, that a QAP solution is interesting if a given assignment x could not be improved by changing x in at most two locations (or, more stringently, in at most three locations). Our approach is completely flexible in this respect; however, the more stringent the conditions of local optimality the user wants to impose, the more expensive the GMVP poll step will be.

Consider, for example, the QAP with three facilities. It may be modeled with three discrete variables ($x^d \in \mathcal{Z}^3$). Not all the points of the integer lattice $\mathcal{Z}^3$ represent feasible assignments, only the permutations of $(1, 2, 3)$. Also, the ordering is not the classical one associated with an inherited metric, since for the set of neighbors $\mathcal{N}(1, 2, 3) = \{(1, 2, 3), (1, 3, 2), (3, 2, 1), (2, 1, 3)\}$ the assignment $(3, 2, 1)$ seems nearer than $(3, 1, 2)$ to $(1, 2, 3)$. Observe that in this example the constraints that define Ω^d (i.e., the set of permutations) are modeled through the definition of the set of neighbors $\mathcal{N}$.

Thus, definition of the set of neighbors $\mathcal{N}$ represents one of the tuning knobs available to the user willing to pay more for a guarantee of a stronger local optimizer. As our thermal example shows, this does not guarantee finding a lower function value, but it does guarantee a wider set of changes that will not produce a better function value. A better way to use this knob is the way we used it in the thermal example—to save evaluating alternatives that are highly unlikely to improve the function value, and thus decrease the cost of the more expensive poll steps in which local exploration is required.

For mixed variable programming, the definition of local optimality must take into account variations of both the continuous and discrete variables. Indeed, in defining $\mathcal{N}(x)$, one would probably need to allow for changes in the continuous as well as the discrete components. That is to say, changing the discrete variables may make no sense without some attendant change in the continuous components as well. We propose the following definition.

DEFINITION 2.1. *The solution $\hat{x} = (\hat{x}^c, \hat{x}^d) \in \Omega$ is said to be a local minimizer of*

f with respect to the set of neighbors $\mathcal{N}(\hat{x})$ if there exists an $\epsilon > 0$ such that

$$f(\hat{x}) \le f(v) \quad \text{for any} \quad v \in \bigcup_{y \in \mathcal{N}(\hat{x})} \left(([\ell, u] \cap B(\epsilon, y^c)) \times y^d \right),$$

where $\mathcal{N}(\hat{x}) \subset \Omega$ is a finite set of points.

We require a notion of continuity with respect to the set of neighbors: If $\{x_k\}$ is a sequence that converges to $\hat{x}$, then $\mathcal{N}(x_k)$ converges to $\mathcal{N}(\hat{x})$; i.e., for any $\epsilon > 0$ and $\hat{y}$ in the set of neighbors $\mathcal{N}(\hat{x})$, there exists a y_k in $\mathcal{N}(x_k)$ that also belongs to the ball $B(\epsilon, \hat{y})$.

This definition of local optimality requires the user to decide how to define the neighbors, and then we produce a point at which we guarantee that there are no better solutions than $\hat{x}$ in any of the balls (in the continuous space and intersected with the box $[\ell, u]$) around the points in the user-defined set of neighbors. Observe that when there are no discrete variables, or else no continuous ones, this definition reduces to the appropriate one presented above.

Of course, one can generally prove only that an optimization algorithm converges to a point satisfying some necessary conditions for optimality. Thus, we prove that our algorithm produces a limit point $\hat{x}$ that satisfies

$$(x^c - \hat{x}^c)^T \nabla^c f(\hat{x}) \ge 0 \qquad \text{for any feasible } (x^c, \hat{x}^d) \tag{2.1}$$

(where $\nabla^c f(x) \in \Re^{n^c}$ denotes the gradient of f with respect to the continuous variables x^c while keeping the discrete x^d fixed), and for any $\hat{y} \in \Omega$ in the set of neighbors $\mathcal{N}(\hat{x})$

$$f(\hat{x}) \le f(\hat{y}). \tag{2.2}$$

In the cases where $f(\hat{y}) < f(\hat{x}) + \xi$ (for a specified $\xi > 0$), there exists a point $\hat{z} \in \Omega$, whose discrete components $\hat{z}^d$ are identical to $\hat{y}^d$, that satisfies $f(\hat{x}) \le f(\hat{z}) \le f(\hat{y})$ and

$$(z^c - \hat{z}^c)^T \nabla^c f(\hat{z}) \ge 0 \qquad \text{for any feasible } (z^c, \hat{z}^d). \tag{2.3}$$

Furthermore, in the cases where $f(\hat{x}) = f(\hat{y})$ and $\hat{y} \ne \hat{z}$,

$$f(\hat{x}) = f(\bar{y}) \tag{2.4}$$

for an infinite number of intermediate points $\bar{y} \in \Omega$ between $\hat{y}$ and $\hat{z}$ (we show in section 4.2 how to construct these intermediate points). Moreover, we present a stronger version of the algorithm that guarantees

$$(y^c - \bar{y}^c)^T \nabla^c f(\bar{y}) \ge 0 \qquad \text{for any feasible } (y^c, \bar{y}^d) \tag{2.5}$$

whenever $f(\hat{x}) = f(\hat{y})$.

2.2. An illustrative application. We illustrate our approach on a thermal insulation system. The problem is thoroughly described in [8], where we show that by considering the two categorical variables we obtain a 65% better objective function value than in the earlier work of Hilal and Boom [6], who considered only the continuous variables.

The setting of the problem is as follows. One wishes to control the heat flow from a hot to a cold surface by inserting some shields (heat intercepts) between them.

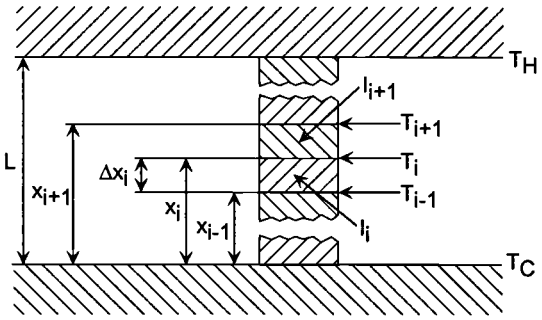


FIG. 2.1. *Schematic of a general thermal insulation system.*

Each shield is kept at a fixed temperature, and the spaces between them are filled by various insulators. The objective is to minimize the power f (with an extra cost for each additional shield) required to keep the shields at their temperatures. This is illustrated in Figure 2.1.

The temperatures of the hot T_H and cold T_C surfaces are given. The decision variables are the number of shields n, their temperatures $T \in \Re^n$, the spacing $\Delta x \in \Re^{n+1}$ between them (or equivalently, the thickness of the insulators), and the types of insulators I between the shields. These last variables are taken from a finite list $\mathcal{I}$ of insulators whose thermal conductivity properties are known. The optimization problem may be formulated as

$$\begin{array}{lll} \min\limits_{\Delta x,T,n,I} & f(\Delta x, T, n, I) & \\ \text{subject to} & \Delta x \geq 0, \quad T_C \leq T_i \leq T_H, & i = 1, 2, \ldots, n, \\ & n \in \mathcal{Z}^+, \quad I_j \in \mathcal{I}, & j = 1, 2, \ldots, n+1. \end{array}$$

The continuous variables are Δx and T, and the categorical ones are n and I. An interesting and challenging aspect of this problem is that the number of decision variables depends on a decision variable n. This does not complicate the theory.

Section 5.2 contains numerical results for this problem with local optimality defined through sets of neighbors $\mathcal{N}$ as follows:

– changing the type of one insulator;
– removing one shield and an adjacent insulator;
– adding a shield and an insulator.

3. Pattern search methods. The underlying structure of a pattern search algorithm is as follows. It is an iterative method that generates a sequence of feasible iterates whose objective function value is nonincreasing. At any given iteration, the objective function is evaluated at a finite number of points on a mesh in order to try to find one that yields a decrease in the objective function value.

Any iteration k of a pattern search method is initiated with the *incumbent solution* x_k, as well as with an enumerable subset $\mathcal{M}_k$ of the domain $\Omega \subset \Re^{n^c} \times \mathcal{Z}^{n^d}$. Construction of the mesh $\mathcal{M}_k$ is formally described in section 3.1, and its fineness, or resolution, is parameterized by a positive real number Δ_k. The goal of each iteration is to obtain a new incumbent solution on the current mesh whose objective function value is strictly less (by any amount) than the old incumbent.

Exploration of the mesh is conducted in one or two phases. First, a finite search, free of any other rules imposed by the algorithm, is performed anywhere on the mesh. Any strategy can be used as long as it searches finitely many points (including none). This part of the algorithm has the advantage that the user can put in place any ad hoc search he/she might favor for improving the incumbent with the knowledge that if this fails, the next phase will provide a fail-safe.

If the search does not succeed in improving the incumbent, the second phase is called. A potentially exhaustive (but always finite) search in a local mesh neighborhood around x_k and around promising points in its set of neighbors is performed. The first phase (called the SEARCH step) provides flexibility to the method and determines in practice the global quality of the solution; a user can do a more extensive, and expensive, search in hopes of finding a better local solution. The second phase (called the POLL step) follows stricter rules and guarantees theoretical convergence to a local minimizer of a quality specified by the user. The set of points visited by this phase is referred to as the *poll set*. Rules for constructing this set are detailed in section 3.2.

If a point with a better objective value than x_k is found in either phase, then the iteration is declared *successful*, the better point becomes the new incumbent, and the next iteration is initiated with a (possibly) coarser (and different) mesh around the new incumbent solution.

Otherwise, the iteration is declared *unsuccessful*. The next iteration is initiated at the same incumbent solution, but with a finer mesh on the continuous variables, and a set of neighbors "closer" (if possible) to the incumbent solution. A key property of the mesh exploration is that if an iteration is unsuccessful, then the current objective function value is less than or equal to the objective function values evaluated at all points in the trial set consisting of all points considered in the search and poll set.

In order to properly present the pattern search algorithm, we first detail in the following sections the construction of the mesh and the poll set.

3.1. The mesh. At any given iteration k, the *current mesh* $\mathcal{M}_k$ is a discrete set of points in Ω from which the algorithm selects the next iterate. The mesh is conceptual; it is not actually constructed. The coarseness or fineness of the mesh is dictated by the strictly positive *mesh size parameter* $\Delta_k \in \Re_+$. Both the mesh and mesh size parameter are updated at every iteration.

The mesh is the direct product of the union of a finite number of lattices in $\Re^{n^c}$ with the integer space $\mathcal{Z}^{n^d}$. Our presentation of the lattices differs from that of Torczon [12], but the sets produced are equivalent. Consider the *basis matrix* $G^B \in \Re^{n^c \times n^c}$, and for j varying from 1 to $j_{max} < \infty$, consider the *generating matrices* $G_j \in \mathcal{Z}^{n^c \times n^c}$; then define the *pattern matrices* $P_j \in \Re^{n^c \times n^c}$ to be the products $G^B G_j$. The continuous variables are chosen from one of the translated (by x_k^c) lattices $\left\{x_k^c + \Delta_k P_j z : z \in Z^{n^c}\right\}$ for $j = 1, 2, \ldots, j_{max}$. The continuous part x_k^c of the current iterate belongs to each of the j_{max} lattices regardless of the value of the parameter Δ_k. The basis matrix G^B is constant over all iterations. However, in practice the generating matrices G_j (and thus P_j) that define the lattices can be determined as the iteration unfolds as long as only a finite number of them is generated.

Each of these lattices is enumerable, and the minimum distance between two distinct points is proportional to the mesh size parameter Δ_k. When an iteration is successful, the continuous part of the next iterate is chosen in any of these lattices and thus belongs to their union $\mathcal{M}(\Delta_k) = \bigcup_{j=1}^{j_{max}} \left\{x_k^c + \Delta_k P_j z : z \in Z^{n^c}\right\}$; the discrete part is chosen in the integer lattice Z^{n^d}.

At iteration k, the current mesh is defined to be the direct product of $\mathcal{M}(\Delta_k) \cap \Omega^c$

by Ω^d

$$\mathcal{M}_k = \left(\mathcal{M}(\Delta_k) \times \mathcal{Z}^{n^d}\right) \cap \Omega.$$

The mesh is completely defined by the current iterate x_k and the mesh size parameter Δ_k. Whether or not the iteration is successful, the next iterate x_{k+1} is always selected in the current mesh $\mathcal{M}_k$.

In the case where the SEARCH step in the current mesh is unsuccessful, a second exploration phase must be conducted by the algorithm in the poll set before the iteration is declared *unsuccessful*. The POLL step verifies whether the incumbent solution is a local mesh minimizer, as defined in the next section.

3.2. The poll set. Polling occurs when the SEARCH step was unable to obtain a point on the current mesh that decreased the incumbent value. Polling is conducted as follows in up to three stages (not necessarily in this order):

– polling with respect to the continuous variables;
– polling on the current set of neighbors $\mathcal{N}$;
– extended polling (in the case where $f(y)$ for some y in the set of neighbors is close to the incumbent value).

The first stage is identical to the typical polling in pattern search algorithms for continuous variables only. The second one is the natural generalization to the discrete variables using the set of neighbors. We introduced the last one to explore around some promising points in the set of neighbors and strengthen the optimality conditions achieved by the limit points.

Polling with respect to the continuous variables requires the use of positive bases, or at least positive spanning sets, on $\Re^{n^c}$. A positive basis is a set of nonzero vectors in $\Re^{n^c}$ whose nonnegative linear combinations span $\Re^{n^c}$, but no proper subset does so. Each positive basis contains at least $n^c + 1$ and at most $2n^c$ vectors. These are referred to as minimal and maximal positive bases (see Davis [4] for characterization of positive bases). We use the following key property of positive spanning sets. For any nonzero vector a in $\Re^{n^c}$ and positive spanning set B on $\Re^{n^c}$, there exists a vector b of B such that

$$a^T b < 0. \tag{3.1}$$

Let $\mathcal{B}$ be a finite set of positive spanning sets on $\Re^{n^c}$ such that every column b of every positive spanning set of $\mathcal{B}$ is of the form $P_j z$ for some $z \in Z^{n^c}$ and $1 \le j \le j_{max}$. The pattern matrices P_j are the same ones used to construct the lattices in section 3.1. In a way similar to [9], we assume that at least one positive spanning set of $\mathcal{B}$ is a maximal positive basis whose columns may be partitioned in a way to form two nonsingular diagonal $n^c \times n^c$ matrices. Let $\overline{\mathcal{B}} \subset \mathcal{B}$ be the set of all these bases. Conceptually, the set $\mathcal{B}$ is fixed throughout all iterations, but it may evolve as the solution process proceeds as long as it remains finite.

The poll points with respect to the continuous variables are obtained by scaling a basis $\mathcal{B}$ by the mesh size parameter as follows: At iteration k, for any mesh point x, define $\mathcal{N}^c(x)$, the *mesh neighborhood of the continuous variables around* x, to be

$$\mathcal{N}^c(x) = \{x + \Delta_k(b, 0) \in \Omega : b \in B_k(x)\} \tag{3.2}$$

for some positive spanning set $B_k(x) \in \mathcal{B}$ that depends on both the iteration number k and the point x. Moreover, in order to avoid the infeasibility problem described

in [9], we require that if one of the components of the current iterate x_k^c is within a tolerance parameter $\varepsilon > 0$ of either its lower or upper bound, then the positive spanning set for this iteration must be chosen in $\overline{\mathcal{B}}$.

This definition ensures that the mesh neighborhood $\mathcal{N}^c(x_k)$ is a subset of the current mesh $\mathcal{M}_k$. Moreover, $\mathcal{N}^c(x_k)$ is constructed using a single positive spanning set chosen from a finite set, and thus there are only a finite number of possible ways to define mesh neighborhoods.

The motivation for introducing positive spanning sets for the continuous variables is that if the gradient $\nabla^c f$ of the function f with respect to the continuous variables is nonzero, then at least one vector of the set defines a descent direction. The original work of Torczon [12] uses a maximal positive basis for unconstrained optimization. It was later generalized in Lewis and Torczon [10] to any positive basis, thus reducing the maximum number of points in the polling set from $2n^c$ to $n^c + 1$. However, for bound constrained optimization, they show that taking the maximal positive bases generated by the coordinate directions guarantees finding a feasible descent direction (if there is one) even on the boundary of the feasible region.

The discrete stage of the POLL step depends on the set of neighbors defined by the user (as in section 2.1). In order to allow for varying of the definition of the set of neighbors for a finite number of iterations, we define the *current set of neighbors* $\mathcal{N}_k$ to be such that $\mathcal{N}_k$ differs from $\mathcal{N}(x_k)$ at most at a finite number of iterations k. This flexibility allows finitely many redefinitions of $\mathcal{N}_k$ to allow the user another knob to adjust the cost of a POLL step (see section 3.3) and the likely quality of the limit point.

If none of the above-mentioned polling points (i.e., those in the mesh neighborhood $\mathcal{N}^c(x_k)$ and in the set of neighbors $\mathcal{N}_k$) yields decrease in the objective function value, a third stage might be required before declaring the iteration unsuccessful. This stage is triggered by the last of our user controlled knobs $\xi > 0$ to pay more for a most likely better final function value. An EXTENDED POLL step must be conducted around each point of the set of neighbors $\mathcal{N}_k$ of x_k at which the function value, even though it is larger, is within ξ of $f(x_k)$. Intuitively, ξ represents a tolerance which is such that if a discrete neighbor y in $\mathcal{N}_k$ provides such a near function value, then the user wishes us to poll in the continuous variables around y since this may produce a new best solution. Our convergence analysis is independent of the value of ξ, but intuitively a larger ξ means extended polling will be carried out at more iterations, which may cost more function evaluations, but should give a better local minimizer. Of course, it would be simple to construct examples showing the opposite behavior, but our thermal example shows how this can work.

More precisely, consider any point y in the set of neighbors $\mathcal{N}_k$. (The variable y should be indexed with the iteration number k and with respect to the set of neighbors $\mathcal{N}_k$, but this would obscure the notation.) In the case where $f(y) > f(x_k) + \xi$ or $f(y) \leq f(v)$ for all v in $\mathcal{N}^c(y)$, the POLL step need not be extended and so we set the index J to 0. In all other cases, y^0 is set to y and for $j \geq 1$ we select the feasible point y^j in the mesh neighborhood $\mathcal{N}^c(y^{j-1})$ iteratively so that $f(y^j) < f(y^{j-1})$ until it is no longer possible (or until $f(y^j) < f(x_k)$ in which case iteration k is successful and x_{k+1} is set to y^j). It follows that the last point (whose index is denoted by J) satisfies $f(y^J) \leq f(v)$ for all v in $\mathcal{N}^c(y^J)$. Define z to be the endpoint y^J of the EXTENDED POLL step. Keep in mind that z depends on the iteration number k and on the neighbor y in $\mathcal{N}_k$. These trial points are illustrated in Figure 3.1, where they are indexed with the iteration number.

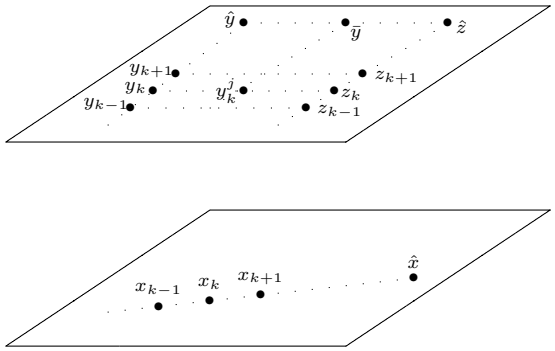


FIG. 3.1. *Limit points of the iterates and the extended poll points.*

With this construction, the function values $f(y) = f(y^0), f(y^1), \ldots, f(y^J) = f(z)$ are monotonically decreasing unless $y = z$. Only at the endpoint z is the function required to be evaluated at every point of its mesh neighborhood $\mathcal{N}^c(z)$. Observe that the index J may be 0, in which case $y = z$. This happens either when $f(y) > f(x_k)+\xi$ or when $f(y) \leq f(v)$ for all v in $\mathcal{N}^c(y)$. The index J is finite since all generated points y^j are distinct and belong to the mesh $\mathcal{M}_k$ intersected with the compact level set $L_\Omega(x_0)$ (see assumption (A1) in section 3.3).

The set of all points visited by the POLL step at iteration k is denoted X_k^ξ and may be explicitly written as

$$X_k^\xi = \mathcal{N}^c(x_k) \bigcup \mathcal{N}_k \bigcup_{\substack{y \in \mathcal{N}_k: \\ f(x_k) \leq f(y) \leq f(x_k)+\xi}} \mathcal{E}(y),$$

where $\mathcal{E}(y)$ is the extended poll set, which contains $\{y^1, y^2, \ldots, y^J\} \cup \mathcal{N}^c(z)$ as well as some points of $\mathcal{N}(y^j)$ for some j's in $\{0, 1, \ldots, J-1\}$.

Figure 3.2 illustrates an instance in which there are two continuous variables and one discrete variable. The set of neighbors of the iterate x_k is assumed to be $\mathcal{N}_k = \mathcal{N}(x_k) = \{x_k, y_1^0, y_2^0\}$, where the subscripts 1 and 2 are added to distinguish the points in the set of neighbors $\mathcal{N}(x_k)$. (Note that the points in $\mathcal{N}_k$ do not have the same values for the continuous variables.) The iterate x_k is a local minimizer of the function f on X_k^ξ if $f(x_k)$ is less than or equal to the function value evaluated at all points in balls around x_k, y_1^0, and y_2^0. The letters a to l in the figure represent mesh neighborhoods of the continuous variables

$$\mathcal{N}^c(y_1^0) = \{d, e, y_1^1\},\ \mathcal{N}^c(y_1^1) = \{a, b, c\},\ \mathcal{N}^c(x_k) = \{f, g, h\},\ \text{and } \mathcal{N}^c(y_2^0) = \{i, j, k, l\}.$$

In this example, since $f(x_k) \leq f(y_1^0) < f(x_k) + \xi < f(y_2^0)$ the poll set X_k^ξ contains points in $\mathcal{N}^c(x_k)$ and $\mathcal{N}^c(y_1^0)$ (among others). Assuming that $f(y_1^1) < f(y_1^0)$ but $f(a) \geq f(y_1^1)$, $f(b) \geq f(y_1^1)$ and $f(c) \geq f(y_1^1)$ lead to the poll set $X_k^\xi = \{f, g, h\} \cup \{x_k, y_1^0, y_2^0\} \cup \mathcal{E}(y)$, where $\mathcal{E}(y) = \{y_1^1\} \cup \{a, b, c\}$. Note that, depending on the order in which the function values are evaluated, it is possible that the extended poll set also contains d or e.

Using the above notation, we can now present the GMVP algorithm.

3.3. The GMVP search algorithm. Our presentation of the pattern search algorithm is closer to that of Booker et al. [2] than to that of Torczon [12]. Consider

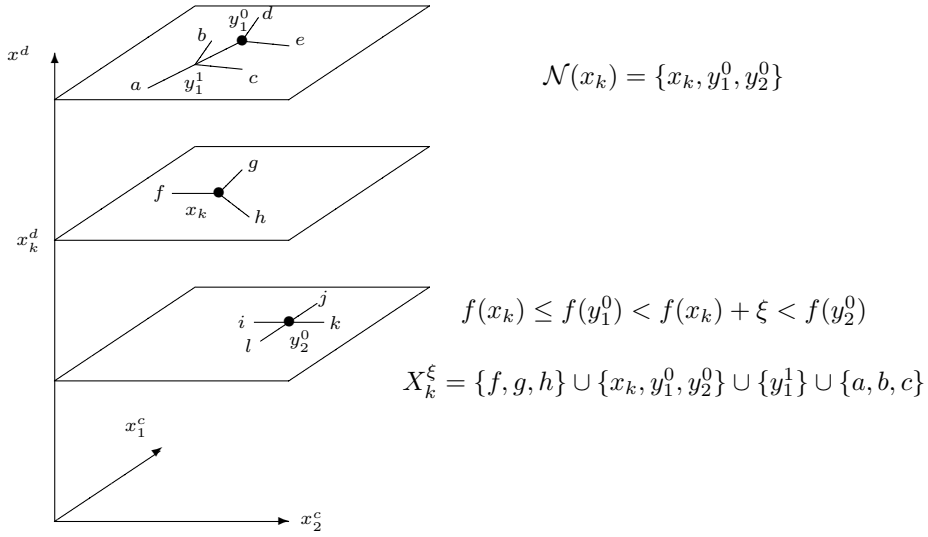


FIG. 3.2. *Construction of the current mesh neighborhood X_k^ξ around x_k.*

the given initial mesh $M_0 \subset \Omega$ with mesh size parameter Δ_0 and initial point x_0 of M_0. Also, let $\xi > 0$ be the objective function change tolerance used to trigger extended polling in the construction of the poll set. Recall that if $f(x_k) \le f(y) \le f(x_k) + \xi$ for some y in the set of neighbors $\mathcal{N}_k$, then the polling step must be extended around y.

Throughout the paper, the following assumptions are made:

(A1) The level set $L_\Omega(x_0) = \{x \in \Omega : f(x) \le f(x_0) + \xi\}$ is compact.

(A2) f is continuously differentiable over a neighborhood of $L_\Omega(x_0)$ when variables in $\mathcal{Z}^{n^d}$ are fixed, i.e., for any $x^d \in \Omega^d$ the function $x^c \mapsto f(x^c, x^d)$ is continuously differentiable in a neighborhood of $\{x^c : (x^c, x^d) \in L_\Omega(x_0)\}$.

At any iteration $k \ge 0$, the general rules for choosing x_{k+1} in the current mesh $\mathcal{M}_k$ and obtaining the next mesh size parameter Δ_{k+1} are as follows.

GENERALIZED MIXED VARIABLE PATTERN SEARCH ALGORITHM (GMVP).

1. SEARCH **step (in current mesh).** Employ some finite strategy to obtain an $x_{k+1} \in \mathcal{M}_k$ satisfying $f(x_{k+1}) < f(x_k)$. If such an x_{k+1} is found, declare the SEARCH step (as well as the iteration) successful, then expand the mesh at step 3.

2. POLL **step.** This step is reached only if the SEARCH step is unsuccessful. If $f(x_k) \le f(x)$ for every x in the poll set X_k^ξ, then declare the POLL step (as well as the iteration) unsuccessful and shrink the mesh at step 4. Otherwise, choose $x_{k+1} \in X_k^\xi$ to be a point such that $f(x_{k+1}) < f(x_k)$, declare the POLL step (as well as the iteration) successful, and expand the mesh at step 3.

3. Mesh expansion (at successful iterations). Let $\Delta_{k+1} = \tau^{m_k^+} \Delta_k$ (for $\tau^{m_k^+} \ge 1$ defined below). Increase k, and initiate the next iteration at step 1.

4. Mesh reduction (at unsuccessful iterations). Set x_{k+1} to x_k and let $\Delta_{k+1} = \tau^{m_k^-} \Delta_k$ (for $0 < \tau^{m_k^-} < 1$ defined below). Increase k, and initiate the next iteration at step 1. □

In the SEARCH and POLL steps, the number of candidate points among which the next iterate can be chosen is finite, since they must belong to the intersection of the enumerable current mesh and the compact set $L_\Omega(x_0)$.

The parameters in the last two steps are the rational number $\tau > 1$ and the

integers (whose absolute values are bounded above by a constant $m_{max} \geq 1$) $m_k^+ \geq 0$ and $m_k^- \leq -1$. In [12], the mesh reduction parameter m_k^- was fixed for all $k \geq 0$. This restriction is relaxed here without affecting the convergence results. We plan to exploit this flexibility in subsequent work to increase the practical convergence speed.

The conditions on these parameters imply the *simple decrease property* used throughout the growing literature on GPS methods: iteration k is successful if and only if $f(x_{k+1}) < f(x_k)$, if and only if $\Delta_{k+1} \geq \Delta_k$, and if and only if $x_{k+1} \neq x_k$. Another important implication of the parameters' definition is that if the iteration k is unsuccessful, then $f(x_k) \leq f(v)$ for all $v \in X_k^\xi$ and thus $f(x_k) \leq f(y)$ for all $y \in \mathcal{N}^c(x_k)$ and, whenever $f(y) \leq f(x_k) + \xi$ for some $y \in \mathcal{N}_k$, then $f(z) \leq f(v)$ for all $v \in \mathcal{N}^c(z)$, where z is the end point of the extended poll step initiated at y. Moreover, Δ_{k+1} is obtained by multiplying Δ_k by a finite positive or negative integer power of τ. Therefore, for any $k \geq 0$, we can write

$$\Delta_k = \Delta_0 \tau^{r_k} \tag{3.3}$$

for some r_k belonging to $\mathcal{Z}$.

Notice that the cost of the POLL step is expected to depend on both ξ and the definition of the set of neighbors $\mathcal{N}$. Thus, the user can pay more function evaluations for a stronger local solution by defining a larger ξ or a larger neighborhood $\mathcal{N}$. Another way that the user can likely improve the quality of the solution is through the SEARCH step. In that step, knowledge of the problem and/or favorite heuristics can be used to improve the solution. The user can also try to evaluate the function at various places in the variable space and design interpolary models or use surrogate functions (as discussed in Booker et al. [2]). The search strategy, which aims at finding the best solution on the current mesh, can be as sophisticated as one wants, but may increase the number of function evaluations.

4. Proof of convergence. This section contains the convergence proof for the GMVP algorithm. We start by studying the behavior of the mesh size parameter Δ_k. The first important result, due to Torczon for the continuous case, is $\liminf_{k \to +\infty} \Delta_k = 0$ which implies the existence of a subsequence of mesh size parameters that converges to zero. A key to our simpler proof is to conclude from this that there is an infinite number of unsuccessful iterations. We analyze converging subsequences of unsuccessful iterates whose mesh size parameters converge to zero. We show that any limit point of such a subsequence satisfies the optimality conditions (2.1)–(2.4). By focusing on unsuccessful iterations, the result for the continuous variables is shown using a much shorter proof than in [12] and [9]. We also present a stronger version of the algorithm that yields a stronger result, i.e., the optimality condition (2.5).

Thus, when we consider the analysis of this class of algorithms, it is the sequences of unsuccessful iterates that we show converge. The terminology *successful/unsuccessful* that made perfect sense in explaining the algorithm suddenly jars us because of the pejorative connotation of the word "unsuccessful." In fact, an iteration is unsuccessful because the corresponding iterate is a local mesh minimizer, and so the discrete resolution of the domain must be refined by reducing Δ_k before we can expect to proceed downhill. Likewise, an iteration is successful because it moves us toward finding a local mesh minimizer. Thus, in a sense, *successful/unsuccessful* could be replaced by *inner/outer* or *minor/major* as labels for the two types of iterations. We hesitate to suggest such a change to well-established terminology too quickly, but we hope this short warning discussion will alleviate the confusion readers have expressed to us.

4.1. Behavior of the mesh size. To show that there is a subsequence of mesh size parameters Δ_k that converges to zero, we first show that these parameters are bounded above by a constant, independent of the iteration number k.

LEMMA 4.1. *There exists a positive integer r^u such that $\Delta_k \leq \Delta_0 \tau^{r^u}$ for any $k \geq 0$.*

Proof. Let Δ be a mesh size parameter large enough so that the union of lattices $\mathcal{M}(\Delta)$ intersects the compact feasible level set $\{x^c : x \in L_\Omega(x_0)\}$ only at the translation parameter x_k^c; i.e., for any $1 \leq j \leq j_{max}, z \in \mathcal{Z}^{n^c}$, and x in $L_\Omega(x_0)$ the solution $x^c + \Delta P_j z$ does not belong to the projection of $L_\Omega(x_0)$ on the continuous variables space unless it equals x_k^c. Therefore, if at iteration k the mesh size parameter Δ_k is greater than or equal to Δ, then

$$\mathcal{M}_k \cap L_\Omega(x_0) \subset \{x_k^c\} \times \Omega^d.$$

Moreover, only a finite number of iterations will follow before the mesh size parameter drops below Δ. Indeed, the continuous part of all these iterates will necessarily be equal to x_k^c, and the discrete part of these iterates can take only a finite number of values because the set $L_\Omega(x_0)$ is bounded. Let d_{max} be the total number of distinct values that the discrete variables may take in $L_\Omega(x_0)$. Therefore, there will be no more than d_{max} successful iterations before the mesh size parameter goes below Δ.

Recall that the expansion mesh size control parameter is bounded above by $\tau^{m_{max}}$. Let r^u be a large enough integer so that $\Delta_0 \tau^{r^u} \geq \Delta(\tau^{m_{max}})^{d_{max}}$. It follows that the mesh size parameter at any iteration will never exceed $\Delta_0 \tau^{r^u}$. ∎

We now study the convergence behavior of the mesh size parameter. The proof of this result is essentially identical to that of Torczon [12] despite the presence of discrete variables.

THEOREM 4.2. *The mesh size parameters satisfy* $\liminf_{k \to +\infty} \Delta_k = 0$.

Proof. Suppose by way of contradiction that there exists a negative integer r^ℓ such that $0 < \Delta_0 \tau^{r^\ell} \leq \Delta_k$ for all $k \geq 0$. Equation (3.3) states that for every $k \geq 0$ there is an $r_k \in \mathcal{Z}$ such that $\Delta_k = \tau^{r_k} \Delta_0$. Combining this with Lemma 4.1 implies that for any $k \geq 0$, r_k takes its value among the integers of the bounded interval $[r^\ell, r^u]$. Therefore, r_k and Δ_k can take only a finite number of values for all $k \geq 0$.

For any k, the continuous part of the next iterate x_{k+1}^c belongs to a lattice and can be written $x_k^c + \Delta_k P_{j_k} z_k$ for some $z_k \in \mathcal{Z}^{n^c}$ and $1 \leq j_k \leq j_{max}$. By substituting $\Delta_k = \Delta_0 \tau^{r_k}$ and $P_j = G^B G_{j_k}$, it follows that for any integer N

$$\begin{aligned} x_N^c &= x_0^c + \sum_{k=1}^{N-1} \Delta_k P_{j_k} z_k \\ &= x_0^c + \Delta_0 G^B \sum_{k=1}^{N-1} \tau^{r_k} G_{j_k} z_k \;=\; x_0^c + \frac{p^{r^\ell}}{q^{r^u}} \Delta_0 G^B \sum_{k=1}^{N-1} p^{r_k - r^\ell} q^{r^u - r_k} G_{j_k} z_k, \end{aligned}$$

where p and q are relatively prime integers satisfying $\tau = \frac{p}{q}$.

Since for any k the term $p^{r_k - r^\ell} q^{r^u - r_k} G_{j_k} z_k$ appearing in this last sum is an integer, it follows that the continuous part of all iterates lies on the translated integer lattice generated by x_0^c and the columns of $\frac{p^{r^\ell}}{q^{r^u}} \Delta_0 G^B$. Moreover, the discrete part of all iterates also lies on the integer lattice $\mathcal{Z}^{n^d}$.

Therefore, since all iterates belong to the compact set $L_\Omega(x_0)$, it follows that there is only a finite number of different iterates, and thus one of them must be

visited infinitely many times. Simple decrease ensures that the mesh size parameters converge to zero, which is a contradiction. ∎

4.2. The main results. Lewis and Torczon [9] show that condition (1.2) holds; i.e., there exists a limit point $\hat{x}$ of the sequence of iterates for which $s^T \nabla f(\hat{x}) \geq 0$ for any feasible direction s. Through a shorter proof, we show a stronger result. We show the existence of a limit point $\hat{x}$ of the sequence of *unsuccessful* iterates that satisfies (2.1) and is a local optimizer with respect to the set of neighbors $\mathcal{N}(\hat{x})$ in the sense of conditions (2.2), (2.3), and (2.4). Recall that iteration k is unsuccessful if and only if $x_{k+1} = x_k$, which is equivalent to $\Delta_{k+1} < \Delta_k$. Thus, the number of unsuccessful iterations is infinite since $\liminf_{k \to +\infty} \Delta_k = 0$ by Theorem 4.2.

Consider the indices of the unsuccessful iterations whose corresponding mesh size parameters go to zero. For any limit point of such a subsequence, there is an iterate x_k arbitrarily close to it for which no trial point of the poll set X_k^{ξ} yields a decrease in the objective function value. The following result details properties of a limit point $\hat{x}$ of the subsequence of unsuccessful iterations whose mesh size parameters converge to zero (Figure 3.1 depicts this result).

PROPOSITION 4.3. *There is a point $\hat{x} \in L_\Omega(x_0)$ and a subset of indices of unsuccessful iterates $K \subset \{k : x_{k+1} = x_k\}$ such that*

$$\lim_{k \in K} \Delta_k = 0, \qquad \lim_{k \in K} x_k = \hat{x}, \qquad \textit{and} \qquad \mathcal{N}_k = \mathcal{N}(x_k) \ \forall k \in K.$$

Moreover, if $\hat{y}$ belongs to the set of neighbors $\mathcal{N}(\hat{x})$, then there exists a $\hat{z} = (\hat{z}^c, \hat{y}^d) \in \Omega$ such that

$$\lim_{k \in K} y_k = \hat{y} \qquad \textit{and} \qquad \lim_{k \in K} z_k = \hat{z},$$

where $z_k \in \Omega$ is the endpoint of the EXTENDED POLL *step initiated at $y_k \in \mathcal{N}(x_k)$ at iteration $k \in K$.*

Proof. Theorem 4.2 guarantees that $\liminf_{k \to +\infty} \Delta_k = 0$; thus there is an infinite subset of indices of unsuccessful iterations $K' \subset \{k : x_{k+1} = x_k\} = \{k : \Delta_{k+1} < \Delta_k\}$ such that the subsequence $\{\Delta_k\}_{k \in K'}$ converges to zero.

Since all iterates x_k lie in the compact set $L_\Omega(x_0)$, we can extract an infinite subset of indices $K'' \subset K'$ such that the subsequence $\{x_k\}_{k \in K''}$ converges. Let $\hat{x}$ in $L_\Omega(x_0)$ be the limit point of such a subsequence. Moreover, since $\mathcal{N}_k$ differs from $\mathcal{N}(x_k)$ at most at a finite number of iterates, we may assume, without any loss of generality, that $x_k^d = \hat{x}^d$ for all $x_k \in K''$.

Let $\hat{y} \in \Omega$ be a point of the set of neighbors $\mathcal{N}(\hat{x})$. Recall that we assumed in section 2.1 a notion of continuity of the sets of neighbors. Therefore, $\hat{y}$ is a limit point of a subsequence $y_k \in \mathcal{N}_k$. Let $\hat{z} \in \Omega$ be a limit point of the sequence $z_k \in \Omega$ of endpoints of the EXTENDED POLL step initiated at y_k. By definition, the endpoint z_k is equal to y_k in the case that the EXTENDED POLL step is not required.

Choose $K \subset K''$ to be such that both $\{y_k\}_{k \in K}$ converges to $\hat{y}$ and $\{z_k\}_{k \in K}$ is convergent (let $\hat{z}$ denote the limit point). ∎

Torczon [12] observes that setting the mesh size increase parameter m_k^+ to zero (in the mesh expansion step of the GPS algorithm) ensures that $\lim_{k \to \infty} \Delta_k = 0$. Thus the mesh is never expanded: at unsuccessful iterations, the mesh size parameter Δ_{k+1} is set to be equal to Δ_k. The same observation holds for our algorithm. It follows that in this case, all the convergence results below hold for *every* limit point of the sequence of unsuccessful iterates.

For the rest of this section, we assume that $\hat{x}$ and K satisfy the conditions of Proposition 4.3. The main results can now be proved. We first show that $\hat{x}$ is a local optimal solution with respect to the set of neighbors $\mathcal{N}(\hat{x}) \subset \Omega$.

THEOREM 4.4. *The limit point $\hat{x}$ satisfies $f(\hat{x}) \leq f(\hat{y})$ for all $\hat{y} \in \mathcal{N}(\hat{x})$.*

Proof. Suppose by way of contradiction that there is a $\hat{y} \in \mathcal{N}(\hat{x})$ such that $f(\hat{x}) > f(\hat{y})$. Continuity of the function f with respect to the continuous variables guarantees the existence of an $\epsilon > 0$ such that if v belongs to the ball $B(\epsilon, \hat{y})$ of radius ϵ centered at $\hat{y}$, then $f(v) < f(\hat{x})$.

Proposition 4.3 guarantees that the subsequences $\{x_k\}_{k\in K}$ and $\{y_k\}_{k\in K}$ (where $y_k \in \mathcal{N}(x_k)$), respectively, converge to $\hat{x}$ and $\hat{y}$. We required in section 2.1 that the set $\mathcal{N}(x_k)$ converges for $k \in K$ to $\mathcal{N}(\hat{x})$ in the sense that if $k \in K$ is large enough, then there exists a $y_k \in \mathcal{N}(x_k)$ such that $y_k \in B(\epsilon, \hat{y})$.

Therefore, there exists an iteration $k \in K$ such that y_k belongs to $\mathcal{N}_k \cap B(\epsilon, \hat{y})$ and satisfies $f(y_k) < f(\hat{x}) \leq f(x_k)$. It follows that the iteration is successful, contradicting the fact that k belongs to $K \subset \{k : x_{k+1} = x_k\}$. ∎

In the case where the inequality in Theorem 4.4 is strict, i.e., $f(\hat{x}) < f(\hat{y})$, then the notion of local optimality for mixed integer programming presented in section 2.1 is verified: There exists an $\epsilon > 0$ such that $f(\hat{x}) \leq f(v)$ for any v in a ball of radius ϵ around $\hat{y}$. This follows from the continuity of the function f with respect to the continuous variables.

Next, we study the gradient of the function f with respect to the continuous variables at the limit point $\hat{x}$. The proof of Theorem 4.7 for the continuous case is much shorter than the original one of Torczon [12]. Its proof, as well as that for the EXTENDED POLL step (Theorem 4.8), relies on the following two lemmas.

The first lemma shows that the gradient is zero in the strict interior of the boundary of the feasible region. These results concern points around which polling is unsuccessful (i.e., p_k in the lemmas will take the value of x_k or z_k).

LEMMA 4.5. *Let $\{p_k\}_{k\in K}$ be a subsequence of unsuccessful poll points and let $\hat{p}$ be a limit point. If the continuous part of the limit point $\hat{p}$ is in the strict interior of the feasible region $[\ell, u]$, then $\nabla^c f(\hat{p}) = 0$.*

Proof. Since $k \in K$ and $\hat{p}$ is strictly feasible, it follows that $\{\Delta_k\}_{k\in K}$ goes to zero and $\{p_k + \Delta_k(b, 0) : b \in B_k(p_k)\}$ is contained in Ω for k large enough. Equation (3.2) and the mean value theorem imply that

$$\begin{aligned} f(p_k) \;\leq\; \min_{v\in\mathcal{N}^c(p_k)} f(v) &= \min_{v\in\{p_k+\Delta_k(b,0):b\in B_k(p_k)\}} f(v) \\ &= \min_{b\in B_k(p_k)} f(p_k + \Delta_k(b,0)) \\ &= \min_{b\in B_k(p_k)} f(p_k) + \Delta_k b^T \nabla^c f(p_k + \alpha_k^b \Delta_k(b,0)) \\ &= f(p_k) + \Delta_k \min_{b\in B_k(p_k)} b^T \nabla^c f(p_k + \alpha_k^b \Delta_k(b,0)) \end{aligned}$$

for some $\alpha_k^b \in [0, 1]$ that depends on both the positive basis vector b and iteration number k. Therefore

$$0 \;\leq\; \min_{b\in B_k(p_k)} b^T \nabla^c f(p_k + \alpha_k^b \Delta_k(b, 0)).$$

Taking the limit for $k \to \infty$ yields $0 \leq \min_{b\in B} b^T \nabla^c f(\hat{p})$ for at least one positive spanning set B of the finite set $\mathcal{B}$ since f is assumed to be continuously differentiable. The positive spanning set property (3.1) guarantees that $\nabla^c f(\hat{p}) = 0$. ∎

The second lemma shows that there are no descent directions for points on the boundary of the feasible region.

LEMMA 4.6. *Let $\{p_k\}_{k\in K}$ be a subsequence of unsuccessful poll points and let $\hat{p}$ be a limit point. If the continuous part of the limit point $\hat{p}$ is on the boundary of the feasible region $[\ell, u]$, then $(p^c - \hat{p}^c)^T \nabla^c f(\hat{p}) \geq 0$ for any feasible $(p^c, \hat{p}^d)$.*

Proof. If the continuous part of p_k is within ε of the boundary of the feasible region $[\ell, u]$, then $B_k(p_k)$ is a maximal positive basis belonging to $\overline{\mathcal{B}}$ constructed from two diagonal matrices (see the discussion preceding the definition of the mesh neighborhood (3.2)).

As in Lemma 4.5, but only for the feasible positive basis directions b of $B_k(p_k)$, we have that

$$0 \;\leq\; b^T \nabla^c f(p_k + \alpha_k^b \Delta_k(b, 0)).$$

The result follows since any feasible direction $(p^c - \hat{p}^c)$ at $\hat{p}$ is a convex combination of some feasible positive basis directions. ∎

We can now state our first main result.

THEOREM 4.7. *The limit point $\hat{x}$ satisfies $(x^c - \hat{x}^c)^T \nabla^c f(\hat{x}) \geq 0$ for any feasible $(x^c, \hat{x}^d)$.*

Proof. The result follows directly from Lemmas 4.5 and 4.6, where x_k plays the role of p_k, and from the results on the sequence $\{x_k\}_{k\in K}$ of Proposition 4.3. ∎

Audet [1] shows through a small example containing two continuous variables and no discrete ones that, in the unconstrained case, this result cannot be strengthened to $\lim_{k\to\infty} \|\nabla^c f(x_k)\| = 0$ since there may be a limit point whose gradient is nonzero. It is also shown there that no second-order optimality conditions can be guaranteed, which is as it should be for an algorithm that uses only function values and no derivatives.

The following result shows that the gradient norm at the endpoints of the EXTENDED POLL converges to zero for $k \in K$.

THEOREM 4.8. *The limit point $\hat{x}$, and any point $\hat{y}$ in the set of neighbors $\mathcal{N}(\hat{x})$ satisfying $f(\hat{y}) < f(\hat{x}) + \xi$, are such that $(z^c - \hat{z}^c)^T \nabla^c f(\hat{z}) \;\geq\; 0$ for any feasible $(z^c, \hat{z}^d)$, where $\hat{z}$ is any limit point of the* EXTENDED POLL *endpoints.*

Proof. The result follows directly from Lemmas 4.5 and 4.6, where z_k plays the role of p_k, and from the results on the sequence $\{z_k\}_{k\in K}$ of Proposition 4.3. ∎

The next result shows that the function is constant at an infinite number of intermediate points between $\hat{y}$ and the endpoint $\hat{z}$ whenever $f(\hat{y}) = f(\hat{x})$. In order to show this result, we add the index k here to avoid confusion. The extended poll points at iteration k initiated at y_k are denoted $y_k^0 = y_k, y_k^1, \ldots, y_k^J = z_k$, where the index J depends on both k and y_k. Again, this is illustrated in Figure 3.1.

PROPOSITION 4.9. *The limit point $\hat{x}$, and any $\hat{y} \in \mathcal{N}(\hat{x})$ satisfying $f(\hat{y}) = f(\hat{x})$, are such that any limit point $\bar{y}$ of the sequence of* EXTENDED POLL *points $\{y_k^j\}$ satisfies $f(\bar{y}) = f(\hat{x})$. Moreover, if $\hat{y} \neq \hat{z}$, then there are infinitely many of these limit points.*

Proof. Let $\hat{y}$ in $\mathcal{N}(\hat{x})$ be such that $f(\hat{y}) = f(\hat{x})$. Let $\bar{y}$ be a limit point distinct from $\hat{y}$ and $\hat{z}$ of the sequence of EXTENDED POLL points $\{y_k^j\}$.

Since $f(\hat{x}) \leq f(y_k^{j+1}) < f(y_k^j)$ for $j = 0, 1, \ldots, J$ and since the subsequence $\left\{f(y_k^0)\right\}_{k\in K}$ converges to $f(\hat{x})$, we conclude that $f(\hat{x}) = f(\bar{y})$.

To show the second part of the result, we first let $d = \|\hat{y} - \hat{z}\|$ be the nonzero distance between $\hat{y}$ and $\hat{z}$. This makes sense because both share the same discrete components. Second, for any scalar p in the open interval $]0, d[$, we define the set

$$Y_p = \left\{ y_k^j : \; k \in K, \; j \in \{0, 1, \ldots, J\}, \; \|y_k^j - \hat{y}\| \leq p, \; \|y_k^{j+1} - \hat{y}\| > p \| \right\}.$$

Since $y_k^0 = y_k \to \hat{y}$ and $y_k^J = z_k \to \hat{z}$, it follows that the set Y_p contains infinitely many points for any p in $]0,1[$. Any limit point $\bar{y}_p$ of Y_p satisfies $\|\bar{y}_p - \hat{y}\| = p$ since Δ_k converges to 0 (in K) and y_k^{j+1} is equal to $y_k^j + \Delta_k b_k^j$ for some vector b_k^j of the basis $B_k(y_k^j)$ of the finite set $\mathcal{B}$. Therefore if $p \neq q$, then $\bar{y}_p \neq \bar{y}_q$ and the result follows. ∎

4.3. Stronger results. Theorem 4.8 may be strengthened under the following (more expensive) version of extended polling.

STRONG EXTENDED POLL STEP:

- $y_k^{j+1} \in \arg\min_{y \in \mathcal{N}^c(y_k^j)} f(y)$ for a given y_k^0 and $j = 0, 1, \ldots, J$ at iteration k;
- the same positive basis in $\overline{\mathcal{B}}$ must be used throughout the strong extended poll step.

This requires performing a complete EXTENDED POLL step; i.e., y_k^{j+1} is chosen only after evaluating the function value at all feasible points of the continuous mesh neighborhood around y_k^j and retaining the one that yields the smallest value (ties are broken arbitrarily).

This also means that the matrix $B_k(y_k^j)$ in $\mathcal{N}^c(y_k^j) = \{y_k^j + \Delta_k(b, 0) \in \Omega : b \in B_k(y_k^j)\}$ does not depend on the index j; it can, however, vary with the iteration number k. This positive basis is maximal and constructed from diagonal matrices. This is to make sure that the basis directions are correctly chosen in the event that the extended poll iterates approach the boundary of the feasible region.

The following result bounds the decrease in the objective function value under precise conditions. We will denote by b_k^j the vector of the positive basis used by the EXTENDED POLL step at the point y_k^j for some $j < J$. The next point is therefore $y_k^{j+1} = y_k^j + \Delta_k b_k^j$.

LEMMA 4.10. *Let $\hat{y} \in \mathcal{N}(\hat{x})$. For any $\eta < 0$, there exist $\delta > 0$ and $\sigma > 0$, both independent of the iteration number k, such that all extended poll iterates y_k^j for which $j < J$, $\Delta_k < \delta$, $y_k^{jd} = \hat{y}^d$ and for which $(b_k^j)^T \nabla^c f(y_k^j) \leq \frac{\eta}{3}$ also satisfy $f(y_k^j) - f(y_k^{j+1}) > \sigma \|y_k^j - y_k^{j+1}\|$.*

Proof. Let $\eta < 0$ be given. Continuous differentiability of the function f with respect to the continuous variables over a neighborhood of the compact set $L_\Omega(x_0)$ implies the existence of $\delta > 0$ such that any $y \in L_\Omega(x_0)$ and $w \in \Omega$ that satisfy $\|w - y\| < \delta \times \max\{\|b\| : b \in B \in \mathcal{B}\}$ also satisfy $\|b^T(\nabla^c f(w) - \nabla^c f(y))\| < \frac{|\eta|}{6}$ for each feasible direction $b \in B \in \mathcal{B}$ at y and, in particular, $b^T \nabla^c f(w) < b^T \nabla^c f(y) - \frac{\eta}{6}$.

Let $\hat{y} \in \mathcal{N}(\hat{x})$ and consider the extended poll iterate $y_k^{j+1} = y_k^j + \Delta_k b_k^j$, where $y_k^{jd} = \hat{y}^d$. Applying the mean value theorem yields

$$f(y_k^{j+1}) = f(y_k^j) + \Delta_k (b_k^j)^T \nabla^c f(w_k^j) \tag{4.1}$$

for some $w_k^j = y_k^j + \alpha_k^j \Delta_k b_k^j$, where α_k^j is a real number in the interval $[0, 1]$.

Assume that y_k^j satisfies $(b_k^j)^T \nabla^c f(y_k^j) \leq \frac{\eta}{3}$; if no such point exists, then the result is trivial. Observe that

$$\Delta_k = \frac{\|y_k^j - y_k^{j+1}\|}{\|b_k^j\|} \;\geq\; \frac{\|y_k^j - y_k^{j+1}\|}{\max\{\|b\| : b \in B \in \mathcal{B}\}}. \tag{4.2}$$

Moreover, if $\Delta_k < \delta$, then $w_k^j \in \Omega$ is within $\delta \times \max\{\|b\| : b \in B \in \mathcal{B}\}$ of $y_k^j \in L_\Omega(x_0)$ since

$$\|w_k^j - y_k^j\| = \alpha_k^j \Delta_k \|b_k^j\| \;<\; 1 \times \delta \|b_k^j\| \;\leq\; \delta \times \max\{\|b\| : b \in B \in \mathcal{B}\}.$$

Therefore, $(b_k^j)^T\nabla^c f(w_k^j) < (b_k^j)^T\nabla^c f(y_k^j) - \frac{\eta}{6} \le \frac{\eta}{3} - \frac{\eta}{6} = \frac{\eta}{6}$, and it follows by (4.1) and (4.2) that

$$f(y_k^{j+1}) - f(y_k^j) < \Delta_k \frac{\eta}{6} \ \le \frac{\eta \|y_k^j - y_k^{j+1}\|}{6 \max\{\|b\| : b \in B \in \mathcal{B}\}}.$$

Setting $\sigma = \frac{|\eta|}{6 \max\{\|b\|:b\in B\in\mathcal{B}\}} > 0$ concludes the proof. ∎

This second lemma relates $(b_k^j)^T\nabla^c f(y_k^j)$ to $(b_k^l)^T\nabla^c f(y_k^l)$ when Δ_k is small.

For the remainder of the section, we assume that the STRONG EXTENDED POLL step is used and that $\bar{y}$ is a limit point of the subsequence $\{y_k^{j(k)}\}_{k\in K}$ for which the continuous part $\bar{y}^c$ is in the strict interior of the feasible region $[\ell, u]$, where $j(k)$ is an index between 1 and J (where J depends on k). Moreover, we assume, without any loss in generality, that all iterates of this subsequence use the same positive basis that we denote by B. Note that the results of Proposition 4.3 concerning the other limit points still hold.

LEMMA 4.11. *For any $\eta < 0$ there exists an $\epsilon > 0$ and a $\delta' > 0$ such that if $\|y_k^j - \bar{y}\| \le \epsilon$ and $\|y_k^l - y_k^j\| \le \epsilon$, and $\Delta_k < \delta'$ for some k, j, and l, then $(b_k^j)^T\nabla^c f(y_k^j) - (b_k^l)^T\nabla^c f(y_k^l) \ge \frac{\eta}{3}$.*

Proof. Let $\eta < 0$ be given. Define $\epsilon > 0$ such that if $\|y_k^j - \bar{y}\| \le \epsilon$ and $\|y_k^l - y_k^j\| \le \epsilon$, then

$$\|b^T(\nabla^c f(y_k^l) - \nabla^c f(y_k^j))\| \le \frac{|\eta|}{12} \tag{4.3}$$

for all $b \in B$.

Using the mean value theorem, define $w_k^l = y_k^l + \alpha_k^l \Delta_k b_k^l$ and $w_k^j = y_k^l + \alpha_k^j \Delta_k b_k^j$ (where both α_k^l and α_k^j are in $[0, 1]$) such that $f(y_k^l + \Delta_k b_k^l) = f(y_k^l) + \Delta_k (b_k^l)^T\nabla^c f(w_k^l)$ and $f(y_k^l + \Delta_k b_k^j) = f(y_k^l) + \Delta_k (b_k^j)^T\nabla^c f(w_k^j)$.

Let $\delta' > 0$ be such that if $\Delta_k < \delta'$ for some $k \in K$, then $y_k^l + \Delta_k b \in \Omega$ for all positive bases directions $b \in B$. This is possible since $\bar{y}^c$ belongs to the strict interior of $[\ell, u]$.

$$\|(b_k^l)^T(\nabla^c f(w_k^l) - \nabla^c f(y_k^l))\| \le \frac{|\eta|}{6} \text{ and } \|(b_k^j)^T(\nabla^c f(w_k^j) - \nabla^c f(y_k^l))\| \le \frac{|\eta|}{12}\ . \tag{4.4}$$

Combining (4.3) (using $b = b_k^j$) with the second inequality of (4.4) yields

$$\|(b_k^j)^T(\nabla^c f(w_k^j) - \nabla^c f(y_k^j))\| \le \frac{|\eta|}{6}. \tag{4.5}$$

In summary, we have shown in the first inequality of (4.4) and in (4.5) that

$$(b_k^l)^T\nabla^c f(w_k^l) = (b_k^l)^T\nabla^c f(y_k^l) + \mu^l \text{ and } (b_k^j)^T\nabla^c f(w_k^j) = (b_k^j)^T\nabla^c f(y_k^j) + \mu^j, \tag{4.6}$$

where $|\mu^l| \le \frac{|\eta|}{6}$ and $|\mu^j| \le \frac{|\eta|}{6}$.

Moreover, since b_k^l is obtained through the STRONG EXTENDED POLL steps, and since $y_k^l + \Delta_k b_k^j$ is feasible, then it follows that $f(y_k^l + \Delta_k b_k^l) \le f(y_k^l + \Delta_k b_k^j)$, and therefore $(b_k^l)^T\nabla^c f(w_k^l) \le (b_k^j)^T\nabla^c f(w_k^j)$. Using the two equalities of (4.6) we get $(b_k^l)^T\nabla^c f(y_k^l) - (b_k^j)^T\nabla^c f(y_k^j) = u^j - u^l \le \frac{|\eta|}{3}$. ∎

The following result strengthens Theorem 4.8 by showing that there are no feasible descent directions at the limit points $\bar{y}$ of Proposition 4.9.

THEOREM 4.12. *If the limit points* $\hat{x}$ *and* $\hat{y} \in \mathcal{N}(\hat{x})$ *obtained under the* STRONG EXTENDED POLL *step satisfy* $f(\hat{y}) = f(\hat{x})$, *then* $\nabla^c f(\bar{y}) = 0$.

Proof. Suppose by way of contradiction that the limit point $\bar{y} \neq \hat{z}$ satisfies $\nabla^c f(\bar{y}) \neq 0$. Set $\eta = \min_{b \in B} b^T \nabla^c f(\bar{y}) < 0$ let δ and σ be from Lemma 4.10, and ϵ and δ' be from Lemma 4.11. Let k be an index in K such that the following six conditions hold:

(i) $\Delta_k \leq \delta$,
(ii) $\Delta_k \leq \delta'$,
(iii) $(b_k^{j(k)})^T \nabla^c f(y_k^{j(k)}) < \frac{2\eta}{3}$,
(iv) $b^T \nabla^c f(z_k) > \frac{\eta}{3}$ for all feasible directions $b \in B$,
(v) $\|y_k^{j(k)} - \bar{y}\| < \epsilon$,
(vi) $f(y_k^{j(k)}) - f(y_k^l) < \sigma\epsilon$ for any $l > j(k)$.

Conditions (i) and (ii) hold since $\Delta_k \to 0$. Condition (iii) holds since $b_k^{j(k)}$ is chosen with the STRONG EXTENDED POLL step. Theorem 4.8 implies condition (iv). Condition (v) follows since $y_k^{j(k)} \to \bar{y}$. Proposition 4.9 guarantees condition (vi) since $f(y_k^{j(k)}) \to f(\bar{\gamma}) = f(\hat{x})$.

Define the index $l(k) = \min\left\{l \geq j(k) : (b_k^l)^T \nabla^c f(y_k^l) > \frac{\eta}{3}\right\}$ (condition (iv) guarantees that $l(k) \leq J$). Therefore, condition (i) and Lemma 4.10 ensure that $f(y_k^j) - f(y_k^{j+1}) > \sigma \|y_k^j - y_k^{j+1}\|$ when $j(k) \leq j < l(k)$. Writing out the telescopic sum leads to

$$\begin{aligned} f(y_k^{j(k)}) - f(y_k^{l(k)}) &= \sum_{j=j(k)}^{l(k)-1} \left(f(y_k^j) - f(y_k^{j+1}) \right) \\ &> \sigma \sum_{j=j(k)}^{l(k)-1} \|y_k^j - y_k^{j+1}\| \;\geq\; \sigma \|y_k^{j(k)} - y_k^{l(k)}\|. \end{aligned}$$

Together with condition (vi) this gives $\|y_k^{j(k)} - y_k^{l(k)}\| \leq \epsilon$. Combining this with conditions (ii)–(v) and with Lemma 4.11 leads to

$$\begin{aligned} \frac{2\eta}{3} &> (b_k^{j(k)})^T \nabla^c f(y_k^{j(k)}) \\ &= (b_k^{l(k)})^T \nabla^c f(y_k^{l(k)}) + \left((b_k^{j(k)})^T \nabla^c f(y_k^{j(k)}) - (b_k^{l(k)})^T \nabla^c f(y_k^{l(k)}) \right) \\ &> \frac{\eta}{3} + \frac{\eta}{3} \;=\; \frac{2\eta}{3}, \end{aligned}$$

which is a contradiction. ∎

In the next section, we illustrate the behavior of the algorithm through two examples.

5. Examples. The first example shows the value of the STRONG EXTENDED POLL versus the cheaper EXTENDED POLL step. This illustrates the difference between Proposition 4.9 and Theorem 4.12. The second example shows how the algorithm behaves on the larger problem presented in section 2.2.

5.1. Illustration of the stronger version of the algorithm. Consider the following example in which there are two continuous variables and a single binary one. In order to simplify notation, the continuous variables x^c are written $x^c = (a, b)$. The objective function is

$$f(x) = f(a, b, x^d) = g(a, b)(1 - x^d) + h(a, b)x^d,$$

where $g(a,b) = a^2+b^2$ and $h(a,b) = a^2v+a(1-b)$. Both variables are constrained to be in the interval $[-2,2]$, but these bounds are never approached by the trial points.

The pattern search algorithm we apply here does not have a SEARCH step; we use only a POLL and an EXTENDED POLL step triggered by $\xi = 1$. The current mesh neighborhood at iteration k is defined to be

$$\mathcal{N}^c(x) = \{x + \Delta_k(0,1,0),\ x + \Delta_k(0,-1,0),\ x + \Delta_k(5,0,0),\ x + \Delta_k(-7,0,0)\}$$

for any $x = (a,b,x^d)$ except when the iterate may be written $x = (2\Delta_k, 1-\Delta_k, 1)$, in which case it is

$$\mathcal{N}^c(x) = \{x + \Delta_k(0,-1,0),\ x + \Delta_k(5,1,0),\ x + \Delta_k(-7,1,0)\}.$$

The set of neighbors of $x = (a,b,x^d)$ is $\mathcal{N}(x) = \{(a,b,1-x^d),(a,b,x^d)\}$. This definition ensures that the discrete variable always remain binary. Iteration k is declared successful and stops as soon as the incumbent is improved and $\Delta_{k+1} = \Delta_k$. Otherwise $\Delta_{k+1} = \frac{\Delta_k}{2}$.

The algorithm is initiated at $x_0 = (1,0,0)$ with $\Delta_0 = \frac{1}{4}$ and with incumbent value $f(x_0) = 1$. The POLL step evaluates the function at the points of $\mathcal{N}^c(x_0) : f(1,\frac{1}{4},0) = \frac{17}{16}, f(1,\frac{-1}{4},0) = \frac{17}{16}, f(\frac{9}{4},0,0) = \frac{81}{16}$, and $f(\frac{-3}{4},0,0) = \frac{9}{16}$. This first iteration is successful.

Iteration 1 starts at $x_1 = (\frac{-3}{4},0,0)$ with $\Delta_0 = \frac{1}{4}$ and $f(x_1) = \frac{9}{16}$. The POLL step computes f in $\mathcal{N}^c(x_1) : f(\frac{-3}{4},\frac{1}{4},0) = \frac{10}{16}, f(\frac{-3}{4},\frac{-1}{4},0) = \frac{10}{16}, f(\frac{1}{2},0,0) = \frac{1}{4}$. This iteration is also successful.

Iteration 2 starts at $x_2 = (\frac{1}{2},0,0)$ with $\Delta_0 = \frac{1}{4}$ and $f(x_2) = \frac{1}{4}$. The POLL step computes f in $\mathcal{N}^c(x_2) : f(\frac{1}{2},\frac{1}{4},0) = \frac{5}{16}, f(\frac{1}{2},\frac{-1}{4},0) = \frac{5}{16}, f(\frac{7}{4},0,0) = \frac{49}{16}$, and $f(\frac{-5}{4},0,0) = \frac{25}{16}$. Before declaring this iteration unsuccessful, polling must be conducted on the set of neighbors $\mathcal{N}(x_2) : f(\frac{1}{2},0,1) = \frac{1}{2}$. This value is within ξ of $f(x_2)$ and so extended polling must be conducted around this last point y_2^0. The EXTENDED POLL step finds $y_2^1 = (\frac{1}{2},\frac{1}{4},1)$ in $\mathcal{N}^c(y_2^0)$ with $f(y_2^1) = \frac{7}{16}$, then $y_2^2 = (\frac{1}{2},\frac{1}{2},1)$ in $\mathcal{N}^c(y_2^1)$ with $f(y_2^1) = \frac{3}{8}$, and $y_2^3 = (\frac{1}{2},\frac{3}{4},1)$ in $\mathcal{N}^c(y_2^2)$ with $f(y_2^1) = \frac{5}{16}$. It does not succeed in improving this last value in $\mathcal{N}^c(y_2^3) : f(\frac{1}{2},\frac{1}{2},1) = \frac{15}{32}, f(\frac{7}{4},1,1) = \frac{49}{16}$, and $f(\frac{-5}{4},1,1) = \frac{25}{16}$. Thus, iteration 2 is unsuccessful and iteration 3 starts at the same point $x_2 = (\frac{1}{2},0,0)$ with $\Delta_0 = \frac{1}{8}$ and $f(x_3) = \frac{1}{4}$.

Table 5.1 shows that the algorithm generates cycles composed of two successful iterations followed by an unsuccessful one. The three iterations detailed above, i.e., the first cycle, appear in the table by letting $\alpha = \frac{1}{4}$. Iteration 3 initiates a new cycle with $\alpha = \frac{1}{8}$.

Figure 5.1 displays the iterates of the EXTENDED POLL step from y_k to z_k. The circles represent the points y_k^j for $j = 0,1,\ldots,J$. All these points are on the same line as the function decreases linearly when the variable a is fixed to 2α. At the last point z_k, the current mesh neighborhood is evaluated using a different positive basis. The set $\mathcal{N}^c(z_k)$ is represented by the three circled crosses. As k goes to infinity, the points $\{y_k^j : j = 0,1,\ldots,J\}$ converge to the line segment from $\hat{y} = (0,0,1)$ to $\hat{z} = (0,1,1)$, which is the thick line in Figure 5.1. The objective function value is equal to zero there. The gradient norm is nonzero at $\hat{y}$ but decreases to zero at $\hat{z}$.

In order to ensure that the gradient norm is zero at all points of $\mathcal{N}(\hat{x})$, the stronger version of the algorithm must be used. By doing this, the EXTENDED POLL step at iteration 2 discovers the point $y^1 = (\frac{-5}{4},0,1)$ of $\mathcal{N}^c(y^0)$, whose function value is $\frac{-5}{4}$. This iteration is successful, and the iterates eventually converge to the global minimizer of f over Ω.

TABLE 5.1
In three consecutive iterations, the iterates go from $x_k = 4\alpha, \Delta_k = \alpha$ *to* $x_{k+3} = 2\alpha, \Delta_{k+3} = \frac{\alpha}{2}$.

x_k	$x_k + \Delta_k(0,1,0)$	$x_k + \Delta_k(0,-1,0)$	$x_k + \Delta_k(5,0,0)$	$x_k + \Delta_k(-7,0,0)$
$(4\alpha,0,0)$ $16\alpha^2$	$(4\alpha,\alpha,0)$ $17\alpha^2$	$(4\alpha,-\alpha,0)$ $17\alpha^2$	$(9\alpha,0,0)$ $81\alpha^2$	$(-3\alpha,0,0)$ $9\alpha^2$
$(-3\alpha,0,0)$ $9\alpha^2$	$(-3\alpha,\alpha,0)$ $10\alpha^2$	$(-3\alpha,-\alpha,0)$ $10\alpha^2$	$(-2\alpha,0,0)$ $4\alpha^2$	$(-10\alpha,0,0)$ $100\alpha^2$
$(2\alpha,0,0)$ $4\alpha^2$	$(2\alpha,\alpha,0)$ $5\alpha^2$	$(2\alpha,-\alpha,0)$ $5\alpha^2$	$(7\alpha,0,0)$ $49\alpha^2$	$(-5\alpha,0,0)$ $25\alpha^2$
EXTENDED	$y_k = y_k^0$	y_k^1	$y_k^2 \quad \ldots$	$z_k = y_k^J$
POLL:	$(2\alpha,0,1)$ 2α	$(2\alpha,\alpha,1)$ $2\alpha(1-\alpha(1-2\alpha))$	$(2\alpha,2\alpha,1)$ $2\alpha(1-2\alpha(1-2\alpha))$	$(2\alpha,1-\alpha,1)$ $2\alpha^2(3-2\alpha))$
$\mathcal{N}^c(z_k)$:	$z_k + \Delta_k(0,-1,0)$ $(2\alpha,1-2\alpha,1)$ $2\alpha^2(4-4\alpha^2)$	$z_k + \Delta_k(-7,1,0)$ $(7\alpha,1,1)$ $49\alpha^2$	$z_k + \Delta_k(5,1,0)$ $(-5\alpha,1,1)$ $25\alpha^2$	

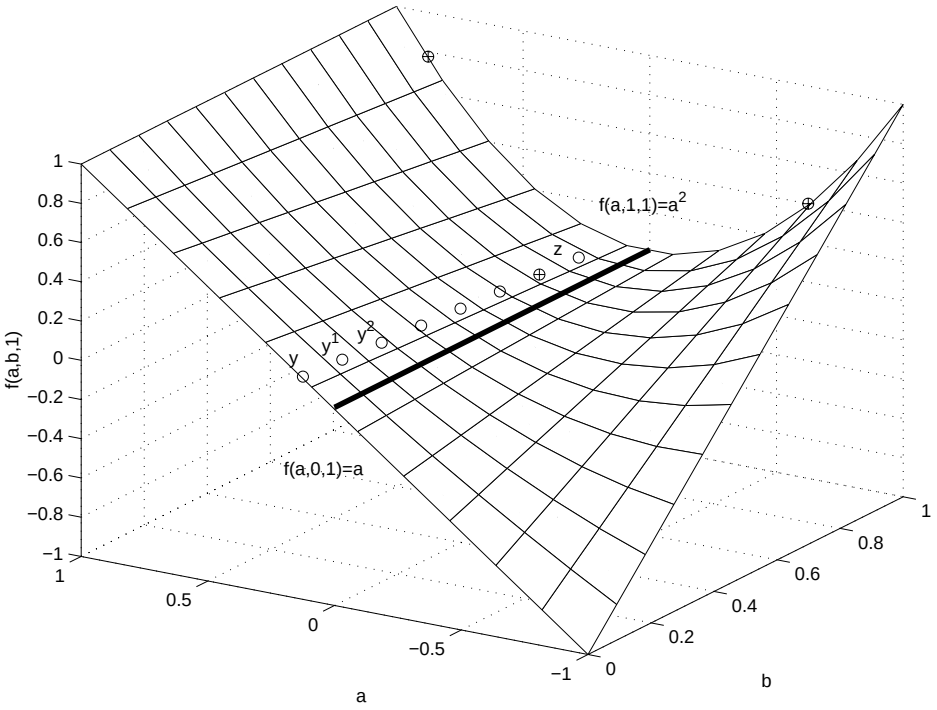


FIG. 5.1. *Extended polling from* $y = (2\alpha, 0, 1)$ *to* $z = (2\alpha, 1-\alpha, 1)$.

5.2. A thermal insulation system. We ran the GMVP algorithm on the example of section 2.2. The behavior was typical of derivative-free algorithms in making rapid improvement of the objective function value and then reaching a plateau, where the objective function value does not decrease significantly.

The initial point was a single shield at temperature $T_1 = 150$ surrounded by $M_1 =$ Teflon and $M_2 =$ nylon of thickness $\Delta x_1 = \Delta x_2 = 50$. An upper bound of 100 was imposed on the number of shields. (This bound was large enough so that it was never reached.) The GMVP algorithm later used the third insulator (epoxy-fiberglass). We

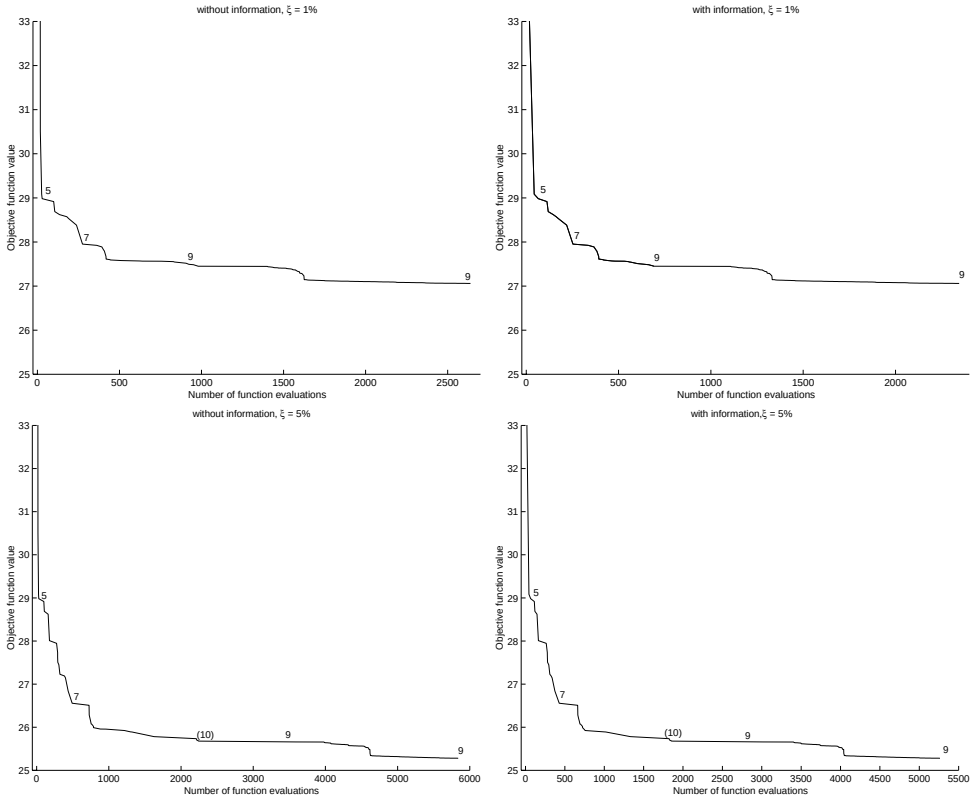


FIG. 5.2. *Progress of the objective function value, without or with extra information, and with* $\xi = 1\%$ *or* 5%.

used the set of discrete neighbors suggested in section 2.2. The algorithm consisted mainly of POLL steps. A simple SEARCH step was invoked if the previous iteration was successful and the incumbent solution (x_k^c, x_k^d) differed from the previous one only in its continuous components: $x_k^c \neq x_{k-1}^c$. This SEARCH step consisted of a point further in the same successful direction: $(x_k^c + 2(x_k^c - x_{k-1}^c), x_k^d)$.

The mesh size parameter was initially set at 10 and not increased. It decreased at local mesh optimizers. Figure 5.2 shows the improvement of the objective function value with the number of function evaluations for four runs. They illustrate the user's control in defining local optimality and in triggering extended polling with respect to the categorical variables. The y-axis is truncated for readability since $f(x_0) \approx 262$. The line on the graph starts at the 18th iteration.

The right-hand graphs use properties of the insulators (such as the fact that Teflon is a much better insulator than nylon at high temperatures) to further restrict the discrete POLL set. The top graphs use $\xi_i = 1\% f(x_i)$ to trigger the EXTENDED POLL step. The bottom graphs use the larger value $\xi_i = 5\% f(x_i)$. As expected, a larger value triggers more EXTENDED POLL steps and uses more objective function evaluations, but finds a better solution. Also, the runs that use the extra information converge using fewer function evaluations.

The numbers on the curves indicate the number of shields at local mesh optimizers, i.e., at unsuccessful iterations. For the top left graph and for $\Delta_k = 10, 5, \frac{5}{4}$, and

$\frac{5}{32}$, there were $5, 7, 9$ and 9 shields, respectively. The number of shields did not increase monotonically. Before reaching five shields, the GMVP algorithm added a third one then immediately removed another. On the bottom graphs, it added a tenth shield and later removed it. (This is indicated by the number in parentheses on the graphs.)

On the top left graph, the algorithm converged to a solution containing 9 shields and a combination of all three types of insulators. The best solution was found after 2639 function evaluations, and an additional 2083 showed it to be a local mesh optimizer—a total of 4722 function evaluations. The top right graph gives the progress using the more restrictive definition of the set of neighbors discussed above. It found the same local mesh optimizer but with 10% fewer function evaluations. It took 2345 evaluations to find the solution and 1861 more to show it to be a local mesh optimizer—a total of 4206 function evaluations.

On the bottom left graph, the algorithm converged after a total of 13,329 function evaluations to a solution that uses the three insulators and whose objective function value is reduced by more than 6%. The same solution was found on the bottom right, but using 10,053 function evaluations (approximately 25% fewer). All four runs produced a solution having 9 shields. The difference in the objective function values suggests the presence of local optimum solutions.

Further computational results on this and related problems can be found with a more engineering slant in [8].

Acknowledgments. The authors would like to thank the referees, David Applegate, and Yin Zhang for discussions which helped improve the quality of this paper. Special thanks are due to our collaborator Michael Kokkolaras who suggested the thermal insulation problem and provided Figure 2.1.

REFERENCES

[1] C. Audet, *Convergence Results for Pattern Search Algorithms are Tight,* Tech. report TR98-24, Department of Computational and Applied Mathematics, Rice University, Houston TX, 1998.

[2] A.J. Booker, J.E. Dennis, Jr., P.D. Frank, D.B. Serafini, V. Torczon, and M.W. Trosset, *A rigorous framework for optimization of expensive functions by surrogates,* Structural Optim. 17 (1999), pp. 1–13.

[3] G.E.P. Box, *Evolutionary operation: A method for increasing industrial productivity,* Appl. Statist., 6 (1957) pp. 81–101.

[4] C. Davis, *Theory of positive linear dependence,* Amer. J. Math., (1954), pp. 448–474.

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

[6] M.A. Hilal and R.W. Boom, *Optimization of mechanical supports for large superconductive magnets*, Adv. Cryogenic Engrg., 22 (1977), pp. 224–232.

[7] R. Hooke and T.A. Jeeves, *Direct search solution of numerical and statistical problems,* J. ACM, 8 (1961), pp. 212–229.

[8] M. Kokkolaras, C. Audet, and J.E. Dennis, Jr., *Mixed Variable Optimization of the Number and Composition of Heat Intercepts in a Thermal Insulation System,* Tech. report TR00-21, Department of Computational and Applied Mathematics, Rice University, Houston TX, 2000.

[9] R.M. Lewis and V. Torczon, *Pattern search algorithms for bound constrained minimization,* SIAM J. Optim., 9 (1999), pp. 1082–1099.

[10] R.M. Lewis and V. Torczon, *Rank Ordering and Positive Basis in Pattern Search Algorithms,* Tech. report TR 96-71, ICASE NASA Research Center, Hampton, VA, 1996.

[11] R.M. Lewis and V. Torczon, *Pattern search methods for linearly constrained minimization,* SIAM J. Optim., 10 (2000), pp. 917–941.

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

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 595–610

APPROXIMATING THE SINGLE-SINK LINK-INSTALLATION PROBLEM IN NETWORK DESIGN*

F. S. SALMAN†, J. CHERIYAN‡, R. RAVI†, AND S. SUBRAMANIAN§

Abstract. We initiate the algorithmic study of an important but NP-hard problem that arises commonly in network design. The input consists of the following:

(1) An undirected graph with one sink node and multiple source nodes, a specified length for each edge, and a specified demand, dem_v, for each source node v.
(2) A small set of cable types, where each cable type is specified by its capacity and its cost per unit length. The cost per unit capacity per unit length of a high-capacity cable may be significantly less than that of a low-capacity cable, reflecting an *economy of scale*; i.e., the payoff for buying at bulk may be very high.

The goal is to design a minimum-cost network that can (simultaneously) route all the demands at the sources to the sink by installing zero or more copies of each cable type on each edge of the graph. An additional restriction is that the demand of each source must follow a single path. The problem is to find a route from each source node to the sink and to assign capacity to each edge of the network such that the total costs of cables installed are minimized. We call this problem the single-sink link-installation problem.

For the general problem, we introduce a new "moat-type" lower bound on the optimal value and we prove a useful structural property of near-optimal solutions: For every instance of our problem, there is a near-optimal solution whose graph is acyclic (with a cost no more than twice the optimal cost). We present efficient approximation algorithms for key special cases of the problem that arise in practice. For points in the Euclidean plane, we give an approximation algorithm with performance guarantee $O(\log(D/u_1))$, where D is the total demand and u_1 is the smallest cable capacity. When the metric is arbitrary, we consider the case where the network to be designed is restricted to be *two level*; i.e., every source-sink path has at most two edges. For this problem, we present an algorithm with performance guarantee $O(\log n)$, where n is the number of nodes in the input graph, and also show that this performance guarantee is nearly best possible.

Key words. minimum-cost capacitated network design, approximation algorithms

AMS subject classifications. 68Q25, 68R10, 90B10, 90B12

PII. S1052623497321432

1. Introduction.

1.1. The problem. An oil company wishes to construct a network of pipelines to carry oil from several remote wells to a major refinery. For each edge of the network, the company can install either zero or more copies of a cheap but thin pipe (say, the diameter is 10 inches and the cost is \$1000 per mile) or zero or more copies of a more expensive but thicker pipe (say, the diameter is 100 inches and the cost is \$2000 per mile). The demand (actually, oil supply) at each of the oil wells is given. The

*Received by the editors May 9, 1997; accepted for publication (in revised form) February 13, 2000; published electronically November 10, 2000. A preliminary version of this paper appears as *Buy-at-bulk network design: Approximating the single-sink edge installation problem*, in the Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, New Orleans, LA, 1997, pp. 619–628.

http://www.siam.org/journals/siopt/11-3/32143.html

†GSIA, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213 (salmanf@mgmt.purdue.edu, ravi+@andrew.cmu.edu). The first author was supported in part by an IBM Corporate Fellowship. The third author was supported in part by NSF CAREER grant CCR-9625297.

‡Department of Combinatorics and Optimization, University of Waterloo, Waterloo, ON, Canada N2L 3G1 (jcheriyan@dragon.uwaterloo.ca). This author was supported in part by NSERC grant OGP0138432 (NSERC code OGPIN 007).

§Wireless Technology Group, Northern Telecom, Dallas, TX 70007 (sai@nortel.com).

goal is to build a minimum-cost network that has sufficient capacity at every edge to transport the oil to the refinery.

Notice one feature of the problem: The cost per unit length versus capacity (available by combination of different pipe types) is a staircase function, reflecting an *economy of scale*. Also, note that several copies of several pipe types may be used in parallel to accommodate the flow on one edge of the network.

The above network design problem is NP-hard. There are reductions from both the Steiner tree problem and the knapsack problem (see section 2.2). These reductions suggest two inherent sources of hardness of our problem. One is the connectivity requirement—our problem is NP-hard even when only one cable type is available. The second is the choice of cables—the problem is NP-hard even on a graph consisting of a single edge. While there are several known approximation algorithms that attack NP-hard minimum-cost connectivity problems [AKR 95, GW 95, WGM+95], to the best of our knowledge there is no (previous) approximation algorithm that considers costs based on the choice of different cable types. Our work gives the first results on this topic.

Our problem of designing a single-sink multisource network at minimum cost is a fundamental and economically significant one that arises in hierarchical design of telecommunication networks. In the lowest level of network design, switching centers (controllers) collect calls from customers (base cells) and in the next level traffic goes between pairs of controllers. Once a set of customers is assigned to a switching center, the single-sink, multisource problem arises. An additional constraint on the telecommunication problem is that the flow of traffic for any demand must follow a single path to the sink in the network [BMW 95]—this arises from limitations on the capacity of routing tables at nodes, and in avoiding complex switching hardware to support bifurcating flow. We call such flow routes *indivisible*.[1] The availability of a small number of cables, strong economies of scale, and the large number of customers are characteristic to the problems in the telecommunication industry.

1.2. Our results. We start by showing NP-hardness of two very simple versions of our problem (section 2.2). Next we formalize two known lower bounds and use them to derive a simple constant factor approximation algorithm for the case with a single cable type using existing ideas.

We continue by proving a structure theorem (Theorem 3.2): For every instance of our problem, there is a near-optimal solution whose graph is acyclic (the cost is no more than two times the optimal value).

The case that appears to be most relevant in the geographic instances of network design is when the graph is defined by points in the two-dimensional Euclidean plane. For this case, we present an approximation algorithm with performance guarantee $O(\log(D/u_1))$, where D denotes the total demand and u_1 denotes the smallest cable capacity (Theorem 4.1). The analysis of the performance guarantees hinges on a new "moat-type" lower bound on the optimal value that we introduce, which is valid also for the general metric case.

In the general case, when the metric is arbitrary, we focus on a restricted version of the problem: Instead of allowing the optimal solution to induce an arbitrary graph, we restrict the graph to be *two level*; i.e., every source-sink path has at most two edges. For this problem, we present an algorithm with nearly best possible performance guarantee $O(\log n)$, where n is the number of nodes (Theorem 5.2).

[1] This is also called unsplittable [Kle 96] or nonbifurcated [Bar 96] in the literature.

1.3. Previous work. The problem we address arises commonly in practical network design and has been widely studied in the operations research literature. One of the first papers on routing flows under a staircase cost function arises from the *Telepak* problem in network design [GR 71]. Specializations of this problem include the fixed charge problem [Bal 61, G71], while generalizations include the minimum concave cost flow problem [Z68, GSS80]. In a survey on network synthesis and design problems, Minoux [Min 89] discusses several variants of the problem and exact solution methods. This body of work does not enforce the indivisible flow constraint.

Balakrishnan, Magnanti, and Wong [BMW 95] address the problem of expanding an existing telecommunication network, where they have the indivisible flow restriction. Magnanti, Mirchandani, and Vachani [MMV 95] study the polyhedral and computational aspects of the design problem with two cables. One feature they highlight is the large gap between heuristic solutions and Lagrangian lower bounds. Because of this, even small instances of the problem cannot be solved to anywhere near optimality by state-of-the-art computational techniques. More recent study of our network design problem with multiple sinks is undertaken in [BG 96, BCGT 98, Bar 96]. These papers develop cutting plane methods by exploiting classes of valid inequalities for an appropriate formulation that only considers one or two cable types.

Mansour and Peleg [MP 94] have results on a variant of our problem. In their model, there are multiple sinks and multiple sources, there is *only one* type of cable, and installing an edge has a fixed cost (similar to our model) as well as a variable cost per unit flow. By applying light-weight distance-preserving spanners [ADD+92], they obtain an $O(\log n)$-approximation algorithm for their network design problem with n nodes. It is easy to apply the method of Mansour and Peleg to the case with only one sink, and *only one* cable type, and to improve the logarithmic approximation to a constant-factor approximation (section 2.3).

2. Preliminaries.

2.1. Formalizing the problem. We are given an underlying undirected graph $G = (V, E), |V| = n$. A subset S of nodes is specified as sources of traffic and a single sink t is specified. Each source node $s_i \in S$ has an integer-valued demand dem_i. All the traffic of the source set is to be routed to t. The edges of G have lengths $\ell : E \to \mathbb{R}^+$. Without loss of generality, we assume that for every pair of nodes v, w, we can use the shortest-path distance $\text{dist}(v, w)$ as the length of the edge between v and w; i.e., we take the metric completion of the given graph. The edges of the network must be installed by purchasing one or more copies from among a small set of cables, where each cable type $i \in \{1, \ldots, q\}$ has a specified capacity u_i and a specified cost c_i per unit length. The indexing of the cables is such that $u_1 \le u_2 \le \cdots \le u_q$, $c_1 \le c_2 \le \cdots \le c_q$, and $c_1/u_1 \ge c_2/u_2 \ge \cdots \ge c_q/u_q$. Let $\sigma_{ij} := \frac{c_i/u_i}{c_j/u_j}$, $i \in \{1, \ldots, q-1\}$, $j > i$. Then a type i cable is σ_{ij} times as expensive as a type j cable, per unit of capacity per unit of length. We refer to σ_{ij} as the "economy of scale" factor between the ith and the jth cable types.

A solution to our network design problem can be characterized by specifying for each source s_i, a path to t and a combination of cables to be used on each arc of the network induced by the paths. We will call the traffic of source s_i commodity i, $i = 1, \ldots, k$. Let P_i be the path for commodity i and let $N = (V_N, A)$ be the graph induced by the union of P_i, $i = 1, \ldots, k$.

Let f_e^i denote the amount of flow of commodity i on edge e of A and f_e denote

the total flow on edge e. That is,

$$f_e^i = \begin{cases} dem_i & \text{if } e \in P_i, \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \qquad f_e = \sum_{i=1}^{k} f_e^i.$$

Let $\nu_1^e, \ldots, \nu_q^e$ be the number of copies of cable types 1 to q to be installed on edge e, where q is the number of available cable types. ($\nu_i^e = 0$ implies that no cable of type i is installed on edge e.)

Let $P = (P_1, \ldots, P_k)$ denote the routing and $\nu^e = (\nu_1^e, \ldots, \nu_q^e)$ denote the choice of cables on edge e that accommodates f_e (induced by P). Then P and $\nu = (\nu^e, e \in A)$ characterize a feasible solution to the problem formulated below:

$$\begin{aligned} \min_{P,\,\nu} \quad & \sum_{e \in A} \left\{ \operatorname{dist}(e) \sum_{j=1}^{q} (c_j \cdot \nu_j^e) \right\} \\ \text{such that} \quad & \sum_{i: e \in P_i} dem_i \; - \; \sum_{j=1}^{q} (u_j \cdot \nu_j^e) \; \leq \; 0, \quad e \in A, \quad (1) \\ & \nu_j^e \in \{0, 1, 2, \ldots\}, \qquad j \in \{1, \ldots, q\}, \quad e \in A, \\ & P_i \text{ is a } s_i - t \text{ path}, \quad i = 1, \ldots, k. \end{aligned}$$

Equivalently, the constraints (1) can be rewritten as

$$f_e \; - \; \sum_{j=1}^{q} (u_j \cdot \nu_j^e) \;\; \leq \;\; 0, \quad e \in A. \quad (2)$$

Note that the optimal choice of the routing depends on the choice of cables, as they determine the cost of edges. Yet the optimal choice of cables on each edge depends on the amount of flow on each edge, which is determined by the routing decision. Hence an optimal solution requires the decision of the routing and the cable choices simultaneously.

2.2. Hardness of the problem. It is easy to see that the single-sink edge-installation problem is NP-hard. There are reductions from both the Steiner tree problem and the knapsack problem.

A special case of our problem when there is only one cable type with capacity large enough to hold all of the demand is equivalent to a Steiner tree problem with the sources and the sink as the terminal nodes. Hence, our problem is NP-hard [GJ 79] even when only one cable type is available with unlimited capacity. In this case, the problem is that of finding a minimum cost routing under fixed costs on the edges.

Another simple special case of our problem with one source node and a single edge is also NP-hard. In this case, the problem reduces to finding the minimum-cost choice of cables on the edge such that the total capacity of the cables covers the demand of the source. This problem is an integer min-knapsack problem with the additional economies of scale restrictions on data. The integer min-knapsack problem was shown to be NP-hard by Lueker [Lue 75] by a transformation from the subset sum problem. This transformation is still valid under the economies of scale restrictions.

2.3. Single cable type case. In section 1.3 we mentioned that a constant factor approximation can be obtained by applying the method of Mansour and Peleg [MP 94]

to the case of single-sink and single cable type. In this subsection we give a proof of this claim. The idea is to route through a light approximate shortest-path tree (LAST) [KRY 93].

DEFINITION 2.1 (see [KRY 93]). *Let G be a graph with nonnegative edge lengths. A tree T rooted at vertex t is called an (α, β)-LAST if the following conditions are satisfied* ($\alpha, \beta \geq 1$):

1. *The distance of every vertex v from t in T is at most α times the distance between v and t in G.*
2. *The length of T is at most β times the length of a minimum spanning tree (MST) of G.*

LEMMA 2.2. *Let C^* be the optimal cost for the network design problem with a single cable type. Let G' be a complete graph on node set $S \cup \{t\}$ where length of an edge is the shortest path length in G between the end points of the edge. Let T be an (α, β)-LAST of G' rooted at the sink node t, and let C_T be the cost of routing dem_i through the $s_i - t$ path in T for all i and using as many copies of the cable as necessary. Then $C_T \leq (\alpha + 2\beta)C^*$.*

Proof. Let $\text{dist}_T(s_i, t)$ denote the length of the $s_i - t$ path P_i in T.

$$C_T = \sum_{e \in T} c_1 \left\lceil \frac{f_e}{u_1} \right\rceil \cdot \text{dist}_T(e) \leq \sum_{e \in T} c_1 \left(\frac{\sum_{i: e \in P_i} dem_i}{u_1} + 1 \right) \cdot \text{dist}_T(e)$$

$$= c_1 \sum_{s_i \in S} \sum_{e \in P_i} \text{dist}_T(e) \frac{dem_i}{u_1} + c_1 \sum_{e \in T} \text{dist}_T(e)$$

$$\leq \frac{c_1}{u_1} \sum_{s_i \in S} \text{dist}_T(s_i, t) dem_i + c_1 \sum_{e \in T} \text{dist}_T(e).$$

Since $\text{dist}_T(s_i, t) \leq \alpha \cdot \text{dist}(s_i, t)$ for all $s_i \in S$ and $\sum_{e \in T} \text{dist}_T(e) \leq \beta\, w(MST(G'))$, where $w(MST(G'))$ is the weight (sum of edge lengths) of an MST of G', we get

$$C_T \leq \alpha \frac{c_1}{u_1} \sum_{s_i \in S} dem_i \cdot \text{dist}(s_i, t) \; + \; \beta c_1 w(MST(G')).$$

The term $\frac{c_1}{u_1} \sum_{s_i \in S} \{dem_i \cdot \text{dist}(s_i, t)\}$ is a lower bound on C^* since dem_i must be routed a distance of at least $\text{dist}(s_i, t)$ and be charged at least at the rate $\frac{c_1}{u_1}$ per unit length. This lower bound is called the *routing lower bound*. In the general case when there are q cable types, $\frac{c_1}{u_1}$ is replaced by the cheapest rate $\frac{c_q}{u_q}$ in the bound. In addition, $\frac{1}{2} c_1\, w(MST(G'))$ is another lower bound on C^*, which is called the *MST lower bound*. The reasoning is as follows. We must connect the nodes in S to t and install at least one copy of the (cheapest) cable on each connecting edge. Then the cost of a Steiner tree with terminal set $S \cup \{t\}$ and cost $\text{dist}(e) \cdot c_1$ on each edge e is a lower bound and the length of an MST on $S \cup \{t\}$ times c_1 is within a factor 2 of this lower bound. These two lower bounds are essentially due to Mansour and Peleg [MP 94], although they study a different model. Thus using the two lower bounds we get $C_T \leq (\alpha + 2\beta)C^*$. Note that to obtain a solution in the original graph G, edges of T are replaced with the corresponding shortest paths in G. Clearly, the new solution also has cost C_T. ∎

Constructing an $(\alpha, 1 + \frac{2}{\alpha - 1})$-LAST as in [KRY 93] for $\alpha = 3$ gives the following corollary.

COROLLARY 2.3. *There is a 7-approximation algorithm for the single-sink edge installation problem with a single cable type.*

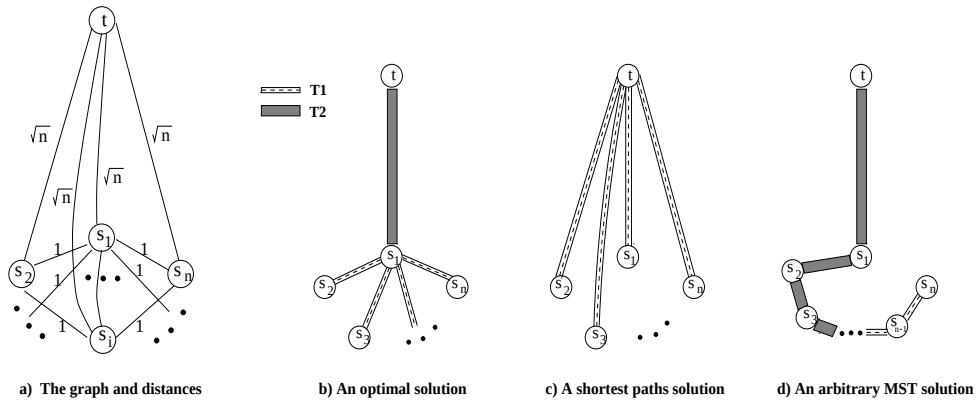


FIG. 2.1. *An example where naive heuristics produce poor solutions.*

Note that when all the nodes in G except the sink node are source nodes, routing through an (α, β)-LAST of G gives an $(\alpha + \beta)$-approximation. Then a $(2\sqrt{2} + 2)$-approximation is obtained using an $(\alpha, 1 + \frac{2}{\alpha - 1})$-LAST of [KRY 93] for $\alpha = \sqrt{2} + 1$.

For the general case of multiple cable types, routing through an (α, β)-LAST and buying as many copies of the cheapest (i.e., the thinnest) cable type as necessary provides an approximate solution with a worst-case bound of $(\alpha\sigma_{1q} + 2\beta)$ times the optimal cost (recall that σ_{1q} is the economies of scale factor between the thinnest and the thickest cables). However, in practice there are strong economies of scale between cable types. Hence, we focus on the case when σ_{1q} is large, possibly larger than polylogarithmic in the number of nodes; i.e., $\sigma_{1q} > (\log n)^{\Omega(1)}$.

2.4. Multiple cable type case—An example where naive heuristics produce poor solutions. Here is an example to show that heuristics based on routing through an MST, a shortest paths tree, or a LAST produce poor solutions. Suppose we have n source nodes $s_1, \ldots, s_n$ each with unit demand, at unit distance to each other and at distance $\sqrt{n}$ to the sink node t (see Figure 2.1(a)). There are only two types of cables, T1 and T2, where a T1 cable has capacity $u_1 = 1$ and costs $c_1 = 1$ per unit length, whereas a T2 cable has capacity $u_2 = n$ and costs $c_2 = \sqrt{n}$ per unit length. (Note that $\sigma_{12} = \sqrt{n}$.) An optimal solution with cost $2n - 1$ is obtained by installing a T2 cable for the edge (t, s_1) and using T1 cables to build a "star" centered at s_1 that has nodes $s_2, \ldots, s_n$ as leafs, i.e., by installing T1 cables on the edges $(s_1, s_2), \ldots, (s_1, s_n)$ (Figure 2.1(b)). A shortest paths tree (with root t) is a poor solution, since it has n edges of length $\sqrt{n}$, implying a cost of $n\sqrt{n}$, which is roughly $\sqrt{n}/2$ times the optimal cost (Figure 2.1(c)). An (arbitrary) MST is a poor solution: For example, the path $t, s_1, s_2, \ldots, s_n$ is an MST, and it requires at least $n - \sqrt{n}$ unit-length edges of capacity $\geq \sqrt{n}$ for a total cost $\geq n\sqrt{n}$, which is roughly $\sqrt{n}/2$ times the optimal cost (Figure 2.1(d)). (Although our optimal solution routes on an MST, this can be avoided by perturbing the distances.) Another heuristic is to use a spanner, or rather a LAST. However, a LAST based on the previous MST, i.e., the path $t, s_1, s_2, \ldots, s_n$, turns out to be even costlier than the MST. One such LAST has edges $(t, s_{i\sqrt{n}})$ and paths $s_{i\sqrt{n}}, s_{i\sqrt{n}+1}, \ldots, s_{(i+1)\sqrt{n}-1}$ for $i \in \{1, 2, \ldots, \sqrt{n}\}$.

3. Structure of solutions. It is possible to have a unique optimal solution such that the graph induced by edges with positive flow contains cycles (see Figure 3.1). However, we show in this section that there is a solution that induces a tree with cost

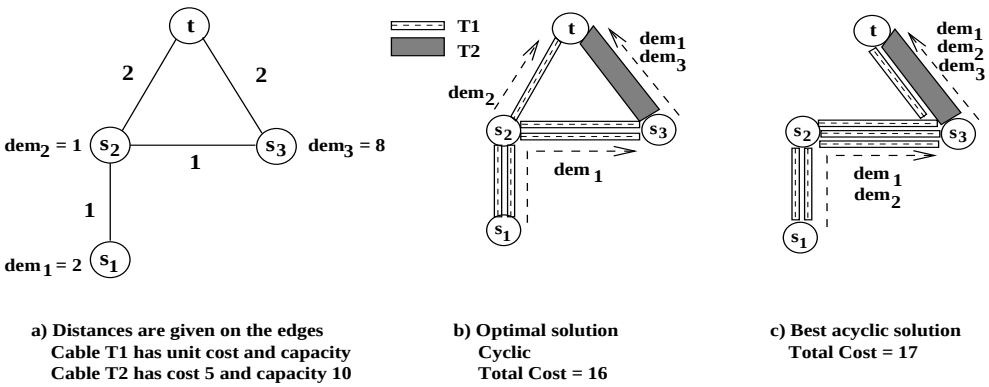


FIG. 3.1. *An example where the unique optimal solution is cyclic.*

at most twice the cost of an optimal solution. The key idea of the proof is to associate with every edge e chosen by an optimal solution an "adversary price" $\mathcal{C}_e$, where $\mathcal{C}_e$ is the length of e times the cheapest cost per unit capacity (per unit length) among all cable types installed on e by the optimal solution and to route the traffic through the cheapest $s_i - t$ paths with respect to the adversary prices.

Let (P^*, ν^*) be an optimal routing and choice of cables. Let $N^* = (V^*, A^*)$ be the graph induced by P^* and C^* be the cost of an optimal solution.

LEMMA 3.1. *Let κ_e be the index of the thickest cable installed on edge e in ν^*. That is, $\kappa_e = \max\{j : \nu_j^{e*} > 0\}$. Associate a cost $\mathcal{C}_e$ for each edge e in A^*, where $\mathcal{C}_e := \text{dist}(e) \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}}$. Then LB_1 defined below is a lower bound on the optimal cost, i.e.,*

$$C^* \geq LB_1 := \sum_{i=1}^{k} \left\{ dem_i \sum_{e \in P_i^*} \mathcal{C}_e \right\}.$$

Proof. The lemma follows since LB_1 corresponds to the cost of a solution where every commodity flows along an optimal path using "cheapest per capacity" cables of an optimal solution fractionally. More rigorously,

$$\begin{aligned} LB_1 &= \sum_{i=1}^{k} \left\{ dem_i \sum_{e \in P_i^*} \left(\text{dist}(e) \frac{c_{\kappa_e}}{u_{\kappa_e}} \right) \right\} \\ &= \sum_{e \in A^*} \left\{ \text{dist}(e) \cdot \left(\sum_{i=1}^{k} f_e^i \right) \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \right\} = \sum_{e \in A^*} \left\{ \text{dist}(e) \cdot f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \right\}. \end{aligned}$$

Let $k(1)$ to $k(l)$ be the indices of cables used in edge e in ν^*, where $k(l) = \kappa_e$. Without loss of generality we can allocate the flow induced by P^* on e, f_e, such that all but the thickest cable are saturated. Then $f_e = u_{k(1)}\nu_{k(1)} + u_{k(2)}\nu_{k(2)} + \cdots + u_{k(l)}(\nu_{k(l)} - 1) + rem_e$, where we use $\nu_{k(i)}$ as a shorthand notation for $\nu_{k(i)}^{e*}$ and rem_e for the remaining flow in the unsaturated cable. Then

$$f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} = \frac{u_{k(1)}}{u_{k(l)}} \cdot c_{k(l)}\nu_{k(1)} + \frac{u_{k(2)}}{u_{k(l)}} \cdot c_{k(l)}\nu_{k(2)} + \cdots + \frac{u_{k(l)}}{u_{k(l)}} \cdot c_{k(l)}(\nu_{k(l)} - 1) + \frac{rem_e}{u_{k(l)}} \cdot c_{k(l)}.$$

By economies of scale, $\frac{u_{k(i)}}{u_{k(l)}} \leq \frac{c_{k(i)}}{c_{k(l)}}$, $i = 1, \ldots, l-1$. Hence, replacing $\frac{u_{k(i)}}{u_{k(l)}}$ by $\frac{c_{k(i)}}{c_{k(l)}}$ for $i = 1, \ldots, l-1$ in the above equation gives

$$f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq c_{k(1)}\nu_{k(1)} + c_{k(2)}\nu_{k(2)} + \cdots + c_{k(l)}(\nu_{k(l)} - 1) + \frac{rem_e}{u_{k(l)}} \cdot c_{k(l)}.$$

Since $\frac{rem_e}{u_{k(l)}} \leq 1$, we have $f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq \sum_{i=1}^{l} c_{k(i)}\nu_{k(i)} = \sum_{j=1}^{q} c_j \nu_j^e$. Thus

$$LB_1 = \sum_{e \in A^*} \text{dist}(e) \cdot f_e \cdot \frac{c_{\kappa_e}}{u_{\kappa_e}} \leq \sum_{e \in A^*} \left\{ \text{dist}(e) \sum_{j=1}^{q} (c_j \cdot \nu_j^e) \right\} = C^*. \qquad \square$$

Now we are ready to prove the following theorem.

THEOREM 3.2. *There exists a routing P that induces a tree T with cable choices ν such that the cost of this solution denoted by C_T satisfies $C_T \leq 2\,C^*$.*

Proof. Let T be a shortest path tree of N^* rooted at t with respect to costs $\mathcal{C}_e$ on $e \in A^*$. Route each commodity i through the unique path P_i from s_i to t in T. For each edge $e \in P$ use as many copies of cable κ_e as necessary, i.e., $\nu_{\kappa_e} = \lceil \frac{f_e}{u_{\kappa_e}} \rceil$ copies of cable κ_e. Then the cost of this feasible solution is

$$C_T = \sum_{e \in T} \text{dist}(e) \cdot c_{\kappa_e} \cdot \nu_{\kappa_e}.$$

Since $\nu_{\kappa_e} \leq \frac{f_e}{u_{\kappa_e}} + 1$,

$$\begin{aligned} C_T &\leq \sum_{e \in T} \text{dist}(e) \cdot c_{\kappa_e} \cdot \frac{f_e}{u_{\kappa_e}} + \sum_{e \in T} \text{dist}(e) \cdot c_{\kappa_e} \\ &= \sum_{i=1}^{k} dem_i \sum_{e \in P_i} \text{dist}(e) \frac{c_{\kappa_e}}{u_{\kappa_e}} + \sum_{e \in T} \text{dist}(e) \cdot c_{\kappa_e}. \end{aligned}$$

As P_i's are shortest paths in N^* with costs $\mathcal{C}_e = \text{dist}(e)\frac{c_{\kappa_e}}{u_{\kappa_e}}$, $e \in A^*$, it follows that

$$C_T \leq \sum_{i=1}^{k} dem_i \sum_{e \in P_i^*} \mathcal{C}_e + \sum_{e \in T} \text{dist}(e) \cdot c_{\kappa_e}.$$

Therefore, the first summand above is at most LB_1 of Lemma 3.1. The second summand is also a lower bound on C^* since C^* includes the cost of at least one copy of the thickest cable on every edge in A^*, and $T \subset N^*$. Therefore, $C_T \leq 2\,C^*$. $\square$

Theorem 3.2 motivates the problem of finding a minimum-cost tree routing as an approximate solution to the general network design problem. However, note that the Steiner tree problem is a special case of the minimum-cost tree routing problem. Thus the minimum-cost tree network design problem is also NP-hard.

4. Euclidean case—An approximation algorithm. In this section, we present an algorithm for our network design problem in the case when the nodes are represented as points in the plane and the length function is the (Euclidean) distance. We have the following theorem.

THEOREM 4.1. *The Euclidean single-sink edge-installation problem can be approximated within a factor* $O(\min\{\log \frac{D}{u_1},\ \log \frac{\ell_{max}}{\ell_{min}}\})$, *where* D *is the total demand,* u_1 *is the capacity of the lowest-capacity cable,* ℓ_{max} *is the longest distance between a pair of nodes, and* ℓ_{min} *is the shortest distance between a pair of nodes.*

The approximation algorithm for the Euclidean case proceeds by successively gridding the plane and constructing the network hierarchically. The performance ratio is proven by collecting several layers of "moat-type" lower bounds and paying for the links laid in each layer of gridding using the lower bounds collected from that layer.

4.1. The algorithm. Let ℓ_{max} (ℓ_{min}) be the longest (shortest) distance between any pair of nodes. The topmost layer of gridding is a single square with minimum side length, centered at the sink node enclosing all the nodes, and hence has side length at most $2\ell_{max}$. We refine a square by partitioning it into four equal subsquares. We continue our refinement until every square in the lowest level of gridding either has side length at most $\ell_{max}/(D/u_1)$ (recall that D is the total demand) or contains at most one source. Thus the number of layers of gridding used overall is $O(\min\{\log (D/u_1),\ \log (\ell_{max}/\ell_{min})\})$.

Based on this gridding, the construction of the network is done recursively by routing all the flow through the centers of the squares at any layer. That is, flow within a square is aggregated at its center. In a generic recursive step, suppose that we have a square of side length 2ℓ that contains points with total demand Dem. This demand can be partitioned into Dem_j for $j = 1, 2, 3, 4$ in each of the four subsquares of side length ℓ into which this square is divided. Assume that each of these demands has been already routed to the center of the subsquare where it arises. We now sketch how to route these demands to the center of the bigger square one level up in the gridding.

If $Dem_j = 0$, then we do not build any edges between the center of square j and the center of the bigger square. Suppose $u_i \le Dem_j < u_{i+1}$ for some $i \in \{0, \ldots, q\}$. There are two cases. In the first case, $u_i \le Dem_j < \rho_i u_i$. We then install $\lceil \frac{Dem_j}{u_i} \rceil$ copies of cable type i from the center of square j to the center of the big square. These cables have length $\frac{\ell}{\sqrt{2}}$. In the second case, $\rho_i u_i \le Dem_j < u_{i+1}$ and we simply use a single copy of cable type $(i+1)$ to route the demand.

We have to be more careful in performing the recursive routing for demands that are near the sink. In particular, consider a demand that is very close to the sink in the northeastern quadrant of the first level of gridding. If this were the only demand, it is too expensive to route it to the center of the northeast square and then reroute it back to the sink. (See Figure 4.1(a).) We route the demand of any square with a corner at the sink directly to the sink. (See Figure 4.1(b).) Thus, under this scheme, if a node v is at a distance d from the sink, its demand can be routed to the center of a square of side length at most $O(d)$.

It is clear that the algorithm runs in polynomial time.

4.2. A moat lower bound. First we define a moat lower bound in its full generality, and then we apply it in the analysis of the above algorithm.

Consider a subset X of the node set V that excludes the sink node t. Let Dem be the total demand of the source nodes in X and let w be the minimum distance of any node in X to t; i.e., $w = \min_{x \in X} \text{dist}(x, t)$. A total of at least Dem flow has to travel a distance of at least w to reach the sink in any network. For any subset of nodes $X \subseteq V - \{t\}$, the ball around X of radius w defines a "moat" of width w separating demand Dem from the sink. The moat lower bound captures the cost of sending the entire

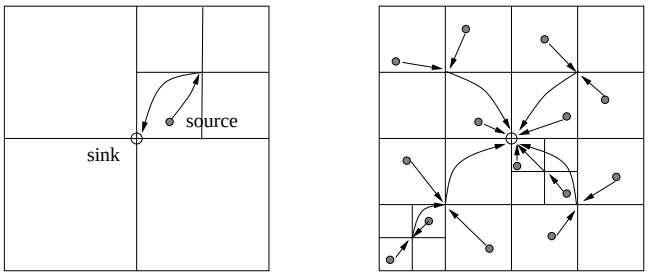


FIG. 4.1. *Part* (a) *illustrates that care is required in routing from the squares closest to the sink. Part* (b) *illustrates the recursive routing strategy. If a square at level* i *does not contain the sink at one of its corners, then we route its demand to the center of the square at level* $i-1$ *enclosing it. The demand of a square that contains the sink at one of its corners is routed directly to the sink.*

flow of value Dem a distance of w even after utilizing the economies of buying at bulk.

Suppose that $u_i \le Dem < u_{i+1}$ for some $i \in \{0, \ldots, q\}$, where we define $u_0 = 0$ and $u_{q+1} = \infty$ for convenience. Define the threshold multiplicity between the ith and the $(i+1)$st cable as $\rho_i = \frac{c_{i+1}}{c_i}$ for $i = 1, \ldots, q-1$. In addition, let $\rho_0 = 1$ and $\rho_q = \infty$. If $u_i \le Dem < \rho_i u_i$ for $i \in \{1, \ldots, q\}$, then a lower bound on the unit-length cable cost crossing this moat is $LB_K = \frac{Dem}{u_i} c_i$. The reason is that to pay the cheaper rate $\frac{c_{i+1}}{u_{i+1}}$, we must buy an integral number of cables and buying a copy of cable type i+1 is costlier than paying $\frac{Dem}{u_i} c_i$. On the other hand, if $\rho_i u_i \le Dem < u_{i+1}$ for $i \in \{0, \ldots, q-1\}$, then the lower bound is $LB_K = c_{i+1}$. As $\frac{Dem}{u_i} c_i \ge c_{i+1}$, buying any combination of cables 1 to i (by paying at least $\frac{Dem}{u_i} c_i$) will be more costly than a single cable of type i+1. We summarize the moat lower bound in the next proposition.

PROPOSITION 4.2. *For any node set* X *excluding* t*, of total demand* Dem*, at minimum distance* w *to* t*,* $LB_K \cdot w$ *is a lower bound on the optimal cost, where*

$$LB_K = \begin{cases} \dfrac{Dem}{u_i} c_i & \text{if } u_i \le Dem < \rho_i u_i \text{ for some } i \in \{1, \ldots, q\}, \\ c_{i+1} & \text{if } \rho_i u_i \le Dem < u_{i+1} \text{ for some } i \in \{0, \ldots, q-1\}. \end{cases}$$

We can also collect lower bounds from several disjoint moats.

PROPOSITION 4.3. *For any set of disjoint moats, the sum of the lower bounds generated by such moats is also a lower bound on the optimal value.*

Let us examine closely the definition of a moat in the Euclidean case. Let X be a set of nodes. A moat around X is defined by a closed line in R^2 that contains X in the inside and leaves the sink node outside. If the moat has width w, then the region of the plane occupied by the moat is the annular region including all points that are outside this line within Euclidean distance w from the line (Figure 4.2(a)). A collection of moats is *disjoint* if the regions occupied by any two moats in the collection do not intersect (Figure 4.2(b)).

4.3. The performance ratio. We now bound the worst-case performance of our algorithm. For each layer of gridding, we use a disjoint collection of moats and bound the cost of cables installed at that layer using the lower bounds accumulated from these moats.

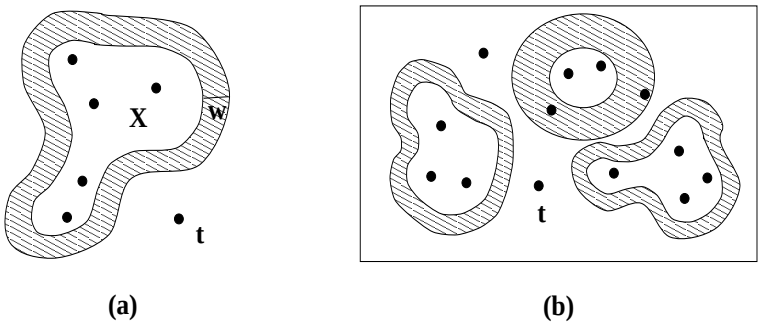


FIG. 4.2. *A moat around node set X of width w is shown in part* (a). *A disjoint moat collection is shown in part* (b).

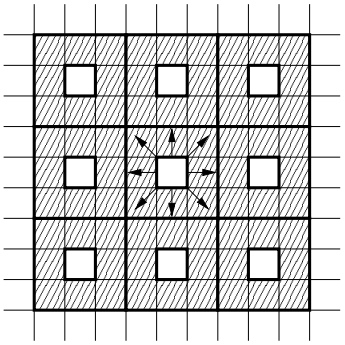


FIG. 4.3. *One class of disjoint moats in a given layer of gridding is shown. The arrows represent the* 8 *translation directions used to define the other* 8 *classes of this layer.*

For a given layer i where each square has side length ℓ, we consider the moat of width ℓ around every square in this layer except the four squares closest to the sink. Of course, a square generates a nonzero lower bound only if it contains nonzero demand. We can split the moats of this layer into 9 classes of disjoint moat collections. One such class is shown in Figure 4.3. Other classes can be obtained by translating all the squares enclosed by the moats in this class by one square in one of the 8 neighboring directions (see Figure 4.3). Let LB_i be the maximum lower bound generated by any of these 9 classes of moats of layer i. By averaging, LB_i is at least 1/9 of the sum of the lower bounds generated by all squares in this layer.

The cost of the cables constructed in one layer of gridding is at most $\sqrt{2}$ times the lower bound generated by all squares in this layer. If a square has demand Dem_j and side length ℓ, the lower bound it generates (LB) is ℓ times LB_K. There are two corresponding cases in the computation of LB_K and the choice of the cables in the algorithm. In the first case $Dem_j \geq u_i$. Thus $\lceil \frac{Dem_j}{u_i} \rceil \leq 2(\frac{Dem_j}{u_i})$ so that unit length cost of cables used is within a factor 2 of the lower bound LB_K defined in Proposition 4.2. Since the cables have length $\ell/\sqrt{2}$, the cost is at most $\sqrt{2}\ell\mathrm{LB}_K = \sqrt{2}\mathrm{LB}$. In the second case, we incur unit length cost of c_{i+1}, which is equal to LB_K. Thus the cost of cables used is $(\ell/\sqrt{2})\mathrm{LB}_K$, which is less than LB. As a result, the cables installed at layer i have cost at most $9\sqrt{2}\mathrm{LB}_i$.

Consider the last layer of gridding, with side length at most $\ell_{\max} u_1/D$. Note that the total cost incurred in routing all the demands to the centers of the squares in this layer is at most $O(\ell_{\max} c_1)$. This is because the demand of each source s_i, dem_i, in each square of the last gridding has to be sent a distance of at most $O(1)$ times the side length at unit length cost of at most $\lceil dem_i/u_1 \rceil c_1$. Since the MST lower bound is at least $1/2\ell_{\max} c_1$, it is clear that the costs incurred in the last layer of gridding can be charged to the MST lower bound (to prevent further gridding).

The number of layers of gridding is $O(\min\{\log \frac{D}{u_1}, \log \frac{\ell_{max}}{\ell_{min}}\})$, and the cost of the cables at each layer is at most $9\sqrt{2}$ times the optimal cost. Therefore, the cost of the solution constructed is within $O(\min\{\log \frac{D}{u_1}, \log \frac{\ell_{max}}{\ell_{min}}\})$ times the optimal cost.

5. Two-level networks. In this section we consider a simpler version of our problem when the network in which the demand is to be routed is two level; i.e., every source-sink path has at most two edges, and the distances arise from a general metric. This problem can be considered to limit the number of aggregation of flows. By requiring the network to be two level, every source node is restricted to aggregate its flow at most once. This special case of the network design problem has been addressed also in [Min 89].

We first show that this problem is as hard as the set cover problem and then give a $O(\log n)$-approximation algorithm.

PROPOSITION 5.1. *The minimum cost two-level network design problem is NP-hard. Furthermore, there is no polynomial time approximation algorithm for the two-level problem whose performance ratio is better than* $(1 - o(1)) \ln n$ *unless* $P = NP$.

Proof. The proof is by reduction from the set cover problem. Consider an instance of a set cover problem with a collection C of subsets of a finite set S, a weight $w(S_i) \in Z^+$ for each set S_i in C, and a positive integer K. The set cover problem is to determine if C contains a subset C' such that total weight of the sets in C' is at most K and every element of S belongs to at least one set in C'. We consider the following instance of the two-level network design problem. Construct the two-level input graph $G = (V, E)$ for the network design problem as follows. Let there be a node s_i corresponding to each element in S with unit demand and a node v_j for each set S_j in C with zero demand. In addition, V contains the sink node t. For each set S_j, E contains an edge of zero length between v_j and all the source nodes corresponding to the elements contained in S_j, as well as an edge of length $w(S_j)$ between v_j and t. Suppose one type of cable with unit cost per unit length and ∞ capacity is available. Then there exists a minimum-cost two-level network solution with cost at most K if and only if there is a set cover C' with weight at most K. Consider a two-level network solution with cost K or less. Since in this solution each demand must follow a single path of at most two edges, each source node is connected to one set containing it. Thus the collection of sets that sends a positive amount of flow to the sink is a cover with total weight at most K. Now let C' be a set cover with weight at most K. We can find a route for each source by connecting the source to one of the sets containing it, in C'. Clearly, the total cost of the cables installed will be at most K.

Note that the above reduction also is approximation preserving, so the current hardness results [F96, RS 97, AS 97] show that there is no polynomial time approximation algorithm for the two-level problem whose performance ratio is better than $(1 - o(1)) \ln n$ unless $P = NP$. □

We present the following result that is nearly best possible.

THEOREM 5.2. *The two-level link-installation problem can be approximated within*

a factor $O(\log n)$, where n is the number of nodes in the input graph.

The key idea of the proof of the above theorem is to define an appropriate (very large size) set cover problem. It is well known that the greedy algorithm yields logarithmically bounded approximate solutions for the set cover problem [Ch 79], but the crucial step in the algorithm is to find a greedy set. In our case, the problem of finding a single greedy choice is computationally hard, but we devise a constant factor approximation for this problem, thereby proving the above theorem; for more details on how a constant factor approximation for a greedy step yields a logarithmic approximation for the set cover problem, see, e.g., [YC 95, Thm. 10].

5.1. The corresponding set cover problem. The two-level problem can be modeled as a set covering problem as follows: The elements to be covered are the sources, each with a demand. The "sets" used to cover them are called *stars*. A star consists of a center (any node in the graph except the sink) and leaves that include the sink node and a subset of the sources. A star has cost equal to the total cost of the *cheapest* choice of cables to route the entire demand of the sources it contains via its center to the sink. Note that a star represents a level of aggregation of flows since the entire demand within a star that is aggregated at the center node is sent through more economical thick cables to the sink. A solution to the two-level problem naturally decomposes into a set of stars (one-level routes define starts with only one leaf, namely, the sink). Hence an optimal solution to the two-level problem is the same as an optimal solution to the set covering problem defined above.

5.2. Finding a greedy star. To implement an iteration of the greedy algorithm, we need to find a greedy star—a star of minimum ratio cost, of the ratio of total cable cost of the star divided by the total demand routed by the star. As there are exponentially many stars, we proceed by approximating the ratio cost within a constant factor. At any given step k of the greedy algorithm, let the total remaining demand to be routed be D^k. We first guess the total demand routed by the minimum ratio star at this stage within a factor of two, and for every such range, we find a cheapest star of roughly this much demand. Formally, consider the demand ranges $[1,2),[2,4),\ldots,[2^{\log \frac{D^k}{2}}, 2^{\log D^k}]$, and for every range, suppose we can compute the minimum *total* cost star whose demand falls within this range. Now suppose that the minimum *ratio* cost star routes a total of D_r demand at total cost C_r, where $D_r \in [2^i, 2^{i+1})$. Let C_i be the minimum cost of any star that routes demand in the range $[2^i, 2^{i+1})$ to the sink. Then the ratio cost of this star is near optimal. In particular, if this star routes demand D_i, then $\frac{C_i}{D_i} \leq \frac{2C_r}{D_r}$ since $C_i \leq C_r$ and $D_r \leq 2D_i$.

One last problem remains—that of finding a star of minimum cost that routes demand in a given range, say, $[2^i, 2^{i+1})$, to the sink. However, this requires solving integer min-knapsack problems by the following reduction. We first guess the center of the star; there are at most n guesses. Then, for each center node, we want to find a set of sources (to connect to the center) that have a total demand of at least 2^i, with total minimum cost of routing the demand of that set to the center. In this sense, we want to fill a knapsack of demand at least 2^i corresponding to this choice of center. The items used to fill the knapsack are the remaining sources s_i each with demand dem_i. The cost of an item s_i is the cost of routing dem_i from s_i to the center. If the edge connecting s_i to the center, say, e, has length $\text{dist}(e)$, this cost is $\text{dist}(e)$ times the value of another integer min-knapsack problem corresponding to the choice of cables for this edge. (Recall that given flow on an edge, finding minimum-cost cables

to cover this flow is a min-knapsack problem.)

The integer min-knapsack problem is NP-hard but can be solved in $O(qDem)$ time by dynamic programming [GG65]. Alternatively, we can transform this integer knapsack problem to a 0-1 knapsack problem with $\hat{n} = \sum_{j=1}^{q} \log\lceil \frac{dem_i}{u_j} \rceil = O(q \log D)$ binary variables in $O(\hat{n})$ time (see [MT 90]). Then applying any of the fully polynomial time approximation schemes for the 0-1 min-knapsack problem we obtain approximate solutions to our problem that obey the worst-case bounds for such schemes. For instance, a $(1+\epsilon)$-approximate solution can be obtained in $O(\hat{n}^2/\epsilon)$ time by the polynomial time approximation scheme of Gens and Levner [GL 79]. Alternatively, a 2-approximation solution can be obtained in $O(\hat{n} \log \hat{n})$ time by the greedy algorithm of Gens and Levner [GL 79].

To summarize, we estimate the costs of the different sources for a given choice of center using the knapsack approximation. Using these costs, we solve yet another knapsack problem that gives an approximately minimum cost of routing a total of at least 2^i demand to this center. The *total* cost of the star, however, must include the cost of routing the demand aggregated at the center to the sink. This is a problem also of choice of cables from the center to the sink for a given value of total demand and can be approximated using the knapsack framework.

We repeat this procedure for every demand range i and every choice of the center node. By choosing the star that has minimum ratio of total cost to demand among all iterations, we get a constant factor approximation to the minimum ratio cost star.

Note that the approximation factor for finding a minimum ratio cost star only multiplies the performance ratio of the greedy algorithm [YC 95]. Since we use a constant factor approximation for finding a minimum ratio star, we get the performance ratio claimed in Theorem 5.2.

6. Open problems. Theorem 3.2 motivates the problem of finding a minimum-cost tree routing as an approximate solution to the general network design problem. Although the problem is still NP-hard, approximating it may be easier than the general case.

In the more general minimum-cost capacitated network design problem arising in the telecommunications industry, special pieces of hardware called *concentrators* [BMW 95, Min 89] are required to aggregate the traffic from several thin cables in a single thick cable. We are given a list of concentrators of various types (inputs being a combination of cable types of total bandwidth equal to the output cable bandwidth), each with an associated fixed cost. Whenever traffic is aggregated, appropriate concentrators have to be used by paying the corresponding fixed cost. Moreover, traffic requirements may be specified between multiple sources and multiple sinks. As before, the flow must be indivisible and routed by purchasing integral copies of cables, whose cost versus capacity is a step function representing an economy of scale. The goal is to find the minimum total cost network.

Approximating this more general problem with concentrators remains open. Note, however, that the fixed costs of concentrators can be incorporated into the approximation algorithm for the 2-level network problem. For a given range of demand to be aggregated, the cost of the concentrator to be installed can be approximated. In the greedy step of the algorithm, after the bicriteria algorithm outputs a star of certain cost and demand, the approximate cost of the concentrator can be added to the total cost of the star.

Subsequent to the appearance of a preliminary version of this paper in [SCR+97], Awerbuch and Azar [AA97] gave a randomized $O(\log^2 n)$-approximation algorithm for

the general edge installation problem with many cable types and many sources and sinks.

REFERENCES

[AKR 95] A. Agrawal, P. Klein, and R. Ravi, *When trees collide: An approximation algorithm for the generalized Steiner problem on networks*, SIAM J. Comput., 24 (1995), pp. 440–456.

[ADD+92] I. Althöfer, G. Das, D. Dobkin, D. Joseph, and J. Soares, *On sparse spanners of weighted graphs*, Discrete Comput. Geom., 9 (1993), pp. 81–100.

[AS 97] S. Aurora and M. Sudan, *Improved low degree testing and its applications*, in Proceedings of the 29th ACM Annual Symposium on Theory of Computing, El Paso, TX, 1997, pp. 485–495.

[AA97] B. Awerbuch and Y. Azar, *Buy at bulk network design*, in Proceedings of the 38th Annual IEEE Symposium on Foundations of Computer Science, Miami Beach, FL, 1997, pp. 542–547.

[BMW 95] A. Balakrishnan, T. Magnanti, and R. T. Wong, *A decomposition algorithm for local access telecommunications network expansion planning*, Oper. Res., 43 (1995), pp. 58–76.

[Bal 61] M. L. Balinski, *Fixed cost transportation problems*, Naval Res. Logist. Quart., 8 (1961), pp. 41–54.

[Bar 96] F. Barahona, *Network design using cut inequalities*, SIAM J. Optim., 6 (1996), pp. 823–837.

[BCGT 98] D. Bienstock, S. Chopra, O. Gunluk, and C.-Y. Tsai, *Minimum cost capacity installation for multicommodity network flows*, Math. Programming, 81 (1998), pp. 177–199.

[BG 96] D. Bienstock and O. Gunluk, *Capacitated network design—Polyhedral structure and computation*, INFORMS J. Comput., 8 (1996), pp. 243–259.

[Ch 79] V. Chvátal, *A greedy heuristic for the set-covering problem*, Math. Oper. Res., 4 (1979), pp. 233–235.

[F96] U. Feige, *A threshold of ln n for approximating set cover*, in Proceedings of the 28th ACM Annual Symposium on Theory of Computing, Philadelphia, PA, 1996, pp. 314–318.

[GSS80] G. Gallo, C. Sandi, and C. Sodini, *An algorithm for the min concave cost flow problem,* European J. Oper. Res., 4 (1980), pp. 248–255.

[GJ 79] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco, 1979.

[GL 79] G. V. Gens and E. V. Levner, *Computational complexity of approximation algorithms for combinatorial problems*, in Mathematical Foundations of Computer Science, Lecture Notes in Comput. Sci. 74, 1979, pp. 292–300.

[GG65] P. C. Gilmore and R. E. Gomory, *Multi-stage cutting stock problems of two and more dimensions*, Oper. Res., 13 (1965), pp. 94–120.

[GW 95] M. X. Goemans and D. P. Williamson, *A general approximation technique for constrained forest problems*, SIAM J. Comput., 24 (1995), pp. 296–317.

[GR 71] M. Goldstein and B. Rothfarb, *The one terminal telepak problem*, Oper. Res., 19 (1971), pp. 156–169.

[G71] P. Gray, *Exact solution of the fixed charge transportation problem*, Oper. Res., 19 (1971), pp. 1529–1538.

[KRY 93] S. Khuller, B. Raghavachari, and N. E. Young, *Balancing minimum spanning and shortest path trees*, Algorithmica, 14 (1993), pp. 305–322.

[Kle 96] J. Kleinberg, *Single-source unsplittable flow*, in Proceedings of the 37th Annual IEEE Symposium on Foundations of Computer Science, Burlington, VT, 1996, pp. 68–77.

[Lue 75] G. S. Lueker, *Two NP-Complete Problems in Nonnegative Integer Programming*, Tech. report 178, Computer Science Laboratory, Princeton University, Princeton, NJ, 1975.

[MMV 95] T. L. Magnanti, P. Mirchandani, and R. Vachani, *Modeling and solving the two-facility capacitated network loading problem*, Oper. Res., 43 (1995), pp. 142–157.

[MP 94] Y. Mansour and D. Peleg, *An Approximation Algorithm for Minimum-Cost Network Design*, Tech. report CS94-22, The Weizman Institute of Science, Rehovot, Israel, 1994.

[MT 90] S. Martello and P. Toth, *Knapsack Problems: Algorithms and Computer Implemen-*

tations, John Wiley & Sons, Chichester, UK, 1990.

[Min 89] M. Minoux, *Network synthesis and optimum network design problems: Models, solution methods and applications*, Networks, 19 (1989), pp. 313–360.

[RMR+93] R. Ravi, M. V. Marathe, S. S. Ravi, D. J. Rosenkrantz, and H. B. Hunt, *Many birds with one stone: Multi-objective approximation algorithms*, in Proceedings of the 25th ACM Symposium on Theory of Computing, San Diego, CA, 1993, pp. 438–448.

[RS 97] R. Raz and S. Safra, *A sub-constant error-probability low-degree test, and a sub-constant error-probability pcp characterization of np*, in Proceedings of the 29th Annual ACM Symposium on Theory of Computing, El Paso, TX, 1997, pp. 314–318.

[SCR+97] F.S. Salman, J. Cherijan, R. Ravi, and S. Subramanian, *Buy-at-bulk network design: Approximating the single-sink edge installation problem*, in Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, New Orleans, LA, 1997, pp. 619–628.

[WGM+95] D. P. Williamson, M. X. Goemans, M. Mihail, and V. V. Vazirani, *A primal-dual approximation algorithm for generalized Steiner network problems*, Combinatorica, 15 (1995), pp. 435–454.

[YC 95] B. Yu and J. Cheriyan, *Approximation algorithms for feasible cut and multicut problems*, in Proc. Algorithms—ESA'95, Third Annual European Symposium, P. Spirakis, ed., Lecture Notes in Comput. Sci. 979, Springer, New York, 1995, pp. 394–408.

[Z68] W. I. Zangwill, *Minimum concave cost flows in certain networks*, Mgmt. Sci., 14 (1968), pp. 429–450.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 611–646

A NEW MATRIX-FREE ALGORITHM FOR THE LARGE-SCALE TRUST-REGION SUBPROBLEM*

MARIELBA ROJAS†, SANDRA A. SANTOS‡, AND DANNY C. SORENSEN§

Abstract. We present a new method for the large-scale trust-region subproblem. The method is matrix-free in the sense that only matrix-vector products are required. We recast the trust-region subproblem as a parameterized eigenvalue problem and compute an optimal value for the parameter. We then find the solution of the trust-region subproblem from the eigenvectors associated with two of the smallest eigenvalues of the parameterized eigenvalue problem corresponding to the optimal parameter. The new algorithm uses a different interpolating scheme than existing methods and introduces a unified iteration that naturally includes the so-called hard case. We show that the new iteration is well defined and convergent at a superlinear rate. We present computational results to illustrate convergence properties and robustness of the method.

Key words. regularization, constrained quadratic optimization, trust region, Lanczos method

AMS subject classifications. 65F15, 65G05

PII. S105262349928887X

1. Introduction.

An important problem in optimization and linear algebra is the *trust-region subproblem*: minimize a quadratic function subject to an ellipsoidal constraint,

$$\min \frac{1}{2}x^T Ax + g^T x \quad \text{subject to (s.t.)} \quad \|x\|_2 \leq \Delta,$$

where $A \in \mathbb{R}^{n\times n}$, $A = A^T$; $x, g \in \mathbb{R}^n$ and $\Delta > 0$. Two significant applications of this basic problem are the regularization or smoothing of discrete forms of ill-posed problems and the trust-region globalization strategy used to force convergence in optimization methods.

A solution x_* to the problem must satisfy an equation of the form $(A + \mu I)x_* = -g$ with $\mu \geq 0$. The parameter μ is the Tikhonov regularization parameter for ill-posed problems and the Levenberg–Marquardt parameter in optimization. The constraint might also involve a matrix $C \neq I$, where C is often constructed to impose a smoothness condition on the solution x_* for ill-posed problems and to incorporate scaling of the variables in optimization. We will not treat this case explicitly here. However, in many applications the matrix C will be nonsingular and therefore with a change of variables we can reduce the problem to the case we are considering.

*Received by the editors September 23, 1999; accepted for publication (in revised form) June 20, 2000; published electronically November 10, 2000.
http://www.siam.org/journals/siopt/11-3/28887.html

†Department of Computational and Applied Mathematics, Rice University, 6100 Main St., Houston, TX 77005-1892 (mrojas@caam.rice.edu). Current address: CERFACS, 42, Avenue Gaspard Coriolis, 31057 Toulouse CEDEX 1, France (mrojas@cerfacs.fr). The research of this author was supported in part by NSF cooperative agreement CCR-9120008 and by the Research Council of Norway.

‡Department of Applied Mathematics, State University of Campinas, CP 6065, 13081-970, Campinas, SP, Brazil (sandra@ime.unicamp.br). The research of this author was supported by FAPESP (93/4907-5), CNPq, FINEP, and FAEP-UNICAMP.

§Department of Computational and Applied Mathematics, Rice University, 6100 Main St., Houston, TX 77005-1892 (sorensen@caam.rice.edu). The research of this author was supported in part by NSF grant CCR-9988393 and in part by the Los Alamos National Laboratory Computer Science Institute (LACSI) through LANL contract 03891-99-23 as part of the prime contract (W-7405-ENG-36) between the Department of Energy and the Regents of the University of California.

If we can afford to compute the Cholesky factorization of matrices of the form $A + \mu I$, then the method proposed by Moré and Sorensen (cf. [10]) is the method of choice to solve the problem. However, in many important applications, factoring or even forming these matrices is prohibitive. This has motivated the development of matrix-free methods that rely only on matrix-vector products. The first method in this class is the method of Steihaug [18] which computes the solution to the problem in a Krylov subspace. This method is very efficient in conjunction with optimization methods; however, it does not compute an optimal solution and cannot handle a special situation known as the *hard case*, which we will describe later. New methods based on matrix-vector products are the ones by Golub and von Matt [3], Sorensen [17], Rendl and Wolkowicz [13], and Pham Dinh and Hoai An [12]. Recently, Lucidi, Palagi, and Roma [8] presented new properties of the trust-region subproblem that provide useful tools for the development of new classes of algorithms for this problem in the large-scale context. Recently, we became aware of a new method proposed by Hager [5], where an SQP approach is used to solve the trust-region subproblem.

Golub and von Matt [3] base their algorithm on the theory of Gauss quadrature and do not include in their analysis the possibility of the hard case. Pham Dinh and Hoai An [12] develop an algorithm based on the difference of convex functions. Their strategy is very inexpensive, due to its projective nature, but needs a restarting mechanism to ensure convergence to a global solution. The approaches of Sorensen [17] and Rendl and Wolkowicz [13] recast the trust-region subproblem as a parameterized eigenvalue problem and design an iteration to find an optimal value for the parameter. The idea of formulating the trust-region subproblem in terms of an eigenvalue problem is also exploited in Gander, Golub, and von Matt [1]. Rendl and Wolkowicz present a primal-dual semidefinite framework for the trust-region subproblem, where a dual simplex-type method is used in the basic iteration and a primal simplex-type method provides steps for the hard-case iteration. At each iteration, the method computes the smallest eigenvalue and corresponding eigenvector of the parameterized problem using a block Lanczos routine. Sorensen's algorithm provides a superlinearly convergent scheme to adjust the parameter and finds the optimal vector x_* from the eigenvector of the parameterized problem, as long as the hard case does not occur. For the hard case, Sorensen's algorithm is linearly convergent. The algorithm uses the implicitly restarted Lanczos method (IRLM) (cf. [16]) to compute the smallest eigenvalue and corresponding eigenvector of the parameterized problem. The IRLM is particularly suitable for large-scale applications since it has low and fixed storage requirements and relies only on matrix-vector products.

In this work we present a new matrix-free algorithm for the large-scale trust-region subproblem. Our algorithm is similar to those proposed in [13, 17] in the sense that we solve the trust-region subproblem through a parameterized eigenvalue problem, but it differs from those approaches in that we do not need two different schemes for the standard case and the hard case. Our algorithm can handle all the cases in the same basic iteration. We achieved this improvement over the methods in [13, 17] by computing two eigenpairs of the parameterized problem and incorporating information about the second eigenpair whenever it is appropriate. This strategy does not substantially increase the computational cost over the method in [17]. We introduce a two-point interpolating scheme that is different from the one in [17]. We show that this new iteration is also convergent and that the convergence rate is superlinear. Moreover, our convergence results naturally include the hard case, since no special iterations are necessary. Such a unified approach is not achieved in either [13] or [17].

The organization of this work is the following. In section 2 we analyze the structure of the problem and motivate the algorithm. In section 3 we give a complete characterization of the hard case with respect to the parameterized eigenproblems. We describe the algorithm in detail in section 4. In section 5 we present the convergence analysis. We describe preliminary numerical experiments in section 6 and present some conclusions in section 7.

2. Structure of the problem. The problem we are interested in solving is

$$\begin{array}{ll} \min & \psi(x) \\ \text{s.t.} & \|x\| \leq \Delta, \end{array} \tag{2.1}$$

where $\psi(x) = \frac{1}{2}x^T Ax + g^T x$; A, g as before and $\|\cdot\| \equiv \|\cdot\|_2$ throughout the paper.

Due to the structure of (2.1), its optimality conditions are both necessary and sufficient, as stated in the next lemma, where we follow [17] in the nonstandard but notationally more convenient use of a nonpositive multiplier.

LEMMA 2.1 (see [15]). *A feasible vector x_* is a solution to* (2.1) *with corresponding Lagrange multiplier λ_* if and only if x_*, λ_* satisfy $(A - \lambda_* I)x_* = -g$ with $A - \lambda_* I$ positive semidefinite, $\lambda_* \leq 0$, and $\lambda_*(\Delta - \|x_*\|) = 0$.*

Proof. For the proof see [15]. □

In order to design efficient methods for solving problem (2.1) we must exploit the tremendous amount of structure of this problem. In particular, the optimality conditions are computationally attractive since they provide a means to reduce the given n-dimensional constrained optimization problem into a zero-finding problem in a single scalar variable. For example, we could define the function $\varphi(\lambda) = \|(A - \lambda I)^{-1}g\|$ and solve the *secular equation* $\varphi(\lambda) = \Delta$, monitoring λ to be no greater than the smallest eigenvalue of A, so that the Cholesky factorization of $A - \lambda I$ is well defined. Using Newton's method to solve $\frac{1}{\varphi(\lambda)} - \frac{1}{\Delta} = 0$ has a number of computationally attractive features (cf. [10]) and we should use this approach when we can afford to compute the Cholesky factorization of $A - \lambda I$. When computing a Cholesky factorization is too expensive, we need to use a different strategy. The introduction of a new parameter will make it possible to convert the original trust-region subproblem into a scalar problem that is suitable for the large-scale setting. The conversion amounts to embedding the given problem into a parameterized bordered matrix eigenvalue problem. Consider the *bordered* matrix

$$B_\alpha = \begin{pmatrix} \alpha & g^T \\ g & A \end{pmatrix}$$

and observe that

$$\frac{\alpha}{2} + \psi(x) = \frac{1}{2}(1, x^T)B_\alpha \begin{pmatrix} 1 \\ x \end{pmatrix}.$$

Therefore, there exists a value of the parameter α such that we can rewrite problem (2.1) as

$$\begin{array}{ll} \min & \frac{1}{2}y^T B_\alpha y \\ \text{s.t.} & y^T y \leq 1 + \Delta^2,\ e_1^T y = 1, \end{array} \tag{2.2}$$

where e_1 is the first canonical unit vector in $\mathbb{R}^{n+1}$. This formulation suggests that we can find the desired solution in terms of an eigenpair of B_α in the following way.

Suppose that $\{\lambda, (1, x^T)^T\}$ is an eigenpair of B_α. Then

$$\begin{pmatrix} \alpha & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix} = \begin{pmatrix} 1 \\ x \end{pmatrix} \lambda \,,$$

which is equivalent to

$$\alpha - \lambda = -g^T x \tag{2.3}$$

and

$$(A \ - \ \lambda \ I)x = -g \,. \tag{2.4}$$

Now, let $\delta_1, \delta_2, \ldots, \delta_d$ be the *distinct* eigenvalues of A in nondecreasing order. Then

$$\alpha - \lambda = -g^T x = \sum_{j=1}^{d} \frac{\beta_j^2}{\delta_j - \lambda}, \tag{2.5}$$

where β_j^2 is the sum of the squares of the expansion coefficients of g in the eigenvector basis, corresponding to all the eigenvectors associated with δ_j.

Observe that as a consequence of Cauchy's interlace theorem (cf. [11, p. 186]), and also from (2.5), the eigenvalues of A interlace the eigenvalues of B_α. In particular, if $\lambda_1(\alpha)$ is the smallest eigenvalue of B_α, then $\lambda_1(\alpha) \leq \delta_1$. This implies that the matrix $A - \lambda_1(\alpha)I$ is always positive semidefinite independently of the value of α.

Equations (2.3)–(2.4) express λ and hence x implicitly in terms of α, suggesting the definition of a convenient function as follows. Let † denote the pseudoinverse of a matrix and let us define

$$\phi(\lambda) \equiv g^T (A \ - \ \lambda \ I)^\dagger g = -g^T x.$$

Therefore,

$$\phi'(\lambda) = g^T [(A \ - \ \lambda \ I)^\dagger]^2 g = x^T x,$$

where differentiation is with respect to λ, and x satisfies $(A \ - \ \lambda \ I)x = -g$. The function ϕ appears in many contexts [2, 9, 19, 20] and Figure 1(a) shows its typical behavior. It is worth noticing that the values of ϕ and ϕ' at an eigenvalue λ of B_α are readily available and contain valuable information with respect to problem (2.1), as long as λ has a corresponding eigenvector with nonzero first component.

Finding the smallest eigenvalue and a corresponding eigenvector of B_α for a given value of α, and then normalizing the eigenvector to have its first component equal to one, will provide a means to evaluate the rational function ϕ and its derivative at appropriate values of λ, namely, at $\lambda = \lambda_1(\alpha) \leq \delta_1$. Moreover, $\lambda_1(\alpha)$ is usually well separated from the rest of the spectrum of B_α, particularly for small values of Δ. In these cases, we expect a Lanczos-type method to be very efficient in computing this eigenvalue and the corresponding eigenvector. If we can adjust α so that the corresponding x satisfies $x^T x = \phi'(\lambda) = \Delta^2$ with $\alpha - \lambda = \phi(\lambda)$, then

$$(A \ - \ \lambda \ I)x = -g \quad \text{and} \quad \lambda(\Delta - \|x\|) = 0$$

with $A \ - \ \lambda \ I$ positive semidefinite. If $\lambda \leq 0$, then x is a boundary solution for the trust-region subproblem. In case we find $\lambda > 0$ with $\|x\| < \Delta$ during the course of

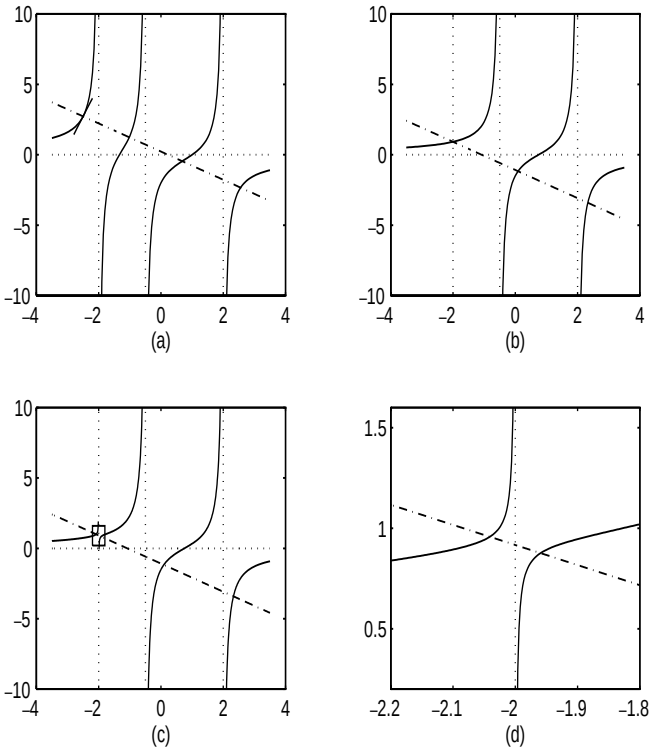


FIG. 1. *Example of the typical pattern of* $\phi(\lambda)$ *(solid) and the straight line* $f(\lambda) = \alpha_* - \lambda$ *(dash-dotted). The three smallest eigenvalues of* A *are* -2, -0.5, *and* 2. (a) *General case with the slope at* λ_* *also plotted;* (b) *exact hard case;* (c) *near hard case;* (d) *detail of box in* (c).

adjusting α, then this implies that the matrix A is positive definite and that $\|A^{-1}g\| < \Delta$. As shown in [10], these two conditions imply that problem (2.1) has an interior solution that satisfies $Ax = -g$.

The availability of the values λ, $\phi(\lambda)$, $\phi'(\lambda)$ makes it possible to use rational interpolation to adjust the parameter using these values as interpolation points. The adjustment of α by means of rational interpolation consists of constructing a rational interpolant $\widehat{\phi}$ and finding a special point $\widehat{\lambda}$ such that $\widehat{\phi}'(\widehat{\lambda}) = \Delta^2$. We then compute the new parameter as $\alpha_+ = \widehat{\lambda} + \widehat{\phi}(\widehat{\lambda})$. In this approach it is necessary to safeguard α_+ to ensure convergence of the iteration. This idea was discussed in [6, 15] and used in [17]. The algorithm in this paper follows this approach.

3. Characterization of the hard case. We assumed in the previous discussion that the smallest eigenvalue of B_α had a corresponding eigenvector with nonzero first component. It remains to consider the possibility that all the eigenvectors associated with $\lambda_1(\alpha)$ have first component zero so that we cannot normalize any of them to have its first component equal to one. In this case, the proposed strategy for solving problem (2.1) breaks down. However, this can happen only when g is orthogonal to $\mathcal{S}_1$, where $\mathcal{S}_j = \{q \mid Aq = \delta_j q\}$, $j = 1, 2, \ldots, d$.

The condition $g \perp \mathcal{S}_1$ is a necessary condition for the occurrence of the so-called hard case. Therefore, we call this situation a potential hard case. Observe that in a potential hard case δ_1 is no longer a pole of ϕ, as Figure 1(b) illustrates. We discuss

the hard case in detail at the end of this section. At this point we will concentrate on the potential hard case, which has intriguing consequences. We will show that in a potential hard case, for all values of α greater than certain critical value $\tilde{\alpha}_1$, all the eigenvectors corresponding to the smallest eigenvalue of B_α will have first component zero. We will also show that for any α, there is always a well-defined eigenvector of B_α, depending continuously on α, that we can safely normalize to have first component one. If $g \not\perp \mathcal{S}_1$ or $g \perp \mathcal{S}_1$ and $\alpha \leq \tilde{\alpha}_1$, then this eigenvector corresponds to $\lambda_1(\alpha)$. If $g \perp \mathcal{S}_1$ and α exceeds the critical value $\tilde{\alpha}_1$ by a small amount, this parameterized vector corresponds to the second smallest eigenvalue of B_α. A complete understanding of this case leads to the main algorithm of this paper. The following results are the basis for this understanding.

LEMMA 3.1. *For any $\alpha \in \mathbb{R}$ and any $q \in \mathcal{S}_j$, $1 \leq j \leq d$, $\{\delta_j, (0, q^T)^T\}$ is an eigenpair of B_α if and only if g is orthogonal to $\mathcal{S}_j$.*

Proof. The proof follows from the observation that $g \perp \mathcal{S}_j$ and $Aq = \delta_j q$ are equivalent to

$$\begin{pmatrix} \alpha & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 0 \\ q \end{pmatrix} = \delta_j \begin{pmatrix} 0 \\ q \end{pmatrix}. \qquad \square$$

If $\mathcal{Z}_1(\alpha)$ is the eigenspace of B_α corresponding to δ_1, Lemma 3.1 establishes that the set $\{(0, q^T)^T \mid q \in \mathcal{S}_1\}$ is a subset of $\mathcal{Z}_1(\alpha)$. Note that while $\mathcal{S}_1$ corresponds to the smallest eigenvalue of A, $\mathcal{Z}_1(\alpha)$ does not necessarily correspond to the smallest eigenvalue of B_α. These subspaces have the same dimension for all but one exceptional value of α. The following result states that there is a unique value of α for which $\dim \mathcal{Z}_1(\alpha) = \dim \mathcal{S}_1 + 1$.

LEMMA 3.2. *Suppose that g is orthogonal to $\mathcal{S}_j$, $1 \leq j \leq d$, and let $p_j = -(A - \delta_j I)^\dagger g$. The pair $\{\delta_j, (1, p_j^T)^T\}$ is an eigenpair of B_α if and only if $\alpha = \tilde{\alpha}_j$, where $\tilde{\alpha}_j = \delta_j - g^T p_j$.*

Proof. First we observe that $g \perp \mathcal{S}_j$ implies that $g \in \mathcal{R}(A - \delta_j I)$ and therefore

$$(A - \delta_j I)p_j = -(A - \delta_j I)(A - \delta_j I)^\dagger g = -g, \tag{3.1}$$

since $(A - \delta_j I)(A - \delta_j I)^\dagger$ is an orthogonal projector onto $\mathcal{R}(A - \delta_j I)$.

Now, let $\alpha = \tilde{\alpha}_j$. Then

$$\begin{pmatrix} \tilde{\alpha}_j & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ p_j \end{pmatrix} = \begin{pmatrix} \tilde{\alpha}_j + g^T p_j \\ g + A p_j \end{pmatrix} = \delta_j \begin{pmatrix} 1 \\ p_j \end{pmatrix},$$

since by definition of $\tilde{\alpha}_j$ we have $\tilde{\alpha}_j + g^T p_j = \delta_j$ and by (3.1), $g + A p_j = \delta_j p_j$.

Suppose now that $\{\delta_j, (1, p_j^T)^T\}$ is an eigenpair of B_α, i.e.,

$$\begin{pmatrix} \alpha & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ p_j \end{pmatrix} = \delta_j \begin{pmatrix} 1 \\ p_j \end{pmatrix}.$$

It follows directly from this relationship that $\alpha = \tilde{\alpha}_j = \delta_j - g^T p_j$. $\square$

The following corollary summarizes the main results from Lemmas 3.1 and 3.2.

COROLLARY 3.1. *Suppose that g is orthogonal to $\mathcal{S}_j$, $1 \leq j \leq d$, and let $\mathcal{Z}_j(\alpha) = \{z \in \mathbb{R}^{n+1} \mid B_\alpha z = \delta_j z\}$. If $\tilde{\alpha}_j = \delta_j + g^T p_j$ with $p_j = -(A - \delta_j I)^\dagger g$, then $\dim \mathcal{Z}_j(\tilde{\alpha}_j) = \dim \mathcal{S}_j + 1$ and for any other value of α, $\dim \mathcal{Z}_j(\alpha) = \dim \mathcal{S}_j$. Moreover, if m_j is the multiplicity of δ_j and $\{q_1, \ldots, q_{m_j}\}$ is an orthogonal basis for $\mathcal{S}_j$, then*

$$\left\{ \begin{pmatrix} 1 \\ p_j \end{pmatrix}, \begin{pmatrix} 0 \\ q_1 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ q_{m_j} \end{pmatrix} \right\}$$

is an orthogonal basis for $\mathcal{Z}_j(\tilde{\alpha}_j)$ *and*

$$\left\{ \begin{pmatrix} 0 \\ q_1 \end{pmatrix}, \ldots, \begin{pmatrix} 0 \\ q_{m_j} \end{pmatrix} \right\}$$

is an orthogonal basis for $\mathcal{Z}_j(\alpha)$, *for* $\alpha \neq \tilde{\alpha}_j$.

The result in Lemma 3.1 was also stated in [17]; the idea behind Lemma 3.2 was presented in [13]. We present here a general formulation of these results given in [14]. In the next results from [14], we establish that there always exists an eigenvector of B_α that we can normalize to have first component equal to one, and we characterize the eigenvalue to which this eigenvector corresponds.

THEOREM 3.1 (see [14]). *Let* $\lambda(\alpha)$ *be the smallest solution of the equation*

$$\phi(\lambda) = \alpha - \lambda.$$

Then, for any value of α, $\lambda(\alpha)$ *is an eigenvalue of* B_α *with a corresponding eigenvector that can be normalized to have first component one.*

Proof. Suppose first that g is orthogonal to $\mathcal{S}_i$, $i = 1, 2, \ldots, \ell$, with $1 \leq \ell < d$. Then

$$\begin{aligned} \phi(\lambda) &= g^T (A - \lambda I)^\dagger g \\ &= \sum_{j=\ell+1}^{d} \frac{\beta_j^2}{\delta_j - \lambda}. \end{aligned}$$

Let $\lambda(\alpha)$ be the smallest solution of the equation $\phi(\lambda) = \alpha - \lambda$. Then $\lambda(\alpha) \in (-\infty, \delta_{\ell+1})$. Since $\phi(\lambda)$ is strictly increasing on its domain and $f(\lambda) = \alpha - \lambda$ is a decreasing straight line, we conclude that $\lambda(\alpha)$ is unique. Since $\lambda(\alpha)$ depends continuously on α, so does $p(\alpha) = -(A - \lambda(\alpha) I)^\dagger g$ and also $v(\alpha) = (1, p(\alpha)^T)^T$. Let us see now that $v(\alpha)$ is an eigenvector of B_α associated with $\lambda(\alpha)$. Consider

$$\begin{pmatrix} \alpha & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ p(\alpha) \end{pmatrix} = \begin{pmatrix} \alpha + g^T p(\alpha) \\ g + Ap(\alpha) \end{pmatrix}$$

and note that

$$\begin{aligned} \alpha + g^T p(\alpha) &= \alpha - \phi(\lambda(\alpha)) \\ &= \lambda(\alpha) \quad \text{by definition of } \lambda(\alpha). \end{aligned}$$

Now, $g \perp \mathcal{S}_i$, $i = 1, 2, \ldots, \ell$, implies that $g \in \mathcal{R}(A - \lambda I)$ for $\lambda \in (-\infty, \delta_{\ell+1})$. Thus, $g \in \mathcal{R}(A - \lambda(\alpha) I)$ and we have $(A - \lambda(\alpha) I)p(\alpha) = -g$. It follows that

$$g + Ap(\alpha) = \lambda(\alpha)p(\alpha)$$

and therefore, $B_\alpha v(\alpha) = \lambda(\alpha)v(\alpha)$.

Suppose now that g is not orthogonal to $\mathcal{S}_1$. Then $\lambda(\alpha) \in (-\infty, \delta_1)$ and this implies $A - \lambda(\alpha) I$ is nonsingular and the previous proof holds with $p(\alpha) = -(A - \lambda(\alpha) I)^{-1} g$. ∎

The following result characterizes the smallest $\ell + 1$ distinct eigenvalues of B_α if g is orthogonal to the eigenspaces corresponding to the smallest ℓ distinct eigenvalues of A. In case g is not orthogonal to $\mathcal{S}_1$, then the lemma characterizes the smallest eigenvalue of B_α. We will denote by $\lambda_j(\alpha)$, $j = 1, 2, \ldots, n + 1$, the eigenvalues of B_α in nondecreasing order.

LEMMA 3.3 (see [14]). *Let* $\{\lambda(\alpha),\ v(\alpha)\}$ *be the eigenpair of* B_α *given by Theorem* 3.1 *and define* $\tilde{\alpha}_i$ *as in Lemma* 3.2. *Then if* $g \not\perp \mathcal{S}_1$, *then* $\lambda_1(\alpha) = \lambda(\alpha)$.

If $g \perp \mathcal{S}_k$, *for* $k = 1, 2, \ldots, \ell$ *and* $1 \le \ell < d$, *then*

(i) *if* $\alpha = \tilde{\alpha}_i$, $1 \le i \le \ell$, *then* $\lambda_j(\alpha) = \delta_j$, $j = 1, 2, \ldots, \ell$. *In this case,* $\lambda_{\ell+1}(\alpha)$ *is the second smallest root of equation* $\phi(\lambda) = \alpha - \lambda$;

(ii) *if* $\alpha < \tilde{\alpha}_1$, *then* $\lambda_1(\alpha) = \lambda(\alpha)$ *and* $\lambda_j(\alpha) = \delta_{j-1}$, $j = 2, \ldots, \ell + 1$;

(iii) *if* $\tilde{\alpha}_{i-1} < \alpha < \tilde{\alpha}_i$, $2 \le i \le \ell$, *then* $\lambda_i(\alpha) = \lambda(\alpha)$, $\lambda_j(\alpha) = \delta_j$ *for* $j = 1, \ldots, i-1$, *and* $\lambda_j(\alpha) = \delta_{j-1}$ *for* $j = i+1, \ldots, \ell+1$;

(iv) *if* $\alpha > \tilde{\alpha}_\ell$, *then* $\lambda_j(\alpha) = \delta_j$, $j = 1, 2, \ldots, \ell$ *and* $\lambda_{\ell+1}(\alpha) = \lambda(\alpha)$.

Proof. These results are a direct consequence of Cauchy's interlace theorem, Lemmas 3.1 and 3.2, and the properties of the functions $\phi(\lambda)$ and $\alpha - \lambda$. ∎

We can expect difficulties in practice when the vector g is nearly orthogonal to the eigenspace $\mathcal{S}_1$. If this happens, there still exists $\lambda_* < \delta_1$ and x_* such that $(A - \lambda_* I)x_* = -g$, $\|x_*\| = \Delta$, with λ_* quite close to δ_1. We call this situation a *near hard case* and Figure 1(c) illustrates it. In the detail shown in Figure 1(d), we can see that in this case, the derivative ϕ' changes rapidly for λ close to δ_1, so the problem of finding λ_* satisfying the correct slope $\phi'(\lambda_*) = \Delta^2$ is very ill-conditioned.

In the remainder of this section we discuss the hard case and present the results that allow us to compute a nearly optimal solution for the trust-region subproblem in this situation. The hard case can occur only when $g \perp \mathcal{S}_1$, the matrix A is indefinite or positive semidefinite and singular, and for certain values of Δ. This case was analyzed in [10] for medium-scale problems and discussed in [13, 17] in the large-scale context. The precise statement is contained in Lemma 3.4 from [15]. We present a slightly different proof here.

LEMMA 3.4 (see [15]). *Assume* g *is orthogonal to* $\mathcal{S}_1$ *and let* $p = -(A - \delta_1 I)^\dagger g$. *If* $\delta_1 \le 0$ *and* $\|p\| < \Delta$, *then the solutions of* (2.1) *consist of the set* $\{x \mid x = p + z\ ,\ z \in \mathcal{S}_1\ ,\ \|x\| = \Delta\}$, *with Lagrange multiplier* $\lambda_* = \delta_1$.

Proof. We need to show that if $x = p + z$ with $z \in \mathcal{S}_1$ and $\|x\| = \Delta$, then $\{x, \delta_1\}$ satisfy the conditions in Lemma 2.1. It follows directly from the hypothesis and the fact that δ_1 is the smallest eigenvalue of A that $A - \delta_1 I$ is positive semidefinite, that $\delta_1 \le 0$, and that $\delta_1(\Delta - \|x\|) = 0$. It remains only to show that $(A - \delta_1 I)x = -g$. To see this, observe

$$\begin{aligned}(A - \delta_1 I)x &= (A - \delta_1 I)(p + z) \\ &= -(A - \delta_1 I)(A - \delta_1 I)^\dagger g\end{aligned}$$

since $z \in \mathcal{N}(A - \delta_1 I)$. Now, since $g \in \mathcal{R}(A - \delta_1 I)$ and $(A - \delta_1 I)(A - \delta_1 I)^\dagger$ is an orthogonal projector onto $\mathcal{R}(A - \delta_1 I)$, we have

$$(A - \delta_1 I)x = -g$$

which completes the proof. ∎

As we can see, it is precisely in the hard case that in the process of adjusting α we will compute values such that $\alpha > \tilde{\alpha}_1$. As Lemma 3.3 establishes, in this case all the eigenvectors corresponding to the smallest eigenvalue of B_α have first component zero. Moreover, in a near hard case the eigenvectors will have very small first components and dividing by these values will introduce large roundoff errors. Theorem 3.1 and Lemma 3.3 suggest a strategy for handling this situation, namely, using the eigenvector of B_α with the desired structure guaranteed by Theorem 3.1 and the corresponding eigenvalue to obtain the interpolation points, so we can proceed with the adjustment

of the parameter α. We will need a safeguarding strategy to enforce convergence of this iteration. We will describe this strategy in the next section where we present the algorithm in detail.

The following results provide the theoretical basis for declaring convergence in the hard case. Results within the same philosophy are presented in [10, 17]. The idea behind these results is to exploit the information available at each iteration and, with practically no additional cost, detect a nearly optimal solution in the hard case or near hard case. Theorem 3.2, Lemma 3.5, and Lemma 3.6 contain these results. Theorem 3.2 establishes that, under certain conditions, the last n components of a special linear combination of eigenvectors of B_α form a nearly optimal solution for problem (2.1). Lemma 3.5 establishes the conditions under which we can compute the special linear combination, and Lemma 3.6 shows how to compute it. Theorem 3.2 follows from a more general result from [14] but we present a different proof here. Lemma 3.5 is a reformulation of a result from [14] and Lemma 3.6 is from [14].

THEOREM 3.2 (see [14]). *Let $\lambda_1(\alpha)$ be the smallest eigenvalue of B_α with a corresponding eigenvector $z_1 = (\nu_1, \widetilde{z}_1^T)^T$. Let $\lambda_i(\alpha)$ be* any *of the remaining n eigenvalues of B_α with a corresponding eigenvector $z_i = (\nu_i, \widetilde{z}_i^T)^T$. Define $Z = [z_1 \;\; z_i]$, $\widetilde{Z} = [\widetilde{z}_1 \;\; \widetilde{z}_i]$, and assume $Z^T Z = I$. Let $\eta > 0$.*

If there exists $t = (\tau_1, \tau_2)^T$, with $\|t\| = 1$ such that

(i) $(e_1^T Z t)^2 = \frac{1}{1+\Delta^2}$, *and*

(ii) $(\lambda_i(\alpha) - \lambda_1(\alpha))\ \tau_2^2\ (1 + \Delta^2) \le -2\eta\psi(\widetilde{x})$ *for* $\widetilde{x} = \frac{\widetilde{Z}t}{e_1^T Z t}$,

then

$$\psi(x_*) \le \psi(\widetilde{x}) \le \frac{1}{1+\eta}\psi(x_*),$$

where x_* is a boundary solution for problem (2.1) *with $\psi(x_*) \le 0$.*

Proof. Since x_* is a boundary solution of (2.1), we have $\psi(x_*) \le \psi(x)\ \forall\ x \in \mathbb{R}^n$ such that $\|x\| = \Delta$. Therefore, in order to prove that $\psi(x_*) \le \psi(\widetilde{x})$, it will suffice to show that $\|\widetilde{x}\| = \Delta$.

Note that $\frac{Zt}{e_1^T Z t} = (1, \widetilde{x}^T)^T$ and therefore

$$\|(1, \widetilde{x}^T)\|^2 = 1 + \|\widetilde{x}\|^2 = \left\| \frac{Zt}{e_1^T Z t} \right\|^2 = \frac{1}{(e_1^T Z t)^2}$$

since $\|t\| = 1$ and $Z^T Z = I$ by hypothesis. Thus, by (i)

$$1 + \|\widetilde{x}\|^2 = 1 + \Delta^2.$$

This implies $\|\widetilde{x}\| = \Delta$ and therefore $\psi(x_*) \le \psi(\widetilde{x})$.

To prove the other part of the inequality, observe that $\alpha + 2\psi(x_*) = (1, x_*^T)\ B_\alpha\ (1, x_*^T)^T$. Thus, by Rayleigh quotient properties

$$\alpha + 2\psi(x_*) \ge \lambda_1(\alpha)\|(1, x_*^T)^T\|^2.$$

Since $\|x_*\| = \Delta$ it follows that $\|(1, x_*^T)^T\|^2 = 1 + \Delta^2$, and therefore

$$\alpha + 2\psi(x_*) \ge \lambda_1(\alpha)(1 + \Delta^2). \tag{3.2}$$

We now show that $\alpha + 2\psi(\widetilde{x}) = \widetilde{\lambda}(1+\Delta^2)$, with $\widetilde{\lambda} = \lambda_1(\alpha)\tau_1^2 + \lambda_i(\alpha)\tau_2^2$. Observe that $\alpha + 2\psi(\widetilde{x}) = (1, \widetilde{x}^T)\ B_\alpha\ (1, \widetilde{x}^T)^T$, and since $(1, \widetilde{x}^T)^T = \frac{1}{e_1^T Zt} Zt$, it follows that

$$\begin{aligned}\alpha + 2\psi(\widetilde{x}) &= t^T Z^T B_\alpha Zt \frac{1}{(e_1^T Zt)^2} \\ &= [\lambda_1(\alpha)\tau_1^2 + \lambda_i(\alpha)\tau_2^2]\ (1+\Delta^2),\end{aligned}$$

by (i) and the fact that $z_1,\ z_i$ are eigenvectors of B_α.

Thus, since $\tau_1^2 + \tau_2^2 = 1$, we have

$$\begin{aligned}\alpha + 2\psi(\widetilde{x}) &= [\lambda_1(\alpha)(1-\tau_2^2) + \lambda_i(\alpha)\tau_2^2]\ (1+\Delta^2) \\ &= [\lambda_1(\alpha) + (\lambda_i(\alpha) - \lambda_1(\alpha))\tau_2^2]\ (1+\Delta^2)\end{aligned}$$

and therefore

$$\begin{aligned}\alpha + 2\psi(\widetilde{x}) - (\lambda_i(\alpha) - \lambda_1(\alpha))\tau_2^2(1+\Delta^2) &= \lambda_1(\alpha)\ (1+\Delta^2) \\ &\le \alpha + 2\psi(x_*) \quad \text{by (3.2).}\end{aligned}$$

If $(\lambda_i(\alpha) - \lambda_1(\alpha))\tau_2^2(1+\Delta^2) \le -2\eta\psi(\widetilde{x})$, then

$$\alpha + 2\psi(\widetilde{x}) + 2\eta\psi(\widetilde{x}) \le \alpha + 2\psi(x_*)$$

and we can conclude $\psi(\widetilde{x}) \le \frac{1}{1+\eta}\psi(x_*)$.

Therefore $\psi(x_*) \le \psi(\widetilde{x}) \le \frac{1}{1+\eta}\psi(x_*)$ as claimed. ∎

It follows directly from this result that

$$\begin{aligned}0 \le \psi(\widetilde{x}) - \psi(x_*) &\le -\frac{\eta}{1+\eta}\psi(x_*), \\ |\psi(\widetilde{x}) - \psi(x_*)| &\le \frac{\eta}{1+\eta}|\psi(x_*)|\ .\end{aligned} \tag{3.3}$$

The inequality (3.3) implies that under the conditions of Theorem 3.2, $\psi(\widetilde{x})$ will be arbitrarily close to $\psi(x_*)$. We will call such $\widetilde{x}$ a quasi-optimal solution for problem (2.1).

The next result establishes conditions for computing the vector t in Theorem 3.2.

LEMMA 3.5 (see [14]). *Let $z_i = (\nu_i, \widetilde{z}_i^T)^T$, with $\nu_i \in \mathbb{R}$, $\widetilde{z}_i \in \mathbb{R}^n$ for $i = 1, 2$. Define the matrices $Z = [z_1\ \ z_2]$ and $\widetilde{Z} = [\widetilde{z}_1\ \ \widetilde{z}_2]$, and assume $Z^T Z = I$. If $\|Z^T e_1\|^2 \ge \frac{1}{\beta}$ for $\beta > 0$, then there exists $t \in \mathbb{R}^2$ with $t \ne 0$ that satisfies*

$$\|Zt\|^2 = \beta(e_1^T Zt)^2. \tag{3.4}$$

Proof. Observe that we can rewrite (3.4) as

$$\begin{aligned}t^T Z^T Zt &= \beta\ (e_1^T Zt)^2 \\ &= \beta\ (t^T Z^T e_1 e_1^T Zt)\end{aligned}$$

which is equivalent to

$$t^T\ [I - \beta\ Z^T e_1 e_1^T Z]\ t = 0 \tag{3.5}$$

since $Z^T Z = I$ by hypothesis. Equation (3.5) will have a nontrivial solution only if the matrix $M = I - \beta\ Z^T e_1 e_1^T Z$ is indefinite or positive semidefinite and singular. So, let

us study the eigenvalues of M. The two eigenpairs of the matrix $M = I - \beta\, Z^T e_1 e_1^T Z$ are given by

$$\{1 - \beta\, e_1^T Z Z^T e_1,\ Z^T e_1\} \quad \text{and} \quad \{1, v\} \quad \text{with } v \perp Z^T e_1\ .$$

Therefore, (3.5) will have nontrivial solutions if $1 - \beta\, e_1^T Z Z^T e_1 \le 0$. In other words, if $\|Z^T e_1\|^2 = e_1^T Z Z^T e_1 \ge \frac{1}{\beta}$, then there exists $t \in \mathbb{R}^2$ with $t \neq 0$ such that t satisfies (3.5). ∎

Note that choosing $\beta = 1 + \Delta^2$ in Lemma 3.5 and normalizing t such that $\|t\| = 1$ will give a vector that satisfies the conditions of Theorem 3.2. The following lemma provides a way of computing such a vector.

LEMMA 3.6 (see [14]). *Let $\beta \in \mathbb{R},\ \beta > 0$, and let $z \in \mathbb{R}^n$. The equation*

$$t^T[I - \beta z z^T]t = 0 \tag{3.6}$$

in t with $t \in \mathbb{R}^n$ has $2(n-1)$ nontrivial solutions if the matrix $M = I - \beta z z^T$ is indefinite and has one nontrivial solution if M is positive semidefinite and singular.

Proof. Let $P \in \mathbb{R}^{n\times n}$ be such that $P^T z = \|z\| e_1$ with $P^T P = I$ and apply this orthogonal transformation to the matrix M to obtain

$$P^T[I - \beta z z^T]P = I - \beta \|z\|^2 e_1 e_1^T.$$

Therefore, the solutions of (3.6) in this new basis are the solutions of

$$y^T \begin{pmatrix} -\theta & 0 \\ 0 & I \end{pmatrix} y = 0,$$

where $\theta = -1 + \beta\|z\|^2 e_1 e_1^T$.

The nontrivial solutions of (3.6) are then given by $t = Py$, where

(1) $y = (1, \sqrt{\theta} e_i^T)^T$ and $y = (-1, \sqrt{\theta} e_i^T)^T$ with e_i the ith canonical vector in $\mathbb{R}^{n-1}$, $i = 1, 2, \ldots, n-1$, if M is indefinite, i.e., if $\theta > 0$, or

(2) $y = e_1$, if M is positive semidefinite and singular, i.e., if $\theta = 0$.

Therefore, (3.6) has $2(n-1)$ nontrivial solutions if M is indefinite and has one nontrivial solution if M is positive semidefinite and singular. ∎

Remark. Suppose $n = 2$ and $z = (\nu_1, \nu_i)^T$ in Lemma 3.6. Then if ${\nu_1}^2 + {\nu_i}^2 > \frac{1}{\beta}$, the vector $t = (\tau_1, \tau_2)^T$ is given by

$$\tau_1 = \frac{\nu_1 - \nu_i\sqrt{\beta({\nu_1}^2 + {\nu_i}^2) - 1}}{({\nu_1}^2 + {\nu_i}^2)\sqrt{\beta}}, \quad \tau_2 = \frac{\nu_1 + \nu_i\sqrt{\beta({\nu_1}^2 + {\nu_i}^2) - 1}}{({\nu_1}^2 + {\nu_i}^2)\sqrt{\beta}}$$

or

$$\tau_1 = \frac{\nu_i + \nu_1\sqrt{\beta({\nu_1}^2 + {\nu_i}^2) - 1}}{({\nu_1}^2 + {\nu_i}^2)\sqrt{\beta}}, \quad \tau_2 = \frac{\nu_1 - \nu_i\sqrt{\beta({\nu_1}^2 + {\nu_i}^2) - 1}}{({\nu_1}^2 + {\nu_i}^2)\sqrt{\beta}}\ .$$

If ${\nu_1}^2 + {\nu_i}^2 = \frac{1}{\beta}$, then t is given by

$$\tau_1 = \frac{\nu_1}{\sqrt{{\nu_1}^2 + {\nu_i}^2}}, \quad \tau_2 = \frac{\nu_i}{\sqrt{{\nu_1}^2 + {\nu_i}^2}}\ .$$

The previous results are the basis for the algorithm in the next section. They provide the necessary tools for handling the hard case and the standard case in the same iteration and for computing a solution in the hard case. Theorem 3.2 and

Lemma 3.5 give mechanisms for approximating the vector $x = p + z$ in Lemma 3.4 from a linear combination of eigenvectors of B_α. Theorem 3.2 also establishes conditions under which the vector x computed in this way is a quasi-optimal solution for problem (2.1). Moreover, Theorem 3.2 guarantees that if the second smallest eigenvalue of B_α belongs to a cluster, as is the case, for example, in discrete ill-posed problems, we can still build the special vector $\hat{x}$ from an eigenvector associated with the smallest eigenvalue of B_α and from an eigenvector associated with *any* eigenvalue of the cluster—not necessarily the second smallest. Observe that Lemmas 3.5 and 3.6, respectively, provide a way of computing the vectors $\hat{x}$ and t needed in Theorem 3.2. We use Theorem 3.2, Lemma 3.5, and Lemma 3.6 in one of the stopping rules in our method. We describe the stopping criteria in section 4.5.

4. The algorithm. Keeping in mind the availability of a well-suited variant of the Lanczos method, namely, the *implicitly restarted Lanczos method* (cf. [16]), we will develop a rapidly convergent iteration to adjust α based on this process. Our goal is to adjust α so that

$$\alpha - \lambda = \phi(\lambda), \qquad \phi'(\lambda) = \Delta^2,$$

where

$$\phi(\lambda) = -g^T x, \qquad \phi'(\lambda) = x^T x,$$

with $(A - \lambda I)x = -g$.

The approach of this work is similar to the one in [17] in the following sense. We compute a function $\widehat{\phi}$ which interpolates ϕ and ϕ' at two properly chosen points. Then, from the interpolating function $\widehat{\phi}$ we determine $\widehat{\lambda}$ satisfying

$$\widehat{\phi}'(\widehat{\lambda}) = \Delta^2 \ . \tag{4.1}$$

Finally, we use $\widehat{\lambda}$ and $\phi(\widehat{\lambda})$ to update the parameter α and compute the next iterates $\{\lambda, x\}$. The new elements in our algorithm are the introduction of safeguards for the sequence in α, the use of the information relative to the second smallest eigenvalue of the matrix B_α, and the introduction of a different interpolating scheme, where the currently available information is exploited to a greater extent. Considering that the interpretation of the primal feasibility equations of [13] can be related to (4.1), the description of our algorithm also has some flavor of the approach in [13], where an inverse interpolation scheme is used to satisfy primal feasibility. However, in the presence of the hard case, we do not need to combine distinct interpolating functions, as in [13], nor switch to another algorithm as in [17]. In this section we will assume that the vector g is nonzero. If $g = 0$, then problem (2.1) reduces to solving an eigenvalue problem for the smallest eigenvalue of A. We shall first describe the components of the algorithm and then present the complete method.

4.1. Interpolating schemes. To begin the iteration, we need a single-point interpolating scheme. We use the approach derived in [17] which gives the following expression for α_1:

$$\alpha_1 = \widehat{\lambda} + \widehat{\phi}(\widehat{\lambda}) = \alpha_0 + \frac{\alpha_0 - \lambda_0}{\|x_0\|} \left(\frac{\Delta - \|x_0\|}{\Delta} \right) \left(\Delta + \frac{1}{\|x_0\|} \right), \tag{4.2}$$

where

$$\widehat{\lambda} = \delta + \frac{g^T x_0}{\|x_0\| \Delta} \ .$$

This method is linearly convergent and may be slow in some cases, so we will use it just to obtain a second pair of iterates, which together with λ_0, x_0 will be the starting values for a two-point method. In the two-point method we use the four pieces of information available at the kth iteration, namely, $\phi(\lambda_{k-1})$, $\phi'(\lambda_{k-1})$, $\phi(\lambda_k)$, and $\phi'(\lambda_k)$ as follows. We compute $\widehat{\lambda}$ such that

$$\frac{1}{\Delta} = \frac{1}{\sqrt{\phi'(\lambda_{k-1})}}\left(\frac{\lambda_k - \widehat{\lambda}}{\lambda_k - \lambda_{k-1}}\right) + \frac{1}{\sqrt{\phi'(\lambda_k)}}\left(\frac{\widehat{\lambda} - \lambda_{k-1}}{\lambda_k - \lambda_{k-1}}\right), \tag{4.3}$$

obtaining

$$\widehat{\lambda} = \frac{\lambda_{k-1}\|x_{k-1}\|(\|x_k\| - \Delta) + \lambda_k\|x_k\|(\Delta - \|x_{k-1}\|)}{\Delta(\|x_k\| - \|x_{k-1}\|)}. \tag{4.4}$$

This is equivalent to defining

$$\widehat{\phi}(\lambda) = \frac{\gamma^2}{\delta - \lambda} + \eta \tag{4.5}$$

for any η and computing $\widehat{\lambda}$ such that $\frac{1}{\sqrt{\widehat{\phi}'(\widehat{\lambda})}} = \frac{1}{\Delta}$. It is easy to verify using (4.3) that

$$\gamma^2 = \frac{(\lambda_k - \lambda_{k-1})^2\|x_{k-1}\|^2\|x_k\|^2}{(\|x_k\| - \|x_{k-1}\|)^2} \qquad \text{and} \qquad \delta = \frac{\lambda_k\|x_k\| - \lambda_{k-1}\|x_{k-1}\|}{\|x_k\| - \|x_{k-1}\|}.$$

Ideally, $\eta = \phi(\widehat{\lambda}) - \frac{\gamma^2}{\delta - \widehat{\lambda}}$, where $\phi(\widehat{\lambda})$ is the value we are going to estimate in order to update α. Using the values $\phi(\lambda_{k-1})$ and $\phi(\lambda_k)$, we first define $\eta_j = \phi(\lambda_j) - \frac{\gamma^2}{\delta - \lambda_j}$, for $j = k-1,\ k$. Then, applying the linear interpolation philosophy on $\lambda_j,\ \eta_j$, and defining the weights by means of the already computed value $\widehat{\lambda}$, we choose

$$\eta = \left(\frac{\lambda_k - \widehat{\lambda}}{\lambda_k - \lambda_{k-1}}\right)\eta_{k-1} + \left(\frac{\widehat{\lambda} - \lambda_{k-1}}{\lambda_k - \lambda_{k-1}}\right)\eta_k.$$

After some manipulation we can express the updating formula for α as

$$\begin{aligned}\alpha_{k+1} &= \widehat{\lambda} + \omega\phi(\lambda_{k-1}) + (1-\omega)\phi(\lambda_k) \\ &\quad + \frac{\|x_{k-1}\|\|x_k\|(\|x_k\| - \|x_{k-1}\|)}{\omega\|x_k\| + (1-\omega)\|x_{k-1}\|}\frac{(\lambda_{k-1} - \widehat{\lambda})(\lambda_k - \widehat{\lambda})}{(\lambda_k - \lambda_{k-1})} \\ &= \omega\alpha_{k-1} + (1-\omega)\alpha_k \\ &\quad + \frac{\|x_{k-1}\|\|x_k\|(\|x_k\| - \|x_{k-1}\|)}{\omega\|x_k\| + (1-\omega)\|x_{k-1}\|}\frac{(\lambda_{k-1} - \widehat{\lambda})(\lambda_k - \widehat{\lambda})}{(\lambda_k - \lambda_{k-1})},\end{aligned} \tag{4.6}$$

where $\omega = \frac{\lambda_k - \widehat{\lambda}}{\lambda_k - \lambda_{k-1}}$, $\alpha_{k-1} = \lambda_{k-1} + \phi(\lambda_{k-1})$, and $\alpha_k = \lambda_k + \phi(\lambda_k)$.

As we discussed in section 3, we need a special strategy to obtain interpolation points in potential hard cases. We describe this strategy in section 4.2.

4.2. Choice of interpolation points. According to Lemma 3.1, if the first component of the eigenvector corresponding to the smallest eigenvalue of B_{α_k} is zero, this will indicate a potential hard case and we will have $\lambda_1(\alpha_k) = \delta_1$. However,

Lemma 3.3 establishes that for α_k slightly larger than $\tilde{\alpha}_1$ there is an eigenvector with significant first component that corresponds to the *second* smallest eigenvalue of B_α. Therefore, we propose to use an eigenpair corresponding to an eigenvalue that is close to the second smallest eigenvalue of the bordered matrix to obtain the interpolation point whenever we detect a potential hard case. As we shall explain, not only can we keep the size of the iterate x_k under control, but we can also ensure convergence of $\{\lambda_k, x_k\}$ to $\{\delta_1, p\}$ by driving the parameter α_k to the value $\tilde{\alpha}_1$ given by Lemma 3.2. Recall that Lemma 3.2 established that there will be an eigenvector with significant first component corresponding to $\lambda_1(\alpha_k)$ precisely when α_k assumes the special value $\tilde{\alpha}_1 = \delta_1 - g^T p$. Moreover, the use of a second eigenvector prevents numerical difficulties in a near-hard-case situation.

There is an easy way to detect a potential hard case during an iteration. Let $(\nu_1, u_1^T)^T$ be a unitary eigenvector of B_{α_k} corresponding to $\lambda_1(\alpha_k)$. Then, we declare ν_1 to be "small," indicating a near hard case has been detected, if the condition $\|g\||\nu_1| \leq \varepsilon\sqrt{1-\nu_1{}^2}$ holds for a given $\varepsilon \in (0,1)$. This is motivated as follows. Since $(A - \lambda_1(\alpha_k)\, I)u_1 = -g\nu_1$, we have

$$\frac{\|(A - \lambda_1(\alpha_k)\, I)u_1\|}{\|u_1\|} = \frac{\|g\||\nu_1|}{\sqrt{1-\nu_1{}^2}}$$

and hence $\|g\||\nu_1| \leq \varepsilon\sqrt{1-\nu_1{}^2}$ ensures that $\|(A - \lambda_1(\alpha_k)\, I)u_1\| \leq \varepsilon\|u_1\|$. In other words, $\{\lambda_1(\alpha_k), u_1\}$ is an approximate eigenpair of A and the eigenvector $(\nu_1, u_1^T)^T$ from the bordered matrix is essentially impossible to normalize. This is approximately the situation described in Lemma 3.1. Of course, this test can be made scale independent by choosing $\varepsilon = \widehat{\varepsilon}\|A\|$, for $\widehat{\varepsilon} \in (0,1)$.

When a near hard case has been detected, we need an alternative way to define the pair $\{\lambda_k, x_k\}$. At each iteration, at essentially no extra cost, we compute an eigenpair corresponding to the smallest eigenvalue of B_{α_k}, which we denote by $\{\lambda_1(\alpha_k), (\nu_1, u_1^T)^T\}$, and also an eigenpair corresponding to an eigenvalue close to the second smallest eigenvalue of B_α, which we denote by $\{\lambda_i(\alpha_k), (\nu_2, u_2^T)^T\}$. If both $|\nu_1|$ and $|\nu_2|$ are small, that is, if $\|g\||\nu_1| \leq \varepsilon\sqrt{1-\nu_1{}^2}$ and $\|g\||\nu_2| \leq \varepsilon\sqrt{1-\nu_2{}^2}$, then we decrease the parameter α_k. According to Theorem 3.1 there always exists an eigenvector of the bordered matrix with significant first component for any value of α and, as we mentioned before, according to Lemma 3.3, as α_k approaches the critical value, this normalizable eigenvector will correspond either to the first or to the second smallest eigenvalue of B_{α_k}. In other words, for values of α_k near the critical value, either $\|g\||\nu_1| > \varepsilon\sqrt{1-\nu_1{}^2}$ or $\|g\||\nu_2| > \varepsilon\sqrt{1-\nu_2{}^2}$ will hold. Hence, after a possible reduction of the parameter α_k, the pair $\{\lambda_k, x_k\}$ is well defined if we compute it by the following procedure:

If $\|g\||\nu_1| \leq \varepsilon\sqrt{1-\nu_1{}^2}$, then set $\lambda_k = \lambda_i(\alpha_k)$ and $x_k = \dfrac{u_2}{\nu_2}$.

Otherwise, set $\lambda_k = \lambda_1(\alpha_k)$ and $x_k = \dfrac{u_1}{\nu_1}$.

Since λ_{k-1} and λ_k are not constrained to $(-\infty, \delta_1]$ but might belong to the interval $(\delta_1, \delta_{\ell+1})$, the value $\widehat{\lambda}$ given by (4.4) may be greater than δ_1. In this case, we set $\widehat{\lambda} = \delta_U$, where δ_U is an upper bound for δ_1. In section 4.3 we will show how to obtain an initial value for δ_U and how to update this value. We will also show how to safeguard α computed by (4.6).

4.3. Safeguarding. We need to introduce safeguarding to ensure global convergence of the iteration. Let λ_*, x_* be an optimal pair for problem (2.1), satisfying the conditions in Lemma 2.1, except when there is only an interior solution, in which case we define $x_* = -(A - \lambda_* I)^\dagger g$ such that $\|x_*\| = \Delta$. Let $\alpha_* = \lambda_* - g^T x_*$. Rendl and Wolkowicz [13] presented the following bounds for the optimal parameter α_*:

$$\delta_1 - \frac{\|g\|}{\Delta} \leq \alpha_* \leq \delta_1 + \|g\|\Delta \,. \tag{4.7}$$

Computing a good approximation to δ_1 can be nearly as expensive as solving the given trust-region subproblem. For this reason, as observed in [13], we shall replace the above bounds by some simple alternatives. First, note that any Rayleigh quotient $\delta_U \equiv \frac{v^T A v}{v^T v}$ gives an upper bound for δ_1. Therefore, if the diagonal of the matrix A is explicitly available, we take $\delta_U = \min\{a_{ii} \,|\, i = 1, \ldots, n\}$; otherwise we take $\delta_U \equiv \frac{v^T A v}{v^T v}$, where v is a random vector. From (4.7) we see that $\alpha_* \leq \alpha_U$, for $\alpha_U = \delta_U + \| g\|\Delta$. Since $\alpha \leq 0$ implies B_α is not positive definite, we set $\alpha_0 = \min\{0, \alpha_U\}$ to ensure that $\lambda_1(\alpha_0) \leq 0$. After solving for $\lambda_1(\alpha_0)$ and setting $\delta_L = \lambda_1(\alpha_0)$ and $\alpha_L = \delta_L - \frac{\|g\|}{\Delta}$, we immediately have that $\alpha_L \leq \alpha_*$, since the interlacing property implies $\delta_L \leq \delta_1$. Using this simple scheme to obtain δ_L and δ_U as initial lower and upper bounds for δ_1, we can start with

$$\alpha_L = \delta_L - \frac{\|g\|}{\Delta} \quad \text{and} \quad \alpha_U = \delta_U + \|g\|\Delta \,. \tag{4.8}$$

We update the upper bound δ_U at each iteration using information from the eigenpair corresponding to the smallest eigenvalue of the bordered matrix in the following way: $\delta_U = \min\{\delta_U, \frac{u_1^T A u_1}{u_1^T u_1}\}$, where $\frac{u_1^T A u_1}{u_1^T u_1} = \lambda_1(\alpha_k) - \nu_1 \frac{g^T u_1}{u_1^T u_1}$. As stated in section 4.2, whenever we detect a potential hard case, $\{\lambda_1(\alpha_k), u_1\}$ approximates an eigenpair of A and $\lambda_1(\alpha_k)$ is a very good approximation to δ_1. Thus, δ_U becomes a sharp estimate of δ_1 in this case.

At every iteration, we update one of the safeguarding bounds α_L or α_U so that we always reduce the length of the interval $[\alpha_L, \alpha_U]$. In case the value α_{k+1} predicted by the interpolating schemes (4.2) or (4.6) does not belong to the current safeguarding interval, we redefine α_{k+1} by means of a linear adjustment based on the upper bound δ_U. If this choice is not in the interval $[\alpha_L, \alpha_U]$, we simply set $\alpha_{k+1} = \frac{\alpha_L + \alpha_U}{2}$.

4.4. Initialization of α. As mentioned in section 4.3, there is a simple choice for initializing α, given by $\alpha_0 = \min\{0, \alpha_U\}$, with α_U as in (4.8). This ensures that $\lambda_1(\alpha_0) \leq 0$ but it has no additional properties. In an attempt to improve this initial guess, we have developed a more sophisticated *hot-start* strategy based on the Lanczos process. To begin, we compute the following j-step Lanczos factorization for the j smallest eigenvalues of A:

$$AV = VT + f e_j^T, \tag{4.9}$$

where $V^T V = I_j$, with I_j the identity matrix of order j ($j \ll n$), $T \in \mathbb{R}^{j \times j}$ tridiagonal, $V^T f = 0$, and e_j denotes the jth canonical unit vector in $\mathbb{R}^j$.

The hot-start strategy consists of first changing variables in (2.1) using $x = Vy$ and solving the j-dimensional problem

$$\begin{aligned} \min \quad & \tfrac{1}{2} y^T T y + g^T V y \\ \text{s.t.} \quad & \|y\| \leq \Delta. \end{aligned}$$

Then, we compute a solution $\{\theta_*, y_*\}$ to this lower dimensional trust-region subproblem by using the algorithm in [10], based on the Cholesky factorization of the tridiagonal matrix $T - \theta I$, $\theta < \delta_1$. The initial value to be used is $\alpha = \theta_* - g^T V y_*$.

We now show that we can use (4.9) to compute an eigenpair corresponding to the smallest eigenvalue of B_{α_0}. Observe

$$\begin{pmatrix} \alpha_0 & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & V \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & V \end{pmatrix} \begin{pmatrix} \alpha_0 & g^T V \\ V^T g & T \end{pmatrix} + \begin{pmatrix} 0 \\ f \end{pmatrix} e_{j+1}^T . \tag{4.10}$$

If we run the standard Lanczos process for A using $v_1 = g/\|g\|$ as the initial vector, then we obtain a tridiagonal matrix on the right-hand side of (4.10). This provides a way of computing the smallest eigenvalue of B_{α_0}.

In numerical experiments, the use of this hot start for α did not substantially improve the performance of the method.

4.5. Stopping criteria. At each iteration we check for a boundary solution, an interior solution, or a quasi-optimal solution according to Theorem 3.2. We can also stop if we reach a maximum number of iterations or if the length of the safeguarding interval is too small. Given the tolerances $\varepsilon_\Delta, \varepsilon_{HC}, \varepsilon_\alpha \in (0,1)$, and $\varepsilon_{Int} \in [0,1)$, we declare convergence of the algorithm according to the following criteria. Let $(\nu_1, u_1^T)^T$ be the eigenvector corresponding to $\lambda_1(\alpha_k)$ and let $\{\lambda_k, x_k\}$ be the current iterates; then we can write the stopping criteria in the following way.

1. *Boundary solution.*

We detect a boundary solution if

$$(\mid \|x_k\| - \Delta \mid \ \leq \ \varepsilon_\Delta * \Delta) \quad \text{and} \quad (\lambda_1(\alpha_k) \leq 0).$$

If this condition is satisfied, the solution is

$$\lambda_* = \lambda_1(\alpha_k) \quad \text{and} \quad x_* = x_k.$$

2. *Interior solution.*

We detect an interior solution if

$$(\|u_1\| \ < \ \Delta|\nu_1|) \quad \text{and} \quad (\lambda_1(\alpha_k) > -\varepsilon_{Int}).$$

In this case, the solution is $\lambda_*,\ x_*$, where $\lambda_* = 0$ and x_* satisfies the linear system $Ax = -g$, with A positive definite. The conjugate gradient method is a natural choice for solving this system for most large-scale problems.

3. *Quasi-optimal solution.*

To declare that we have found a quasi-optimal solution, we first compute t and $\widetilde{x}$ as in Lemma 3.5, provided that the conditions of the lemma are satisfied. If $t = (\tau_1, \tau_2)^T$ and $\widetilde{x}$ satisfy condition (ii) of Theorem 3.2, then $\widetilde{x}$ is a quasi-optimal solution for problem (2.1) and we set $\lambda_* = \widetilde{\lambda}$ and $x_* = \widetilde{x}$.

4. *The safeguarding interval is too small.*

If $|\alpha_U - \alpha_L| \leq \varepsilon_\alpha \max\{|\alpha_L|, |\alpha_U|\}$, then we stop the iteration and set $\lambda_* = \lambda_1(\alpha_k)$. If this criterion is satisfied and we do not have a boundary solution, then we are in the hard case and α_* is within ε_α of $\tilde{\alpha}_1$. If ν_1 is large enough, we set $p = \frac{u_1}{\nu_1}$. Since $\|p\| < \Delta$ in this case, we compute x_* as $x_* = p + \tau z$ such that $\|x_*\| = \Delta$, where the vector z is an approximate eigenvector associated with the smallest eigenvalue of A. Of the two possible choices for τ, we choose the one with smaller magnitude since this

value minimizes $\psi(p+\tau z)$ (see [10, p. 558]). This choice of τ is given by

$$\tau = \frac{\Delta^2 - \|p\|^2}{p^T z + sign(p^T z)\sqrt{(p^T z)^2 - (\Delta^2 - \|p\|^2)}} .$$

The vector z is usually available in potential hard cases since in those cases the eigenvectors corresponding to the smallest eigenvalue of B_{α_k} will often have a small first component. In the rather unlikely situation where this vector is not available, we increase the parameter and solve an eigenproblem for the smallest eigenvalue of the bordered matrix. This strategy will provide an approximate vector in $\mathcal{S}_1$ as Lemma 3.3 guarantees.

If ν_1 is too small or zero, we cannot compute a solution. This situation can arise in practice because the eigensolver might not provide the eigenvector with significant first component that the theory guarantees. We have not encountered this case in our experiments.

4.6. The algorithm. Let us now put all these pieces together and present LSTRS, our algorithm for the large-scale trust-region subproblem. We describe steps 2.1 and 2.5 of Algorithm 4.1 separately. In step 2.1 we adjust the parameter α_k so that the eigenvector corresponding to the smallest eigenvalue, or to an eigenvalue equal or close to the second smallest eigenvalue of B_{α_k}, has a significant first component. We might reduce the interval $[\alpha_L, \alpha_U]$ during this adjustment. In step 2.5 we correct the parameter predicted by the interpolation schemes in case it does not belong to the current safeguarding interval $[\alpha_L, \alpha_U]$. We try a linear adjustment first and adopt the middle point of the current interval as a last resort. Figure 2 shows Algorithm 4.1, while Figures 3 and 4 show steps 2.1 and 2.5, respectively.

5. Convergence analysis.

5.1. Iterates are well defined.

LEMMA 5.1. *The iterates generated by Algorithm* 4.1 *are well defined.*

Proof. In order to define the current iterate x_k in Algorithm 4.1, we must ensure that we can safely normalize an eigenvector, corresponding to either the smallest eigenvalue or a value equal or close to the second smallest eigenvalue of B_{α_k}, to have first component one. This is accomplished in step 2.1, where we adjust the parameter α_k until we can normalize one of these two eigenvectors to have first component one. Theorem 3.1 and Lemma 3.3 guarantee that the adjusting procedure in step 2.1 yields a value of α such that there exists an eigenvector that has significant first component and is associated with the smallest eigenvalue or a value equal or close to the second smallest eigenvalue of B_α. ∎

5.2. Local convergence.

5.2.1. Preliminary results.

LEMMA 5.2. *Let λ_k, x_k be the iterates at iteration k of Algorithm* 4.1. *Then*

$$g \in \mathcal{R}(A - \lambda_k I).$$

Proof. If λ_k, x_k are the iterates at iteration k of Algorithm 4.1, then

$$\begin{pmatrix} \alpha_k & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ x_k \end{pmatrix} = \lambda_k \begin{pmatrix} 1 \\ x_k \end{pmatrix}.$$

Therefore, $(A - \lambda_k I)x_k = -g$, which implies that $g \in \mathcal{R}(A - \lambda_k I)$. ∎

ALGORITHM 4.1. *LSTRS.*

Input: $A \in \mathbb{R}^{n\times n}$, $g \in \mathbb{R}^n$, $\Delta > 0$, $\varepsilon_\Delta, \varepsilon_\nu, \varepsilon_{HC}, \varepsilon_\alpha \in (0,1)$, $\varepsilon_{Int} \in [0,1)$.
Output: λ_*, x_* *satisfying conditions of Lemma* 2.1.

1. Initialization
  1.1 Compute $\delta_U \geq \delta_1$, initialize α_U using (4.8),
    set $\alpha_0 = \min\{0, \alpha_U\}$
  1.2 Compute eigenpairs $\{\lambda_1(\alpha_0), (\nu_1, u_1^T)^T\}$, $\{\lambda_i(\alpha_0), (\nu_2, u_2^T)^T\}$
    corresponding to smallest eigenvalue and an eigenvalue
    close to second smallest eigenvalue of B_{α_0}
  1.3 Initialize α_L using (4.8)
  1.4 Set $k = 0$
2. **repeat**
  2.1 Adjust α_k
  2.2 Update $\delta_U = \min\left\{\delta_U, \dfrac{u_1^T A u_1}{u_1^T u_1}\right\}$
  2.3 **if** $\|g\||\nu_1| > \varepsilon_\nu\sqrt{1-\nu_1{}^2}$ **then**
    set $\lambda_k = \lambda_1(\alpha_k)$ and $x_k = \dfrac{u_1}{\nu_1}$
    if $\|x_k\| < \Delta$ **then** $\alpha_L = \alpha_k$ **end if**
    if $\|x_k\| > \Delta$ **then** $\alpha_U = \alpha_k$
    else set $\lambda_k = \lambda_i(\alpha_k)$, $x_k = \dfrac{u_2}{\nu_2}$ and $\alpha_U = \alpha_k$ **end if**
    end if
  2.4 Compute α_{k+1} by interpolation scheme
    using (4.2) if $k = 0$ or (4.4) and (4.6) otherwise
  2.5 Safeguard α_{k+1}
  2.6 Set $k = k + 1$
until convergence

FIG. 2. *LSTRS: A method for the large-scale trust-region subproblem.*

LEMMA 5.3. *Let $\lambda_* \leq \delta_1$ be the Lagrange multiplier corresponding to a boundary solution of problem* (2.1). *Then*

$$g \in \mathcal{R}(A - \lambda_* I).$$

Proof. If $\lambda_* < \delta_1$, then $A - \lambda_* I$ is nonsingular and $g \in \mathcal{R}(A - \lambda_* I)$. If $\lambda_* = \delta_1$, then $g \perp \mathcal{N}(A - \lambda_* I)$ must hold and therefore $g \in \mathcal{R}(A - \lambda_* I)$. ∎

Remark. Since $(A - \lambda I)(A - \lambda I)^\dagger$ and $(A - \lambda I)^\dagger(A - \lambda I)$ are orthogonal projectors onto $\mathcal{R}(A - \lambda I)$, we have that

$$g = (A - \lambda I)(A - \lambda I)^\dagger g = (A - \lambda I)^\dagger(A - \lambda I)g \tag{5.1}$$

for any λ such that $g \in \mathcal{R}(A - \lambda I)$. In particular, Lemmas 5.2 and 5.3 imply that (5.1) holds for $\lambda = \lambda_k$ and $\lambda = \lambda_*$.

5.2.2. Technical lemmas. We present several technical lemmas that allow us to prove our local convergence result. We will use the following notation:

$$A_k \equiv A - \lambda_k I \quad \text{and} \quad A_* \equiv A - \lambda_* I. \tag{5.2}$$

Step 2.1. *Adjust* α_k.

Input: $\varepsilon_\nu, \varepsilon_\alpha \in (0,1)$, α_L, α_U, α_k with $\alpha_k \in [\alpha_L, \alpha_U]$.
Output: $\alpha_k, \{\lambda_1(\alpha_k), (\nu_1, u_1^T)^T\}$ and $\{\lambda_i(\alpha_k), (\nu_2, u_2^T)^T\}$.

· `Set` $\alpha = \alpha_k$
· **if** $k > 0$ **then**
  `compute eigenpairs` $\{\lambda_1(\alpha), (\nu_1, u_1^T)^T\}$ `and` $\{\lambda_i(\alpha), (\nu_2, u_2^T)^T\}$`,`
  `corresponding to smallest eigenvalue and an eigenvalue`
  `close to second smallest eigenvalue of` B_α
 end if
· **while**
  $\|g\||\nu_1| \le \varepsilon_\nu \sqrt{1-\nu_1{}^2}$ **and** $\|g\||\nu_2| \le \varepsilon_\nu \sqrt{1-\nu_2{}^2}$
  and $|\alpha_U - \alpha_L| > \varepsilon_\alpha * \max\{|\alpha_L|, |\alpha_U|\}$ **do**
  $\alpha_U = \alpha$
  $\alpha = (\alpha_L + \alpha_U)/2$
  `Compute` $\{\lambda_1(\alpha), (\nu_1, u_1^T)^T\}$ `and` $\{\lambda_i(\alpha), (\nu_2, u_2^T)^T\}$
 end while
· `Set` $\alpha_k = \alpha$

FIG. 3. *Adjustment of* α.

Step 2.5. *Safeguard* α_{k+1}.

Input: α_{k+1} computed by step 2.4 of Algorithm 4.1, $\delta_U \ge \delta_1$, α_L, α_U,
$\phi_i = -g^T x_i$, and $\phi_i' = \|x_i\|^2$, for $i = k-1, k$.
Output: Safeguarded value for α_{k+1}.

if $\alpha_{k+1} \notin [\alpha_L, \alpha_U]$
  if $k = 0$ **then** $\alpha_{k+1} = \delta_U + \phi_k + \phi_k'(\delta_U - \lambda_k)$
  else if $\|x_k\| < \|x_{k-1}\|$ **then** $\alpha_{k+1} = \delta_U + \phi_k + \phi_k'(\delta_U - \lambda_k)$
  else $\alpha_{k+1} = \delta_U + \phi_{k-1} + \phi_{k-1}'(\delta_U - \lambda_{k-1})$
  end if
  if $\alpha_{k+1} \notin [\alpha_L, \alpha_U]$ **then** `set` $\alpha_{k+1} = (\alpha_L + \alpha_U)/2$ **end if**
end if

FIG. 4. *Safeguarding of* α.

The first lemma establishes a key relationship satisfied by the iterates computed by Algorithm 4.1.

LEMMA 5.4. *Let* λ_k, x_k *be the iterates at iteration* k *of Algorithm* 4.1. *Then*

$$x_k = -(A - \lambda_k I)^\dagger g.$$

Proof. First note that if λ_k, x_k are the iterates at iteration k of Algorithm 4.1, then they satisfy

$$\begin{pmatrix} \alpha_k & g^T \\ g & A \end{pmatrix} \begin{pmatrix} 1 \\ x_k \end{pmatrix} = \lambda_k \begin{pmatrix} 1 \\ x_k \end{pmatrix}.$$

Therefore

$$(A - \lambda_k I)x_k = -g. \tag{5.3}$$

In order to prove the result we need to consider two cases.

Case 1: $\lambda_k \neq \delta_i,\ i = 1, 2, \ldots, d.$

In this case we have that $A - \lambda_k I$ is nonsingular, $(A - \lambda_k I)^{-1} = (A - \lambda_k I)^\dagger$, and from (5.3) we conclude

$$x_k = -(A - \lambda_k I)^\dagger g.$$

Case 2: $\lambda_k = \delta_i$, $1 \leq i \leq d$.

If $\lambda_k = \delta_i$, then (5.3) implies that $g \perp \mathcal{S}_i$. This follows from the observation that for any $q \in \mathcal{S}_i$, we have $0 = q^T(A - \delta_i I)x_k = -q^T g$. Corollary 3.1 now implies that $\alpha_k = \tilde{\alpha}_i$ and

$$\begin{aligned} x_k &= p_i \\ &= -(A - \delta_i I)^\dagger g, \end{aligned}$$

since $(1, x_k^T)^T$ is an eigenvector of B_{α_k}. This concludes the proof. ∎

Before presenting the next lemma, which provides useful relationships for the convergence analysis, we introduce the following definition.

DEFINITION 5.1. *Let λ_i, x_i and λ_j, x_j be the iterates computed by Algorithm* 4.1 *at iterations i and j, respectively. Then we define*

$$\rho(i, j) \equiv x_i^T A_j^\dagger x_i + x_j^T A_i^\dagger x_j. \tag{5.4}$$

We can substitute any of the iterates by $\lambda_*,\ y$, with $y = -A_*^\dagger g$. We denote this by $\rho(*, j)$ and $\rho(i, *)$, respectively.

Observe that if $A = QDQ^T$ is an eigendecomposition of A, i.e., Q is an orthogonal matrix and D is a diagonal matrix with the eigenvalues of A on the diagonal, we can write $\rho(i, j)$ in the following way:

$$\rho(i, j) = g^T Q D_i^\dagger (D_i^\dagger + D_j^\dagger) D_j^\dagger Q^T g,$$

where $D_i = D - \lambda_i I$ and $D_j = D - \lambda_j I$. From this expression we obtain

$$\rho(i, j) = \sum_{k=1}^{d} \frac{\beta_k^2 (2\delta_k - \lambda_i - \lambda_j)}{(\delta_k - \lambda_i)^2 (\delta_k - \lambda_j)^2}, \tag{5.5}$$

where β_k^2 is the sum of the expansion coefficients of g in the eigenvector basis, corresponding to all the eigenvectors associated with δ_k. As before, we assume that $\delta_1, \delta_2, \ldots, \delta_d$ are the distinct eigenvalues of A in nondecreasing order.

LEMMA 5.5. *Let $\lambda_i, x_i,\ \lambda_j, x_j$, and λ_k, x_k be the iterates computed by Algorithm* 4.1 *at iterations i, j, and k, respectively. Then*

(i) $(x_i - x_j)^T g = (\lambda_j - \lambda_i) x_i^T x_j$;

(ii) $(x_i - x_j)^T x_k = (\lambda_i - \lambda_j) x_j^T A_i^\dagger x_k$;

(iii) $x_i^T x_i - x_j^T x_j = (\lambda_i - \lambda_j)\rho(i, j)$, *with $\rho(i, j)$ given by* (5.4).

Moreover, (i)–(iii) *also hold if we substitute any of the pairs above by $\lambda_*,\ y$, where λ_* is the Lagrange multiplier corresponding to a boundary solution of problem* (2.1) *and $y = -A_*^\dagger g$.*

Proof. Let us first prove (i). Observe that by Lemma 5.4

$$(x_i - x_j)^T g = (A_j^\dagger g - A_i^\dagger g)^T g.$$

Therefore, using (5.1) and the fact that $A_i, A_i^\dagger, A_j, A_j^\dagger$ commute, we have

$$\begin{aligned}(x_i - x_j)^T g &= g^T (A_j^\dagger - A_i^\dagger) g \\ &= g^T A_i^\dagger (A_i - A_j) A_j^\dagger g \\ &= (\lambda_j - \lambda_i) x_i^T x_j.\end{aligned}$$

To prove (ii), we use (5.1), Lemma 5.4, and the fact that $A_i, A_i^\dagger, A_j, A_j^\dagger$ commute, obtaining

$$\begin{aligned}(x_i - x_j)^T x_k &= g^T (A_i^\dagger - A_j^\dagger) x_k \\ &= g^T A_j^\dagger (A_j - A_i) A_i^\dagger x_k \\ &= (\lambda_i - \lambda_j) x_j^T A_i^\dagger x_k.\end{aligned}$$

Finally, let us prove (iii). By (5.1), Lemma 5.4, and the fact that $A_i, A_i^\dagger, A_j, A_j^\dagger$ commute, we have

$$\begin{aligned}x_i^T x_i - x_j^T x_j &= g^T [(A_i^\dagger)^2 - (A_j^\dagger)^2] g \\ &= g^T [(A_i^\dagger)^2 A_j^2 - A_i^2 (A_j^\dagger)^2] g \\ &= g^T (A_i^\dagger)^2 (A_j - A_i)(A_j + A_i)(A_j^\dagger)^2 g \\ &= (\lambda_i - \lambda_j) x_i^T (A_i^\dagger + A_j^\dagger) x_j \\ &= (\lambda_i - \lambda_j) \rho(i,j).\end{aligned}$$

Observe that (i)–(iii) hold for λ_*, y, since (5.1) holds for λ_*, $y = -A_*^\dagger g$, and A_* commutes with the matrices above. This observation concludes the proof. ∎

Using the updating formula (4.6), we obtained the following result relating $\lambda_{k+1} - \lambda_*$ with $\lambda_{k-1} - \lambda_*$ and $\lambda_k - \lambda_*$. This lemma provides a key relationship for establishing the local convergence properties of Algorithm 4.1.

LEMMA 5.6. *Let $\lambda_* \le \delta_1$ be the Lagrange multiplier corresponding to a boundary solution of problem* (2.1)*, with $g \neq 0$. Let λ_{k+1}, x_{k+1} be the $(k+1)$st iterates computed by Algorithm* 4.1 *using the two-point interpolating scheme given by* (4.6) *to update α. Then, there exists a neighborhood $\mathcal{B}$ of λ_* such that if λ_{k-1}, $\lambda_k \in \mathcal{B}$, then λ_{k+1} satisfies*

$$|\lambda_{k+1} - \lambda_*| \le \mathcal{C} \, |\lambda_{k-1} - \lambda_*| |\lambda_k - \lambda_*| \tag{5.6}$$

with $\mathcal{C}$ independent of k.

Proof. Let $y = -A_*^\dagger g$ and $\alpha_* = \lambda_* - g^T y$. We divide the proof into two cases $\|y\| = \Delta$ and $\|y\| < \Delta$. In each case, we first find an appropriate neighborhood of λ_* and then prove (5.6) for λ_{k-1}, λ_k in that neighborhood.

Case 1: $\|y\| = \Delta$.

We will first find a neighborhood $\mathcal{B}$ of λ_* such that if λ_{k-1}, $\lambda_k \in \mathcal{B}$, then $\widehat{\lambda} \in \mathcal{B}$, with $\widehat{\lambda}$ given by (4.4). In this part of the proof we define the numbers ℓ and m in the following way.

Let $0 \le \ell < n$ and assume that $g \perp \mathcal{S}_i$, $i = 1, 2, \ldots, \ell$, where $\ell = 0$ indicates that $g \not\perp \mathcal{S}_1$. Let $m = 0$ if $\lambda_* < \delta_1$ and $m = \ell$ if $\lambda_* = \delta_1$. Define

$$r_1 = \frac{\delta_{m+1} - \lambda_*}{2} \quad \text{and} \quad \mathcal{B}_1 = \{\lambda \mid |\lambda - \lambda_*| \le r_1\}$$

and suppose that $\lambda_{k-1}, \lambda_k \in \mathcal{B}_1$. Then by (4.4), Lemma 5.5(iii), and the fact that $\|y\| = \Delta$, we have

$$\begin{aligned} \widehat{\lambda} - \lambda_* &= \lambda_k - \lambda_* + \frac{\|x_{k-1}\|(\|x_{k-1}\| + \|x_k\|)(\Delta - \|x_k\|)}{\rho(k-1,k)\Delta} \\ &= (\lambda_k - \lambda_*)\left[1 - \frac{\rho(*,k)\|x_{k-1}\|(\|x_{k-1}\| + \|x_k\|)}{\rho(k-1,k)\Delta(\Delta + \|x_k\|)}\right]. \end{aligned} \tag{5.7}$$

We will prove now that $|\widehat{\lambda} - \lambda_*| \le |\lambda_k - \lambda_*|\vartheta$, with $\vartheta \in (0,1)$.

Let $\Delta_{max} = \max_{\lambda \in \mathcal{B}_1} \|(A - \lambda I)^\dagger g\|$ and $\Delta_{min} = \min_{\lambda \in \mathcal{B}_1} \|(A - \lambda I)^\dagger g\|$. Therefore

$$\frac{\|x_{k-1}\|(\|x_{k-1}\| + \|x_k\|)}{\Delta(\Delta + \|x_k\|)} \ge \left(\frac{\Delta_{min}}{\Delta_{max}}\right)^2. \tag{5.8}$$

In view of (5.5) we have that for $\lambda_{k-1},\ \lambda_k \in \mathcal{B}_1$

$$\rho(*,k) \ge \frac{(2\delta_{\ell+1} - \lambda_* - \lambda_k)}{(\delta_d - \lambda_*)^2(\delta_d - \lambda_k)^2}\|g\|^2 \ .$$

Since $\delta_{m+1} \le \delta_{\ell+1}$ and since $\frac{-\delta_{m+1}+\lambda_*}{2} \le \lambda_* - \lambda_k \le \frac{\delta_{m+1}-\lambda_*}{2}$, we have

$$\begin{aligned} \rho(*,k) &\ge \frac{(2\delta_{m+1} - \lambda_* - \lambda_k)}{(\delta_d - \lambda_*)^2(\delta_d - \lambda_k)^2}\|g\|^2 \\ &= \frac{2(\delta_{m+1} - \lambda_*) + (\lambda_* - \lambda_k)}{(\delta_d - \lambda_*)^2[(\delta_d - \lambda_*) + (\lambda_* - \lambda_k)]^2}\|g\|^2 \\ &\ge \frac{2}{3}\|g\|^2 \frac{(\delta_{m+1} - \lambda_*)}{(\delta_d - \lambda_*)^4} \ . \end{aligned} \tag{5.9}$$

Using similar manipulations we obtain

$$\begin{aligned} \rho(k-1,k) &\le \frac{(2\delta_d - \lambda_k - \lambda_{k-1})\|g\|^2}{(\delta_{\ell+1} - \lambda_k)^2(\delta_{\ell+1} - \lambda_{k-1})^2} \\ &\le \frac{(2\delta_d - \lambda_k - \lambda_{k-1})\|g\|^2}{(\delta_{m+1} - \lambda_k)^2(\delta_{m+1} - \lambda_{k-1})^2} \\ &\le \frac{3 \cdot 2^4\|g\|^2(\delta_d - \lambda_*)}{(\delta_{m+1} - \lambda_*)^4}. \end{aligned} \tag{5.10}$$

It follows from (5.7), (5.8), (5.9), and (5.10) that

$$|\widehat{\lambda} - \lambda_*| \le |\lambda_k - \lambda_*|\left|1 - \left(\frac{\Delta_{min}}{\Delta_{max}}\right)^2 \frac{1}{72}\frac{(\delta_{m+1} - \lambda_*)^5}{(\delta_d - \lambda_*)^5}\right| \equiv |\lambda_k - \lambda_*|\vartheta$$

with $\vartheta \in (0,1)$. Therefore $\widehat{\lambda} \in \mathcal{B}_1$ whenever $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}_1$.

Now, we use these results to establish (5.6). Let the neighborhood $\mathcal{B}$ be given by $\mathcal{B}_1$ and let $\lambda_{k-1}, \lambda_k \in \mathcal{B}$; therefore $\widehat{\lambda} \in \mathcal{B}$, and to prove (5.6) we need to consider two possibilities $\widehat{\lambda} < \delta_1$ and $\widehat{\lambda} \geq \delta_1$.

Case 1.1: $\|y\| = \Delta$ and $\widehat{\lambda} < \delta_1$.

In this case, we use formulas (4.4) and (4.6), obtaining

$$\alpha_{k+1} = T_1 + T_2,$$

where

$$T_1 = \frac{\alpha_{k-1}\|x_{k-1}\|(\|x_k\| - \Delta) + \|x_k\|(\Delta - \|x_{k-1}\|)}{\Delta(\|x_k\| - \|x_{k-1}\|)}$$

and

$$T_2 = \frac{\|x_k\|\|x_{k-1}\|(\Delta - \|x_k\|)(\Delta - \|x_{k-1}\|)(\lambda_k - \lambda_{k-1})}{\Delta(\|x_k\| - \|x_{k-1}\|)}.$$

We will now find an upper bound for $|T_2|$. From Lemma 5.5(iii), we have

$$T_2 = \frac{\|x_k\|\|x_{k-1}\|(\Delta - \|x_k\|)(\Delta - \|x_{k-1}\|)(\|x_k\| + \|x_{k-1}\|)}{\Delta\rho(k-1,k)},$$

and since $\|y\| = \Delta$ we can write

$$T_2 = \frac{\|x_k\|\|x_{k-1}\|(\|x_k\| + \|x_{k-1}\|)}{\rho(k-1,k)} \frac{(\|y\| - \|x_k\|)(\|y\| - \|x_{k-1}\|)}{\Delta}.$$

Using Lemma 5.5(iii) we obtain

$$T_2 = \frac{\|x_k\|\|x_{k-1}\|(\|x_k\| + \|x_{k-1}\|)}{\rho(k-1,k)} \frac{(\lambda_* - \lambda_{k-1})(\lambda_* - \lambda_k)\rho(k-1,*)\rho(k,*)}{\Delta(\Delta + \|x_k\|)(\Delta + \|x_{k-1}\|)}.$$

Now, since $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}$ and since $\delta_{m+1} \leq \delta_{\ell+1}$,

$$\rho(k-1,k) \geq \frac{2}{3}\|g\|^2\ \frac{(\delta_{m+1} - \lambda_*)}{(\delta_d - \lambda_*)^4}, \tag{5.11}$$

$$\rho(*,k),\ \rho(*,k-1) \leq 10\ \frac{\delta_d - \lambda_*}{(\delta_{m+1} - \lambda_*)^4}\ . \tag{5.12}$$

Since for $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}$ we also have $\Delta_{min} \leq \|x_{k-1}\|, \|x_k\| \leq \Delta_{max}$, we obtain

$$|T_2|\ \leq\ \mathcal{C}_2\ |\lambda_* - \lambda_{k-1}|\ |\lambda_* - \lambda_k|. \tag{5.13}$$

We will use this estimate in a moment. First, we need to relate it to $\lambda_{k+1} - \lambda_*$. To do this, consider

$$\alpha_{k+1} - \alpha_* = \frac{(\alpha_{k-1} - \alpha_*)\|x_{k-1}\|(\|x_k\| - \Delta) + (\alpha_k - \alpha_*)\|x_k\|(\Delta - \|x_{k-1}\|)}{\Delta(\|x_k\| - \|x_{k-1}\|)} + T_2\ . \tag{5.14}$$

From Lemma 5.5(i), the definition of α_*, and since $\alpha_j - \lambda_j = -g^T x_j$ for $j > 0$, we have

$$\begin{aligned} \alpha_j - \alpha_* &= \lambda_j - \lambda_* - g^T(x_j - y) \\ &= (\lambda_j - \lambda_*)(1 + x_j^T y). \end{aligned} \tag{5.15}$$

Using (5.15) along with Lemma 5.5, (5.14) becomes

$$\begin{aligned}(\lambda_{k+1}-\lambda_*)(1+x_{k+1}^T y) &= \frac{(\lambda_{k-1}-\lambda_*)\|x_{k-1}\|(\|x_k\|-\Delta)(1+x_{k-1}^T y)}{\Delta(\|x_k\|-\|x_{k-1}\|)} \\ &\quad - \frac{(\lambda_k-\lambda_*)\|x_k\|(\|x_{k-1}\|-\Delta)(1+x_k^T y)}{\Delta(\|x_k\|-\|x_{k-1}\|)} + T_2 \\ &= \frac{(\lambda_{k-1}-\lambda_*)(\lambda_k-\lambda_*)T_3}{\Delta(\|x_k\|-\|x_{k-1}\|)(\|x_k\|+\Delta)(\|x_{k-1}\|+\Delta)} + T_2,\end{aligned}$$

where

$$\begin{aligned}T_3 = &(\|x_{k-1}\|-\|x_k\|)(\|x_{k-1}\|+\|x_k\|+\Delta)(1+x_k^T y)\rho(k-1,*) \\ &+ y^T(x_{k-1}-x_k)\|x_{k-1}\|(\|x_{k-1}\|+\Delta)\rho(k,*) \\ &+ (\rho(k,*)-\rho(k-1,*))\|x_{k-1}\|(\|x_{k-1}\|+\Delta)(1+x_k^T y)\ .\end{aligned}$$

Now, by Lemma 5.5(ii) we have

$$y^T(x_{k-1}-x_k) = (\lambda_{k-1}-\lambda_k)y^T A_k^\dagger x_{k-1} \tag{5.16}$$

and by Lemma 5.5(iii)

$$\begin{aligned}\rho(k,*)-\rho(k-1,*) &= x_k^T A_*^\dagger x_k + y^T A_k^\dagger y - x_{k-1}^T A_*^\dagger x_{k-1} - y A_{k-1}^\dagger y \\ &= g^T A_k^\dagger A_*^\dagger A_k^\dagger g - g^T A_{k-1}^\dagger A_*^\dagger A_{k-1}^\dagger g + y^T(A_k^\dagger - A_{k-1}^\dagger)y \\ &= g^T((A_k^\dagger)^2-(A_{k-1}^\dagger)^2)y + (\lambda_k-\lambda_{k-1})y^T A_k^\dagger A_{k-1}^\dagger y \\ &= (\lambda_k-\lambda_{k-1})(y^T A_{k-1}^\dagger A_k^\dagger y + x_{k-1}^T(A_k^\dagger + A_{k-1}^\dagger)A_k^\dagger y).\end{aligned} \tag{5.17}$$

Therefore by (5.12), (5.16), (5.17), and since $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}$, we have

$$|T_3| \le \mathcal{C}_3|\lambda_k-\lambda_{k-1}|\ .$$

We may now combine the estimates we have established for $T_1,\ T_2$, and T_3 to give

$$\begin{aligned}|\lambda_{k+1}-\lambda_*||1+x_{k+1}^T y| &\le \mathcal{C}_3\frac{|\lambda_{k-1}-\lambda_*|\ |\lambda_k-\lambda_*|\ |\lambda_k-\lambda_{k-1}|}{\Delta|(\|x_k\|-\|x_{k-1}\|)(\|x_k\|+\Delta)(\|x_{k-1}\|+\Delta)|} + |T_2| \\ &\le (\mathcal{C}_4+\mathcal{C}_2)|\lambda_{k-1}-\lambda_*|\ |\lambda_k-\lambda_*|\end{aligned}$$

since $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}$, and (5.13) holds. Let us see now that $\frac{1}{1+x_{k+1}^T y} < 1$. Note that

$$\begin{aligned}x_{k+1}^T y &= g^T A_{k+1}^\dagger A_*^\dagger g \\ &= \sum_{j=\ell+1}^{d}\frac{\beta_j^2}{(\delta_j-\lambda_{k+1})(\delta_j-\lambda_*)} \ge \frac{\|g\|^2}{(\delta_d-\lambda_{k+1})(\delta_d-\lambda_*)}\ .\end{aligned} \tag{5.18}$$

From this expression we can conclude $x_{k+1}^T y > 0$ since $\lambda_* < \delta_{m+1} \le \delta_d$ and also since $\lambda_{k+1} < \delta_{m+1} \le \delta_d$, by the way we compute the iterates in Algorithm 4.1.

We can now claim (5.6) when $\|y\| = \Delta$ and $\widehat{\lambda} < \delta_1$.

Case 1.2: $\|y\| = \Delta$ and $\widehat{\lambda} \ge \delta_1$.

In this case we must use the ideal safeguard, setting $\widehat{\lambda} = \delta_1$. Before proceeding with the proof, we point out that this can occur only when $\lambda_* = \delta_1$. To see this recall that, if $\lambda_* < \delta_1$, we have $\mathcal{B} = \{\lambda \mid |\lambda - \lambda_*| \leq \frac{\delta_1 - \lambda_*}{2}\}$, and we proved that $\lambda_{k-1},\ \lambda_k \in \mathcal{B}$ implies $\widehat{\lambda} \in \mathcal{B}$. Therefore

$$\widehat{\lambda} - \lambda_* \leq \frac{\delta_1 - \lambda_*}{2},$$

$$\widehat{\lambda} \leq \frac{\delta_1 + \lambda_*}{2} < \delta_1$$

if $\lambda_* < \delta_1$.

To continue with the proof we write the formula for α_{k+1} in this case as

$$\alpha_{k+1} = T_4 + T_5,$$

where

$$T_4 = \omega\alpha_{k-1} + (1-\omega)\alpha_k$$

and

$$T_5 = \frac{\|x_k\|\|x_{k-1}\|(\|x_k\| - \|x_{k-1}\|)}{\omega\|x_k\| + (1-\omega)\|x_{k-1}\|} \frac{(\lambda_{k-1} - \delta_1)(\lambda_k - \delta_1)}{(\lambda_k - \lambda_{k-1})}.$$

Since $\widehat{\lambda} = \delta_1$, we have $\omega = \frac{\lambda_k - \delta_1}{\lambda_k - \lambda_{k-1}}$ and therefore

$$\begin{aligned} \omega\|x_k\| + (1-\omega)\|x_{k-1}\| &= \frac{(\lambda_k - \delta_1)\|x_k\| + (\delta_1 - \lambda_{k-1})\|x_{k-1}\|}{\lambda_k - \lambda_{k-1}} \\ &= \frac{(\lambda_k - \delta_1)\rho(k-1,k) + \|x_{k-1}\|(\|x_{k-1}\| + \|x_k\|)}{\|x_{k-1}\| + \|x_k\|} \end{aligned} \tag{5.19}$$

by Lemma 5.5(iii).

Using (5.19), (5.4), and Lemma 5.5(iii), we obtain

$$T_5 = \frac{\|x_k\|\|x_{k-1}\|\rho(k,k-1)(\lambda_{k-1} - \delta_1)(\lambda_k - \delta_1)}{(\lambda_k - \delta_1)\rho(k-1,k) + \|x_{k-1}\|(\|x_{k-1}\| + \|x_k\|)}\ .$$

By (5.11) and the hypothesis that $\lambda_{k-1},\ \lambda_k \in \mathcal{B}$, we have

$$|T_5| \leq \mathcal{C}_5|\lambda_{k-1} - \delta_1||\lambda_k - \delta_1|. \tag{5.20}$$

We now write

$$\begin{aligned} \alpha_{k+1} - \alpha_* &= \omega\alpha_{k-1} + (1-\omega)\alpha_k - \alpha_* + T_5 \\ &= \omega(\alpha_{k-1} - \alpha_*) + (1-\omega)(\alpha_k - \alpha_*) + T_5. \end{aligned}$$

Equation (5.14) and the fact that $\lambda_* = \delta_1$ yield

$$|\lambda_{k+1} - \lambda_*||1 + y^T x_{k+1}| \leq \frac{|\lambda_{k-1} - \lambda_*||\lambda_k - \lambda_*|}{|\lambda_k - \lambda_{k-1}|}\ |y^T(x_k - x_{k-1})| + |T_5|\ .$$

Observe that $|y^T(x_k - x_{k-1})| \leq |g^T(A - \lambda_k I)^\dagger(A - \lambda_{k-1} I)^\dagger(A - \lambda_* I)^\dagger g|$, and we can compute an upper bound for this term using the Cauchy–Schwarz inequality, continuity of $\|\cdot\|$, and that $\lambda_{k-1},\ \lambda_k \in \mathcal{B}$. Therefore, by (5.20) and since $\lambda_{k-1},\ \lambda_k \in \mathcal{B}$

$$|\lambda_{k+1} - \lambda_*||1 + y^T x_{k+1}| \leq (\mathcal{C}_6 + \mathcal{C}_5)|\lambda_{k-1} - \lambda_*||\lambda_k - \lambda_*|.$$

Using (5.18), we can now establish (5.6) when $\|y\| = \Delta$ and $\widehat{\lambda} \geq \delta_1$.

Case 2: $\|y\| < \Delta$.

In this situation, we are in the hard case and therefore $\lambda_* = \delta_1$ and $g \perp \mathcal{S}_i$, $i = 1, 2, \ldots, \ell$, with $1 \leq \ell < d$. For this case we will find a neighborhood $\mathcal{B}$ of λ_* such that $\lambda_{k-1},\ \lambda_k \in \mathcal{B}$ implies $\widehat{\lambda} > \delta_1$.

Let the function $\varphi(\lambda) \equiv \|(A - \lambda I)^\dagger g\|$. Then $\varphi(\lambda)$ is strictly increasing in $(-\infty, \delta_{\ell+1})$, and there exist $\lambda_a,\ \lambda_b$ such that $\varphi(\lambda_a) = \frac{\Delta_y}{2}$ and $\varphi(\lambda_b) = \frac{\Delta + \Delta_y}{2}$, with $\Delta_y = \|y\|$.

Let

$$r_2 = \min\left\{\frac{\delta_1 - \lambda_a}{2}, \frac{\delta_1 - \lambda_b}{2}, \frac{\delta_{\ell+1} - \delta_1}{2}\right\} \quad \text{and} \quad \mathcal{B}_2 = \{\lambda \mid |\lambda - \lambda_*| \leq r_2\}.$$

Then for $\lambda_{k-1},\ \lambda_k \in \mathcal{B}_2$

$$\frac{\Delta_y}{2} \leq \|x_{k-1}\|,\ \|x_k\| \leq \frac{\Delta + \Delta_y}{2},$$
$$\text{and} \quad \delta_1 - r_2 \leq \lambda_{k-1},\ \lambda_k \leq \delta_1 + r_2 < \delta_{\ell+1}\ .$$

Now observe that using (4.4) and Lemma 5.5(iii) we can write

$$\begin{aligned}
\widehat{\lambda} - \lambda_* &= \widehat{\lambda} - \delta_1 \\
&= \frac{(\lambda_{k-1} - \delta_1)\|x_{k-1}\|(\|x_k\| - \Delta) + (\lambda_k - \delta_1)\|x_k\|(\Delta - \|x_{k-1}\|)}{\Delta(\|x_k\| - \|x_{k-1}\|)} - \delta_1 \\
&= \lambda_k - \delta_1 + \frac{\|x_{k-1}\|(\Delta - \|x_k\|)(\|x_{k-1}\| + \|x_k\|)}{\Delta\rho(k, k-1)} \\
&\geq \lambda_k - \delta_1 + \frac{{\Delta_y}^2(\Delta - \Delta_y)}{4\Delta\rho(k, k-1)}\ . \qquad (5.21)
\end{aligned}$$

Observe now that for $\lambda_{k-1},\ \lambda_k \in \mathcal{B}_2$

$$\rho(k, k-1) \leq \frac{3 \cdot 2^4 \|g\|^2(\delta_d - \delta_1)}{(\delta_{\ell+1} - \delta_1)^4}\ . \qquad (5.22)$$

Using (5.21) and (5.22) we obtain

$$\widehat{\lambda} - \delta_1 \geq \lambda_k - \delta_1 + \frac{{\Delta_y}^2(\Delta - \Delta_y)(\delta_{\ell+1} - \delta_1)^4}{3 \cdot 2^6 \Delta \|g\|^2(\delta_d - \delta_1)}\ .$$

Let

$$\zeta \equiv \frac{{\Delta_y}^2(\Delta - \Delta_y)(\delta_{\ell+1} - \delta_1)^4}{3 \cdot 2^6 \Delta \|g\|^2(\delta_d - \delta_1)}.$$

Observe that ζ is well defined since $\delta_d = \delta_1$ would imply $g \in \mathbb{R}^{n\ \perp} = \{0\}$. Observe also that $\zeta > 0$, since $\Delta > \Delta_y$ and $\delta_{\ell+1} > \delta_1$. Then, for $\lambda_k \geq \delta_1 - \zeta$ we have $\widehat{\lambda} \geq \delta_1$. So, let

$$r_3 = \min\{r_2, \zeta\} \quad \text{and} \quad \mathcal{B}_3 = \{\lambda \mid |\lambda - \delta_1| \leq r_3\}.$$

It follows that for $\lambda_{k-1},\ \lambda_k\ \in\ \mathcal{B}_3$, we have $\widehat{\lambda} \geq \delta_1$ and we must use the ideal safeguard, setting $\widehat{\lambda} = \delta_1$. The proof now proceeds as in Case 1.2, where the neighborhood $\mathcal{B}$ is given by $\mathcal{B}_3$, and we use (5.22) instead of (5.10).

The analysis of the two cases concludes the proof. ∎

Note that the assumption in Lemma 5.6 that the trust-region constraint is binding at the solution includes the possibility of the hard case, since in this case $x_* = -A_*^{\dagger} g + z$, with $z \in \mathcal{S}_1$ and $\|x_*\| = \Delta$.

5.2.3. Local convergence result.

THEOREM 5.1. *Let $\lambda_* \leq \delta_1$ be the Lagrange multiplier corresponding to a boundary solution of problem* (2.1)*, with $g \neq 0$. Let $\{\lambda_k\}$, $\{x_k\}$ be the sequences of iterates generated by Algorithm* 4.1 *using the two-point interpolating scheme given by* (4.6) *to update α. There exists a neighborhood $\mathcal{B}$ of λ_* such that if $\lambda_{i-1},\ \lambda_i \in \mathcal{B}$, then for $k \geq i - 1$*

(i) *$\{\lambda_k\}$ remains in $\mathcal{B}$ and converges q-superlinearly to λ_*;*

(ii) *$\{x_k\}$ converges q–superlinearly to $y = -(A\ -\ \lambda_*\ I)^{\dagger} g$.*

Proof. First we show that $\{\lambda_k\}$ converges to λ_* and that the rate of convergence is superlinear.

Let $r \in \mathbb{R}, r > 0$, and $\mathcal{B} = \{\lambda \mid |\lambda - \lambda_*| < r\}$ be the neighborhood of λ_* stated in Lemma 5.6 and suppose that $\lambda_{i-1},\ \lambda_i\ \in\ \mathcal{B}$, for $i \geq 1$. Then, Lemma 5.6 implies that there exists $\mathcal{C}$ such that

$$|\lambda_{i+1} - \lambda_*| \leq \mathcal{C}\ |\lambda_i - \lambda_*|\ |\lambda_{i-1} - \lambda_*|. \tag{5.23}$$

Let $\hat{r} = \min\{r, \frac{1}{2\mathcal{C}}\}$, define $\widehat{\mathcal{B}} = \{\lambda \mid |\lambda - \lambda_*| < \hat{r}\}$, and observe that $\widehat{\mathcal{B}} \subset \mathcal{B}$. Suppose $\lambda_{i-1},\ \lambda_i\ \in\ \widehat{\mathcal{B}}$; then $\lambda_{i-1},\ \lambda_i\ \in\ \mathcal{B}$, and (5.23) holds.

Observe now that for $\lambda_{i-1},\ \lambda_i\ \in\ \widehat{\mathcal{B}}$ we have $\mathcal{C}|\lambda_{i-1} - \lambda_*| \leq \frac{1}{2}$ and therefore

$$|\lambda_{i+1} - \lambda_*| \leq \frac{1}{2}|\lambda_i - \lambda_*|$$

which implies $\lambda_{i+1}\ \in\ \widehat{\mathcal{B}} \subset \mathcal{B}$.

It follows inductively that if $\lambda_{i-1},\ \lambda_i\ \in\ \mathcal{B}$, then $\lambda_k \in \mathcal{B}$ for $k \geq i - 1$ and this implies

$$|\lambda_k - \lambda_*| \leq \frac{1}{2^{k-i+1}}|\lambda_{i-1} - \lambda_*|$$

and therefore $\lambda_k \to \lambda_*$ as $k \to \infty$.

To see that the rate of convergence is q-superlinear, observe that, by (5.23), for $k \geq i$ we have

$$\frac{|\lambda_{k+1} - \lambda_*|}{|\lambda_k - \lambda_*|} = \mathcal{C}|\lambda_{k-1} - \lambda_*|,$$

which goes to zero as k goes to infinity.

In the second part of the proof we show that the sequence $\{x_k\}$ converges superlinearly to $y = -(A - \lambda_* I)^\dagger g$.

Recall from Lemma 5.4 that $x_k = -A_k^\dagger g$ and let us study $x_k - y$ which is given by

$$\begin{aligned} x_k - y &= (A - \lambda_* I)^\dagger g - (A - \lambda_k I)^\dagger g \\ &= (A - \lambda_* I)^\dagger ((A - \lambda_k I) - (A - \lambda_* I))(A - \lambda_k I)^\dagger g \\ &= (\lambda_* - \lambda_k)(A - \lambda_* I)^\dagger (A - \lambda_k I)^\dagger g \end{aligned}$$

using (5.1) and rearranging terms. Taking norms on both sides we have

$$\begin{aligned} \|x_k - y\| &= |\lambda_k - \lambda_*| \, \|(A - \lambda_* I)^\dagger\| \|(A - \lambda_k I)^\dagger\| \, \|g\| \\ &\leq \widehat{\mathcal{C}} |\lambda_k - \lambda_*| \end{aligned} \tag{5.24}$$

for a positive constant $\widehat{\mathcal{C}}$, since $\lambda_k \in \mathcal{B}$, $\lambda_* \leq \delta_1$, and $\|g\|$ is constant. Therefore, since $\lambda_k \to \lambda_*$ as $k \to \infty$, we have that $x_k \to y$ as $k \to \infty$.

To see that the rate of convergence is q-superlinear, observe that (5.23) and (5.24) imply

$$\frac{\|x_{k+1} - y\|}{\|x_k - y\|} \leq |\lambda_{k-1} - \lambda_*|$$

which goes to zero as k goes to infinity. This completes the proof. ∎

5.2.4. Near hard case. The next lemma provides a relationship between the function ϕ and the interpolating function (4.5). We will use this relationship in the analysis of the near hard case.

LEMMA 5.7. *At iteration k of Algorithm* 4.1 *the interpolating function* (4.5) *satisfies*

$$\widehat{\phi}(\widehat{\lambda}) - \phi(\lambda_k) = (\widehat{\lambda} - \lambda_k) \left[x_k^T x_{k-1} + \frac{\|x_{k-1}\| \|x_k\| \rho(k, k-1)(\widehat{\lambda} - \lambda_{k-1})}{(\lambda_k - \widehat{\lambda})\rho(k, k-1) + \|x_{k-1}\|(\|x_k\| + \|x_{k-1}\|)} \right],$$

with $\rho(k, k-1)$ as in (5.4) *and $\widehat{\lambda}$ given by* (4.4).

Proof. By (4.5) and Lemma 5.5(iii), we have

$$\begin{aligned} \widehat{\phi}(\widehat{\lambda}) &= \frac{\gamma^2}{\delta - \widehat{\lambda}} + \omega\phi(\lambda_{k-1}) - \omega \frac{\gamma^2}{\delta - \lambda_{k-1}} \\ &\quad + (1-\omega)\phi(\lambda_k) - (1-\omega)\frac{\gamma^2}{\delta - \lambda_k} \\ &= \phi(\lambda_k) + \omega g^T (x_k - x_{k-1}) + \frac{\omega\gamma^2(\widehat{\lambda} - \lambda_{k-1})}{(\delta - \widehat{\lambda})(\delta - \lambda_{k-1})} \\ &\quad + \frac{(1-\omega)\gamma^2(\widehat{\lambda} - \lambda_k)}{(\delta - \widehat{\lambda})(\delta - \lambda_k)} \\ &= \phi(\lambda_k) + (\widehat{\lambda} - \lambda_k) x_k^T x_{k-1} + \frac{\gamma^2(\widehat{\lambda} - \lambda_k)(\widehat{\lambda} - \lambda_{k-1})}{(\delta - \widehat{\lambda})(\delta - \lambda_k)(\delta - \lambda_{k-1})} \\ &= \phi(\lambda_k) + (\widehat{\lambda} - \lambda_k) x_k^T x_{k-1} \\ &\quad + \frac{\|x_{k-1}\| \|x_k\| (\|x_k\| - \|x_{k-1}\|)(\widehat{\lambda} - \lambda_k)(\widehat{\lambda} - \lambda_{k-1})}{(\omega\|x_k\| + (1-\omega)\|x_{k-1}\|)(\lambda_k - \lambda_{k-1})}, \end{aligned}$$

where $\omega = \frac{\lambda_{k-1} - \widehat{\lambda}}{\lambda_k - \lambda_{k-1}}$. Thus, the result follows from Lemma 5.5. ∎

A few comments are in order concerning the near hard case. As mentioned in section 3, finding $\lambda_* < \delta_1$ in a near hard case is a very ill-conditioned process. The difference $\delta_1 - \lambda_*$ can be very small to the extent of being undetectable within the given tolerances. The smaller the value $\delta_1 - \lambda_*$, the harder it is to determine $\{\lambda_*, x_*\}$. Furthermore, rounding errors generally will convert an exact hard case into a near hard case. Although δ_1 is still a pole of ϕ when g is not exactly orthogonal to $\mathcal{S}_1$, the weight of such a pole is very small in comparison to the other poles because the expansion coefficients of g in the basis of eigenvectors of A are practically zero for those eigenvectors associated with δ_1. The strategy that we follow in Algorithm 4.1 for dealing with this case consists of building an interpolating function that *ignores* the pole δ_1 at early stages, using the eigenpair corresponding to the second smallest eigenvalue of B_{α_k} to obtain the interpolation points. In addition, we use the second eigenpair to compute a vector that might be a quasi-optimal solution for the trust-region subproblem as established in Theorem 3.2. Moreover, as that theorem and related results established, it is not necessary to compute an eigenpair corresponding to the second smallest eigenvalue. This is especially useful when the vector g is orthogonal or nearly orthogonal to several eigenspaces corresponding to the smallest eigenvalues of A, and those eigenvalues are clustered.

If we use information concerning a second eigenpair, then we will have $\lambda_k > \delta_1$. This occurs because the first component ν_1 of the eigenvector $(\nu_1, u_2^T)^T$ associated with $\lambda_1(\alpha_k)$ is too small so that $\|u_1/\nu_1\| = \|x_k\|$ becomes excessively large. Therefore $\{\lambda_k, x_k\}$ is defined as $\{\lambda_i(\alpha_k), u_2/\nu_2\}$. Intuitively, this is a good strategy since in the exact hard case this would continuously select the correct eigenvector that will approach $(1, p_1^T)^T$ when α tends to the value $\tilde{\alpha}_1$, stated in Lemma 3.2, from either side.

Now, at iteration k the parameter α_k is updated as $\alpha_{k+1} = \widehat{\lambda} + \widehat{\phi}(\widehat{\lambda})$ with $\widehat{\lambda} \leq \delta_U$, where either $\widehat{\lambda} < \delta_U$, $\widehat{\phi}'(\widehat{\lambda}) = \Delta^2$, or $\widehat{\lambda} = \delta_U$, $\widehat{\phi}'(\delta_U) < \Delta^2$. By the same arguments of the proof of Case 1 in Lemma 5.6, there exists a neighborhood $\mathcal{B}$ of λ_* such that if $\lambda_{k-1}, \lambda_k \in \mathcal{B}$, then $\widehat{\lambda} \in \mathcal{B}$, with $|\widehat{\lambda} - \lambda_k| = \vartheta|\lambda_* - \lambda_k|$, for $\vartheta \in (0,1)$. In other words, eventually the safeguarding $\widehat{\lambda} = \delta_U$ is no longer necessary. If $\lambda_{k-1}, \lambda_k \in \mathcal{B}$, then Lemma 5.7 implies that $|\widehat{\phi}(\widehat{\lambda}) - \phi(\lambda_k)| = \vartheta|\widehat{\lambda} - \lambda_k||\lambda_* - \lambda_k|$. The agreement between $\widehat{\lambda}$ and λ_k and between $\widehat{\phi}(\widehat{\lambda})$ and $\phi(\lambda_k)$ drive α_k toward $\alpha_* = \lambda_* + \phi(\lambda_*)$. As α_k approaches α_*, the reduction of the safeguarding interval $[\alpha_L, \alpha_U]$ at every iteration provides a means to avoid the numerical difficulties associated with a near hard case, and eventually there is no need to use a second eigenpair of B_{α_k}. At early stages, however, it might be that $\widehat{\lambda} = \delta_U$. Although $\phi(\delta_1)$ is infinite, the interpolating function value $\widehat{\phi}(\delta_U)$ is finite. Using $\alpha_{k+1} = \delta_U + \widehat{\phi}(\delta_U)$ is essential in keeping the process under control.

5.3. Global convergence.

THEOREM 5.2. *Algorithm* 4.1 *is globally convergent.*

Proof. The goal of Algorithm 4.1 is to solve the trust-region subproblem either by determining the existence of an interior solution or by computing an optimal value α_* for the parameter α, such that the solution to the parameterized eigenvalue problem for B_{α_*} can be used to compute a boundary solution for the trust-region subproblem. The global convergence of Algorithm 4.1 is achieved by keeping α_k in an interval that contains the optimal parameter α_*.

We first recall that the initial safeguarding interval $[\alpha_L, \alpha_U]$ contains the optimal value α_*. Starting with that interval, the updating procedure for α_L and α_U guaran-

tees that α_* remains in the interval and that the safeguarding interval is reduced at each iteration.

Therefore, since $\alpha_k = \lambda_k - g^T x_k$, after a finite number of iterations either the sequence $\{\lambda_k\}$ reaches the neighborhood of λ_* of Theorem 5.1 that guarantees convergence, or the length of the safeguarding interval $|\alpha_U - \alpha_L|$ goes to zero with $\alpha_L \leq \alpha_* \leq \alpha_U$. □

6. Numerical experiments. In this section we present numerical experiments to demonstrate the viability of our approach and to illustrate different aspects of our method. We implemented Algorithm 4.1 (LSTRS) in MATLAB 5.3 using a Mexfile interface to access the IRLM [16] implemented in ARPACK [7]. We ran our experiments on a Sun Ultrasparc 10 with a 300 MHz processor and 256 megabytes of RAM, running Solaris 5.6. The floating point arithmetic was IEEE standard double precision with machine precision $2^{-52} \approx 2.2204 \cdot 10^{-16}$. We present five sets of experiments. In the first and second sets we study the sensitivity of LSTRS to different tolerances for the trust-region radius and to different sizes of the trust-region radius, respectively, for problems where the hard case is not present. In order to put our method in context, we include the number of matrix-vector products required by the conjugate gradient method to solve systems of the form $(A - \lambda I)x = -g$. The third set of experiments illustrates the local superlinear rate of convergence. The fourth set shows the behavior of LSTRS in the hard case. In the fifth set we provide a comparison with the semidefinite programming approach presented in [13].

The following tolerances are fixed in all the experiments: $\varepsilon_\nu = 10^{-2}$, $\varepsilon_\alpha = 10^{-8}$, $\varepsilon_{Int} = 10^{-8}$. We will indicate the values for the rest of the parameters when we describe each particular set of experiments.

6.1. Different tolerances. In the first experiment, we show the behavior of the method when different levels of accuracies of the norm of the trust-region solution are required. The matrix A in (2.1) was $A = L - 5I$, where L is the standard two-dimensional (2D) discrete Laplacian on the unit square based upon a 5-point stencil with equally spaced mesh points. The shift of $-5I$ was introduced to make A indefinite. The order of A was $n = 1024$. We solved a sequence of 20 related problems, differing only by the vector g, randomly generated with entries uniformly distributed on $(0, 1)$. We solved each of these problems for a fixed trust-region radius $\Delta = 100$ and for $\varepsilon_\Delta = 10^{-4}, 10^{-6}, 10^{-8}$, where ε_Δ is the relative accuracy of the norm of the computed solution with respect to Δ. The initial δ_U was the minimum of the diagonal of A and $\alpha_0 = \delta_U$. The tolerance for a quasi-optimal solution was set to $\varepsilon_{HC} = 10^{-16}$ in order to allow the method to compute a boundary solution; otherwise the quasi-optimal stopping criterion would be satisfied first.

For $\varepsilon_\Delta = 10^{-4}, 10^{-6}$ the number of Lanczos basis vectors was limited to 9, and 6 shifts (i.e., 6 matrix-vector products) were applied on each implicit restart, while for $\varepsilon_\Delta = 10^{-8}$, the number of vectors was 20 with 14 shifts on each implicit restart. The maximum number of restarts allowed was 45 for $\varepsilon_\Delta = 10^{-4}, 10^{-6}$ and 100 for $\varepsilon_\Delta = 10^{-8}$. More basis vectors were needed for $\varepsilon_\Delta = 10^{-8}$, since in this case the eigenvalues were computed to a higher accuracy. We chose v_1, the initial vector for the IRLM, in the following way. In the first iteration of LSTRS, $v_1 = (1, 1, \ldots, 1)/\sqrt{n+1}$ and subsequently, v_1 was the first column of the matrix V containing the Lanczos vectors computed by the IRLM for the previous bordered matrix. This choice standardized the initial vector along the set of tests and performed better than a randomly generated vector, or the eigenvector corresponding to the smallest eigenvalue of B_{α_k}, or the vector $(0, g^T)^T$. Note that the

TABLE 1
Average behavior for different tolerances.

ε_Δ	LSTRS IT	LSTRS MV	CG MV	$\frac{LSTRS\ MV}{CG\ MV}$
10^{-4}	6.00	64.00	43.45	1.47
10^{-6}	7.50	83.00	57.00	1.46
10^{-8}	6.55	183.40	71.55	2.56

TABLE 2
Average behavior for different tolerances allowing quasi-optimal solutions.

ε_Δ	LSTRS IT	LSTRS MV	CG MV	$\frac{LSTRS\ MV}{CG\ MV}$
10^{-4}	5.00	54.00	48.80	1.11
10^{-6}	5.40	62.00	62.90	0.99
10^{-8}	5.40	64.75	76.95	0.84

last two options have the additional disadvantage of preventing the IRLM from finding the eigenspace of B_α corresponding to δ_1 whenever a potential hard case is present. As in [17], we relaxed the accuracy required in the eigenvalue solution and made it proportional to the relative accuracy in the computed solution. Specifically, $\|B_\alpha q - q\lambda\| < \varepsilon_{Lan}$, where $\varepsilon_{Lan} = \max\{\min\{\varepsilon_{Lan}, |\frac{\Delta - \|x\|}{\Delta}|\}, \varepsilon_{max}\}$ and $\varepsilon_{max} = 0.125, 0.1, 0.075$ for $\varepsilon_\Delta = 10^{-4},\ 10^{-6},\ 10^{-8}$, respectively.

In Table 1 we report the average number of iterations of LSTRS (LSTRS IT), the average number of matrix-vector products required by LSTRS (LSTRS MV), and the average number of matrix-vector products required by the conjugate gradient method (CG MV) to solve the system $(A - \lambda_* I)x = -g$ to the same accuracy ε_Δ in the norm of the computed solution of LSTRS. The value of λ_* was the optimal value computed by LSTRS.

We observe that for $\varepsilon_\Delta = 10^{-4}, 10^{-6}$ the behavior in [17] is reproduced: a trust-region solution requires fewer than twice as many matrix-vectors products on average than the number needed to solve a *single* linear system to the same accuracy using conjugate gradients. For $\varepsilon_\Delta = 10^{-8}$, even though LSTRS requires more matrix-vector products, the cost of LSTRS is less than three times the cost of solving one system by conjugate gradients.

If we repeat the experiment, setting the tolerance for a quasi-optimal solution to $\varepsilon_{HC} = 10^{-6}$, we obtain the results in Table 2, where we observe the low number of matrix-vector products required by LSTRS. In this experiment we used nine Lanczos basis vectors for all cases and allowed a maximum of 45 restarts.

6.2. Different trust-region radii. The second experiment illustrates the behavior of LSTRS for different sizes of the trust-region radius. The matrix A in (2.1) was of the form $A = UDU^T$ with D diagonal and $U = I - 2uu^T$, $u^T u = 1$. The elements of D were randomly selected from a uniform distribution on $(-5, 5)$. Both vectors u and g were randomly generated with entries uniformly distributed on $(-0.5, 0.5)$ and then u was normalized to have unit length. The order of A was $n = 1000$. We solved a sequence of 10 problems generated with different seeds, for a fixed tolerance $\varepsilon_\Delta = 10^{-6}$ and Δ varying from 100 to 0.0001 by a factor of 10, for a total of 70 problems. The initial δ_U was set to -4.5 and $\alpha_0 = \min\{0, \alpha_U\}$. The tolerance for a quasi-optimal solution was set to $\varepsilon_{HC} = 10^{-6}$.

The parameters for the IRLM were the following. For $\Delta = 100, 10$ the number of Lanczos basis vectors was 30, and 20 shifts were applied on each implicit restart, while

TABLE 3
Average behavior for different trust-region radii.

Δ	100	10	1	0.1	0.01	0.001	0.0001
IT	9.9	7.7	4.6	4	3.8	3	3
LSTR MV	1032.4	354	46.9	38.8	37.2	29.4	29.4
CG MV	921.4	410	29.3	28.7	27.2	12	12.5
$\frac{\|g+(A\ -\ \lambda_*\ I)x_*\|}{\|g\|}$	10^{-3}	10^{-3}	10^{-2}	10^{-11}	10^{-2}	10^{-2}	10^{-13}
$\left\|\frac{\Delta-\|x_*\|}{\Delta}\right\|$	10^{-16}	10^{-16}	10^{-8}	10^{-9}	10^{-12}	10^{-10}	10^{-7}

for $\Delta \leq 1$, the number of vectors was nine with six shifts on each implicit restart. The maximum number of restarts was 150 and 45, respectively. The difference in the number of basis vectors is due to the fact that for larger radii the hard case and near hard case are more likely to occur, and therefore the smallest eigenvalues of the bordered matrix become more clustered and the IRLM needs more space and iterations to compute the desired eigenpairs to the required accuracy. The initial vector for the IRLM was chosen as in section 6.1. We relaxed the accuracy required in the eigenvalue solution in the following way. The initial values for ε_{Lan} were $0.03, 0.1$, and 0.25 for $\Delta = 100, 10$, and $\Delta < 10$, respectively. The value of ε_{Lan} was kept the same until $|\frac{\Delta-\|x_k\|}{\Delta}| < 0.1$, when $\varepsilon_{Lan} = 0.015, 0.05$, and 0.125 for $\Delta = 100, 10$ and $\Delta < 10$, respectively. The results of the experiment are shown in Table 3, where we also report the average number of matrix-vector products required by the conjugate gradient method to solve the systems $(A\ -\ \lambda_k\ I)x = -g$ for λ_k generated by LSTRS.

As observed in [17], the conjugate gradient method has a much easier time for smaller values of Δ.

6.3. Superlinear convergence. The purpose of the third experiment was to verify superlinear convergence. The matrix A was again set to $A = L - 5I$ with L the 2D discrete Laplacian on the unit square, but now $n = 256$. The vector g was randomly generated with entries uniformly distributed on $(-0.5, 0.5)$. We studied problems with and without hard case. To generate the hard case, we orthogonalized the vector g randomly generated as before against the eigenvector q corresponding to the smallest eigenvalue of A. We accomplished this by setting $g \leftarrow g - q(q^T g)$. For the problem without hard case the trust-region radius was $\Delta = 10$ and $\varepsilon_\Delta = 10^{-11}$. For the problem with hard case the radius was $\Delta = 100$ and $\varepsilon_{HC} = 10^{-11}$. The eigenproblems were solved with the MATLAB routine `eig`. The results are shown in Table 4, where we report the quantity $|\frac{\Delta-\|x_k\|}{\Delta}|$ for the problem without hard case and the quantity $\frac{(\lambda_i(\alpha)-\lambda_1(\alpha))\ \tau_2^2\ (1+\Delta^2)}{-2\eta\psi(\widetilde{x})}$ from Theorem 3.2 for the problem with hard case.

The quantity $\|(A\ -\ \lambda_*\ I)x_* + g\|/\|g\|$ was of order 10^{-14} for problem (a) and 10^{-7} for problem (b). An asterisk $*$ in the hard case means that we could not check for a quasi-optimal solution since the conditions of Lemma 3.5 were not satisfied.

6.4. The hard case. The fourth experiment illustrates the behavior of the method in the hard case. The matrix A was of the form $A = UDU^T$, with $D = \text{diag}(d_1, \ldots, d_n)$ and $U = I - 2uu^T$, $u^T u = 1$. The elements of D were randomly generated with a uniform distribution on $(-5, 5)$ then sorted in nondecreasing order and d_i set to -5 for $i = 1, 2, \ldots, \ell$, allowing multiplicity ℓ for the smallest eigenvalue of A. Both vectors u and g were randomly generated with entries selected from a

TABLE 4
Verification of superlinear convergence for problems without hard case (a) *and with hard case* (b).

k	$\left\lvert \frac{\Delta - \|x_k\|}{\Delta} \right\rvert$
0	8.739485e-01
1	1.101152e+01
2	7.790406e-01
3	5.987336e-01
4	1.247129e-01
5	2.593978e-02
6	3.410990e-04
7	1.038581e-06
8	4.049703e-11
9	8.704149e-15

(a)

k	$\frac{(\lambda_i(\alpha_k) - \lambda_1(\alpha_k))\ \tau_2^2\ (1+\Delta^2)}{-2\eta\widetilde{\psi(x)}}$
0	1.694375e-01
1	*
2	4.112269e-02
3	9.276102e-03
4	6.306448e-04
5	5.851597e-06
6	4.159997e-09
7	2.116485e-09
8	7.976599e-10
9	1.267130e-12

(b)

uniform distribution on $(-0.5, 0.5)$ and then u was normalized to have unit length. The order of A was $n = 1000$.

In this case, the eigenvectors of the matrix A are of the form $q_i = e_i - 2uu_i,\ i = 1, \ldots, n$, with e_i the ith canonical vector in $\mathbb{R}^n$ and u_i the ith component of the vector u. This provides complete control in the generation of the hard case. In fact, if $\ell = 1$, the vector g was orthogonalized against q_1 computed by the formula given above. For $\ell > 1$, g was computed as the sum of the vectors in an orthonormal basis for the orthogonal complement of $\mathcal{S}_1$. After this, a noise vector s was added to g and $g \leftarrow \frac{g+s}{\|g+s\|}$. Both hard case and near hard case were generated by adding noise vectors of norms 10^{-8} and 10^{-2}, respectively. To ensure that the hard case really occurred, we computed $\Delta_{min} = \|(A - d_1 I)^{\dagger} g\|$ and set $\Delta = 2\Delta_{min}$. The problems were solved to the level $\varepsilon_{HC} = 10^{-6}$. The initial δ_U was set to -4.5 and $\alpha_0 = \min\{0, \alpha_U\}$.

The parameters for the IRLM were chosen as follows: for the hard case, 9 Lanczos basis vectors with 6 shifts on each implicit restart and a maximum of 45 restarts; for the near hard case, 18 Lanczos basis vectors with 12 shifts on each implicit restart and a maximum of 90 restarts. The different number of basis vectors is due to the fact that in the near hard case the smallest eigenvalues of the bordered matrix become more clustered and the IRLM needs more space in order to compute the desired eigenpairs. The tolerance ε_{Lan} was fixed at 10^{-2}.

In Table 5(a), (b) we summarize the average results for a sequence of 10 problems, generated with different seeds, for problems with hard case and near hard case, respectively.

6.5. Comparison with the semidefinite programming (SDP) approach. Finally, we compared LSTRS with the SDP approach of [13]. In this experiment, we solved two different families of problems. For each family, we generated 10 problems of each type (easy and hard case) with different seeds and solved them with Algorithm 4.1 (LSTRS) and the SDP approach of [13]. In all cases, the eigenproblems were solved with the function `eig` of MATLAB, so that the eigenpairs available to both methods had the same level of accuracy and also to avoid the inconsistencies associated with having two different eigensolvers. We report the average number of iterations (IT), average magnitude of the residual $\|(A - \lambda_* I)x_* + g\|/\|g\|$, and average relative accuracy in the norm of the trust-region solution, $|\Delta - \|x_*\||/\Delta$. Since we were using the function `eig` as the eigensolver, we are also reporting the average number of calls

TABLE 5
(a) *The hard case and* (b) *near hard case, when* $\mathcal{S}_1$ *has dimension* $\ell \geq 1$.

ℓ	MV	IT	$\frac{\|(A - \lambda_* I)x_* + g\|}{\|g\|}$
1	683.9	9.6	10^{-2}
5	790.4	12.1	10^{-2}
10	1301.9	22.6	10^{-2}

(a)

ℓ	MV	IT	$\frac{\|(A - \lambda_* I)x_* + g\|}{\|g\|}$
1	1153.5	10	10^{-3}
5	1039.9	9.7	10^{-3}
10	1063.5	9.7	10^{-3}

(b)

TABLE 6
Comparison with SDP approach. First set of problems, $\varepsilon_{HC} = 10^{-8}$.

			IT	SOLVES	$\frac{\|g+(A - \lambda_* I)x_*\|}{\|g\|}$	$\left\lvert \frac{\Delta - \|x_*\|}{\Delta} \right\rvert$
$A = L - 5I$	Easy case	LSTRS	5.0	5.0	10^{-13}	10^{-7}
		SDP	4.8	5.8	10^{-3}	10^{-3}
	Hard case	LSTRS	8.0	9.7	10^{-9}	10^{-16}
		SDP	9.1	10.1	10^{-7}	10^{-7}

to the eigensolver (SOLVES) to provide a means of comparing the amount of work needed by each method. It is important to point out that in large-scale applications the computational effort will concentrate on solving the eigenvalue problems, and therefore in such situations we should also compare the cost of solving each eigenvalue problem.

In the first family of problems, the matrix A was $A = L - 5I$ of order $n = 256$ and the vector g was randomly generated with entries uniformly distributed on $(0, 1)$. As in section 6.3, we orthogonalized g against the eigenvector of A corresponding to δ_1 to generate the hard case. For both easy and hard cases we added a noise vector to g, of norm 10^{-8}. The trust-region radius was $\Delta = 100$. We used $\varepsilon_\Delta = 10^{-6}$ and we ran the experiments with $\varepsilon_{HC} = 10^{-8}$ and $\varepsilon_{HC} = 10^{-6}$. We report these results in Tables 6 and 7, respectively.

In the second family of problems, A, g, and Δ_{min} were generated exactly as in section 6.4, where $A = UDU^T$ of order $n = 256$. For the easy case, $\Delta = 0.1\Delta_{min}$ and for the hard case $\Delta = 5\Delta_{min}$. The tolerances used for Algorithm 4.1 were $\varepsilon_\Delta = 10^{-6}$ and $\varepsilon_{HC} = 10^{-6}$. The results are reported in Table 8.

The previous tests indicate a marginal advantage to our algorithm in most cases. We believe this is partially due to the fact that in the SDP approach it is necessary to compute the smallest eigenvalue of A in order to begin the major iteration, while our approach avoids this extra calculation. From the comparative results, we can see that LSTRS obtained solutions with improved feasibility over the ones computed by the SDP approach. Moreover, LSTRS required slightly less computational effort overall to compute the solutions, especially in the hard case.

7. Conclusions. We have presented a new algorithm for the large-scale trust-region subproblem. The algorithm is based upon embedding the trust-region problem into a family of parameterized eigenvalue problems as developed in [17]. The main

TABLE 7
Comparison with SDP approach. First set of problems, $\varepsilon_{HC} = 10^{-6}$.

			IT	SOLVES	$\frac{\|g+(A-\lambda_* I)x_*\|}{\|g\|}$	$\left\lvert\frac{\Delta-\|x_*\|}{\Delta}\right\rvert$
$A = L - 5I$	Easy case	LSTRS	4.5	4.5	10^{-2}	10^{-8}
		SDP	4.8	5.8	10^{-3}	10^{-3}
	Hard case	LSTRS	7.0	8.7	10^{-7}	10^{-16}
		SDP	9.1	10.1	10^{-7}	10^{-7}

TABLE 8
Comparison with SDP approach. Second set of problems.

			IT	SOLVES	$\frac{\|g+(A-\lambda_* I)x_*\|}{\|g\|}$	$\left\lvert\frac{\Delta-\|x_*\|}{\Delta}\right\rvert$
$A = UDU^T$	Easy case	LSTRS	7.8	8.7	10^{-3}	10^{-14}
		SDP	4.4	5.4	10^{-4}	10^{-4}
	Hard case	LSTRS	6.4	12.5	10^{-3}	10^{-8}
		SDP	13.8	14.8	10^{-5}	10^{-5}

contribution of this paper has been to give a better understanding of the hard-case condition and to utilize this understanding to develop a better treatment of this case. As a result, we have designed a unified algorithm that naturally incorporates both the standard and hard cases.

We have proved that the iterates for this new algorithm converge either to an optimal pair for the trust-region subproblem or to a pair that can be used to construct a quasi-optimal solution. We have proved that the rate of convergence is superlinear and we have demonstrated this computationally for both the standard and hard cases. This result represents a major improvement over the performance of the method originally presented in [17]. That approach used a different iteration for the hard case that was linearly convergent. In practice this behavior seemed to occur often and greatly detracted from the performance. We have also compared our method to the SDP approach presented in [13], obtaining better results in terms of feasibility.

Our motivation for developing the LSTRS method came from some important large-scale applications. In particular, the regularization of ill-posed problems such as those arising in seismic inversion [21] provides an important class of trust-region subproblems. It was shown in [14] that near hard cases are common for this class of problems, where the vector g is nearly orthogonal to eigenspaces corresponding to *several* of the smallest eigenvalues of A. The work in [14] also reports the successful application of LSTRS to the regularization of discrete forms of ill-posed problems from inverse problems, including problems with field data.

Further work should include an analysis of the quasi-optimal solutions computed by LSTRS, the use of LSTRS within a trust-region method for the solution of large-scale optimization problems, and an analysis of such a method in light of the work in [4].

Acknowledgments. We would like to thank Trond Steihaug for insightful discussions about this material. M. Rojas in particular would like to thank Professor Steihaug for arranging an extended visit to the University of Bergen and for the support received there. Finally, we thank the two anonymous referees for their useful suggestions.

REFERENCES

[1] W. Gander, G.H. Golub, and U. von Matt, *A constrained eigenvalue problem*, in Numerical Linear Algebra, Digital Signal Processing and Parallel Algorithms, NATO Adv. Sci. Inst. Ser. F, Comput. Systems Sci. 70, G.H. Golub and P. Van Dooren, eds., Springer-Verlag, Berlin, Heidelberg, 1991, pp. 677–686.

[2] G.H. Golub, *Some modified matrix eigenvalue problems,* SIAM Rev., 15 (1973), pp. 318–334.

[3] G.H. Golub and U. von Matt, *Quadratically constrained least squares and quadratic problems,* Numer. Math., 59 (1991), pp. 561–580.

[4] N.I.M. Gould, S. Lucidi, M. Roma, and Ph. L. Toint, *Solving the trust-region subproblem using the Lanczos method,* SIAM J. Optim., 9 (1999), pp. 504–525.

[5] W.W. Hager, *Minimizing a Quadratic over a Sphere*, manuscript, 1999.

[6] M.D. Hebden, *An Algorithm for Minimization Using Exact Second Derivatives*, Technical Report T.P. 515, Atomic Energy Research Establishment, Harwell, England, 1973.

[7] R.B. Lehoucq, D.C. Sorensen, and C. Yang, *ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods*, SIAM, Philadelphia, 1998.

[8] 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.

[9] A. Melman, *Numerical solution of a secular equation,* Numer. Math., 69 (1995), pp. 483–493.

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

[11] B.N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice–Hall, Englewood Cliffs, NJ, 1980.

[12] Pham Dinh Tao and Le Thi Hoai An, *A D.C. optimization algorithm for solving the trust-region subproblem*, SIAM J. Optim., 8 (1998), pp. 476–505.

[13] F. Rendl and H. Wolkowicz, *A semidefinite framework for trust region subproblems with applications to large scale minimization*, Math. Programming, 77 (1977), pp. 273–299.

[14] M. Rojas, *A Large-Scale Trust-Region Approach to the Regularization of Discrete Ill-Posed Problems*, Ph.D. thesis, Technical Report TR98-19, Department of Computational and Applied Mathematics, Rice University, Houston, TX, May 1998.

[15] D.C. Sorensen, *Newton's method with a model trust region modification,* SIAM J. Numer. Anal., 19 (1982), pp. 409–426.

[16] D.C. Sorensen, *Implicit application of polynomial filters in a K-step Arnoldi method,* SIAM J. Matrix Anal. Appl., 13 (1992), pp. 357–385.

[17] D.C. Sorensen, *Minimization of a large-scale quadratic function subject to a spherical constraint*, SIAM J. Optim., 7 (1997), pp. 141–161.

[18] T. Steihaug, *The conjugate gradient method and trust regions in large scale optimization*, SIAM J. Numer. Anal., 20 (1983), pp. 626–637.

[19] R.J. Stern and H. Wolkowicz, *Indefinite trust region subproblems and nonsymmetric eigenvalue perturbations,* SIAM J. Optim., 5 (1995), pp. 286–313.

[20] R.J. Stern and J.J. Ye, *Variational analysis of an extended eigenvalue problem,* Linear Algebra Appl., 220 (1995), pp. 391–418.

[21] W.W. Symes, *A differential semblance criterion for inversion of multioffset seismic reflection data,* J. Geophy. Res., 98 (1993), pp. 2061–2073.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 647–674

EXPLOITING SPARSITY IN SEMIDEFINITE PROGRAMMING VIA MATRIX COMPLETION I: GENERAL FRAMEWORK*

MITUHIRO FUKUDA†, MASAKAZU KOJIMA†, KAZUO MUROTA‡, AND KAZUHIDE NAKATA§

Abstract. A critical disadvantage of primal-dual interior-point methods compared to dual interior-point methods for large scale semidefinite programs (SDPs) has been that the primal positive semidefinite matrix variable becomes fully dense in general even when all data matrices are sparse. Based on some fundamental results about positive semidefinite matrix completion, this article proposes a general method of exploiting the aggregate sparsity pattern over all data matrices to overcome this disadvantage. Our method is used in two ways. One is a conversion of a sparse SDP having a large scale positive semidefinite matrix variable into an SDP having multiple but smaller positive semidefinite matrix variables to which we can effectively apply any interior-point method for SDPs employing a standard block-diagonal matrix data structure. The other way is an incorporation of our method into primal-dual interior-point methods which we can apply directly to a given SDP. In Part II of this article, we will investigate an implementation of such a primal-dual interior-point method based on positive definite matrix completion, and report some numerical results.

Key words. semidefinite programming, primal-dual interior-point method, matrix completion problem, chordal graph

AMS subject classifications. 90C22, 90C51, 05C50

PII. S1052623400366218

1. Introduction.

Let $\mathbb{R}^n$ denote the n-dimensional Euclidean space, and $\mathcal{S}^n$ the space of $n \times n$ symmetric matrices with the Frobenius inner product $\boldsymbol{X} \bullet \boldsymbol{Y} = \sum_{i=1}^n \sum_{j=1}^n X_{ij} Y_{ij}$ for $\boldsymbol{X},\ \boldsymbol{Y} \in \mathcal{S}^n$. We will use the notation $\boldsymbol{X} \in \mathcal{S}^n_+$ and $\boldsymbol{X} \in \mathcal{S}^n_{++}$ to designate that $\boldsymbol{X} \in \mathcal{S}^n$ is positive semidefinite and positive definite, respectively. Given $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 0, 1, \ldots, m)$ and $\boldsymbol{b} \in \mathbb{R}^m$, we are concerned with the standard equality form semidefinite program (SDP)

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{A}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{A}_p \bullet \boldsymbol{X} = b_p \ (p = 1, 2, \ldots, m),\ \boldsymbol{X} \in \mathcal{S}^n_+ \end{array}\right\}, \tag{1.1}$$

and its dual

$$\left.\begin{array}{ll} \text{maximize} & \displaystyle\sum_{p=1}^m b_p z_p \\ \text{subject to} & \displaystyle\sum_{p=1}^m \boldsymbol{A}_p z_p + \boldsymbol{Y} = \boldsymbol{A}_0,\ \boldsymbol{Y} \in \mathcal{S}^n_+ \end{array}\right\}. \tag{1.2}$$

In recent years, many interior-point methods have been proposed for SDPs. Among others, primal-dual interior-point methods have been studied intensively and

*Received by the editors January 5, 2000; accepted for publication (in revised form) June 1, 2000; published electronically November 10, 2000.

http://www.siam.org/journals/siopt/11-3/36621.html

†Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-Okayama, Meguro-ku, Tokyo 152-8552, Japan (mituhiro@is.titech.ac.jp, kojima@is.titech.ac.jp). The first author was supported by Ministry of Education, Science, Sports and Culture of Japan.

‡Research Institute for Mathematical Sciences, Kyoto University, Kyoto 606-8502, Japan (murota@kurims.kyoto-u.ac.jp).

§Department of Applied Physics, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8565, Japan (nakata@zzz.t.u-tokyo.ac.jp).

extensively [1, 13, 15, 16, 20, 21, 24, 27]. They generate a sequence $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{z}^k) \in \mathcal{S}^n \times \mathcal{S}^n \times \mathbb{R}^m\}$ such that $\boldsymbol{X}^k \in \mathcal{S}^n_{++}$ and $\boldsymbol{Y}^k \in \mathcal{S}^n_{++}$. At each iteration, they first compute a search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z}) \in \mathcal{S}^n \times \mathcal{S}^n \times \mathbb{R}^m$, and then they choose a step length $\alpha^k > 0$ such that the next iterate defined by

$$(\boldsymbol{X}^{k+1}, \boldsymbol{Y}^{k+1}, \boldsymbol{z}^{k+1}) = (\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{z}^k) + \alpha^k (d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z}) \tag{1.3}$$

still satisfies $\boldsymbol{X}^{k+1} \in \mathcal{S}^n_{++}$ and $\boldsymbol{Y}^{k+1} \in \mathcal{S}^n_{++}$.

The computation of a search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$ is usually reduced to an $m \times m$ square system of linear equations $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$, which is often called the Schur complement equation. Here the coefficient matrix $\boldsymbol{B}$ (hence the search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$) varies with the individual method. See [15, 21, 27] for more details on various search directions used in primal-dual interior-point methods. The size m of the matrix $\boldsymbol{B}$ coincides with the number of equality constraints in the primal SDP (1.1) so that m can be as large as $n(n+1)/2$ even if the constraint matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m$ are assumed to be linearly independent. For a fixed n, as m becomes larger, more CPU time is spent in

(a) the computation of the coefficient matrix $\boldsymbol{B}$, and
(b) the computation of the solution $d\boldsymbol{z}$ of $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$.

See [5, 23]. Fujisawa, Kojima, and Nakata [7] proposed an efficient method for computing the coefficient matrix $\boldsymbol{B}$ when the data matrices $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 1, 2, \ldots, m)$ are sparse. Also, the computation of $\boldsymbol{B}$ can be carried out efficiently when the data matrices $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 1, 2, \ldots, m)$ are of rank 1 or 2 [2, 12].

In general, the matrix $\boldsymbol{B}$ is fully dense. Therefore, as m becomes larger, it becomes more difficult to apply direct methods such as the Cholesky factorization to the computation of the solution $d\boldsymbol{z}$ of $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$. If m is larger than 10,000, it is even impossible to store the coefficient matrix in standard workstations. [19, 23] studied the use of iterative methods such as the conjugate gradient method to overcome the storage problem for such large and dense systems of linear equations.

Another difficulty in applying primal-dual interior-point methods to large scale SDPs arises from the fact that

(c) the $n \times n$ primal positive semidefinite matrix variable $\boldsymbol{X}$ is fully dense in general even when all the data matrices $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 0, 1, \ldots, m)$ are sparse.

On the other hand, the dual positive semidefinite matrix variable $\boldsymbol{Y}$, which is computed by

$$\boldsymbol{Y} = \boldsymbol{A}_0 - \sum_{p=1}^{m} \boldsymbol{A}_p z_p,$$

inherits the sparsity of the data matrices $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 0, 1, \ldots, m)$. This difference has been a critical disadvantage of primal-dual interior-point methods compared to the dual interior-point method [2] which generates a sequence $\{(\boldsymbol{Y}^k, \boldsymbol{z}^k)\}$ only in the dual space.

The purpose of the current paper is to resolve the difficulty (c). Let V denote the set $\{1, 2, \ldots, n\}$ of row/column indices of the data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \ldots, \boldsymbol{A}_m$. For every pair of subsets S and T of V, we use the notation $\boldsymbol{X}_{ST}$ for the submatrix of $\boldsymbol{X}$ obtained by deleting all rows $i \notin S$ and all columns $j \notin T$. To outline the basic idea behind our method, we introduce the *aggregate sparsity pattern* E of the data matrices given by

$$E = \{(i, j) \in V \times V : [\boldsymbol{A}_p]_{ij} \neq 0 \text{ for some } p \in \{0, 1, 2, \ldots, m\}\}. \tag{1.4}$$

Here $[\boldsymbol{A}_p]_{ij}$ denotes the (i,j)th entry of $\boldsymbol{A}_p$. Geometrically, it is convenient to identify the aggregate sparsity pattern E with the *aggregate sparsity pattern matrix* $\boldsymbol{A}$ having unspecified nonzero numerical values in E. Since the matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \ldots, \boldsymbol{A}_m$ are all symmetric, $(i,j) \in E$ if and only if $(j,i) \in E$; hence the corresponding matrix $\boldsymbol{A}$ is symmetric. (In section 2, we will represent the aggregate sparsity pattern E in terms of a graph.)

Assume that a collection of nonempty subsets $C_1, C_2, \ldots, C_\ell$ of V satisfies the following two conditions:

(i) $E \subseteq F \equiv \bigcup_{r=1}^{\ell} C_r \times C_r$.

(ii) Any partial symmetric matrix $\boldsymbol{X}$ with entries $X_{ij} = \bar{X}_{ij} \in \mathbb{R}$ $((i,j) \in F)$ has a *positive (semi)definite matrix completion* (i.e., given any $\bar{X}_{ij} \in \mathbb{R}$ $((i,j) \in F)$, there exists a positive (semi)definite $\boldsymbol{X} \in \mathcal{S}^n$ such that $X_{ij} = \bar{X}_{ij} \in \mathbb{R}$ $((i,j) \in F)$) if and only if the submatrices $\bar{\boldsymbol{X}}_{C_r C_r}$ $(r = 1, 2, \ldots, \ell)$ are all positive (semi)definite.

From condition (i), we observe that values of the objective and constraint linear functions $\boldsymbol{A}_p \bullet \boldsymbol{X}$ $(p = 0, 1, \ldots, m)$ involved in the SDP (1.1) are completely determined by values of entries X_{ij} $((i,j) \in F)$ and independent of values of entries X_{ij} $((i,j) \notin F)$. In other words, if two $\boldsymbol{X},\ \boldsymbol{X}' \in \mathcal{S}^n$ satisfy $X_{ij} = X'_{ij}$ $((i,j) \in F)$, then

$$\boldsymbol{A}_p \bullet \boldsymbol{X} = \boldsymbol{A}_p \bullet \boldsymbol{X}' \ (p = 0, 1, \ldots, m).$$

The remaining entries X_{ij} $((i,j) \notin F)$ affect only whether $\boldsymbol{X}$ is positive (semi)definite. Now we know by condition (ii) whether we can assign some appropriate values to those remaining entries X_{ij} $((i,j) \notin F)$ so that the resulting whole matrix $\boldsymbol{X}$ becomes positive (semi)definite. Therefore, the SDP (1.1) is equivalent to

$$\left.\begin{array}{ll} \text{minimize} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_0]_{ij} X_{ij} \\ \text{subject to} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_p]_{ij} X_{ij} = b_p \ (p = 1, 2, \ldots, m), \\ & \boldsymbol{X}_{C_r C_r} \in \mathcal{S}_+^{C_r} \ (r = 1, 2, \ldots, \ell) \end{array}\right\}.$$

Here $\mathcal{S}_+^{C_r}$ denotes the set of $\sharp C_r \times \sharp C_r$ positive semidefinite symmetric matrices with entries specified in $C_r \times C_r$, and $\sharp C_r$ denotes the number of elements of C_r.

Section 2 is devoted to some fundamental results on the positive (semi)definite matrix completion problem. In particular, we present a characterization of the positive (semi)definite matrix completion in terms of chordal graphs based on the paper [11] by Grone et al. and relate it to the perfect elimination ordering for the Cholesky factorization with no fill-in. Based on the former characterization, we describe in section 3 a general method of choosing a collection of subsets $C_1, C_2, \ldots, C_\ell$ satisfying conditions (i) and (ii) above. The latter perfect elimination ordering leads us to a sparse factorization formula for the maximum-determinant positive definite matrix completion in the latter part of section 2. A variation of this formula, which we will call the sparse clique-factorization formula, plays an essential role in the primal-dual interior-point method based on positive definite matrix completion which we describe in section 5.

As an illustrative example, consider the simple case

$$E = \{(i,n),\ (n,i),\ (i,i)\ : i = 1, 2, \ldots, n\},$$

i.e., the case where each $\boldsymbol{A}_p$ has possible nonzero entries only in its nth row, its nth column, and its diagonal. Let S_r $(r = 1, 2, \ldots, \ell)$ be a partition of $\{1, 2, \ldots, n-1\}$, i.e., $\bigcup_{r=1}^{\ell} S_r = \{1, 2, \ldots, n-1\}$ and $S_r \cap S_s = \emptyset$ $(1 \le r < s \le \ell)$. Let $C_r = S_r \cup \{n\}$ $(r = 1, 2, \ldots, \ell)$ and $F = \bigcup_{r=1}^{\ell} C_r \times C_r$. Then conditions (i) and (ii) hold. (We will discuss more general cases in detail in section 3.) In this case, we obtain the problem below which is equivalent to the SDP (1.1):

$$\left.\begin{array}{ll} \text{minimize} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_0]_{ij} X_{ij} \\ \text{subject to} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_p]_{ij} X_{ij} = b_p \ (p = 1, 2, \ldots, m), \\ & \begin{pmatrix} \boldsymbol{X}_{S_r S_r} & \boldsymbol{X}_{S_r n} \\ \boldsymbol{X}_{n S_r} & X_{nn} \end{pmatrix} \in \mathcal{S}_+^{C_r} \ (r = 1, 2, \ldots, \ell) \end{array}\right\}.$$

Since the entry X_{nn} is involved commonly in all the ℓ positive semidefinite constraints above, we need to introduce additional $\ell-1$ variables U_{rr} $(r = 1, 2, \ldots, \ell-1)$ to rewrite the problem above as a standard SDP. Consequently, we obtain an SDP

$$\left.\begin{array}{ll} \text{minimize} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_0]_{ij} X_{ij} \\ \text{subject to} & \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_p]_{ij} X_{ij} = b_p \ (p = 1, 2, \ldots, m), \\ & \begin{pmatrix} \boldsymbol{X}_{S_r S_r} & \boldsymbol{X}_{S_r n} \\ \boldsymbol{X}_{n S_r} & U_{rr} \end{pmatrix} \in \mathcal{S}_+^{C_r} \ (r = 1, 2, \ldots, \ell-1), \\ & \begin{pmatrix} \boldsymbol{X}_{S_\ell S_\ell} & \boldsymbol{X}_{S_\ell n} \\ \boldsymbol{X}_{n S_\ell} & X_{nn} \end{pmatrix} \in \mathcal{S}_+^{C_\ell}, \\ & U_{rr} = X_{nn} \ (r = 1, 2, \ldots, \ell-1) \end{array}\right\}.$$

Thus we have converted the SDP (1.1) having an $n \times n$ positive semidefinite matrix variable $\boldsymbol{X}$ into an SDP having ℓ smaller size positive semidefinite matrix variables. We can use several software packages [4, 6, 28] of primal-dual interior-point methods incorporating a standard block-diagonal matrix data structure to solve this type of SDP quite efficiently.

The conversion mentioned above considerably reduces the size of the positive semidefinite matrix variables when we take a larger ℓ and smaller size S_r $(r = 1, 2, \ldots, \ell)$. Intuitively, it becomes easier to solve the resulting SDP as the size of each positive semidefinite matrix variable $\boldsymbol{X}_{C_r C_r}$ gets smaller. However, it is also necessary to take into account the increase in the number of equality constraints. For example, if we take $\ell = n-1$ and $S_r = \{r\}$ $(r = 1, 2, \ldots, n-1)$, then the conversion yields $n-2$ additional equality constraints of the form $U_{rr} = X_{nn}$ $(r = 1, 2, \ldots, n-2)$, which, in turn, causes an increase in the CPU time to solve the system of linear equations $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$. Therefore, we need to balance two factors: reduction in the sizes of positive semidefinite matrix variables and increase in the number of equality constraints. We will present more details on the conversion method in section 4. Some numerical examples are presented in section 7 which show how the balance of the two factors is important.

In section 5, we propose a primal-dual interior-point method based on positive definite matrix completion which we can directly apply to the primal-dual pair of SDPs (1.1) and (1.2) without increasing the number of equality constraints. The

method generates a sequence $\{(\boldsymbol{X}^k, \boldsymbol{Y}^k, \boldsymbol{z}^k)\}$ of interior points of the primal-dual pair of SDPs (1.1) and (1.2), but we perform all matrix computations in $\boldsymbol{X}^k$, $\boldsymbol{Y}^k$, and their inverses essentially in partial matrices with entries specified in F by fully utilizing the sparse clique-factorization formula for the maximum-determinant positive definite matrix completion given in section 2.2. This method is more promising than the conversion method given in section 4. A practical implementation of this method and its numerical experiments will be the main subjects of part II [22] of this article.

Section 6 discusses linear transformations in the primal and the dual spaces which enhance the aggregate sparsity pattern of data matrices of SDPs. In particular, we will show that an appropriate congruence transformation in the primal space makes it possible for us to apply our methods given in sections 4 and 5 to SDP relaxations of the graph equipartition problem and the maximum clique problem.

Sections 4, 5, and 6 can be read independently.

Finally, section 7 is devoted to some numerical examples on the conversion method given in section 4.

2. Theoretical background on positive semidefinite matrix completion. In this section, we review some fundamental results about the positive semidefinite matrix completion problem.

2.1. Chordal graph. Some graph-theoretic concepts needed in the subsequent discussion are introduced here. Particular emphasis is laid on chordal graphs.

We denote by $G(V, E)$ an undirected graph with the vertex set V and the edge set $E \subseteq V \times V$, where $(u, v) \in V \times V$ is identified with $(v, u) \in V \times V$. It is assumed throughout this paper that a graph has no loops, that is, $(v, v) \notin E$ for any $v \in V$. Two vertices $u, v \in V$ are said to be *adjacent* if $(u, v) \in E$. The set of the vertices adjacent to $v \in V$ is denoted by $\mathrm{Adj}(v) = \{u \in V : (u, v) \in E\}$.

A graph is called *complete* if every pair of vertices is adjacent. For a subset V' of the vertex set V of a graph $G(V, E)$, the *induced subgraph* on V' is a graph $G(V', E')$ with the vertex set V' and the edge set $E' = E \cap (V' \times V')$. A *clique* of a graph is an induced subgraph which is complete, and a clique is *maximal* if its vertices do not constitute a proper subset of another clique. In our succeeding discussions, we often call $C \subseteq V$ a clique of $G(V, E)$ whenever it induces a clique of $G(V, E)$. A vertex is called *simplicial* if its adjacent vertices induce a clique.

A graph $G(V, E)$ is said to be *chordal* if every cycle of length ≥ 4 has a chord (an edge joining two nonconsecutive vertices of the cycle). Chordal graphs have been studied extensively in many different contexts. See [3, 10, 18] for the background materials as well as the proofs of the statements given below.

The most fundamental property of a chordal graph is that it has a simplicial vertex, say v_1. Then the subgraph induced on $V \setminus \{v_1\}$ is again chordal, and therefore it has a simplicial vertex, say v_2. By repeating this, we can construct an ordering of the vertices $(v_1, v_2, \ldots, v_n)$ (where $n = |V|$) such that $\mathrm{Adj}(v_i) \cap \{v_{i+1}, v_{i+2}, \ldots, v_n\}$ induces a clique for each $i = 1, 2, \ldots, n-1$. Such an ordering of the vertices is called a *perfect elimination ordering*. The existence of a perfect elimination ordering characterizes chordality as follows.

THEOREM 2.1 (Fulkerson and Gross [8]). *A graph is chordal if and only if it has a perfect elimination ordering.*

It is known that a perfect elimination ordering of a chordal graph can be found efficiently in linear time in the number of vertices and edges of the graph [26].

Maximal cliques of a chordal graph can be enumerated easily with reference to a perfect elimination ordering $(v_1, v_2, \ldots, v_n)$. A maximal clique containing v_1 is unique, which is given by $\{v_1\} \cup \mathrm{Adj}(v_1)$, and a maximal clique not containing v_1 is a maximal clique of the subgraph induced on $\{v_2, v_3, \ldots, v_n\}$. Then it follows that the family $\{C_r \subseteq V : r = 1, 2, \ldots, \ell\}$ of (the vertex sets of) the maximal cliques is given as the maximal members of $\{v_i\} \cup (\mathrm{Adj}(v_i) \cap \{v_{i+1}, v_{i+2}, \ldots, v_n\})$ for $i = 1, 2, \ldots, n$. More specifically, we have

$$C_r = \{v_i\} \cup (\mathrm{Adj}(v_i) \cap \{v_{i+1}, v_{i+2}, \ldots, v_n\})$$

for $i = \min\{j : v_j \in C_r\}$. This shows, in particular, that the number ℓ of maximal cliques is bounded by n.

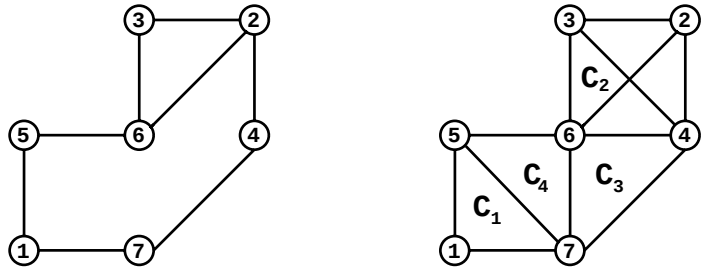
It is known that the maximal cliques can be indexed in such a way that for each $r = 1, 2, \ldots, \ell - 1$ it holds that

$$\exists s \geq r + 1 : \quad C_r \cap (C_{r+1} \cup C_{r+2} \cup \cdots \cup C_\ell) \subsetneq C_s. \tag{2.1}$$

The property (2.1) is called the *running intersection property*. An ordering of the maximal cliques satisfying the running intersection property (2.1) induces a perfect elimination ordering of the vertices. Note first that $S_1 = C_1 \backslash (C_2 \cup C_3 \cup \cdots \cup C_\ell)$ is nonempty and all the vertices in S_1 are simplicial. This means that we can start a perfect elimination ordering by numbering the vertices in S_1 with $1, 2, \ldots, |S_1|$. For each $r = 1, 2, \ldots, \ell$ in general we number the vertices in $S_r = C_r \backslash (C_{r+1} \cup \cdots \cup C_\ell)$ with $\sum_{s=1}^{r-1} |S_s| + 1, \sum_{s=1}^{r-1} |S_s| + 2, \ldots, \sum_{s=1}^{r-1} |S_s| + |S_r|$. We can thus obtain a perfect elimination ordering of the vertices, in which the vertices in S_r are given consecutive numbers for each r. Throughout this paper, we assume that $(v_1, v_2, \ldots, v_n)$ is a perfect elimination ordering induced in this way from an ordering of the maximal cliques satisfying the running intersection property (2.1).

The structure of the family of maximal cliques can be represented most conveniently in terms of a tree, called a *clique tree*, of which the vertices are maximal cliques. In particular, the ordering of the maximal cliques with the running intersection property (2.1) can be represented by an orientation (of the edges) of the clique tree to a rooted tree. The use of clique trees will be discussed in part II of this article where the implementation issues are treated.

In numerical linear algebra, chordal graphs have been studied in relation to the Gaussian elimination (Cholesky factorization) of sparse positive definite matrices. Given a positive definite matrix $\boldsymbol{X}$, we consider a graph $G(V, E)$ that represents the sparsity pattern of the matrix $\boldsymbol{X}$. Namely, V is the set of row/column indices and $E = \{(i, j) : X_{ij} \neq 0, i \neq j\}$. Let $\boldsymbol{X} = \boldsymbol{L}\boldsymbol{L}^T$ be the Cholesky factorization, where $\boldsymbol{L}$ is a lower-triangular matrix. The sparsity pattern of $\boldsymbol{L}$ can be represented similarly by a graph $G(V, F)$ defined by $F = \{(i, j) : L_{ij} \neq 0 \text{ or } L_{ji} \neq 0,\ i \neq j\}$. Under the generic assumption that no numerical cancellations occur in the elimination process, the sparsity pattern of $\boldsymbol{L}$ is determined by that of the matrix $\boldsymbol{X}$, and accordingly the graph $G(V, F)$ is determined by the graph $G(V, E)$ and the ordering of the vertices. In particular, we have $F \supseteq E$, where the added edges (belonging to $F \setminus E$) correspond to the fill-in. Moreover, the graph $G(V, F)$ is a chordal graph by Theorem 2.1. Given a graph $G(V, E)$ in general (not necessarily chordal), we say that a graph $G(V, F)$ is a *chordal extension* of $G(V, E)$ if $G(V, F)$ is chordal and $F \supseteq E$.



(a) *graph* $G(V, E)$ (b) *chordal extension* $G(V, F)$

FIG. 2.1. *Graph* $G(V, E)$ *and its chordal extension* $G(V, F)$*;* $(1, 2, \ldots, 7)$ *and* $(3, 2, 4, 6, 7, 1, 5)$ *are perfect elimination orderings for* $G(V, F)$*, and* C_1, C_2, C_3 *and* C_4 *are maximal cliques of* $G(V, F)$*.*

EXAMPLE 2.2. *The chordal extension is illustrated here. Let* $\boldsymbol{X}$ *be a* 7×7 *positive definite symmetric matrix with the nonzero pattern given by*

$$\boldsymbol{X} = \begin{pmatrix} e & & & & e & & e \\ & e & e & e & & e & \\ & e & e & & & e & \\ & e & & e & & & e \\ e & & & & e & e & \\ & e & e & & e & e & \\ e & & & e & & & e \end{pmatrix},$$

where e *denotes nonzero entries. The associated graph* $G(V, E)$ *is depicted in Figure* 2.1(a)*, where* $V = \{1, 2, \ldots, 7\}$*. The Cholesky factorization of* $\boldsymbol{X}$ *yields fill-in at* $(i, j) = (5, 7), (3, 4), (4, 6), (6, 7)$*, and the chordal extension* $G(V, F)$ *is shown in Figure* 2.1(b)*. The matrix pattern for* $G(V, F)$ *is*

$$\tilde{\boldsymbol{X}} = \begin{pmatrix} e & & & & e & & e \\ & e & e & e & & e & \\ & e & e & f & & e & \\ & e & f & e & & f & e \\ e & & & & e & e & f \\ & e & e & f & e & e & f \\ e & & & e & f & f & e \end{pmatrix},$$

where f *denotes fill-in. The natural ordering* $(1, 2, \ldots, 6, 7)$ *is a perfect elimination ordering of the chordal graph* $G(V, F)$*, whereas* $(7, 6, \ldots, 2, 1)$ *is not. The perfect elimination ordering is not unique; for instance,* $(3, 2, 4, 6, 7, 1, 5)$ *is another perfect elimination ordering. The chordal graph* $G(V, F)$ *has four maximal cliques,* $C_1 = \{1, 5, 7\}$*,* $C_2 = \{2, 3, 4, 6\}$*,* $C_3 = \{4, 6, 7\}$*,* $C_4 = \{5, 6, 7\}$*. Note that the running intersection property* (2.1) *holds with respect to this ordering of the maximal cliques.*

The fill-in in the Cholesky factorization, and hence the resulting chordal extension $G(V, F)$, depends on the ordering of the row/column indices. It is a major issue in sparse matrix computation to find a permutation matrix $\boldsymbol{P}$ (representing an ordering) such that $\boldsymbol{P}\boldsymbol{X}\boldsymbol{P}^T$ yields as little fill-in as possible. Using the graph terminology this amounts to finding a sparse chordal extension of a given graph, since any minimal chordal extension $G(V, F)$ of $G(V, E)$ can be obtained through the Cholesky factor-

ization process for some ordering. The problem of finding a permutation matrix $\boldsymbol{P}$ that results in the minimum number of fill-in, or equivalently, the problem of finding a chordal extension with the minimum number of edges, is known to be NP-complete. Several heuristic algorithms such as the minimum-degree ordering and the nested dissection have been proposed for this problem [9]. In the most favorable case, where the given graph $G(V, E)$ is chordal, the perfect elimination ordering yields the Cholesky factorization with no fill-in.

2.2. Positive semidefinite matrix completion. A *partial symmetric matrix* means a symmetric matrix in which only part of the entries are specified. More precisely, an $n \times n$ partial symmetric matrix $\bar{\boldsymbol{X}}$ is given as a collection of real numbers $(\bar{X}_{ij} = \bar{X}_{ji} : (i, j) \in F)$ for some $F \subseteq V \times V$ such that $(i, j) \in F$ if and only if $(j, i) \in F$, where $V = \{1, 2, \ldots, n\}$. A *completion* of a partial symmetric matrix $\bar{\boldsymbol{X}}$ means a symmetric matrix $\boldsymbol{X}$ (of the same size) such that $X_{ij} = \bar{X}_{ij}$ for $(i, j) \in F$. The *positive (semi)definite matrix completion problem* is to find a positive (semi)definite matrix which is a completion of a given partial symmetric matrix. See [14, 17] for surveys on matrix completion problems.

In considering this problem we may assume, without loss of generality, that the diagonal entries are all specified, i.e.,

$$F \supseteq \{(i, i) : i = 1, 2, \ldots, n\}, \tag{2.2}$$

since unspecified diagonal entries, if any, may be given sufficiently large values to realize positive (semi)definiteness. We adopt the convention (2.2) throughout this section.

We use the following notation:

- $\mathcal{S}^n(F, ?)$: the set of $n \times n$ partial symmetric matrices with entries specified in F;
- $\mathcal{S}^n_+(F, ?)$: the set of $n \times n$ partial symmetric matrices with specified entries in F which can be completed to positive semidefinite symmetric matrices; i.e., $\mathcal{S}^n_+(F, ?) = \{\bar{\boldsymbol{X}} \in \mathcal{S}^n(F, ?) : \exists \boldsymbol{X} \in \mathcal{S}^n_+, \bar{X}_{ij} = X_{ij} \text{ for } (i, j) \in F\}$;
- $\mathcal{S}^n_{++}(F, ?)$: the set of $n \times n$ partial symmetric matrices with specified entries in F which can be completed to positive definite symmetric matrices; i.e., $\mathcal{S}^n_{++}(F, ?) = \{\bar{\boldsymbol{X}} \in \mathcal{S}^n(F, ?) : \exists \boldsymbol{X} \in \mathcal{S}^n_{++}, \bar{X}_{ij} = X_{ij} \text{ for } (i, j) \in F\}$;
- $\mathcal{S}^n(F, 0)$: the set of $n \times n$ symmetric matrices with vanishing entries outside F; i.e., $\mathcal{S}^n(F, 0) = \{\boldsymbol{X} \in \mathcal{S}^n : X_{ij} = 0 \text{ if } (i, j) \notin F\}$;
- $\mathcal{S}^n_+(F, 0)$: the set of $n \times n$ positive semidefinite symmetric matrices with vanishing entries outside F; i.e., $\mathcal{S}^n_+(F, 0) = \mathcal{S}^n_+ \cap \mathcal{S}^n(F, 0) = \{\boldsymbol{X} \in \mathcal{S}^n_+ : X_{ij} = 0 \text{ if } (i, j) \notin F\}$;
- $\mathcal{S}^n_{++}(F, 0)$: the set of $n \times n$ positive definite symmetric matrices with vanishing entries outside F; i.e., $\mathcal{S}^n_{++}(F, 0) = \mathcal{S}^n_{++} \cap \mathcal{S}^n(F, 0) = \{\boldsymbol{X} \in \mathcal{S}^n_{++} : X_{ij} = 0 \text{ if } (i, j) \notin F\}$;
- $\mathcal{S}^C, \mathcal{S}^C_+, \mathcal{S}^C_{++}$: the sets of $\sharp C \times \sharp C$ symmetric matrices, positive semidefinite symmetric matrices, positive definite symmetric matrices, respectively, with rows and columns indexed by $C \subseteq V$, where $\sharp C$ means the number of elements of C.

For $E, F \subseteq V \times V$ in general, we define

$$F^\circ = F \setminus \{(i, i) : i = 1, 2, \ldots, n\}, \tag{2.3}$$

$$E^\bullet = E \cup \{(i, i) : i = 1, 2, \ldots, n\}. \tag{2.4}$$

Then, the structure F of a partial symmetric matrix can be represented by a graph $G(V,E)$ with $E = F^\circ$. Conversely, a graph $G(V,E)$ is associated with the class of partial symmetric matrices $\mathcal{S}^n(E^\bullet,?)$.

Suppose we are given a partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(F,?)$, and let $G(V,E)$ be the associated graph, where $E = F^\circ$. Denote by $\{C_r \subseteq V : r = 1,2,\ldots,\ell\}$ the family of all maximal cliques of $G(V,E)$. An obvious necessary condition for $\bar{\boldsymbol{X}}$ to have a positive semidefinite matrix completion is that each $\bar{\boldsymbol{X}}_{C_rC_r}$ is positive semidefinite, i.e.,

$$\bar{\boldsymbol{X}}_{C_rC_r} \in \mathcal{S}^{C_r}_{+} \qquad (r = 1,2,\ldots,\ell), \tag{2.5}$$

where it is noted that all the entries of the submatrix $\bar{\boldsymbol{X}}_{C_rC_r}$ are specified. Similarly, an obvious necessary condition for $\bar{\boldsymbol{X}}$ to have a positive definite matrix completion is that each $\bar{\boldsymbol{X}}_{C_rC_r}$ is positive definite, i.e.,

$$\bar{\boldsymbol{X}}_{C_rC_r} \in \mathcal{S}^{C_r}_{++} \qquad (r = 1,2,\ldots,\ell). \tag{2.6}$$

We refer to (2.5) and (2.6) as the *clique-PSD condition* and the *clique-PD condition*, respectively.

The following two theorems are most fundamental concerning the positive (semi) definite matrix completion problem.

THEOREM 2.3 (Grone et al. [11, Theorem 7]). *Let $G(V,E)$ be a graph.*

(i) *Any partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(E^\bullet,?)$ satisfying the clique-PSD condition* (2.5) *can be completed to a positive semidefinite symmetric matrix $\boldsymbol{X}$ if and only if $G(V,E)$ is chordal.*

(ii) *Any partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(E^\bullet,?)$ satisfying the clique-PD condition* (2.6) *can be completed to a positive definite symmetric matrix $\boldsymbol{X}$ if and only if $G(V,E)$ is chordal.*

THEOREM 2.4 (Grone et al. [11, Theorem 2]). *Suppose that a partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(F,?)$ has a positive definite matrix completion. Then there exists a unique positive definite matrix completion $\boldsymbol{X} = \hat{\boldsymbol{X}}$ that maximizes the determinant, i.e., such that*

$$\det(\hat{\boldsymbol{X}}) = \max\{\det(\boldsymbol{X}) : \boldsymbol{X} \text{ is a positive definite matrix completion of } \bar{\boldsymbol{X}}\}.$$

Moreover, such $\hat{\boldsymbol{X}}$ is characterized by the condition

$$[\hat{\boldsymbol{X}}^{-1}]_{ij} = 0 \quad ((i,j) \notin F), \quad \text{i.e.,} \quad \hat{\boldsymbol{X}}^{-1} \in \mathcal{S}^n(F,0).$$

We refer to the completion $\hat{\boldsymbol{X}}$ in Theorem 2.4 as the *maximum-determinant positive definite matrix completion* of $\bar{\boldsymbol{X}}$.

The sufficiency part in Theorem 2.3 can be restated in the following form convenient for our subsequent use.

THEOREM 2.5. *Let $G(V,E)$ be a chordal graph.*

(i) *A partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(E^\bullet,?)$ can be completed to a positive semidefinite symmetric matrix $\boldsymbol{X}$ if and only if it satisfies the clique-PSD condition* (2.5).

(ii) *A partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n(E^\bullet,?)$ can be completed to a positive definite symmetric matrix $\boldsymbol{X}$ if and only if it satisfies the clique-PD condition* (2.6).

In what follows we shall give a concrete expression of the maximum-determinant positive definite matrix completion in case Theorem 2.5(ii) above. This expression forms the basis of our computational scheme for sparse semidefinite programs, to be described in section 5. Also it serves as a constructive proof of the "if" part in (ii), while the "only if" part is obvious.

We start with a fundamental lemma showing an elementary construction of the maximum-determinant positive definite matrix completion.

LEMMA 2.6. *Let S and T be disjoint nonempty subsets of V and $\bar{\boldsymbol{X}}$ be a partial symmetric matrix with the entries in $(S\times T)\cup(T\times S)$ unspecified, i.e., $\bar{\boldsymbol{X}}\in\mathcal{S}^n(F,?)$ for $F=(V\times V)\setminus((S\times T)\cup(T\times S))$. Then $\bar{\boldsymbol{X}}$ admits a positive definite matrix completion if and only if the two submatrices*

$$\begin{pmatrix} \bar{\boldsymbol{X}}_{SS} & \bar{\boldsymbol{X}}_{SU} \\ \bar{\boldsymbol{X}}_{US} & \bar{\boldsymbol{X}}_{UU} \end{pmatrix} \text{ and } \begin{pmatrix} \bar{\boldsymbol{X}}_{UU} & \bar{\boldsymbol{X}}_{UT} \\ \bar{\boldsymbol{X}}_{TU} & \bar{\boldsymbol{X}}_{TT} \end{pmatrix} \tag{2.7}$$

are both positive definite, where $U=V\setminus(S\cup T)$. If this is the case, the matrix $\hat{\boldsymbol{X}}$ defined by

$$\hat{\boldsymbol{X}}=\begin{pmatrix} \bar{\boldsymbol{X}}_{SS} & \bar{\boldsymbol{X}}_{SU} & \bar{\boldsymbol{X}}_{SU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{UT} \\ \bar{\boldsymbol{X}}_{US} & \bar{\boldsymbol{X}}_{UU} & \bar{\boldsymbol{X}}_{UT} \\ \bar{\boldsymbol{X}}_{TU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{US} & \bar{\boldsymbol{X}}_{TU} & \bar{\boldsymbol{X}}_{TT} \end{pmatrix} \tag{2.8}$$

has the following properties: (i) *$\hat{\boldsymbol{X}}$ is positive definite,* (ii) *$(\hat{\boldsymbol{X}}^{-1})_{ST}=\boldsymbol{O}$,* (iii) *$\hat{\boldsymbol{X}}$ is the unique maximizer of the determinant among all positive definite matrix completions of $\bar{\boldsymbol{X}}$. Here we adopt the convention $\bar{\boldsymbol{X}}_{SU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{UT}=\boldsymbol{O}$ and $\bar{\boldsymbol{X}}_{TU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{US}=\boldsymbol{O}$ if $U=\emptyset$.*

Proof. The necessity of the positive definiteness of the two submatrices in (2.7) is obvious. For the sufficiency, we note

$$\begin{aligned} &\begin{pmatrix} \boldsymbol{I} & -\bar{\boldsymbol{X}}_{SU}\bar{\boldsymbol{X}}_{UU}^{-1} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{I} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} \end{pmatrix}\hat{\boldsymbol{X}}\begin{pmatrix} \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} \\ -\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{US} & \boldsymbol{I} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{I} \end{pmatrix} \\ &=\begin{pmatrix} \bar{\boldsymbol{X}}_{SS}-\bar{\boldsymbol{X}}_{SU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{US} & \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{O} & \bar{\boldsymbol{X}}_{UU} & \bar{\boldsymbol{X}}_{UT} \\ \boldsymbol{O} & \bar{\boldsymbol{X}}_{TU} & \bar{\boldsymbol{X}}_{TT} \end{pmatrix}, \end{aligned} \tag{2.9}$$

in which

$$\bar{\boldsymbol{X}}_{SS}-\bar{\boldsymbol{X}}_{SU}\bar{\boldsymbol{X}}_{UU}^{-1}\bar{\boldsymbol{X}}_{US}\in\mathcal{S}_{++}^{S}$$

by the positive definiteness of the first matrix in (2.7). Hence (i) follows. Let $\boldsymbol{D}$ denote the matrix on the right-hand side of (2.9). Then (ii) can be shown as

$$(\hat{\boldsymbol{X}}^{-1})_{ST}=\begin{pmatrix} \boldsymbol{I} & \boldsymbol{O} & \boldsymbol{O} \end{pmatrix}\boldsymbol{D}^{-1}\begin{pmatrix} \boldsymbol{O} \\ \boldsymbol{O} \\ \boldsymbol{I} \end{pmatrix}=\boldsymbol{O}.$$

Finally, (iii) follows from (ii) by Theorem 2.4. □

A recursive application of Lemma 2.6 in accordance with the perfect elimination ordering yields an explicit construction of the positive definite matrix completion

in Theorem 2.5(ii). For simplicity of notation, let us assume that $(1, 2, \ldots, n)$ is a perfect elimination ordering of the chordal graph $G(V, E)$. Suppose (recursively) that the $(n-1) \times (n-1)$ submatrix corresponding to $\{2, 3, \ldots, n\}$ has been completed to a positive definite matrix. Then we can apply Lemma 2.6 with $S = \{1\}$, $T = \{i : i > 1\} \setminus \mathrm{Adj}(1)$ and $U = \mathrm{Adj}(1)$ to obtain a positive definite matrix completion of the whole matrix. Note that the first matrix in (2.7) is positive definite by the assumed clique-PD condition (2.6), and the second by the recursive assumption. Let $\hat{\boldsymbol{X}}$ be the completion obtained by the recursive application of this procedure.

We shall show in Lemma 2.7 below that the matrix $\hat{\boldsymbol{X}}$ constructed above is indeed the maximum-determinant positive definite matrix completion of $\bar{\boldsymbol{X}}$, and moreover, that it admits a factorization

$$\boldsymbol{P}\hat{\boldsymbol{X}}\boldsymbol{P}^T = \boldsymbol{L}_1^T \boldsymbol{L}_2^T \cdots \boldsymbol{L}_{n-1}^T \boldsymbol{D} \boldsymbol{L}_{n-1} \cdots \boldsymbol{L}_2 \boldsymbol{L}_1 \tag{2.10}$$

with "sparse" triangular matrices $\boldsymbol{L}_k$ $(k = 1, 2, \ldots, n-1)$ and a positive definite diagonal matrix $\boldsymbol{D}$, where $\boldsymbol{P} = \boldsymbol{I}$ under our tentative assumption that $(1, 2, \ldots, n)$ is a perfect elimination ordering. We define

$$U_k = \mathrm{Adj}(k) \cap \{i : i > k\} \qquad (k = 1, 2, \ldots, n). \tag{2.11}$$

It follows from the repeated use of (2.9) that $\boldsymbol{L}_k$ is a lower-triangular matrix

$$\boldsymbol{L}_k = \begin{pmatrix} \boldsymbol{I}_{k-1} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0}^T & 1 & 0 & \cdots & 0 \\ \boldsymbol{0}^T & [\boldsymbol{L}_k]_{k+1,k} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{0}^T & [\boldsymbol{L}_k]_{nk} & 0 & \cdots & 1 \end{pmatrix} \tag{2.12}$$

with unit diagonal entries $[\boldsymbol{L}_k]_{ii} = 1$ $(i = 1, 2, \ldots, n)$ and other possible nonzero entries at $\{(i, k) : i \in U_k\}$ in the kth column; to be specific,

$$[\boldsymbol{L}_k]_{ij} = \begin{cases} 1 & (i = j), \\ [\bar{\boldsymbol{X}}_{U_k U_k}^{-1} \bar{\boldsymbol{X}}_{U_k k}]_{ik} & (i \in U_k, j = k), \\ 0 & (\text{otherwise}) \end{cases} \tag{2.13}$$

for $k = 1, 2, \ldots, n-1$. Expressions of the diagonal entries of $\boldsymbol{D}$ are also known from (2.9) as

$$D_{kk} = \begin{cases} \bar{X}_{kk} - \bar{\boldsymbol{X}}_{kU_k} \bar{\boldsymbol{X}}_{U_k U_k}^{-1} \bar{\boldsymbol{X}}_{U_k k} & (k = 1, 2, \ldots, n-1), \\ \bar{X}_{nn} & (k = n). \end{cases} \tag{2.14}$$

We have $D_{kk} > 0$ for $k = 1, 2, \ldots, n$ by the clique-PD condition (2.6). Henceforth we refer to (2.10) as the *sparse factorization formula*.

It is mentioned that the sparse factorization formula (2.10) of $\hat{\boldsymbol{X}}$ depends on the perfect elimination ordering, represented by $\boldsymbol{P}$, used in the construction, whereas $\hat{\boldsymbol{X}}$ itself is independent of it because of the uniqueness of the maximum-determinant positive definite matrix completion. Note also that the factorization (2.10) is equivalent to

$$\boldsymbol{P}\hat{\boldsymbol{X}}^{-1}\boldsymbol{P}^T = \boldsymbol{L}_1^{-1} \boldsymbol{L}_2^{-1} \cdots \boldsymbol{L}_{n-1}^{-1} \boldsymbol{D}^{-1} \boldsymbol{L}_{n-1}^{-T} \cdots \boldsymbol{L}_2^{-T} \boldsymbol{L}_1^{-T}, \tag{2.15}$$

which is the product form of the (LDL^T) Cholesky factorization of $\boldsymbol{P}\hat{\boldsymbol{X}}^{-1}\boldsymbol{P}^T$.

LEMMA 2.7. *Let $G(V,E)$ be a chordal graph and $\bar{\boldsymbol{X}} \in \mathcal{S}^n(E^\bullet,?)$ be a partial symmetric matrix satisfying the clique-PD condition* (2.6). *Let $\boldsymbol{P}$ be a permutation matrix representing a perfect elimination ordering of $G(V,E)$ in such a way that $(1,2,\ldots,n)$ is a perfect elimination ordering for $\boldsymbol{P}\bar{\boldsymbol{X}}\boldsymbol{P}^T$. Then the maximum-determinant positive definite matrix completion $\hat{\boldsymbol{X}}$ of $\bar{\boldsymbol{X}}$ can be expressed in terms of the sparse factorization formula* (2.10), *where $\boldsymbol{L}_k$ is a lower-triangular matrix given by* (2.12) *and* (2.13) *and $\boldsymbol{D}$ is a positive definite diagonal matrix given by* (2.14).

Proof. The positive definiteness of $\hat{\boldsymbol{X}}$ follows from the factorization formula (2.10) together with the positive definiteness of $\boldsymbol{D}$. For the maximum-determinant property it suffices, by Theorem 2.4, to show that $[\hat{\boldsymbol{X}}^{-1}]_{ij} = 0$ for $(i,j) \notin E^\bullet$. Referring to (2.15) we define $\boldsymbol{M} = \boldsymbol{L}_1^{-1}\boldsymbol{L}_2^{-1}\cdots\boldsymbol{L}_{n-1}^{-1}$, which is a lower-triangular matrix with unit diagonal entries. The kth column of $\boldsymbol{M}$ coincides, except for the diagonal entry, with the negative of the kth column of $\boldsymbol{L}_k$. Therefore, $\boldsymbol{M}$ has nonzero off-diagonal entries only at $(i,j) \in E$. Suppose that $[\hat{\boldsymbol{X}}^{-1}]_{ij} \neq 0$ and assume $\boldsymbol{P} = \boldsymbol{I}$ in (2.15). Then $M_{ik} \neq 0$ and $M_{jk} \neq 0$ for some $k \leq \min(i,j)$. Hence $(k,i) \in E^\bullet$ and $(k,j) \in E^\bullet$. This means $(i,j) \in E^\bullet$ because $(1,2,\ldots,n)$ is a perfect elimination ordering. □

REMARK 2.8. *Here is a minor remark on the computations of* (2.13) *and* (2.14). *For each k, the subset $\{k\}\cup U_k$ induces a clique in $G(V,E)$, and the maximal members of such cliques are exactly the maximal cliques of $G(V,E)$, which are denoted as $\{C_r \subseteq V : r = 1,2,\ldots,\ell\}$. Moreover, for each r, those subsets U_k which are contained in C_r form a nested family; define $K_r = \{k : U_k \subseteq C_r\}$. Hence, the Cholesky factorizations of $\bar{\boldsymbol{X}}_{U_kU_k}$ for all $k \in K_r$ needed in the computations in* (2.13) *and* (2.14) *are embedded in the Cholesky factorization of $\bar{\boldsymbol{X}}_{C_rC_r}$ with an appropriate ordering.*

The sparse factorization formula (2.10) can be made conceptually more transparent and practically more efficient if it is constructed with reference to an ordering of maximal cliques rather than to a perfect elimination ordering of vertices. Let $(C_1, C_2, \ldots, C_\ell)$ be an ordering of maximal cliques that enjoys the running intersection property (2.1). A similar argument based on Lemma 2.6 yields a variant of the sparse factorization formula of the form

$$\boldsymbol{P}\hat{\boldsymbol{X}}\boldsymbol{P}^T = \boldsymbol{L}_1^T\boldsymbol{L}_2^T\cdots\boldsymbol{L}_{\ell-1}^T\boldsymbol{D}\boldsymbol{L}_{\ell-1}\cdots\boldsymbol{L}_2\boldsymbol{L}_1, \tag{2.16}$$

where $\boldsymbol{L}_r$ $(r = 1,2,\ldots,\ell-1)$ are "sparse" triangular matrices and $\boldsymbol{D}$ is a positive definite block-diagonal matrix consisting of ℓ diagonal blocks. We will call (2.16) the *sparse clique-factorization formula*. The concrete expressions of $\boldsymbol{L}_r$ $(r = 1,2,\ldots,\ell-1)$ and $\boldsymbol{D}$ can be obtained as straightforward extensions of (2.12) $\sim$ (2.14). Namely, define

$$\begin{aligned} S_r &= C_r\backslash(C_{r+1}\cup C_{r+2}\cup\cdots\cup C_\ell) \quad (r = 1,2,\ldots,\ell), \\ U_r &= C_r\cap(C_{r+1}\cup C_{r+2}\cup\cdots\cup C_\ell) \quad (r = 1,2,\ldots,\ell). \end{aligned}$$

Then the factors in (2.16) are given by

$$[\boldsymbol{L}_r]_{ij} = \begin{cases} 1 & (i=j), \\ [\bar{\boldsymbol{X}}_{U_rU_r}^{-1}\bar{\boldsymbol{X}}_{U_rS_r}]_{ij} & (i\in U_r, j\in S_r), \\ 0 & (\text{otherwise}) \end{cases} \tag{2.17}$$

for $r = 1,2,\ldots,\ell-1$, and

$$\boldsymbol{D} = \begin{pmatrix} \boldsymbol{D}_{S_1S_1} & & & \\ & \boldsymbol{D}_{S_2S_2} & & \\ & & \ddots & \\ & & & \boldsymbol{D}_{S_\ell S_\ell} \end{pmatrix} \tag{2.18}$$

with

$$\boldsymbol{D}_{S_rS_r} = \begin{cases} \bar{\boldsymbol{X}}_{S_rS_r} - \bar{\boldsymbol{X}}_{S_rU_r}\bar{\boldsymbol{X}}^{-1}_{U_rU_r}\bar{\boldsymbol{X}}_{U_rS_r} & (r = 1, 2, \ldots, \ell-1), \\ \bar{\boldsymbol{X}}_{S_\ell S_\ell} & (r = \ell). \end{cases} \tag{2.19}$$

It should be remarked that we can compute all nonzero submatrices $\bar{\boldsymbol{X}}^{-1}_{U_rU_r}\bar{\boldsymbol{X}}_{U_rS_r}$ and $\boldsymbol{D}_{S_rS_r}$ above in parallel although we need an induction argument to derive the sparse clique-factorization formula (2.16).

3. Chordal extension of aggregate sparsity pattern. In this section, we apply the discussions given in the previous section to the standard equality form SDP (1.1). Let E denote the aggregate sparsity pattern of the data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m$ given in (1.4). We first choose a chordal extension $G(V, F^\circ)$ of the graph $G(V, E^\circ)$. Let $\{C_r \subseteq V : r = 1, 2, \ldots, \ell\}$ be the family of maximal cliques of the graph $G(V, F^\circ)$, where $F \supseteq E$. Then (i) the values of the objective and constraint linear functions $\boldsymbol{A}_p \bullet \boldsymbol{X}$ $(p = 0, 1, \ldots, m)$ of the SDP (1.1) are determined by X_{ij} $((i,j) \in F)$ regardless of X_{ij} $((i,j) \notin F)$, and (ii) any $\boldsymbol{X} \in \mathcal{S}^n(F, ?)$ has a positive semidefinite (or positive definite, respectively) matrix completion if and only if the submatrices $\boldsymbol{X}_{C_rC_r}$ $(r = 1, 2, \ldots, \ell)$ are positive semidefinite (or positive definite, respectively)—the clique-PSD condition (2.5) (or the clique-PD condition (2.6), respectively). Therefore we can replace the constraint and the objective function $\boldsymbol{A}_0 \bullet \boldsymbol{X}$ of the SDP (1.1) by the constraint

$$\sum_{(i,j)\in F} [\boldsymbol{A}_p]_{ij} X_{ij} = b_p \ (p = 1, 2, \ldots, m) \ \text{ and } \ \boldsymbol{X}_{C_rC_r} \in \mathcal{S}^{C_r}_+ \ (r = 1, 2, \ldots, \ell) \tag{3.1}$$

and the objective function $\sum_{(i,j)\in F}[\boldsymbol{A}_0]_{ij}X_{ij}$, respectively. More precisely, if $\boldsymbol{X} = \bar{\boldsymbol{X}} \in \mathcal{S}^n$ satisfies the constraint of (1.1), then the partial symmetric matrix $\boldsymbol{X}' \in \mathcal{S}^n(F, ?)$ with entries $X'_{ij} = \bar{X}_{ij}$ $((i,j) \in F)$ satisfies (3.1) and their objective values $\boldsymbol{A}_0 \bullet \boldsymbol{X}$ and $\sum_{(i,j)\in F}[\boldsymbol{A}_0]_{ij}X'_{ij}$ coincide with each other. Conversely, any partial symmetric matrix $\boldsymbol{X}' \in \mathcal{S}^n(F, ?)$ satisfying (3.1) has a positive semidefinite matrix completion $\boldsymbol{X} \in \mathcal{S}^n$ that satisfies the constraint of (1.1) and has the same objective value as $\boldsymbol{X}' \in \mathcal{S}^n_+(F, ?)$.

We will propose two methods with the use of (3.1) for solving the SDP (1.1). The first one is a conversion of the SDP (1.1) having a single matrix variable $\boldsymbol{X} \in \mathcal{S}^n_+$ into an SDP having ℓ matrix variables in $\mathcal{S}^{C_r}_+$ $(r = 1, 2, \ldots, \ell)$ in section 4. The other is a primal-dual interior-point method based on positive definite matrix completion in section 5. Roughly speaking, matrix operations such as finding the Cholesky factorization of $\boldsymbol{X}$, the minimum eigenvalue of $\boldsymbol{X}$, and matrix-matrix multiplications, are replaced by the corresponding matrix operations on smaller matrices in $\mathcal{S}^{C_r}$ $(r = 1, 2, \ldots, \ell)$ in both methods. There are also overheads depending on the maximal cliques C_r $(r = 1, 2, \ldots, \ell)$. In particular, the number of additional equality constraints required in the former method is determined by the intersections of two distinct maximal cliques C_r and C_s $(r < s)$, while the amount of arithmetic operations to compute the search direction in the latter method depends not only on the maximal cliques C_r $(r = 1, 2, \ldots, \ell)$, but also on the number m of equality constraints and the sparsity pattern of data matrices $\boldsymbol{A}_p$ $(p = 0, 1, 2, \ldots, m)$. The effectiveness of both methods relies entirely on a suitable choice of a chordal extension $G(V, F^\circ)$ of the graph $G(V, E^\circ)$. (Through simple numerical examples in section 7, we will see how crucial a better choice of a chordal extension is to the conversion method.) It seems quite

difficult, however, to determine (or even define) an "optimal" chordal extension that would minimize the amount of computational work in each method because various consequences of the use of (3.1), including those mentioned above, are too complicated to be evaluated accurately. In addition, even if we could set up an appropriate objective function to be minimized over the chordal extensions of the graph $G(V, E^\circ)$, such a minimization problem would be a very difficult combinatorial optimization problem.

As we have seen in the previous section, the chordal extension is closely related to the Cholesky factorization. Specifically, the chordal extension that minimizes the total number of edges in $G(V, F^\circ)$ is obtained via the Cholesky factorization of the aggregate sparsity pattern matrix $\boldsymbol{A}$ with the minimum fill-in. Therefore it seems reasonable (or at least attractive) in practice to employ various existing heuristic methods, such as the minimum-degree ordering for less fill-in, the (nested) dissection ordering for less fill-in, and the reverse Cuthill–McKee ordering for reducing bandwidth, developed for the Cholesky factorization [9]. We briefly illustrate below how we construct a chordal extension $G(V, F^\circ)$ of the graph $G(V, E^\circ)$ using some of those existing methods.

Suppose that we have reordered the row/column indices symmetrically by applying a dissection ordering so that the resulting aggregate sparsity pattern matrix $\boldsymbol{A}$ has the following bordered block-diagonal form:

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_{S_1S_1} & \boldsymbol{O} & \cdots & \boldsymbol{O} & \boldsymbol{A}_{S_1S_0} \\ \boldsymbol{O} & \boldsymbol{A}_{S_2S_2} & \cdots & \boldsymbol{O} & \boldsymbol{A}_{S_2S_0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{A}_{S_\ell S_\ell} & \boldsymbol{A}_{S_\ell S_0} \\ \boldsymbol{A}_{S_0S_1} & \boldsymbol{A}_{S_0S_2} & \cdots & \boldsymbol{A}_{S_0S_\ell} & \boldsymbol{A}_{S_0S_0} \end{pmatrix}$$

and

$$E \subseteq \left(\bigcup_{r=1}^{\ell} S_r \times S_r\right) \bigcup \left(\bigcup_{r=0}^{\ell} S_r \times S_0\right) \bigcup \left(S_0 \times \bigcup_{r=0}^{\ell} S_r\right).$$

Let

$$C_r = S_0 \cup S_r \ (r = 1, 2, \ldots, \ell) \ \text{ and } \ F = \bigcup_{r=1}^{\ell} C_r \times C_r. \tag{3.2}$$

Obviously $E \subseteq F$. We also see that $G(V, F^\circ)$ is a chordal extension of $G(V, E^\circ)$ and that $\{C_r \subseteq V : r = 1, 2, \ldots, \ell\}$ forms the family of maximal cliques of $G(V, F^\circ)$. Furthermore, $(1, 2, \ldots, n)$ is a perfect elimination ordering, and the running intersection property (2.1) holds for any $s \geq r + 1$.

Another chordal extension can be obtained through the reordering of row/column indices by the reverse Cuthill–McKee ordering that yields the aggregate sparsity pattern matrix $\boldsymbol{A}$ having a small bandwidth:

$$A_{ij} = 0 \ \text{ if } |j - i| > \beta \quad \text{and} \quad E = \{(i, j) \in V \times V : |i - j| \leq \beta\},$$

where β is a small positive integer. In this case, we can take a collection of subsets $C_1, C_2, \ldots, C_\ell$ and $F \supseteq E$ such that

$$C_r = \{i \in V : (r-1)\kappa < i \leq \beta + r\kappa\} \ (r = 1, 2, \ldots, \ell) \ \text{ and } \ F = \bigcup_{r=1}^{\ell} C_r \times C_r, \tag{3.3}$$

where κ denotes a positive integer and ℓ the smallest positive integer satisfying $\beta+\ell\kappa \geq n$. Then $G(V, F^\circ)$ is a chordal extension of $G(V, E^\circ)$ and $\{C_r \subseteq V : r = 1, 2, \ldots, \ell\}$ forms the family of maximal cliques of $G(V, F^\circ)$. In this case, $(1, 2, \ldots, n)$ is a perfect elimination ordering, and the running intersection property (2.1) holds for $s = r + 1$.

It is not difficult to extend the discussions above to more sophisticated cases where the aggregate sparsity pattern matrix $\boldsymbol{A}$ forms a nested bordered block-diagonal matrix or a bordered band matrix. In our succeeding paper [22], we will discuss in more detail how we choose a chordal extension of $G(V, E^\circ)$ in general.

In the remainder of this paper, we assume that

- an appropriate chordal extension $G(V, F^\circ)$ of $G(V, E^\circ)$ and the family $\{C_r \subseteq V : r = 1, 2, \ldots, \ell\}$ of maximal cliques of $G(V, F^\circ)$ are available to us, and
- $(v_1, v_2, \ldots, v_n)$ is a perfect elimination ordering induced from an ordering of the maximal cliques satisfying the running intersection property (2.1).

Hence, in view of the discussions in the previous section, we can factorize the maximum-determinant positive definite matrix completion $\hat{\boldsymbol{X}}$ of each $\bar{\boldsymbol{X}} \in \mathcal{S}^n(F; ?)$ as in the sparse factorization formula (2.10) (and also as in the sparse clique-factorization formula (2.16)), and any $\boldsymbol{Y} \in \mathcal{S}^n_{++}(F; 0)$ is factorized as $\boldsymbol{Y} = \boldsymbol{R}\boldsymbol{R}^T$ for some $n \times n$ lower-triangular matrix $\boldsymbol{R}$ without any fill-in. We also know that the number ℓ of maximal cliques of $G(V, F^\circ)$ does not exceed n.

REMARK 3.1. *We also assume in the remainder of the paper that* $(i, i) \in E$ $(i = 1, 2, \ldots, n)$. *Assume, to the contrary, that some* $(i, i) \notin E$, *for example,*

$$(i, i) \notin E \ (i = 1, 2, \ldots, k) \quad \textit{and} \quad (j, j) \in E \ (j = k+1, k+2, \ldots, n).$$

Then we can rewrite the SDP (1.1) *as*

$$\left.\begin{array}{ll} \textit{minimize} & \boldsymbol{A}'_0 \bullet \boldsymbol{X}' + 2\displaystyle\sum_{i=1}^{k}\sum_{j=i+1}^{n} [\boldsymbol{A}_0]_{ij} X_{ij}, \\ \textit{subject to} & \boldsymbol{A}'_p \bullet \boldsymbol{X}' + 2\displaystyle\sum_{i=1}^{k}\sum_{j=i+1}^{n} [\boldsymbol{A}_p]_{ij} X_{ij} = b_p \ (p = 1, 2, \ldots, m), \\ & X_{ij} \in \mathbb{R} \ (i = 1, 2, \ldots, k, \ i < j \leq n), \\ & \boldsymbol{X}' = \begin{pmatrix} X_{k+1,k+1} & X_{k+1,k+2} & \cdots & X_{k+1,n} \\ X_{k+2,k+1} & X_{k+2,k+2} & \cdots & X_{k+2,n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n,k+1} & X_{n,k+2} & \cdots & X_{nn} \end{pmatrix} \in \mathcal{S}^U_+ \end{array}\right\},$$

where

$$\boldsymbol{A}'_p = \begin{pmatrix} [\boldsymbol{A}_p]_{k+1,k+1} & [\boldsymbol{A}_p]_{k+1,k+2} & \cdots & [\boldsymbol{A}_p]_{k+1,n} \\ [\boldsymbol{A}_p]_{k+2,k+1} & [\boldsymbol{A}_p]_{k+2,k+2} & \cdots & [\boldsymbol{A}_p]_{k+2,n} \\ \vdots & \vdots & \ddots & \vdots \\ [\boldsymbol{A}_p]_{n,k+1} & [\boldsymbol{A}_p]_{n,k+2} & \cdots & [\boldsymbol{A}_p]_{nn} \end{pmatrix} \in \mathcal{S}^U \ (p = 0, 1, \ldots, m),$$

and $U = \{k+1, k+2, \ldots, n\}$. *In the transformed problem above, none of* $X_{ij} \in \mathbb{R}$ $(i = 1, 2, \ldots, k, \ i < j \leq n)$ *are involved in the positive semidefinite constraint* $\boldsymbol{X}' \in \mathcal{S}^U_+$, *and therefore they are free variables. We can easily adapt the methods described in*

sections 4 *and* 5 *for the SDP* (1.1) *satisfying the assumption* $(i,i) \in E$ $(i = 1, 2, \ldots, n)$ *to the transformed problem.*

4. Conversion to an SDP having multiple but smaller size positive semidefinite matrix variables. In the previous section, we have shown that the SDP (1.1) is equivalent to the problem of minimizing the objective function $\sum_{(i,j)\in F}[\boldsymbol{A}_0]_{ij}X_{ij}$ over the constraint (3.1). This problem involves less variables and smaller size positive semidefinite constraints than the original SDP (1.1). This feature certainly makes the conversion attractive in practice because such a problem is expected to be solved more easily. It should be noted, however, that two distinct positive semidefinite constraints $\boldsymbol{X}_{C_rC_r} \in \mathcal{S}_+^{C_r}$ and $\boldsymbol{X}_{C_sC_s} \in \mathcal{S}_+^{C_s}$ in (3.1) share variables X_{ij} $((i,j) \in (C_r \cap C_s) \times (C_r \cap C_s))$ whenever $C_r \cap C_s \neq \emptyset$. Hence, the problem is not a standard SDP. In this section, we show how to convert the problem to a standard SDP to which we can apply interior-point methods, and we discuss some advantages and disadvantages of the resulting SDP.

For every $r = 1, 2, \ldots, \ell$, let

$$E_r = \{(i,j) \in C_r \times C_r : (i,j) \in C_s \times C_s \ \text{ for some } s < r \ \}.$$

By definition, $E_1 = \emptyset$, and if $(i,j) \in E_r$ then the positive semidefinite constraint $\boldsymbol{X}_{C_rC_r} \in \mathcal{S}_+^{C_r}$ shares variables X_{ij} $((i,j) \in E_r)$ with the positive semidefinite constraint $\boldsymbol{X}_{C_sC_s} \in \mathcal{S}_+^{C_s}$ for some $s < r$. To make such a pair of dependent positive semidefinite constraints independent, we introduce auxiliary variables U_{ij}^r $((i,j) \in E_r,\ r = 2, 3, \ldots, \ell)$, and we rewrite the constraint (3.1) as

$$\left.\begin{array}{l} \displaystyle\sum_{(i,j)\in F} [\boldsymbol{A}_p]_{ij}X_{ij} = b_p \ (p = 1, 2, \ldots, m), \\ U_{ij}^r = X_{ij} \ ((i,j) \in E_r,\ i \geq j,\ r = 2, 3, \ldots, \ell), \\ \boldsymbol{X}^r \in \mathcal{S}_+^{C_r} \ (r = 1, 2, \ldots, \ell) \end{array}\right\}, \tag{4.1}$$

where

$$[\boldsymbol{X}^r]_{ij} = \begin{cases} U_{ij}^r & \text{if } (i,j) \in E_r, \\ X_{ij} & \text{otherwise.} \end{cases}$$

Then we may regard the minimization of the objective function $\sum_{(i,j)\in F}[\boldsymbol{A}_0]_{ij}X_{ij}$ over the constraint (4.1) as a standard SDP. In fact, if we further introduce a block-diagonal symmetric matrix variable of the form

$$\boldsymbol{X}' = \begin{pmatrix} \boldsymbol{X}^1 & \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{X}^2 & \boldsymbol{O} & \cdots & \boldsymbol{O} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \cdots & \boldsymbol{X}^\ell \end{pmatrix},$$

and if we appropriately rearrange all the coefficients of the linear equality constraints in (4.1) and the objective function $\sum_{(i,j)\in F}[\boldsymbol{A}_0]_{ij}X_{ij}$ to reconstruct data matrices with the same block-diagonal structure as $\boldsymbol{X}'$, we obtain a standard equality form SDP.

There are two major advantages of this conversion. First, when the sizes of all positive semidefinite matrix variables in (4.1) are small, their Cholesky factorizations, computation of their minimum eigenvalues, and matrix multiplications require less

CPU time than those of the original positive semidefinite matrix variable $\boldsymbol{X}$ in (1.1). Second, once we have converted the SDP (1.1) into the SDP with the block-diagonal positive semidefinite matrix variable $\boldsymbol{X}'$, we can apply effectively any interior-point method incorporating a block-diagonal matrix data structure [4, 6, 28] for SDPs.

We should note, however, that the conversion above from the SDP (1.1) to the SDP with the block-diagonal symmetric matrix variable $\boldsymbol{X}'$ increases the number of equality constraints from m to the number

$$m' = m + \sum_{r=2}^{\ell} \sharp\{(i,j) \in E_r : i \geq j\}.$$

When we apply interior-point methods to a standard form SDP having m equality constraints, we solve a system of linear equations with a fully dense $m \times m$ coefficient matrix $\boldsymbol{B}$ to generate a search direction at each iteration. This requires $\mathcal{O}(m^3)$ arithmetic operations. So the increase in the number of equality constraints in the converted problem may worsen the total computational efficiency. Therefore, the reduction in the sizes of positive semidefinite matrix variables should be properly balanced with the increase in the number of equality constraints in (4.1) when we choose a chordal extension $G(V, F^\circ)$ of $G(V, E^\circ)$. In section 7, we will show by simple numerical examples how this balance is crucial.

5. Primal-dual interior-point method based on positive definite matrix completion. One disadvantage of the conversion of the SDP (1.1) to the SDP with multiple but smaller size positive semidefinite matrix variables (4.1) is an increase in the number of equality constraints. In this section, we propose a primal-dual interior-point method based on positive semidefinite matrix completion which exploits the mechanism of positive definite completion to compute the search directions and step lengths and which does not add any equality constraints in the original SDP formulation. Various search directions [1, 13, 15, 16, 20, 21, 24, 27] have been proposed so far for primal-dual interior-point methods. Among others, we restrict ourselves to the HRVW/KSH/M search direction [13, 16, 20], although we can adapt some of the discussions below to some other search directions.

There are two places below where we effectively utilize the equivalence between the constraint on the symmetric matrix variable $\boldsymbol{X} \in \mathcal{S}^n$ of the original problem (1.1) and the constraint (3.1) on the partial symmetric matrix $\boldsymbol{X} \in \mathcal{S}^n(F, ?)$ with entries specified in F. One is the computation of a search direction and the other is the computation of a step length. Recall that E denotes the aggregate sparsity pattern of the data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \ldots, \boldsymbol{A}_m$, and that $G(V, F^\circ)$ denotes a chordal extension of $G(V, E^\circ)$.

Let $(\bar{\boldsymbol{X}}, \bar{\boldsymbol{Y}}, \bar{\boldsymbol{z}})$ be a point obtained at the kth iteration of a primal-dual interior-point method using the HRVW/KSH/M search direction ($k \geq 1$) or given initially ($k = 0$). We assume that $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$ and $\bar{\boldsymbol{Y}} \in \mathcal{S}^n_{++}(E, 0)$. Here the feasibility of the point $(\bar{\boldsymbol{X}}, \bar{\boldsymbol{Y}}, \bar{\boldsymbol{z}})$ is not assumed; $\bar{\boldsymbol{X}}$ and $(\bar{\boldsymbol{Y}}, \bar{\boldsymbol{z}})$ need not satisfy the equality constraints of the SDPs (1.1) and (1.2), respectively.

In order to compute the HRVW/KSH/M search direction, we use the whole matrix values for both $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$ and $\bar{\boldsymbol{Y}} \in \mathcal{S}^n_{++}(E, 0)$, so that we need to make a positive definite matrix completion of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$. Let $\hat{\boldsymbol{X}} \in \mathcal{S}^n_{++}$ be the maximum-determinant positive definite matrix completion of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$. See section 2.2.

Then we compute the HRVW/KSH/M search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$ by solving the system of linear equations

$$\left.\begin{array}{l} \boldsymbol{A}_p \bullet d\boldsymbol{X} = g_p \ (p = 1, 2, \ldots, m),\ d\boldsymbol{X} \in \mathcal{S}^n, \\ \sum_{p=1}^{m} \boldsymbol{A}_p dz_p + d\boldsymbol{Y} = \boldsymbol{H},\ d\boldsymbol{Y} \in \mathcal{S}^n(E, 0),\ d\boldsymbol{z} \in \mathbb{R}^m, \\ \widetilde{d\boldsymbol{X}}\bar{\boldsymbol{Y}} + \hat{\boldsymbol{X}} d\boldsymbol{Y} = \boldsymbol{K},\ d\boldsymbol{X} = (\widetilde{d\boldsymbol{X}} + \widetilde{d\boldsymbol{X}}^T)/2 \end{array}\right\}, \tag{5.1}$$

where $g_p = b_p - \boldsymbol{A}_p \bullet \hat{\boldsymbol{X}} \in \mathbb{R}$ $(p = 1, 2, \ldots, m)$ (the primal residual), $\boldsymbol{H} = \boldsymbol{A}_0 - \sum_{p=1}^{m} \boldsymbol{A}_p \bar{z}_p - \bar{\boldsymbol{Y}} \in \mathcal{S}^n(E, 0)$ (the dual residual), $\boldsymbol{K} = \mu \boldsymbol{I} - \hat{\boldsymbol{X}}\bar{\boldsymbol{Y}}$ (an $n \times n$ constant matrix), and $\widetilde{d\boldsymbol{X}}$ denotes an $n \times n$ auxiliary matrix variable. The search direction parameter μ is usually chosen to be $\beta \hat{\boldsymbol{X}} \bullet \bar{\boldsymbol{Y}}/n$ for some $\beta \in [0, 1]$. We can reduce the system of linear equations (5.1) to

$$\left.\begin{array}{l} \boldsymbol{B} d\boldsymbol{z} = \boldsymbol{s},\ d\boldsymbol{Y} = \boldsymbol{H} - \sum_{p=1}^{m} \boldsymbol{A}_p dz_p, \\ \widetilde{d\boldsymbol{X}} = (\boldsymbol{K} - \hat{\boldsymbol{X}} d\boldsymbol{Y})\bar{\boldsymbol{Y}}^{-1},\ d\boldsymbol{X} = (\widetilde{d\boldsymbol{X}} + \widetilde{d\boldsymbol{X}}^T)/2 \end{array}\right\}, \tag{5.2}$$

where

$$\left.\begin{array}{l} B_{pq} = \text{Trace } \boldsymbol{A}_p \hat{\boldsymbol{X}} \boldsymbol{A}_q \bar{\boldsymbol{Y}}^{-1} \ (p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m), \\ s_p = g_p - \text{Trace } \boldsymbol{A}_p (\boldsymbol{K} - \hat{\boldsymbol{X}} \boldsymbol{H}) \bar{\boldsymbol{Y}}^{-1} \ (p = 1, 2, \ldots, m) \end{array}\right\}.$$

Note that $\boldsymbol{B}$ is a positive definite symmetric matrix.

Now recall that the maximum-determinant positive definite matrix completion $\hat{\boldsymbol{X}}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$ is expressed in terms of the sparse clique-factorization formula (2.16). Since we have assumed that $(1, 2, \ldots, n)$ is a perfect elimination ordering of the chordal graph $G(V, F^\circ)$ as in section 2.1, we can take the identity $\boldsymbol{I}$ for the permutation matrix $\boldsymbol{P}$ in (2.16). Hence, the sparse clique-factorization formula (2.16) turns out to be

$$\hat{\boldsymbol{X}} = \boldsymbol{L}_1^T \boldsymbol{L}_2^T \ldots \boldsymbol{L}_{\ell-1}^T \boldsymbol{D} \boldsymbol{L}_{\ell-1} \ldots \boldsymbol{L}_2 \boldsymbol{L}_1, \tag{5.3}$$

where $\boldsymbol{L}_r$ $(r = 1, 2, \ldots, \ell - 1)$ and $\boldsymbol{D}$ are given by (2.17), (2.18), and (2.19). Also $\bar{\boldsymbol{Y}} \in \mathcal{S}^n_{++}(E, 0)$ is factorized as $\bar{\boldsymbol{Y}} = \boldsymbol{N}\boldsymbol{N}^T$ without any fill-in except for entries in $F \backslash E$, where $\boldsymbol{N}$ is a lower-triangular matrix. We can effectively utilize these factorizations of $\hat{\boldsymbol{X}}$ and $\bar{\boldsymbol{Y}}$ for the computation of the search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$. In particular, the coefficients B_{pq} $(p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m)$ in the system (5.2) of linear equations are computed by

$$\begin{aligned} B_{pq} = \text{Trace } & \boldsymbol{A}_p (\boldsymbol{L}_1^T \boldsymbol{L}_2^T \ldots \boldsymbol{L}_{\ell-1}^T \boldsymbol{D} \boldsymbol{L}_{\ell-1} \ldots \boldsymbol{L}_2 \boldsymbol{L}_1) \boldsymbol{A}_q (\boldsymbol{N}^{-T} \boldsymbol{N}^{-1}) \\ & (p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m). \end{aligned}$$

If we utilize those factorizations also for the computation of s_p $(p = 1, 2, \ldots, m)$ and $\widetilde{d\boldsymbol{X}}$, we do not need to store the whole dense matrix $\hat{\boldsymbol{X}}$ in the memory but only its sparse clique-factorizations in terms of $\boldsymbol{L}_1$, $\boldsymbol{L}_2, \ldots,$ $\boldsymbol{L}_{\ell-1}$ and $\boldsymbol{D}$. As we will see below in the computation of a step length and a next iterate, we need the partial symmetric matrix with entries $[d\boldsymbol{X}]_{ij}$ specified in F, but not the whole search direction matrix $d\boldsymbol{X} \in \mathcal{S}^n$ in the primal space (hence the partial symmetric matrix

with entries $[\widetilde{d\boldsymbol{X}}]_{ij}$ specified in F but not the whole matrix $\widetilde{d\boldsymbol{X}}$). Hence, it is possible to carry out all the matrix computations above using only partial matrices with entries specified in F. Therefore, we can expect to save both CPU time and memory in our computation of the search direction. To clarify the distinction between the whole primal search direction matrix $d\boldsymbol{X} \in \mathcal{S}^n$ and the corresponding partial symmetric matrix with entries specified in F in the discussions below, we use the notation $d\hat{\boldsymbol{X}}$ for the former whole matrix in $\mathcal{S}^n$ and $d\bar{\boldsymbol{X}}$ for the latter partial symmetric matrix in $\mathcal{S}^n(F;?)$. Now, suppose that we have computed the HRVW/KSH/M search direction $(d\bar{\boldsymbol{X}}, d\boldsymbol{Y}, d\boldsymbol{z}) \in \mathcal{S}^n \times \mathcal{S}^n(E,0) \times \mathbb{R}^m$. We describe how to compute a step length $\alpha > 0$ and the next iterate $(\boldsymbol{X}', \boldsymbol{Y}', \boldsymbol{z}') \in \mathcal{S}^n \times \mathcal{S}^n(E,0) \times \mathbb{R}^m$. Usually we compute the maximum $\hat{\alpha}$ of α's satisfying

$$\hat{\boldsymbol{X}} + \alpha d\hat{\boldsymbol{X}} \in \mathcal{S}^n_+ \ \text{ and } \ \bar{\boldsymbol{Y}} + \alpha d\boldsymbol{Y} \in \mathcal{S}^n_+, \tag{5.4}$$

and let $(\boldsymbol{X}', \boldsymbol{Y}', \boldsymbol{z}') = (\hat{\boldsymbol{X}}, \bar{\boldsymbol{Y}}, \bar{\boldsymbol{z}}) + \gamma\hat{\alpha}(d\hat{\boldsymbol{X}}, d\boldsymbol{Y}, d\boldsymbol{z})$ for some $\gamma \in (0,1)$. Then $\boldsymbol{X}' \in \mathcal{S}^n_{++}$ and $\boldsymbol{Y}' \in \mathcal{S}^n_{++}(E,0)$. The computation of $\hat{\alpha}$ is necessary to know how long we can take the step length along the search direction $(d\hat{\boldsymbol{X}}, d\boldsymbol{Y}, d\boldsymbol{z})$. The computation of $\hat{\alpha}$ is usually carried out by calculating the minimum eigenvalues of the matrices

$$\hat{\boldsymbol{M}}^{-1} d\hat{\boldsymbol{X}} \hat{\boldsymbol{M}}^{-T} \ \text{ and } \ \boldsymbol{N}^{-1} d\boldsymbol{Y} \boldsymbol{N}^{-T},$$

where $\hat{\boldsymbol{X}} = \hat{\boldsymbol{M}}\hat{\boldsymbol{M}}^T$ and $\bar{\boldsymbol{Y}} = \boldsymbol{N}\boldsymbol{N}^T$ denote the factorizations of $\hat{\boldsymbol{X}}$ and $\bar{\boldsymbol{Y}}$, respectively.

Instead of (5.4), we propose to employ

$$\bar{\boldsymbol{X}}_{C_rC_r} + \alpha d\bar{\boldsymbol{X}}_{C_rC_r} \in \mathcal{S}^{C_r}_+ \ (r = 1,2,\ldots,\ell) \ \text{ and } \ \bar{\boldsymbol{Y}} + \alpha d\boldsymbol{Y} \in \mathcal{S}^n_+(E,0). \tag{5.5}$$

Recall that $\{C_r \subseteq V : r = 1,2,\ldots,\ell\}$ denotes the family of maximal cliques of $G(V, F^\circ)$ and $\ell \le n$. Let $\bar{\alpha}$ be the maximum of α's satisfying (5.5), and let

$$(\boldsymbol{X}', \boldsymbol{Y}', \boldsymbol{z}') = (\bar{\boldsymbol{X}}, \bar{\boldsymbol{Y}}, \bar{\boldsymbol{z}}) + \gamma\bar{\alpha}(d\bar{\boldsymbol{X}}, d\boldsymbol{Y}, d\boldsymbol{z}) \in \mathcal{S}^n(F,?) \times \mathcal{S}^n_{++}(E,0) \times \mathbb{R}^m$$

for some $\gamma \in (0,1)$. By Theorem 2.3, $\boldsymbol{X}' \in \mathcal{S}^n(F,?)$ has a positive definite matrix completion, so that the point $(\boldsymbol{X}', \boldsymbol{Y}', \boldsymbol{z}') \in \mathcal{S}^n_{++}(F,?) \times \mathcal{S}^n_{++}(E,0) \times \mathbb{R}^m$ can be the next iterate. In this case, the computation of $\bar{\alpha}$ is reduced to the computation of the minimum eigenvalues of the matrices

$$\bar{\boldsymbol{M}}_r^{-1} d\bar{\boldsymbol{X}}_{C_rC_r} \bar{\boldsymbol{M}}_r^{-T} \ (r = 1,2,\ldots,\ell) \ \text{ and } \ \boldsymbol{N}^{-1} d\boldsymbol{Y} \boldsymbol{N}^{-T},$$

where $\bar{\boldsymbol{X}}_{C_rC_r} = \bar{\boldsymbol{M}}_r \bar{\boldsymbol{M}}_r^T$ denotes a factorization of $\bar{\boldsymbol{X}}_{C_rC_r}$ $(r = 1,2,\ldots,\ell)$. Thus the computation of the minimum eigenvalue of $\hat{\boldsymbol{M}}^{-1} \hat{\boldsymbol{X}} \hat{\boldsymbol{M}}^{-T}$ has been replaced by the computation of the minimum eigenvalues of ℓ smaller submatrices $\bar{\boldsymbol{M}}_r^{-1} d\bar{\boldsymbol{X}}_{C_rC_r} \bar{\boldsymbol{M}}_r^{-T}$ $(r = 1,2,\ldots,\ell)$.

We mention some important effects of the maximum-determinant positive definite matrix completion $\hat{\boldsymbol{X}} \in \mathcal{S}^n_{++}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F,?)$ on the theoretical and practical convergence of the primal-dual interior-point method with the modification above. We first observe that

$$\boldsymbol{X} \bullet \bar{\boldsymbol{Y}} = \hat{\boldsymbol{X}} \bullet \bar{\boldsymbol{Y}} \ \text{ and } \ \det \boldsymbol{X} \le \det \hat{\boldsymbol{X}}$$

for any positive definite matrix completion $\boldsymbol{X} \in \mathcal{S}^n_{++}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F,?)$. This implies that $\hat{\boldsymbol{X}} \in \mathcal{S}^n_{++}$ minimizes the value of the primal-dual potential function

$$\rho \log \boldsymbol{X} \bullet \bar{\boldsymbol{Y}} - \log\det(\boldsymbol{X}\bar{\boldsymbol{Y}})$$

over all positive definite matrix completions $\boldsymbol{X} \in \mathcal{S}^n_{++}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F,?)$, where ρ is a positive number. If we combine this fact with the primal-dual interior-point potential reduction method given in the paper [16] for SDPs, it is easy to design a polynomial-time primal-dual interior-point potential reduction method based on positive definite matrix completion for SDPs.

We also see that $\hat{\boldsymbol{X}}$ optimizes (maximizes) a centrality measure $\frac{(\det(\boldsymbol{X}\bar{\boldsymbol{Y}}))^{1/n}}{(\boldsymbol{X} \bullet \bar{\boldsymbol{Y}})/n}$ over all positive definite matrix completions $\boldsymbol{X} \in \mathcal{S}^n_{++}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}_{++}(F,?)$. Thus the maximum-determinant positive definite matrix completion is expected to work positively in both theoretical and practical convergence. It is not necessarily true, however, that $\hat{\boldsymbol{X}} \in \mathcal{S}^n_{++}$ optimizes (minimizes) the standard centrality measure $\|\boldsymbol{X}^{1/2}\bar{\boldsymbol{Y}}\boldsymbol{X}^{1/2} - \boldsymbol{X} \bullet \bar{\boldsymbol{Y}}/n\|$ over all positive definite matrix completions $\boldsymbol{X} \in \mathcal{S}^n_{++}$ of $\bar{\boldsymbol{X}} \in \mathcal{S}_{++}(F,?)$. Here $\|\cdot\|$ denotes the Frobenius norm of a matrix.

Another positive effect of our modification is that the maximum $\bar{\alpha}$ of α's satisfying (5.5) is larger than or equal to the maximum $\hat{\alpha}$ of α's satisfying (5.4). So we are able to choose a larger step length if we use (5.5) instead of (5.4).

6. Linear transformation in the primal and dual spaces. When we are given an SDP to be solved, we may be able to transform it into a sparser SDP to which we more effectively apply the conversion method in section 4 and/or the primal-dual interior-point method based on positive definite matrix completion in section 5. As we will see later in this section, certain semidefinite programming relaxations of some combinatorial optimization problems including the graph equipartition problem and the maximum clique problem are such cases.

We introduce a general framework for transformation of a given SDP which induces an equivalence class of SDPs. For every $\boldsymbol{\mathcal{A}} = (\boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m) \in \prod_{p=0}^m \mathcal{S}^n$ and $\boldsymbol{b} \in \mathbb{R}^m$, we use the notation $\mathrm{P}(\boldsymbol{\mathcal{A}}, \boldsymbol{b})$ for the standard equality form SDP (1.1) and the notation $\mathrm{D}(\boldsymbol{\mathcal{A}}, \boldsymbol{b})$ for its dual (1.2).

Let $\boldsymbol{P}$ be an arbitrary $n \times n$ nonsingular matrix. Performing the congruence transformation $\boldsymbol{X} = \boldsymbol{P}\boldsymbol{X}'\boldsymbol{P}^T$ from $\boldsymbol{X}$ to $\boldsymbol{X}'$ in the primal space, we obtain an SDP $\mathrm{P}(\boldsymbol{\mathcal{A}}^{\mathrm{p}}, \boldsymbol{b})$ and its dual $\mathrm{D}(\boldsymbol{\mathcal{A}}^{\mathrm{p}}, \boldsymbol{b})$, where

$$\boldsymbol{\mathcal{A}}^{\mathrm{p}} = (\boldsymbol{P}^T\boldsymbol{A}_0\boldsymbol{P}, \boldsymbol{P}^T\boldsymbol{A}_1\boldsymbol{P}, \boldsymbol{P}^T\boldsymbol{A}_2\boldsymbol{P}, \ldots, \boldsymbol{P}^T\boldsymbol{A}_m\boldsymbol{P}) \in \prod_{k=0}^m \mathcal{S}^n.$$

Let $\boldsymbol{D}$ be an $m \times m$ arbitrary nonsingular matrix and $\boldsymbol{\zeta}$ an arbitrary vector in $\mathbb{R}^m$. Performing the affine transformation

$$\boldsymbol{z} = \boldsymbol{D}\boldsymbol{z}' - \boldsymbol{\zeta}$$

from $\boldsymbol{z}$ to $\boldsymbol{z}'$ in the dual space, we obtain an SDP $\mathrm{D}(\boldsymbol{\mathcal{A}}^{\mathrm{d}}, \boldsymbol{b}^{\mathrm{d}})$ and the corresponding primal SDP $\mathrm{P}(\boldsymbol{\mathcal{A}}^{\mathrm{d}}, \boldsymbol{b}^{\mathrm{d}})$, where

$$\begin{aligned}
&\boldsymbol{b}^{\mathrm{d}} = \boldsymbol{D}^T\boldsymbol{b} \in \mathbb{R}^m,\\
&\boldsymbol{A}^{\mathrm{d}}_0 = \boldsymbol{A}_0 + \sum_{p=1}^m \boldsymbol{A}_p\zeta_p \in \mathcal{S}^n,\ \ \boldsymbol{A}^{\mathrm{d}}_k = \sum_{p=1}^m \boldsymbol{A}_p D_{pk} \in \mathcal{S}^n\ \ (k = 1, 2, \ldots, m),\\
&\boldsymbol{\mathcal{A}}^{\mathrm{d}} = (\boldsymbol{A}^{\mathrm{d}}_0, \boldsymbol{A}^{\mathrm{d}}_1, \boldsymbol{A}^{\mathrm{d}}_2, \ldots, \boldsymbol{A}^{\mathrm{d}}_m) \in \prod_{k=0}^m \mathcal{S}^n.
\end{aligned}$$

If we perform the primal transformation and the dual transformation simultaneously, we obtain another primal-dual pair of SDPs $\mathrm{P}(\boldsymbol{\mathcal{A}}^{\mathrm{pd}}, \boldsymbol{b}^{\mathrm{pd}})$ and $\mathrm{D}(\boldsymbol{\mathcal{A}}^{\mathrm{pd}}, \boldsymbol{b}^{\mathrm{pd}})$,

where

$$\boldsymbol{b}^{\rm pd} = \boldsymbol{D}^T\boldsymbol{b} \in \mathbb{R}^m,\ \boldsymbol{A}_0^{\rm pd} = \boldsymbol{P}^T\boldsymbol{A}_0\boldsymbol{P} + \sum_{p=1}^{m} \boldsymbol{P}^T\boldsymbol{A}_p\boldsymbol{P}\zeta_p \in \mathcal{S}^n,$$

$$\boldsymbol{A}_k^{\rm pd} = \sum_{p=1}^{m} \boldsymbol{P}^T\boldsymbol{A}_p\boldsymbol{P}D_{pk} \in \mathcal{S}^n\ (k = 1, 2, \ldots, m),$$

$$\boldsymbol{\mathcal{A}}^{\rm pd} = (\boldsymbol{A}_0^{\rm pd}, \boldsymbol{A}_1^{\rm pd}, \boldsymbol{A}_2^{\rm pd}, \ldots, \boldsymbol{A}_m^{\rm pd}) \in \prod_{k=0}^{m} \mathcal{S}^n.$$

By construction, all the primal-dual pairs, P($\boldsymbol{\mathcal{A}}, \boldsymbol{b}$) and D($\boldsymbol{\mathcal{A}}, \boldsymbol{b}$), P($\boldsymbol{\mathcal{A}}^{\rm p}, \boldsymbol{b}^{\rm p}$) and D($\boldsymbol{\mathcal{A}}^{\rm p}, \boldsymbol{b}^{\rm p}$), P($\boldsymbol{\mathcal{A}}^{\rm d}, \boldsymbol{b}^{\rm d}$) and D($\boldsymbol{\mathcal{A}}^{\rm d}, \boldsymbol{b}^{\rm d}$), P($\boldsymbol{\mathcal{A}}^{\rm pd}, \boldsymbol{b}^{\rm pd}$) and D($\boldsymbol{\mathcal{A}}^{\rm pd}, \boldsymbol{b}^{\rm pd}$), are equivalent to each other. The important issue here is how we choose $\boldsymbol{P}$, $\boldsymbol{D}$, and $\boldsymbol{\zeta}$ to

- improve the aggregate sparsity pattern of the data matrices, and also
- reduce the total number of nonzeros in the data matrices, which affects the computation of the coefficient matrix $\boldsymbol{B}$ of the linear equation in (5.2) to determine a search direction. See also [7].

It should be noted that any transformation using an $m \times m$ nonsingular matrix $\boldsymbol{D}$ and an m-dimensional vector $\boldsymbol{\zeta}$ in the dual space never changes the aggregate sparsity pattern of the data matrices, but it may be useful to decrease the total number of nonzeros in the data matrices, especially when some data matrices are 0-1 or integral (see also (D) of section 8 for further discussion on this transformation). Below, we will show two cases in which an appropriate congruence transformation $\boldsymbol{P}$ in the primal space improves the aggregate sparsity pattern of data matrices.

First consider a structured SDP with data matrices having the following sparsity pattern:

$$\boldsymbol{A}_0 = \begin{pmatrix} * & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ * & * & * & * \end{pmatrix},\ \boldsymbol{A}_1 = \begin{pmatrix} \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & * & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ * & * & * & * \end{pmatrix},$$

$$\boldsymbol{A}_2 = \begin{pmatrix} \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & * & * \\ * & * & * & * \end{pmatrix},\ \boldsymbol{A}_p = \begin{pmatrix} \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & * \\ * & * & * & * \end{pmatrix}\ (p = 3, 4, \ldots, m).$$

Here $*$ denotes a (possibly) nonzero matrix. In this case, the aggregate sparsity pattern matrix turns out to be a bordered block-diagonal matrix

$$\begin{pmatrix} * & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & * & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & * & * \\ * & * & * & * \end{pmatrix}.$$

Since each of the first three nonzero blocks in the diagonal is due to $\boldsymbol{A}_0$, $\boldsymbol{A}_1$, and $\boldsymbol{A}_2$, respectively, and no other data matrices $\boldsymbol{A}_p$ ($p = 3, 4, \ldots, m$) contain any nonzeros in those diagonal blocks, we can choose a nonsingular matrix $\boldsymbol{P}$ of the form

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{P}_{11} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{P}_{22} & \boldsymbol{O} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{P}_{33} & \boldsymbol{O} \\ \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{O} & \boldsymbol{P}_{44} \end{pmatrix}$$

such that the transformed data matrices $\boldsymbol{P}^T\boldsymbol{A}_p\boldsymbol{P}$ $(p = 0, 1, \ldots, m)$ get the aggregate sparsity pattern

$$\begin{pmatrix} \diamond & \boldsymbol{O} & \boldsymbol{O} & * \\ \boldsymbol{O} & \diamond & \boldsymbol{O} & * \\ \boldsymbol{O} & \boldsymbol{O} & \diamond & * \\ * & * & * & * \end{pmatrix}.$$

Here each $\diamond$ denotes a diagonal matrix. Thus, the aggregate sparsity pattern has been improved along the diagonal.

Now we consider the SDP relaxation of the graph equipartition problem, which is formulated as

$$\left.\begin{array}{ll} \text{minimize} & \boldsymbol{A}_0 \bullet \boldsymbol{X} \\ \text{subject to} & \boldsymbol{E}_p \bullet \boldsymbol{X} = \dfrac{1}{4}\ (p = 1, 2, \ldots, n), \\ & \boldsymbol{E} \bullet \boldsymbol{X} = 0,\ \boldsymbol{X} \in \mathcal{S}^n_+ \end{array}\right\}.$$

Here $\boldsymbol{A}_0 = \mathrm{diag}(\boldsymbol{C}\boldsymbol{e}) - \boldsymbol{C}$, $\boldsymbol{C}$ denotes an $n \times n$ symmetric cost matrix, $\mathrm{diag}(\boldsymbol{C}\boldsymbol{e})$ denotes the diagonal matrix whose entries are $\boldsymbol{C}\boldsymbol{e}$, $\boldsymbol{E}_p$ denotes the $n \times n$ matrix with all entries 0 except $[\boldsymbol{E}_p]_{pp} = 1$, and $\boldsymbol{E}$ denotes the $n \times n$ matrix with all entries 1. When the graph under consideration is sparse, the matrix $\boldsymbol{C}$ (hence the matrix $\boldsymbol{A}_0$) is sparse. But the aggregate sparsity pattern matrix is fully dense due to the only fully dense matrix $\boldsymbol{E}$. To improve the sparsity pattern, we perform the congruence transformation using

$$\boldsymbol{P} = \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -1 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & -1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & -1 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 1 \end{pmatrix}$$

to the data matrices $\boldsymbol{A}_0$, $\boldsymbol{E}_p$ $(p = 1, 2, \ldots, n)$ and $\boldsymbol{E}$ in the primal space to obtain

$$\boldsymbol{P}^T\boldsymbol{A}_0\boldsymbol{P}, \boldsymbol{P}^T\boldsymbol{E}_p\boldsymbol{P}\ (p = 1, 2, \ldots, n) \ \text{ and } \boldsymbol{P}^T\boldsymbol{E}\boldsymbol{P}.$$

Then all entries of $\boldsymbol{P}^T\boldsymbol{E}\boldsymbol{P}$ vanish except $[\boldsymbol{P}^T\boldsymbol{E}\boldsymbol{P}]_{11} = 1$. We can also verify that

$$\begin{aligned} &(\text{the total number of nonzeros of the matrices } \boldsymbol{P}^T\boldsymbol{A}_0\boldsymbol{P},\ \boldsymbol{P}^T\boldsymbol{E}_p\boldsymbol{P}\ (p = 1, 2, \ldots, n)) \\ &\quad \le 4 \times (\text{the total number of nonzeros of the matrices } \boldsymbol{A}_0,\ \boldsymbol{E}_p\ (p = 1, 2, \ldots, n)). \end{aligned}$$

Therefore, if $\boldsymbol{A}_0$ is sparse this transformation reduces the total number of nonzeros in data matrices and improves the aggregate sparsity pattern.

We can apply the same congruence transformation above to the SDP relaxation of the maximum clique problem.

7. Numerical examples. In this section, we give three numerical examples which show the effectiveness, advantages, and disadvantages of the conversion method described in section 4. This conversion can be interpreted as a preprocessing scheme to the existing software [4, 6, 28] which can handle standard equality form SDPs (1.1)

TABLE 7.1
Sizes of the equivalent SDPs to the 1*-bordered diagonal SDP and the tridiagonal SDP.*

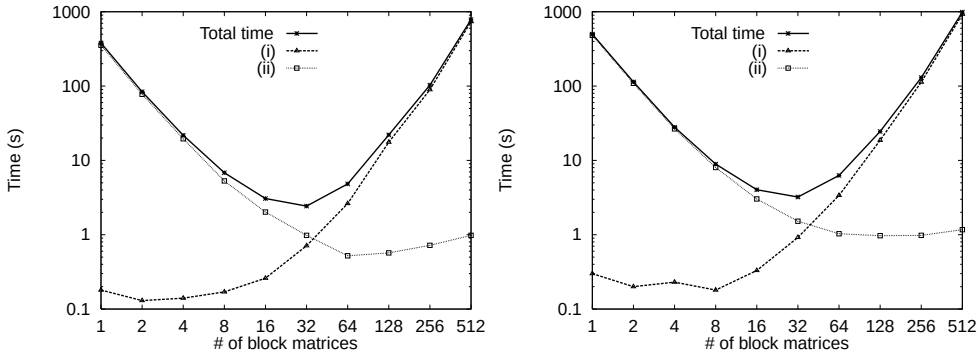
k	# block matrices (2^k)	Dimension of each block	# constraints m'
0	1	513×513	$79 + 0$
1	2	257×257	$79 + 1$
2	4	129×129	$79 + 3$
3	8	65×65	$79 + 7$
4	16	33×33	$79 + 15$
5	32	17×17	$79 + 31$
6	64	9×9	$79 + 63$
7	128	5×5	$79 + 127$
8	256	3×3	$79 + 255$
9	512	2×2	$79 + 511$

and (1.2) with block-diagonal data matrices. In particular, we use SDPA 5.0 [6] to solve the SDPs of this section on a DEC Alpha machine (300MHz with 256MB of memory). The first two examples illustrate remarkable effectiveness of the conversion method, and also the importance of determining an "optimal" chordal extension of the aggregate sparsity pattern for a given SDP. The third example exhibits a crucial disadvantage of employing this conversion compared with the primal-dual interior-point method based on positive definite matrix completion proposed in section 5.

We start by describing the first two examples which are randomly generated SDPs with high sparsity and special structures. The first problem is the example given in section 1. Let V denote the set $\{1, 2, \ldots, n\}$ of row/column indices of the data matrices $\boldsymbol{A}_p$ $(p = 0, 1, \ldots, m)$, and let $E_{\rm b} = \{(i, n), (n, i), (i, i) : i \in V\}$ be the aggregate sparsity pattern of the data matrices. We call this example the 1-*bordered diagonal SDP*. In the second example, the aggregate sparsity pattern is replaced by $E_{\rm t} = \{(i, j) \in V \times V : |i - j| \leq 1\}$ instead. We call this example the *tridiagonal SDP*. Notice that the graphs associated with the aggregate sparsity patterns, $G(V, E_{\rm b}^{\circ})$ and $G(V, E_{\rm t}^{\circ})$, are already chordal. Nevertheless, we can consider other chordal extensions which include them, namely, the graphs corresponding to the bordered block-diagonal matrix (3.2), and the graphs corresponding to (3.3), respectively. For both examples, we fixed the dimensions of the symmetric matrices $\boldsymbol{A}_p \in \mathcal{S}^n$ $(p = 0, 1, \ldots, m)$ to be equal to $n = 2^9 + 1 = 513$ and the numbers of equality constraints in the primal SDP formulation to be equal to $m = 79$. For each of the matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m$ of the 1-bordered diagonal SDP (tridiagonal SDP, respectively), we randomly generated three nonzero entries at some $(i, j) \in E_{\rm b}$ ($(i, j) \in E_{\rm t}$, respectively), and, for $\boldsymbol{A}_0$, we generated nonzero elements for all (i, j)th entries in $E_{\rm b}$ ($E_{\rm t}$, respectively).

Since a similar discussion for the 1-bordered diagonal SDP will be also valid for the tridiagonal SDP, we focus on the former example for the moment. According to the notation in sections 1, 3, and 4, for each $k \in \{0, 1, \ldots, 9\}$, let us define $S_0 = \{n\}$, and $S_r = \{1 + (r-1)2^{9-k}, 2 + (r-1)2^{9-k}, \ldots, r2^{9-k}\}$ $(r = 1, 2, \ldots, 2^k)$. Defining now $C_r = S_0 \cup S_r$ $(r = 1, 2, \ldots, 2^k)$ and $F_{\rm b} = \bigcup_{r=1}^{2^k} C_r \times C_r$, $G(V, F_{\rm b}^{\circ})$ will be a chordal extension of $G(V, E_{\rm b}^{\circ})$. Using the formula (4.1), we can convert the 1-bordered diagonal SDP to equivalent SDPs whose sizes are specified in Table 7.1. Observe that $k = 0$ gives the original SDP.

Figure 7.1(a) shows in log scale the total time (solid line) to solve the equivalent SDPs listed in Table 7.1 using SDPA. Most of the total time is spent in the following two major subroutines in SDPA:



(a) 1-*bordered diagonal SDP* (b) *tridiagonal SDP*

FIG. 7.1. *Computational time for the* 1*-bordered diagonal SDP and the tridiagonal SDP (total time;* (i) *time to compute search directions;* (ii) *time to compute step lengths).*

(i) Time to compute the search direction $(d\boldsymbol{X}, d\boldsymbol{Y}, d\boldsymbol{z})$—by calculating the coefficient matrix $\boldsymbol{B} \in \mathcal{S}^{m'}_{++}$ and solving the Schur complement equation $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$—dashed line in Figure 7.1;

(ii) time to compute the step length for the search direction—by computing the Cholesky factorization and the eigenvalues, and performing matrix operations for each small block matrix $(\mathcal{O}((\#C_r)^3))$—dotted line in Figure 7.1.

Figure 7.1(a) shows that we have to select a "good" chordal extension $G(V, F^{\circ}_{\rm b})$ in order to balance the time spent in (i), which mainly depends on the number of equality constraints m', and the time spent in (ii), which mainly depends on the dimensions of the small block matrices. This balance is crucial to reduce the total computational time to solve the SDP. For the 1-bordered diagonal SDP, a partition of the original problem into 32 small block matrices of 17×17 dimension each $(k = 5)$ gives the "optimal" conversion, and it reduces the total computational time by a factor of approximately 150.

A similar discussion can be made for the tridiagonal SDP. In this case, given $k \in \{0, 1, \ldots, 9\}$, a chordal extension of the graph associated with the aggregate sparsity pattern $G(V, E^{\circ}_{\rm t})$ is chosen such that the maximal cliques for it are given by (3.3) with $\beta = 1$ and $\kappa = 2^{9-k}$. The sizes of each equivalent SDP to the tridiagonal SDP are given in Table 7.1. The computational time is shown in Figure 7.1(b). Notice the similarity between the computational time for these two examples with extremely sparse data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \ldots, \boldsymbol{A}_m$.

We observe the following two points from these numerical examples:

(a) The problem of detecting an "optimal" chordal extension of the aggregate sparsity pattern for an SDP is extremely important in order to balance the time spent in (i) and (ii) and therefore reduce the total computational time;

(b) the conversion to multiple block matrices of smaller size (section 4) is extremely efficient when very sparse data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \ldots, \boldsymbol{A}_m$ have special sparsity patterns and the number of added constraints $(m' - m)$ in the equivalent SDP is relatively small.

The last example comes from the topology optimization problem of truss structures [25], and we call it the *topology optimization SDP* here. The aggregate sparsity pattern for the data matrices $\boldsymbol{A}_p$ $(p = 0, 1, \ldots, 392)$ after diminishing the bandwidth

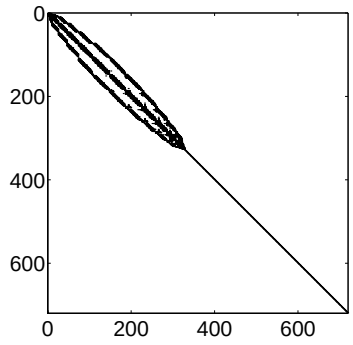


FIG. 7.2. *Aggregate sparsity pattern for the data matrices of the topology optimization SDP.*

TABLE 7.2
Sizes of the equivalent SDPs to the topology optimization SDP and the computational time to solve them.

# block matrices	Dimension of each block	# constraints m'	Total time	(i)	(ii)
1(+1)	327×327 ($+392 \times 392$)	392	237 s	69 s	164 s
2(+1)	186×186 ($+392 \times 392$)	$392 + 1035$	715 s	657 s	56 s
3(+1)	137×137 ($+392 \times 392$)	$392 + 2070$	3190 s	2550 s	36 s
4(+1)	115×115 ($+392 \times 392$)	$392 + 3105$	9032 s	8995 s	49 s

( ) indicates the block corresponding to the diagonal matrix.

by the reverse Cuthill–McKee ordering is shown in Figure 7.2. This matrix consists of two diagonal blocks: a 327×327 block matrix with a small bandwidth, and a 392×392 diagonal matrix. Since SDPA can handle the latter diagonal matrix quite efficiently, we will consider only the block matrix with the small bandwidth. We define the chordal extension of the graph associated with the sparsity pattern of this block matrix as $G(V, F_{\rm top}^{\circ})$, where $F_{\rm top} = \{(i,j) \in V \times V : |i-j| \leq 45\}$ and $V = \{1, 2, \ldots, 327\}$. The maximal cliques corresponding to this chordal extension are given in (3.3).

The sizes of the SDPs resulting from the topology optimization SDP by the conversion method and the computational time to solve them are shown in Table 7.2. The time to compute the search directions (i) grows drastically compared to the decrease in the time to compute the step lengths (ii) in this case, because we have to add $45 \cdot (45+1)/2$ new variables and equality constraints if we increase the number of block matrices by one. See (4.1). In this case, it is much better to solve the original SDP instead of converting it.

The last example shows the following fundamental drawback of the conversion method:

(c) A large number of additional equality constraints are often required in the converted SDP.

Although we might be able to utilize more sophisticated ordering such as the nested dissection ordering to decrease the number of additional equality constraints, this drawback exhibits a certain limitation of the conversion method for practical use. In section 5, we have proposed a method to compute the search directions and the step length in the primal-dual interior-point method based on positive definite matrix completion. That method does not add any equality constraints as the conversion method of section 4 does and therefore avoids the above drawback (c). In part II [22]

of this article, we will continue discussing the technical details and implementation of this method, and we present its numerical results applied to larger classes of SDPs.

8. Concluding discussion. We have proposed two kinds of methods for a large-scale sparse SDP exploiting the aggregate sparsity pattern E over its data matrices. One is a conversion of such an SDP into an SDP having multiple but smaller size positive semidefinite matrix variables. The other is a primal-dual interior-point method based on maximum-determinant positive definite matrix completion. Concerning practical implementation of these two methods, however, there remain many significant and interesting issues which we need to investigate further. Among others, we mention the following:

(A) How do we find an effective chordal extension $G(V, F^\circ)$ of $G(V, E^\circ)$? This issue is common to both methods. In part II [22] of this article, we will study more extensively how we can utilize some of the existing ordering methods, such as the minimum-degree ordering, the (nested) dissection ordering and the reverse Cuthill–McKee ordering, developed for the Cholesky factorization.

(B) The computation of the search direction, which we discussed in section 5, for the latter method is also a very important issue. In part II [22], we will explore in more detail (i) how we efficiently construct the product form representation (5.3) of the maximum-determinant positive definite matrix completion $\hat{\boldsymbol{X}}$ of a partial symmetric matrix $\bar{\boldsymbol{X}} \in \mathcal{S}^n_{++}(F, ?)$, and (ii) how we compute the coefficients B_{pq} $(p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m)$ of the key linear equation $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$ in (5.2) by utilizing the representation (5.3) effectively.

(C) Our methods still need to solve the Schur complement equation $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$. As we have mentioned in the introduction, the coefficient matrix $\boldsymbol{B}$ is fully dense, in general, so that it becomes more difficult to apply direct methods to the equation as its size (= the number of equality constraints in the primal SDP (1.1)) becomes larger. To solve a large-scale SDP having not only a large size matrix variable but also a large number of equality constraints, we can incorporate iterative methods [19, 23] to solve the Schur complement equation into our methods.

(D) The linear transformation in the primal and the dual spaces described in section 6 may be regarded as a preprocessing or preconditioning technique for SDPs. Since the transformation in the dual space does not affect the aggregate sparsity pattern of data matrices of a given SDP to be solved, without damaging the computational efficiency much, we may be able to use the transformation for numerical stability, which is another major purpose of preprocessing besides computational efficiency. In particular, if we apply the dual transformation using an $m \times m$ nonsingular matrix $\boldsymbol{D}$ and a $\boldsymbol{\zeta} \in \mathbb{R}^m$ to an SDP with data matrices $\boldsymbol{A}_0, \boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_m$, the coefficients B_{pq} $(p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m)$ of the key linear equation $\boldsymbol{B}d\boldsymbol{z} = \boldsymbol{s}$ in (5.2) turn out to be

$$B_{pq} = \text{Trace } \left(\sum_{k=1}^{m} \boldsymbol{A}_k D_{kp}\right) \hat{\boldsymbol{X}} \left(\sum_{k=1}^{m} \boldsymbol{A}_k D_{kq}\right) \bar{\boldsymbol{Y}}^{-1}$$
$$(p = 1, 2, \ldots, m,\ q = 1, 2, \ldots, m).$$

Thus the transformation may work as a preconditioning for iterative methods such as the conjugate gradient method and the conjugate residual method (see [19, 23]). It should be noted that the matrix $\boldsymbol{D}$ has enough parameters to

control the eigenvalues of the matrix $\boldsymbol{B}$, although we need to investigate an effective choice of the matrix $\boldsymbol{D}$.

Theoretically, it is an interesting issue to see whether we can design a polynomial-time and/or locally superlinearly convergent primal-dual path-following interior-point method based on the maximum-determinant positive definite matrix completion.

Acknowledgment. The authors thank Dr. Akihisa Tamura of Kyoto University for helpful comments.

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] S. J. Benson, Y. Ye, and X. Zhang, *Solving large-scale sparse semidefinite programs for combinatorial optimization*, SIAM J. Optim., 10 (2000), pp. 443–461.

[3] J. R. S. Blair and B. Peyton, *An introduction to chordal graphs and clique trees*, in Graph Theory and Sparse Matrix Computation, A. George, J. R. Gilbert, and J. W. H. Liu, eds., Springer-Verlag, New York, 1993, pp. 1–29.

[4] B. Borchers, CSDP 2.3 *user's guide*, Optim. Methods Softw., 11–12 (1999), pp. 597–611. Available at http://www.nmt.edu/~borchers/csdp.html.

[5] K. Fujisawa, M. Fukuda, M. Kojima, and K. Nakata, *Numerical evaluation of* SDPA *(semidefinite programming algorithm)*, in High Performance Optimization, H. Frenk, K. Roos, T. Terlaky, and S. Zhang, eds., Kluwer Academic Publishers, Dordrecht, 1999, pp. 267–301.

[6] K. Fujisawa, M. Kojima, and K. Nakata, *SDPA (Semidefinite Programming Algorithm) User's Manual*, Technical Report B-308, Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, Japan, December 1995 (revised August 1996). Available at ftp://ftp.is.titech.ac.jp/pub/OpRes/software/SDPA.

[7] K. Fujisawa, M. Kojima, and K. Nakata, *Exploiting sparsity in primal-dual interior-point methods for semidefinite programming*, Math. Program., 79 (1997), pp. 235–253.

[8] D. R. Fulkerson and O. A. Gross, *Incidence matrices and interval graphs*, Pacific J. Math., 15 (1965), pp. 835–855.

[9] A. George and J. W. H. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[10] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, New York, 1980.

[11] R. Grone, C. R. Johnson, E. M. Sá, and H. Wolkowicz, *Positive definite completions of partial hermitian matrices*, Linear Algebra Appl., 58 (1984), pp. 109–124.

[12] C. Helmberg and F. Rendl, *Solving quadratic* $(0,1)$*-problems by semidefinite programming and cutting planes*, Math. Program., 82 (1998), pp. 291–315.

[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] C. R. Johnson, *Matrix completion problems: A survey*, Proc. Sympos. Appl. Math., 40 (1990), pp. 171–198.

[15] M. Kojima, M. Shida, and S. Shindoh, *Search directions in the* SDP *and the monotone* SDLCP*: Generalization and inexact computation*, Math. Program., 85 (1999), pp. 51–80.

[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] M. Laurent, *Cuts, matrix completions and graph rigidity*, Math. Program., 79 (1997), pp. 255–283.

[18] J. G. Lewis, B. W. Peyton, and A. Pothen, *A fast algorithm for reordering sparse matrices for parallel factorization*, SIAM J. Sci. Stat. Comput., 10 (1989), pp. 1146–1173.

[19] C.-J. Lin and R. Saigal, *An incomplete Cholesky factorization for dense symmetric positive definite matrices*, BIT, 40 (2000), pp. 536–558.

[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 and Y. Zhang, *A unified analysis for a class of long-step primal-dual path-following interior-point algorithms for semidefinite programming*, Math. Program.,

81 (1998), pp. 281–299.

[22] K. Nakata, K. Fujisawa, M. Fukuda, M. Kojima, and K. Murota, *Exploiting sparsity in semidefinite programming via matrix completion* II: *Implementation and numerical results*, in preparation.

[23] K. Nakata, K. Fujisawa, and M. Kojima, *Using the conjugate gradient method in interior-point methods for semidefinite programs*, in Proc. Inst. Statist. Math., 46 (1998), pp. 297–316 (in Japanese).

[24] Yu. E. Nesterov and M. J. Todd, *Primal-dual interior-point methods for self-scaled cones*, SIAM J. Optim., 8 (1998), pp. 324–364.

[25] M. Ohsaki, K. Fujisawa, N. Katoh, and Y. Kanno, *Semi-definite programming for topology optimization of trusses under multiple eigenvalue constraints*, Comput. Methods Appl. Mech. Engrg., 180 (1999), pp. 203–217.

[26] D. J. Rose, R. E. Tarjan, and G. S. Lueker, *Algorithmic aspects of vertex elimination on graphs*, SIAM J. Comput., 5 (1976), pp. 266–283.

[27] 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.

[28] K.C. Toh, M. J. Todd, and R. H. Tütüncü, SDPT3—*a* MATLAB *software package for semidefinite programming, Version* 1.3, Optim. Methods Softw., 11–12 (1999), pp. 545–581. Available at http://www.math.nus.edu.sg/~mattohkc.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 675–690

EXISTENCE THEOREMS FOR GENERALIZED NONCOERCIVE EQUILIBRIUM PROBLEMS: THE QUASI-CONVEX CASE*

FABIÁN FLORES-BAZÁN†

Abstract. We provide a characterization of the nonemptiness of the solution set for generalized noncoercive equilibrium problems (an extension of generalized quasi-variational inequalities) defined in reflexive Banach spaces in the quasi-convex case. In addition, several necessary and sufficient conditions for the set of solutions to these problems to be nonempty and bounded are also given. Our approach is based on recession notions which proved to be very useful in the study of noncoercive minimization problems. In fact, we find some particular cones as estimates for the recession cone of the solution set. These cones (for the ones containing the latter set) are proved to be sharp enough to encompass several special situations found in the literature.

Key words. generalized quasi-variational inequalities, complementarity problems, equilibrium problems, nonconvex optimization, recession functions and cones, convex analysis

AMS subject classifications. 49J40, 49J45, 49J52, 90C26, 90C30, 90C31, 90C33, 90C34, 90C48, 90C99

PII. S1052623499364134

1. Introduction and formulation of the problem.

Several problems arising in optimization, such as fixed-point problems, problems of (Nash) economic equilibrium, complementarity problems, and quasi-variational inequalities, with the latter modeling various situations in mechanics, for instance, have the same mathematical formulation, which may be stated as follows: Given a closed convex set K and real-valued functions $f : K \times K \to \mathbb{R}$, $\varphi : K \times X \to \mathbb{R} \cup \{+\infty\}$,

$$\text{find } \bar{x} \in K \text{ such that } f(\bar{x}, y) + \varphi(\bar{x}, y) \geq \varphi(\bar{x}, \bar{x}) \quad \forall\, y \in K. \tag{1.1}$$

The appearance of the function φ is in order to include some classes of generalized quasi-variational inequalities, and since our interest is also to include (quasi) variational inequalities set in the context of the calculus of variations (see [BNS, JM, GT, BBGT, BO, AGT, A1]), we consider the case when K is a subset of a reflexive Banach space X. Thus, X will be endowed with its weak topology.

A model case we have in mind is $f(x, y) = \langle F(x) - x^*, y - x\rangle$, $\varphi(x, y) = i_{Q(x)}(y)$. Here, i_C denotes the indicator function of the set C; i.e., $i_C(y) = 0$ if $y \in C$ and $i_C(y) = +\infty$ otherwise, and F (resp., Q) is a single-valued (resp., set-valued) map from K into its topological dual X^* (resp., the subsets of X). From this model, one can realize, on one hand, the close relation to minimization problems in the context of the calculus of variations [BNS, JM, GT, BO] or mathematical programming [HP, AC, C] and, on the other hand, which basic assumptions on f and φ arise. We will impose the following two assumptions on f according to whether $\varphi \equiv 0$:

(f_1) For every $x \in K$, every $y \in K$, $f(x, y) \geq 0$ implies $f(y, x) \leq 0$ and

(f_1') for every $x \in K$, every $y \in K$, $f(x, y) + f(y, x) \leq 0$.

Problems like (1.1) have been considered in [JM] as a generalization to those studied in [BNS], where only the case $\varphi \equiv 0$ is discussed. The particular case $\varphi(x, x) =$

*Received by the editors November 15, 1999; accepted for publication (in revised form) June 27, 2000; published electronically November 10, 2000.

http://www.siam.org/journals/siopt/11-3/36413.html

†Departamento de Ingeniería Matemática, Universidad de Concepción, Casilla 160-C, Concepción, Chile (fflores@ing-mat.udec.cl). The work of this author is based on research material supported in part by CONICYT-Chile through FONDECYT 199-0348 and FONDAP-Matemáticas Aplicadas.

0 is treated in [BO] and in subsequent papers (see, for instance, [HS] and references therein). The case when K is bounded was first studied in [F2] under a stronger assumption on f with $\varphi \equiv 0$. In [BO] the authors use the term *equilibrium* problems.

In all the above-mentioned papers the coerciveness condition is avoided. This is done by assuming the existence of a bounded set such that no element outside this set can be a candidate for a solution. Therefore, in this case, the solution set will be bounded [BNS, JM, K2, HP, HS]. A variant of this idea is used in Corollary 3.1 in [HP] and also in [BO], allowing the solution set to be unbounded.

In this paper the noncoerciveness condition will be treated by using the notion of recession functions and cones. This technique was also employed in [BBGT, A1] for minimization problems and in [GT, AGT] for variational inequalities. We point out that the case when $\varphi \equiv 0$ has also been studied, in our framework, in [CCR] under the convexity assumption on $f(x, \cdot)$, but its result yields existence of solutions, as in [BNS, JM, HS], whenever the solution set is bounded. Our results apply also to situations in which the solution set may be unbounded.

As in the convex minimization problem, one is led to study the behavior of the convex function to be minimized along the directions on which its recession function is nonpositive. These directions are obtained as weak limits of sequences $(\frac{x_n}{||x_n||})$ with (x_n) being any unbounded minimizing sequence, as expected. This consideration has been taken into account in Theorem 9.2 in [R], Proposition 4.4 in [AC], partially in [GT, BBGT], and slightly improved in [A1], where it is assumed that the recession function is nonnegative, but, as we will see in sections 3 and 4 (see also [FS] for the convex case) it will not be the case. On the contrary, we will devote our study to the set of directions where the recession function is nonpositive. This is motivated by the observation, in case K is convex and the function h to be minimized on K is also convex, that $S^\infty = R$ provided $S \neq \emptyset$, where S is the set of minimizers of h and $R = \{v \in K^\infty : \ h^\infty(v) \leq 0\}$. Clearly, this set is a closed, convex cone. Thus, in case $R \neq \{0\}$, some additional conditions on R must be imposed to guarantee the existence of solutions. Now, the following question in the nonconvex case, at least when h is quasi-convex, arises: Is R the right set to have the equality $R = S^\infty$? The answer is certainly no, as the example $h(x) = \sqrt{x}$, $x \geq 0$, shows. Notice that, in this case, h^∞ is still defined (see [BBGT] or the next section). Thus, our first task will be to find an alternative set for R in a minimization problem or, more generally, for generalized quasi-variational inequalities or generalized equilibrium problems. Once this is done, we shall impose assumptions on R yielding existence of solutions to problem (1.1).

Our abstract existence theorems extend the results established in [FS] to the quasi-convex case. If K is bounded or the usual coerciveness assumption (3.4) or (3.5), to be stated presently, is satisfied, our theorems recover well-known classical existence results for nonlinear variational inequalities (see [KS]), and these results are extended to generalized quasi-variational inequalities or equilibrium problems.

The purpose of this paper is to provide necessary and sufficient conditions for the solution set to be nonempty. If, in addition, the set of solutions is required to be bounded, the necessary and sufficient conditions become more precise. In this case, we extend a result due to Crouzeix in [C] and that established in [DH].

To gain insight into the mathematical difficulties and to obtain the most information, we start by studying the case $\varphi(x, y) = h(y)$ in section 3, once we introduce some basic definitions in section 2. The general case as stated in (1.1) is discussed in section 4.

Some algorithms for the classical variational inequality $\langle F(x), y - x\rangle \geq 0$, based

on recession methods, have been developed in [A2].

A class of vector minimization problems is discussed in [F] following the same approach used in this paper.

2. Basic definitions and preliminaries. Throughout this paper X will be a reflexive Banach space. For any given weakly closed set C in X (actually, the recession notion to be considered is blind to weak closure), we define the recession cone of C as the weakly closed set

$$C^\infty = \Big\{x \in X : \ \exists\, t_n \downarrow 0, \exists\, x_n \in C, \ \ t_n x_n \rightharpoonup x\Big\}.$$

Here "$\rightharpoonup$" stands for the weak convergence. It results in that $\emptyset^\infty = \emptyset$. For any given function $h : X \to \mathbb{R} \cup \{+\infty\}$, the recession function of h is defined as the function h^∞ such that

$$\text{epi } h^\infty = (\text{epi } h)^\infty.$$

Consequently, it is not difficult to prove that

$$h^\infty(y) = \inf\left[\liminf_{n\to+\infty} \ t_n h\left(\frac{x_n}{t_n}\right) : \ t_n \downarrow 0, x_n \rightharpoonup y\right].$$

In the case where C is convex and closed it is known that, given $x_0 \in C$,

$$C^\infty = \Big\{x \in X : \ x_0 + tx \in C \quad \forall\, t > 0\Big\},$$

the cone does not depend on $x_0 \in K$, and when h is a convex and l.s.c. function, we have

$$h^\infty(x) = \lim_{\lambda\to+\infty} \frac{h(x_0+\lambda x) - h(x_0)}{\lambda} = \sup_{\lambda>0} \frac{h(x_0+\lambda x) - h(x_0)}{\lambda} \ \ \forall\, x_0 \in \text{dom } h,$$

where, as usual, dom $h = \{x \in X : \ h(x) < +\infty\}$ and the epigraph of h is the set epi $h = \{(x,t) \in X \times \mathbb{R} : \ h(x) \le t\}$. In what follows, given a set $K \subset X$ and a function $f : K \times K \to \mathbb{R}$, f^∞ will denote the recession function of f with respect to its second argument, i.e., the recession function of the function $y \in K \mapsto f(x,y)$ for any fixed $x \in K$. Certainly, we extend the function $f(x,\cdot)$ to all X by setting $f(x,y) = +\infty$ if $y \in X \setminus K$.

We list some basic results on recession cones in the following proposition that will be useful in what follows.

PROPOSITION 2.1. *The following hold:*

(a) $K_1 \subset K_2$ *implies* $K_1^\infty \subset K_2^\infty$;

(b) $(K+x)^\infty = K^\infty$ *for all* $x \in X$;

(c) *let* (K_i), $i \in I$, *be any family of nonempty sets in* X; *then*

$$\left(\bigcap_{i\in I} K_i\right)^\infty \subset \bigcap_{i\in I} (K_i)^\infty.$$

If, in addition, $\cap_i K_i \neq \emptyset$ *and each set* K_i, $i \in I$, *is closed and convex, then we obtain an equality in the previous inclusion.*

DEFINITION 2.2. *A function* $f : X \to \mathbb{R} \cup \{+\infty\}$ *with* dom f *being a convex set*

(i) *is said to be strictly quasi-convex if, given any u, v in X, $f(u) \neq f(v)$, one has $f(z) < \max\{f(u), f(v)\}$ for all $z \in\ ]u, v[$;*

(ii) *is said to be quasi-convex if each of its level sets is a convex set or, equivalently, if $f(tx + (1-t)y) \leq \max\{f(x), f(y)\}$ for all x, y in X and all $t \in [0, 1]$.*

Simple examples show that there are functions that are strictly quasi-convex but not quasi-convex. However, it is well known that strict quasi-convexity and lower semicontinuity imply quasi-convexity [K1].

3. The case $\varphi(x, y) = h(y)$. Given a closed, convex set $K \subset X$, and functions $f : K \times K \to \mathbb{R}$, $h : X \to \mathbb{R} \cup \{+\infty\}$ such that $K \cap \operatorname{dom} h \neq \emptyset$. We consider the problem

$$\text{find } \bar{x} \in K \ \text{ such that } \ f(\bar{x}, y) + h(y) \geq h(\bar{x}) \quad \forall\ y \in K. \tag{3.1}$$

Here, our basic assumptions are the following:

(f_0) $f(x, x) = 0$ for all $x \in K$;

(f_1) for every $x \in K \cap \operatorname{dom} h$ and every $y \in K \cap \operatorname{dom} h$, $f(x, y) + h(y) \geq h(x)$ implies $f(y, x) + h(x) \leq h(y)$;

(f_2) for every $z \in K$, $f(z, \cdot) + h(\cdot)$ is l.s.c. and strictly quasi-convex in K (hence quasi-convex);

(f_3) for every x, y in K, the function $t \in\ [0, 1] \mapsto f(ty + (1-t)x, y)$ is u.s.c. at $t = 0$;

(h) the function $h : X \to \mathbb{R} \cup \{+\infty\}$ is l.s.c. with dom h convex.

We start by recalling the well-known lemma due to Ky Fan [F1], which will be used in the proof of an existence result to problem (3.1) in the case where K is bounded. Although such an existence result is a particular case of Lemma 1 in [O], we present a proof of this result for the readers' convenience.

LEMMA 3.1 (see [F1]). *Let Y be an arbitrary set in a topological vector space X. Assume that for every $y \in Y$, $F(y)$ is closed in X and the following two conditions are satisfied:*

(a) *The convex hull of any finite set $\{y_1, \ldots, y_n\}$ of Y is contained in $\bigcup_{i=1}^n F(y_i)$;*

(b) *$F(y)$ is compact for at least one $y \in Y$.*

Then

$$\bigcap_{y \in Y} F(y) \neq \emptyset.$$

A result similar to the next lemma, in the case where $f(\cdot, y)$ is u.s.c., may be found in [F2].

LEMMA 3.2. *Let $K \subset X$ be a closed, convex, and bounded set. Assume functions f, h satisfy conditions (f_0), (f_1), (f_2), (f_3), and (h). Then problem* (3.1) *admits a solution.*

Proof. First we recall that every closed, bounded, convex set in a reflexive Banach space is weakly compact. We will apply Lemma 3.1 to the sets

$$F(y) = \Big\{x \in K :\ f(y, x) + h(x) \leq h(y)\Big\}, \quad y \in K \cap \operatorname{dom} h.$$

Let us verify the hypothesis of the lemma. Each of these sets is nonempty, closed, bounded, and convex because of (f_2) and the remark following Definition 2.2. Take any finite set $y_1, \ldots, y_n$ in $K \cap \operatorname{dom} h$. Notice that when $y \in K \setminus \operatorname{dom} h$, $F(y) = K$. If $z = \sum_{i=1}^n \alpha_i y_i \notin \bigcup_{i=1}^n F(y_i)$ for some $\alpha = (\alpha_1, \ldots, \alpha_n) \in [0, 1]^n$ with $\sum_{i=1}^n \alpha_i = 1$,

then $f(y_i,z)+h(z)>h(y_i)$ for all $i=1,\dots,n$. Thus, by (f_1), $f(z,y_i)+h(y_i)<h(z)$ for all $i=1,\dots,n$. Since $f(z,\cdot)+h(\cdot)$ is quasi-convex, we have

$$h(z)=f(z,z)+h(z)\le \max_{1\le i\le n}\{f(z,y_i)+h(y_i)\}.$$

The latter leads to a contradiction with the previous strict inequalities. Hence, as a consequence of Lemma 3.1, we obtain

$$\bigcap_{y\in K\cap \mathrm{dom}\ h} F(y)\neq\emptyset.$$

Take any $\bar{x}$ in this intersection; then $f(y,\bar{x})+h(\bar{x})\le h(y)$ for all $y\in K\cap\mathrm{dom}\ h$. Let $y\in K\cap\mathrm{dom}\ h$ be arbitrary, $y\neq\bar{x}$, and let $x_t=ty+(1-t)\bar{x}$, $t\in\ ]0,1[$. Obviously $x_t\in K\cap\mathrm{dom}\ h$ and $f(x_t,\bar{x})+h(\bar{x})\le h(x_t)$ for all $t\in\ ]0,1[$. If $f(x_t,y)+h(y)<h(x_t)$ for some $t\in\ ]0,1[$, then, in case $f(x_t,\bar{x})+h(\bar{x})<h(x_t)$, we obtain by quasi-convexity

$$h(x_t)=f(x_t,x_t)+h(x_t)\le\max\{f(x_t,\bar{x})+h(\bar{x}),f(x_t,y)+h(y)\}<h(x_t),$$

which is a contradiction; in case $f(x_t,\bar{x})+h(\bar{x})=h(x_t)(>f(x_t,y)+h(y))$, we also get a contradiction because of the strict quasi-convexity of $f(x_t,\cdot)+h(\cdot)$. Thus, we conclude that $f(x_t,y)+h(y)\ge h(x_t)$ for all $t\in\ ]0,1[$. By assumption (f_3) and (h), one obtains $f(\bar{x},y)+h(y)\ge h(\bar{x})$. □

We now consider the case when K is unbounded. For that purpose we introduce the following sets:

$$R_0=\bigcap_{y,z\in K}\Big\{v\in K^\infty:\ f(y,z+\lambda v)+h(z+\lambda v)\le\max\{f(y,z)+h(z),h(y)\}\ \ \forall\,\lambda>0\Big\},$$

$$R_1=\bigcap_{y\in K}\Big\{v\in K^\infty:\ f(y,y+\lambda v)+h(y+\lambda v)\le h(y)\ \ \forall\,\lambda>0\Big\}.$$

Obviously, both sets are closed convex cones. In addition, we also consider the set

$$R=\Big\{v\in K^\infty:\ (f(y,\cdot)+h(\cdot))^\infty(v)\le 0\ \ \forall\,y\in K\Big\},$$

which is a weakly closed cone not necessarily convex: recall that for fixed $y\in K$, $f(y,\cdot)$ is extended to all X by putting $f(y,x)=+\infty$ if $x\in X\setminus K$. Let us denote by S the solution set to problem (3.1). Then

$$S=\Big\{x\in K:\ f(x,y)+h(y)\ge h(x)\ \ \forall\,y\in K\Big\}.$$

Taking into account the assumptions on f and h (see the last part of the proof of Lemma 3.2), we have

$$S=\Big\{x\in K:\ f(y,x)+h(x)\le h(y)\ \forall\,y\in K\Big\}=\bigcap_{y\in K}\Big\{x\in K:\ f(y,x)+h(x)\le h(y)\Big\}.$$

Setting for any fixed $y\in K\cap\mathrm{dom}\ h$

$$S_0(y)=\Big\{x\in X:\ f(y,x)+h(x)\le h(y)\Big\},$$

we have

$$S = \bigcap_{y \in K \cap \operatorname{dom} h} K \cap S_0(y).$$

Therefore, because of the assumptions on f, the set S is convex and closed. Then, by Proposition 2.1,

$$S^\infty = \left(\bigcap_{y \in K \cap \operatorname{dom} h} K \cap S_0(y) \right)^\infty \subset \bigcap_{y \in K \cap \operatorname{dom} h} (K \cap S_0(y))^\infty \subset \bigcap_{y \in K \cap \operatorname{dom} h} (S_0(y))^\infty \cap K^\infty, \tag{3.2}$$

where equalities hold if $S \neq \emptyset$. On the other hand,

$$\begin{aligned} R_1 &= \bigcap_{y \in K \cap \operatorname{dom} h} \Big\{ v \in K^\infty : \ f(y, y + \lambda v) + h(y + \lambda v) \leq h(y) \ \ \forall\ \lambda > 0 \Big\} \\ &= \bigcap_{y \in K \cap \operatorname{dom} h} \Big\{ v \in K^\infty : \ y + \lambda v \in S_0(y) \ \forall\ \lambda > 0 \Big\} \\ &= \bigcap_{y \in K \cap \operatorname{dom} h} \Big\{ v \in K^\infty : \ v \in (S_0(y))^\infty \ \Big\} = \bigcap_{y \in K \cap \operatorname{dom} h} K^\infty \cap (S_0(y))^\infty. \end{aligned}$$

Hence, $S^\infty \subset R_1$. Since $(S_0(y))^\infty \subset \{v \in X : \ (f(y, \cdot) + h(\cdot))^\infty(v) \leq 0\}$ for all $y \in K$, we conclude $R_1 \subset R$. Thus, we have the following theorem.

THEOREM 3.3. *Let $K \subset X$ be a closed convex set. Under assumptions (f_i), $i = 0, 1, 2, 3$, and (h), we have that*

(a) *R_0 and R_1 are closed convex cones; R is a weakly closed cone;*

(b) *$S^\infty \subset R_0 \subset R_1 \subset R$;*

(c) *if the solution set S is nonempty, then $S^\infty = R_1 = R_0$. Thus, $S + R_1 = S$. Furthermore, if S is bounded, then $R_1 = R_0 = \{0\}$;*

(d) (see [FS]) *if $f(x, \cdot) + h(\cdot)$ is convex for all $x \in K$, then $R_1 = R$.*

Proof. Part (a) and the fact that $R_0 \subset R_1 \subset R$ were already proved. Let us prove that $S^\infty \subset R_0$. If $S = \emptyset$, there is nothing to do. Let $v \in S^\infty$, then $\exists\ t_n \downarrow 0$, $\exists\ x_n \in S$ such that $t_n x_n \rightharpoonup v$. Then $v \in K^\infty$. For any fixed $y \in K \cap \operatorname{dom} h$, we have $f(x_n, y) + h(y) \geq h(x_n)$ for all n. Let us fix any $\lambda > 0$ and $z \in K$; then the quasi-convexity of $f(y, \cdot) + h(\cdot)$ implies

$$\begin{aligned} f(y, (1 - \lambda t_n) z + \lambda t_n x_n) + h((1 - \lambda t_n) z + \lambda t_n x_n) & \\ &\leq \max\{f(y, z) + h(z), f(y, x_n) + h(x_n)\} \\ &\leq \max\{f(y, z) + h(z), h(y)\}. \end{aligned}$$

By the l.s.c. property, $v \in R_0$ and the proof of part (b) is complete. Part (c) follows from (b) and the remark in the previous paragraph. Finally, part (d) is a consequence of the representation of the recession function in the convex case. □

We now establish the first main existence theorem of this section.

THEOREM 3.4. *If assumptions (f_i), $i = 0, 1, 2, 3$, and (h) on the functions f and h hold, then problem (3.1) admits at least a solution if and only if (f_4) is satisfied, where*

(f_4) if the sequence $x_n \in K \cap \operatorname{dom} h$, $||x_n|| \to +\infty$ is such that $\frac{x_n}{||x_n||} \rightharpoonup v$ with $v \in R_0$ and for all $y \in K \cap \operatorname{dom} h$, n_y exists such that

$$f(x_n, y) + h(y) \geq h(x_n) \quad \text{when} \quad n \geq n_y,$$

then there exists $u \in K \cap \operatorname{dom} h$ *such that for* n *sufficiently large* $||u|| < ||x_n||$ *and*

$$f(x_n, u) + h(u) \leq h(x_n).$$

Proof. The "only if" part is as follows. For every $n \in \mathbb{N}$, set $K_n = \{x \in K : ||x|| \leq n\}$. We may suppose, without loss of generality, that $K_n \cap \operatorname{dom} h \neq \emptyset$ for all $n \in \mathbb{N}$. Take any solution $x_n \in K_n \cap \operatorname{dom} h$ to the problem

$$\text{find } \bar{x} \in K_n \cap \operatorname{dom} h \text{ such that } f(\bar{x}, y) + h(y) \geq h(\bar{x}) \ \forall\, y \in K_n. \tag{3.3}$$

Such x_n exists by virtue of Lemma 3.2. Certainly, if $||x_n|| < n$ for some n, then standard arguments show that x_n is also a solution to problem (3.1). If, on the contrary, $||x_n|| = n$ for all $n \in \mathbb{N}$, we will prove that x_n, for n large enough, will be also a solution to our problem (3.1). To that end, we reason as follows. First of all, we may suppose that $\frac{x_n}{||x_n||} \rightharpoonup v$. Then $v \in K^\infty$. For any fixed $y \in K \cap \operatorname{dom} h$, we have $f(x_n, y) + h(y) \geq h(x_n)$ for all n sufficiently large ($n \geq n_y > ||y||$). Let us fix any $\lambda > 0$ and $z \in K$; then the quasi-convexity of $f(y, \cdot) + h(\cdot)$ implies

$$\begin{aligned} f\left(y, \left(1 - \frac{\lambda}{||x_n||}\right) z + \frac{\lambda}{||x_n||} x_n\right) + h\left(\left(1 - \frac{\lambda}{||x_n||}\right) z + \frac{\lambda}{||x_n||} x_n\right) \\ \leq \max\{f(y, z) + h(z), f(y, x_n) + h(x_n)\} \\ \leq \max\{f(y, z) + h(z), h(y)\}. \end{aligned}$$

By the l.s.c. property, $v \in R_0$. Thus, assumption (f_4) asserts that, for n sufficiently large, there exists $u \in K \cap \operatorname{dom} h$ such that $||u|| < ||x_n||$ and

$$f(x_n, u) + h(u) \leq h(x_n).$$

We claim that x_n is also a solution to problem (3.1). If not, there exists $y \in K \cap \operatorname{dom} h$, $||y|| > n$ such that $f(x_n, y) + h(y) < h(x_n)$. Since $||u|| < ||x_n|| = n$, we have $f(x_n, u) + h(u) = h(x_n) (> f(x_n, y) + h(y))$ and, moreover, we can find $z \in \,]u, y[$ such that $||z|| < n$. Thus, by the strict quasi-convexity of $f(x_n, \cdot) + h(\cdot)$,

$$f(x_n, z) + h(z) < \max\{f(x_n, u) + h(u), f(x_n, y) + h(y)\} = h(x_n),$$

which gives a contradiction since x_n is a solution to (3.3). Consequently, $f(x_n, y) + h(y) \geq h(x_n)$, proving our claim.

To prove the "if" part, take any sequence (x_n) in K, $||x_n|| \to +\infty$ and any solution $\bar{x}$ to problem (3.1). Then condition (f_4) is satisfied by setting $u = \bar{x}$ and we use (f_1). □

This theorem, with $h = 0$, was established in [FS, Theorem 3.7] under the convexity assumption on $f(x, \cdot)$; see also [CCR, BO]. On the other hand, our assumption (f_4) permits the solution set to be unbounded. This makes our assumption different from most existent results; see [BNS] for convex problems, [HP] (resp., [HS]) for variational inequalities in finite (resp., infinite) dimensional spaces. In fact, in [FS] it is shown that (f_4) is strictly weaker than the corresponding assumptions imposed in the above-mentioned papers.

When $X = \mathbb{R}^n$, assumption (f_4) is satisfied vacuously if $R_0 = \{0\}$. Contrary to this case, only the condition $R_0 = \{0\}$ is not sufficient to guarantee existence of solutions, unless a further assumption is imposed (see Theorem 3.6). A condition implying $R_0 = \{0\}$ is given in Remark 3.8. A more general condition is expressed in the next lemma.

LEMMA 3.5. *Assume K is a closed convex set and for every $x \in K$, $f(x,\cdot)+h(\cdot)$ is l.s.c. and strictly quasi-convex. The following assertions hold:*

(a) *If $R_0 \subset -R_0$ (resp., $R_1 \subset -R_1$), then $f(y, y+\lambda v)+h(y+\lambda v) = h(y)$ for all $\lambda \in \mathbb{R}$, all $v \in R_0$ (resp., all $v \in R_1$), and all $y \in K$;*

(b) *if $f(x,\cdot)+h(\cdot)$ is convex for all $x \in K$, then R_1 is also convex and $R = R_1$. In this case, R is a subspace if and only if $R \subset -R$.*

Proof. Assume that $R_0 \subset -R_0$ (the case $R_1 \subset -R_1$ is similar); then every $v \in R_0$ satisfies $-v \in R_0$. Thus one obtains, in particular, $f(y, y+\lambda v)+h(y+\lambda v) \le h(y)$ and $f(y, y-\lambda v)+h(y-\lambda v) \le h(y)$ for all $\lambda > 0$ and all $y \in K$. By using the quasi-convexity of $f(y,\cdot)+h(\cdot)$, one has

$$h(y) = f(y,y)+h(y) \le \max\{f(y,y-\lambda v)+h(y-\lambda v), f(y,y+\lambda v)+h(y+\lambda v)\} \le h(y).$$

Thus, either $f(y, y-\lambda v)+h(y-\lambda v) = h(y)$ or $f(y, y+\lambda v)+h(y+\lambda v) = h(y)$. In case one of them is zero and the other is negative, we use the strict quasi-convexity of $f(y,\cdot)+h(\cdot)$ to get a contradiction. Therefore, $f(y, y+\lambda v)+h(y+\lambda v) = h(y)$ for all $\lambda \in \mathbb{R}$ and all $y \in K$. Part (b) follows directly since the recession cone of a convex set is also convex. The last part follows in a similar way. ∎

By many aspects, among them numerical, one may be interested in knowing when the solution set is bounded; thus a "coercivity" condition has to arise. To that purpose, we introduce the following compactness condition meaningful only in spaces of infinite dimension. Such a condition, when applied to problems in mechanics, is usually a consequence of some embedding theorems.

(f_5) Any sequence $x_n \in K \cap \operatorname{dom} h$ with $||x_n|| \to +\infty$ such that for all $y \in K \cap \operatorname{dom} h$, n_y exists such that

$$f(x_n, y)+h(y) \ge h(x_n) \quad \text{when} \quad n \ge n_y$$

admits a subsequence (x_{n_k}) such that $\frac{x_{n_k}}{||x_{n_k}||}$ converges strongly.

This additional condition is required in infinite dimensional spaces since $S^\infty = \{0\}$ does not imply, in general, the boundedness of S.

Apart from the finite dimensional case, assumption (f_5) is usually satisfied when dealing with variational inequalities [GT, AGT] or minimization problems arising in mechanics [BBGT, CCR], where it is a consequence of some embedding theorems. To our knowledge, an assumption like (f_5) appears for the first time in [GT] in the context of variational inequalities. However, there are instances where assumption (f_5) does not hold (see Example 3.9 below).

We thus have the second main existence theorem.

THEOREM 3.6. *Let K be a closed convex set. Assume function f satisfies assumptions (f_i), $i = 0,1,2,3$, (h), and (f_5). Then the following three conditions are equivalent:*

(a) $R_0 = \{0\}$;

(b) $R_1 = \{0\}$;

(c) *the solution set to problem* (3.1) *is nonempty and bounded.*

Proof. (a) $\Longrightarrow$ (c): $R_0 = \{0\}$, together with assumption (f_5), implies the boundedness of the sequence constructed in the proof of Theorem 3.4, since otherwise R_0 would contain a nonnull element. Thus, by assumptions (f_i), $i = 1,2,3$, a solution will be obtained as the weak limit of (x_n), and the boundedness of the solution set is implied again by assumption (f_5). The implications (c) $\Longrightarrow$ (b) and (b) $\Longrightarrow$ (a) are consequences of Theorem 3.3. ∎

We now present another sufficient and necessary condition for the solution set to be nonempty and bounded. The sufficiency of this condition has its origin, as far as we know, in [K2]; for a historical reference we quote [HP]. This condition regards the existence of a bounded set such that no element outside this set is a candidate for solution. The necessity of such a condition was proved in [DH] for classical variational inequalities. Sometimes it may be difficult to find such a bounded set; in this case, Remark 3.8 (see also (3.5)) plays a fundamental role.

THEOREM 3.7. *Let K be a closed convex set. Assume functions f and h satisfy assumptions (f_i), $i = 0, 1, 2, 3$, and (h). Then the solution set to* (3.1) *is nonempty and bounded if and only if the following "coercivity" condition holds:*

$$\exists\ r > 0 \quad \forall\ x \in K \setminus K_r \quad \exists\ y \in K_r \cap \operatorname{dom} h :\ f(x, y) + h(y) < h(x),$$

where $K_r = \{x \in K :\ ||x|| \le r\}$ is such that $K_r \cap \operatorname{dom} h \neq \emptyset$.

Proof. Assume the solution set S to be nonempty and bounded. The reasoning is as in [DH]. Let $\bar{x} \in S$. If the coercivity condition does not hold, then, in particular, for $n > \sup_{x\in S} ||x|| + 1$, there exists $x \in K \setminus K_n$ such that for all $y \in K_n$ one has $f(x, y) + h(y) \ge h(x)$. Notice that $x \in \operatorname{dom} h$. Take $\lambda \in\ ]0, 1[$ such that, setting $z = \bar{x} + \lambda(x - \bar{x})$, we have $n - 1 \le ||z|| < n$. Clearly $z \in K \cap \operatorname{dom} h$. We claim that z is a solution to problem (3.3). In fact, for all $y \in K_n$, $f(y, z) + h(z) \le \max\{f(y, \bar{x}) + h(\bar{x}), f(y, x) + h(x)\} \le h(y)$ by quasi-convexity of $f(y, \cdot) + h(\cdot)$. By assumption (f_3) (see the proof of Lemma 3.2), we conclude that the claim is true. We now prove that z is actually a solution to problem (3.1). If not, there exists $y_1 \in K \setminus K_n$ satisfying $f(z, y_1) + h(y_1) < h(z)$. Choose $\tilde{z} = \alpha y_1 + (1 - \alpha)z$ with $\alpha \in\ ]0, 1[$ such that $\tilde{z} \in K_n$. Then, by strict quasi-convexity, we have

$$f(z, \tilde{z}) + h(\tilde{z}) < \max\{f(z, y_1) + h(y_1), f(z, z) + h(z)\} = h(z),$$

which contradicts the fact that z is a solution to (3.3), proving that $z \in S$. Since $||z|| \ge n - 1 > \sup_{x\in S} ||x||$, we conclude the coercivity condition has to hold.

Let us prove the solution set S is nonempty and bounded. We construct a sequence (x_n) as solutions of the problem (3.1) restricted to K_n (see Lemma 3.2). The coercivity assumption implies that such a sequence is bounded and, therefore, there exists $\bar{x} \in K$ such that $x_n \rightharpoonup \bar{x}$. It follows from (f_1) and (f_2) that $\bar{x}$ is a solution to (3.1). The boundedness of S is again a consequence of the coercivity condition. ∎

Remark 3.8. We notice that if K is bounded or the problem is coercive in the sense that

$$\liminf_{||x||\to+\infty, x\in K} \{-f(x, y_0) + h(x)\} > h(y_0) \ \text{ for some } \ y_0 \in K \cap \operatorname{dom} h, \tag{3.4}$$

then no sequence in (f_4) or (f_5) exists. Thus, (f_4) and (f_5) hold vacuously. Hence, if the functions f and h satisfy (f_i), $i = 0, 1, 2, 3$, and (3.4), then the solution set is nonempty and bounded. As a consequence $R_0 = R_1 = \{0\}$. In fact, the nonemptiness of the solution set follows from Theorem 3.4, and the boundedness is a consequence of (3.4).

We point out that there are functions h ($f = 0$) satisfying $R_0 = \{0\}$ without being coercive in the sense of (3.4). Such a situation is exhibited in Example 3.9. On the other hand, condition (3.4) is implied by the usual assumption

$$\lim_{||x||\to+\infty, x\in K} \frac{-f(x, y_0) + h(x)}{||x||} = +\infty. \tag{3.5}$$

This condition was used in [GY] in case $f(x,y) = \langle F(x), y - x\rangle$ with h being a convex l.s.c. function (see also [GT]), while in case $h = 0$ condition (3.4) is used with non-monotone operator F. When $h(x) = \langle x^*, x\rangle$, $x^* \in X^*$, (3.5) amounts to writing

$$\lim_{||x||\to+\infty, x\in K} \frac{-f(x,y_0)}{||x||} = +\infty,$$

which is considered in [HP] in the finite dimensional case. We refer to [KS] for more general spaces.

The next example shows an instance where Theorem 3.4 can be applied without satisfying a "compactness condition" as required in [A1] or Theorem 3.6.

Example 3.9. See also [FS]. Take

$$K = X = l^2 = \left\{ x = (x_i) : \ ||x|| = \sum_{i\geq 1} |x_i|^2 < +\infty \right\} \ \text{ and } h(x) = \sum_{i\geq 1} \frac{|x_i|}{i^2}.$$

Let us consider $f = 0$. Denote by e_n the unit element in l^2; i.e., e_n is the sequence having 1 in the nth position and zero in all other positions. Then consider the sequence $x_n = ne_n$ in l^2; such a sequence does not satisfy the compactness condition required in [A1]. Nevertheless, function h has a minimum point as a consequence of our Theorem 3.4 (take $u = 0$), although it may also be obtained directly. It is worthwhile to mention that, in this case, $R = R_0 = \{0\}$. However, we have

$$\liminf_{||x||\to+\infty, x\in K} h(x) \leq h(y_0) \ \text{ for every } y_0 \in l^2,$$

that is, it is not coercive in the sense of (3.4).

Remark 3.10. The phenomenon presented in the preceding example is related to ill-posed problems. Notice that the sequence above is minimizing and there exists only one minimizer for h, but such a sequence does not converge strongly. This cannot happen in finite dimensional spaces.

Theorems 3.4 and 3.6 can be considered as extensions to the quasi-convex case of the results established in [FS] and therefore also of that proved in [C]. In particular, Theorem 3.4 extends several results appearing in the literature, among them the result proved in [GT]. More precisely, in that paper the case $\langle F(x) - x^*, y - x\rangle + h(y) \geq h(x)$ with F monotone and h convex is studied, and some applications to unilateral problems are also exhibited. In this case,

$$R_1 = \left\{ v \in K^\infty : \ \frac{h(y+\lambda v) - h(y)}{\lambda} \leq \langle x^* - F(y), v\rangle \ \ \forall\ \lambda > 0,\ \forall\ y \in K \right\}.$$

Here we do not require the convexity of h. Related results can also be found in [AGT] for this particular situation.

If K is bounded or the coerciveness assumption (3.4) or (3.5) is satisfied, one can recognize from our Theorems 3.4 and 3.6 the classical results for nonlinear variational inequalities [KS]. Certainly, our theorems allow wider generality, not only to noncoercive variational inequalities, but also to generalized quasi-variational inequalities or equilibrium problems. For instance, in the case where R_0 is such that $R_0 \subset -R_0$, we obtain an existence result once assumption (f_5) is satisfied without further restrictions on the data. More precisely, on combining Theorem 3.4, Lemma 3.5, and Theorem 3.3, we have the following corollary.

COROLLARY 3.11. *Assume functions f and h satisfy assumptions (f_i), $i = 0, 1, 2, 3$, (f_5), and (h). If either $R_0 \subset -R_0$ or $R_1 \subset -R_1$, then problem* (3.1) *admits at least a solution. As a consequence $R_0 = R_1$.*

Proof. We proceed as in Theorem 3.4 to construct a sequence (x_n) with $\frac{x_n}{||x_n||} \rightharpoonup v \in R_0$. Applying (f_5), we get $\frac{x_n}{||x_n||} \to v$, $v \neq 0$. Then, the result is obtained as a consequence of Lemma 3.5 and Theorem 3.4. ∎

We have to point out that, in the general nonconvex case, the condition $R_0 \subset -R_0$ or $R_1 \subset -R_1$ is not enough to guarantee the existence of a solution to the corresponding problem, as Example 3.13 in [BBGT] shows.

Example 3.12. Let K be a convex closed set in X and let $F : X \to X^*$, $g : X \to X$ be single-valued maps such that the function f defined as $f(x, y) = \langle F(x), g(y) - g(x)\rangle$, $x, y \in K$, satisfies assumptions (f_i), $i = 1, \ldots, 4$. It is requested to

$$\text{find } \bar{x} \in K \text{ such that } \langle F(\bar{x}), g(y) - g(\bar{x})\rangle \geq 0 \quad \forall\, y \in K.$$

In this case,

$$R_1 = \Big\{v \in K^\infty :\ \langle F(y), g(y + \lambda v) - g(y)\rangle \leq 0 \quad \forall\, \lambda > 0,\ \forall\, y \in K\Big\}.$$

In case $g(\xi) = \xi$, this cone reduces to $R_1 = K^\infty \cap (F(K))^0$, where $(F(K))^0$ is the polar cone of $F(K)$.

Remark 3.13. The case $f \equiv 0$ corresponding to minimization problems deserves special attention by virtue of Theorem 3.4 in [BBGT], Theorem 2.3 in [BT], and Theorem 2.1 in [A1]. In this particular situation, our result obtained is a refinement of to the case when h is a quasi-convex function: first, we do not need to know a priori the nonnegativeness of h^∞; and second, when imposing the additional condition $h^\infty \geq 0$, our assumption (f_4) is required to hold on the set R_0 which is strictly contained in $\ker h^\infty = R$ used in the above-mentioned papers. The latter is shown by taking the function $h(y) = \sqrt{y}$, $y \geq 0$, where $R_0 = R_1 = \{0\}$ and $R = [0, +\infty[$.

Example 3.14. Let $A : \mathbb{R}^n \to \mathbb{R}^p$ be a linear mapping. Let $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper, strict quasi-convex, and l.s.c. function and consider the set K as

$$K = \Big\{x \in \mathbb{R}^n :\ y = Ax\Big\},\ K \cap \operatorname{dom} h \neq \emptyset,$$

which is closed and convex. In this case it is not difficult to prove that

$$K^\infty = \Big\{v \in \mathbb{R}^n :\ Av = 0\Big\}.$$

We are interested in the following problem:

$$\min\{h(x) :\ Ax = y\}. \tag{3.6}$$

In this case,

$$R_1 = \Big\{v \in \mathbb{R}^n :\ Av = 0,\ h(y + \lambda v) \leq h(y)\ \ \forall\, \lambda > 0,\ \forall\, y \in K\Big\}.$$

Then, $R_1 \subset -R_1$ if and only if for all $v \in \mathbb{R}^n$ with $Av = 0$, for all $\lambda > 0$ and all $y \in K$: $h(y + \lambda v) \leq h(y)$ implies $h(y - \lambda v) \leq h(y)$. Under this assumption the solution set to problem (3.6) is nonempty. In addition, the set of solutions will be nonempty and bounded if and only if $R_1 = \{0\}$.

The case when h is convex was studied in [R, Theorems 9.2 and 27.3] and [AC]. The results of these papers are extended in [FS].

4. The general case. Now, in addition to the set K and function f as before, we are given a function $\varphi : K \times X \to \mathbb{R} \cup \{+\infty\}$. The problem to be discussed in this section is

$$\text{find } \bar{x} \in K \quad f(\bar{x}, y) + \varphi(\bar{x}, y) \geq \varphi(\bar{x}, \bar{x}) \quad \forall\, y \in K. \tag{4.1}$$

Since every $\bar{x} \in K$ such that $K \cap \operatorname{dom} \varphi(\bar{x}, \cdot) = \emptyset$ is a solution to (4.1), we will assume that $K \cap \operatorname{dom} \varphi(x, \cdot) \neq \emptyset$ for all $x \in K$. We need the following assumptions:

$(\tilde{f}_0)$ $f(x, x) = 0$ for all $x \in K$;

$(\tilde{f}_1)$ $f(x, y) + f(y, x) \leq 0$ for all $x, y \in K$;

$(\tilde{f}_2)$ for every $(x, y) \in K \times K$, the function $f(x, \cdot) + \varphi(y, \cdot)$ is strictly quasi-convex and l.s.c. in K (these two conditions imply quasi-convexity);

$(\tilde{f}_3)$ for every $y \in K$, $\varphi(y, \cdot)$ is l.s.c. in X with $\operatorname{dom} \varphi(y, \cdot)$ being a convex set;

$(\tilde{f}_4)$ for every $x, y \in K$, the function $t \in [0, 1] \mapsto f(ty + (1 - t)x, y)$ is u.s.c. at $t = 0$.

LEMMA 4.1. *Let K be a convex, closed, and bounded set in X. Under assumptions $(\tilde{f}_i)$, $i = 0, \ldots, 3$, we have*

$$\bigcap_{w \in K} F_x(w) \neq \emptyset \quad \forall\, x \in K,$$

where $F_x(w) = \{y \in K : f(w, y) + \varphi(x, y) \leq \varphi(x, w)\}$. If, in addition, $(\tilde{f}_4)$ is imposed, then every $\bar{y}$ in this intersection satisfies

$$f(\bar{y}, w) + \varphi(x, w) \geq \varphi(x, \bar{y}) \quad \forall\, w \in K. \tag{4.2}$$

Proof. In the case where $x \in K$ is such that $K \cap \operatorname{dom} \varphi(x, \cdot) = \emptyset$, then $\varphi(x, w) = +\infty = \varphi(x, y)$ for all $w, y \in K$. Thus $F_x(w) = K$ for all $w \in K$ and hence

$$\bigcap_{w \in K} F_x(w) = K \neq \emptyset.$$

Therefore, we consider $x \in K$ such that $K \cap \operatorname{dom} \varphi(x, \cdot) \neq \emptyset$. In this case, one has

$$\bigcap_{w \in K} F_x(w) = \bigcap_{w \in K \cap \operatorname{dom}\, \varphi(x, \cdot)} F_x(w),$$

where each set $F_x(w)$ is closed, convex, and nonempty for all $w \in K \cap \operatorname{dom} \varphi(x, \cdot)$. An application of Lemma 3.1 (see the proof of Lemma 3.2) allows us to conclude that

$$\bigcap_{w \in K} F_x(w) = \bigcap_{w \in K \cap \operatorname{dom}\, \varphi(x, \cdot)} F_x(w) \neq \emptyset \text{ if } K \cap \operatorname{dom} \varphi(x, \cdot) \neq \emptyset.$$

Let us prove the second part. In the case where $K \cap \operatorname{dom} \varphi(x, \cdot) = \emptyset$ there is nothing to prove. Thus, we consider the case $K \cap \operatorname{dom} \varphi(x, \cdot) \neq \emptyset$. Take any $\bar{y} \in F_x(w)$ for all $w \in K \cap \operatorname{dom} \varphi(x, \cdot)$; then

$$f(w, \bar{y}) + \varphi(x, \bar{y}) \leq \varphi(x, w) \quad \forall\, w \in K \cap \operatorname{dom} \varphi(x, \cdot).$$

Thus $\bar{y} \in K \cap \operatorname{dom} \varphi(x, \cdot)$. Certainly (4.2) is trivially satisfied if $w \in K \setminus \operatorname{dom} \varphi(x, \cdot)$. Therefore, we need only to consider $w \in K \cap \operatorname{dom} \varphi(x, \cdot)$. Let $x_t = tw + (1 - t)\bar{y}$ for $t \in\ ]0, 1]$. Then $x_t \in K \cap \operatorname{dom} \varphi(x, \cdot)$, $w \neq \bar{y}$. Hence, $f(x_t, \bar{y}) + \varphi(x, \bar{y}) \leq \varphi(x, x_t)$

for all $t \in\ ]0,1[$. We claim that $f(x_t,w)+\varphi(x,w) \geq \varphi(x,x_t)$ for all $t \in\ ]0,1[$. If on the contrary $f(x_t,w)+\varphi(x,w) < \varphi(x,x_t)$ for some $t \in\ ]0,1[$, the quasi-convexity of $f(x_t,\cdot)+\varphi(x,\cdot)$ leads to a contradiction in case $f(x_t,\bar{y})+\varphi(x,\bar{y}) < \varphi(x,x_t)$. In the case when $f(x_t,\bar{y})+\varphi(x,\bar{y}) = \varphi(x,x_t)$, we use the strict quasi-convexity to get a contradiction. It follows that the claim is proved, i.e., $f(x_t,w)+\varphi(x,w) \geq \varphi(x,x_t)$ for all $t \in\ ]0,1[$. By letting $t \downarrow 0$ and using $(\tilde{f}_3)$ and $(\tilde{f}_4)$, one obtains $f(\bar{y},w)+\varphi(x,w) \geq \varphi(x,\bar{y})$, which proves the lemma. □

Now, our purpose is to deal with the unbounded case. To that end, we introduce the following sets:

$$R_0 = \Bigg\{ v \in K^\infty :\ \exists\ x_n \in K,\ ||x_n|| \to +\infty,\ \frac{x_n}{||x_n||} \rightharpoonup v,\ f(y,z+\lambda v) + \limsup_{n\to+\infty} \varphi\left(x_n, \left(1-\frac{\lambda}{||x_n||}\right) z + \frac{\lambda}{||x_n||} x_n\right) \leq \liminf_{n\to+\infty} \max\{f(y,z)+\varphi(x_n,z), \varphi(x_n,y)\} \quad \forall\ y,z \in K,\ \forall\ \lambda > 0 \Bigg\},$$

$$R_1 = \Bigg\{ v \in K^\infty :\ \exists\ x_n \in K,\ ||x_n|| \to +\infty,\ \frac{x_n}{||x_n||} \rightharpoonup v,\ f(y,y+\lambda v) + \limsup_{n\to+\infty} \varphi\left(x_n, \left(1-\frac{\lambda}{||x_n||}\right) y + \frac{\lambda}{||x_n||} x_n\right) \leq \liminf_{n\to+\infty} \varphi(x_n,y)\ \ \forall\ y \in K,\ \forall\ \lambda > 0 \Bigg\},$$

It is easy to see that both sets reduce to the ones introduced in section 3.

$(\tilde{f}_5)$ For every closed, bounded, and convex set $C \subset K$, for every filter (or generalized sequence) (x_α), (z_α) in K such that $x_\alpha \rightharpoonup x$, $z_\alpha \rightharpoonup z$, $x,z \in C$, and

$$f(w,z_\alpha)+\varphi(x_\alpha,z_\alpha) \leq \varphi(x_\alpha,w)\ \ \forall\ w \in C,\ \ \forall\ \alpha,$$

one has

$$f(w,z)+\varphi(x,z) \leq \varphi(x,w)\ \ \forall\ w \in C.$$

$(\tilde{f}_6)$ If the sequence $x_n \in K$, $||x_n|| \to +\infty$, is such that $\frac{x_n}{||x_n||} \rightharpoonup v$ with $v \in R_0$ and for all $y \in K$, n_y exists such that

$$f(x_n,y)+\varphi(x_n,y) \geq \varphi(x_n,x_n)\ \ \text{when } n \geq n_y,$$

then there exists $u \in K$ such that for n sufficiently large $||u|| < ||x_n||$ and

$$f(x_n,u)+\varphi(x_n,u) \leq \varphi(x_n,x_n).$$

For $n \in \mathbb{N}$, set $K_n = K \cap \{x :\ ||x|| \leq n\}$ which may be assumed nonempty for all $n \in \mathbb{N}$. Let us consider the set-valued map $S_n : K_n \to K_n$ defined by

$$S_n(x) = \bigcap_{w \in K_n} F_x^n(w),$$

where $F_x^n(w) = \Big\{ y \in K_n :\ f(w,y)+\varphi(x,y) \leq \varphi(x,w) \Big\}$.

We have the following result.

LEMMA 4.2. *Let us fix $n \in \mathbb{N}$ under the assumptions of Lemma 4.1 for K_n, instead of K, that the set-valued map S_n has nonempty, convex, closed values. If, in addition, $(\tilde{f}_5)$ is satisfied, then S_n has a fixed point; i.e., there exists $x_n \in K_n$ such that $x_n \in S_n(x_n)$.*

Proof. The first part is a straightforward consequence of the previous lemma and the assumptions on f. The existence of a fixed point follows from the well-known Katutani fixed-point theorem, since the map S_n is u.s.c. by assumption $(\tilde{f}_5)$. □

THEOREM 4.3. *Under assumptions $(\tilde{f}_i)$, $i = 0, \ldots, 6$, problem (4.1) admits at least a solution.*

Proof. For every $n \in \mathbb{N}$, we consider K_n as above. By the previous lemma, take $x_n \in K_n$ such that $x_n \in S^n(x_n)$. Then, for every $n \in \mathbb{N}$,

$$f(y, x_n) + \varphi(x_n, x_n) \leq \varphi(x_n, y) \ \ \forall\ y \in K_n.$$

(i) Assume first that $\sup_n ||x_n|| < +\infty$; then we may suppose that $x_n \rightharpoonup x$, $x \in K$. Let $y \in K \cap \operatorname{dom} \varphi(x, \cdot)$ be arbitrary. Take $n_0 > \max\{||y||, \sup_n ||x_n||\}$. Then we have

$$f(w, x_n) + \varphi(x_n, x_n) \leq \varphi(x_n, w) \ \ \forall\ w \in K_{n_0},\ \forall\ n \geq n_0.$$

Assumption $(\tilde{f}_5)$ implies

$$f(w, x) + \varphi(x, x) \leq \varphi(x, w) \ \ \forall\ w \in K_{n_0}.$$

The latter, together with $(\tilde{f}_4)$, give $f(x, w) + \varphi(x, w) \geq \varphi(x, x)$ for all $w \in K_{n_0}$. It follows, in particular, that $f(x, y) + \varphi(x, y) \geq \varphi(x, x)$, proving that x is a solution to problem (4.1).

(ii) Let us now assume that $\sup_n ||x_n|| = +\infty$. Up to a subsequence, we may suppose that $||x_n|| \to +\infty$ and $\frac{x_n}{||x_n||} \rightharpoonup v$. We will prove that x_n, for n large enough, will be also a solution to (4.1). To that purpose, we reason as follows. For any fixed $y \in K$, we have $f(x_n, y) + \varphi(x_n, y) \geq \varphi(x_n, x_n)$ for all n sufficiently large. On the other hand, for all n large enough, the quasi-convexity (see $(\tilde{f}_2)$) implies

$$f\left(y, \left(1 - \frac{\lambda}{||x_n||}\right) z + \frac{\lambda}{||x_n||} x_n\right) + \varphi\left(x_n, \left(1 - \frac{\lambda}{||x_n||}\right) z + \frac{\lambda}{||x_n||} x_n\right)$$

$$\leq \max\{f(y, z) + \varphi(x_n, z), f(y, x_n) + \varphi(x_n, x_n)\} \leq \max\{f(y, z) + \varphi(x_n, z), \varphi(x_n, y)\}.$$

By the lower semicontinuity property, one concludes that $v \in R_0$. Thus, by assumption $(\tilde{f}_6)$, there exist $u \in K$ such that $||u|| < ||x_n||$ and $f(x_n, u) + \varphi(x_n, u) \leq \varphi(x_n, x_n)$ for n sufficiently large. We claim that x_n is also a solution to problem (4.1). If not, there exists $y \in K \cap \operatorname{dom} \varphi(x_n, \cdot)$, $||y|| > ||x_n||$, such that $f(x_n, y) + \varphi(x_n, y) < \varphi(x_n, x_n)$. Since $||u|| < ||x_n||$, we have $f(x_n, u) + \varphi(x_n, u) = \varphi(x_n, x_n)$ by a previous inequality. In addition, we can find $z \in\ ]u, y[$ such that $||z|| < ||x_n||$. Thus, by the strict quasi-convexity of $f(x_n, \cdot) + \varphi(x_n, \cdot)$,

$$f(x_n, z) + \varphi(x_n, z) < \max\{\varphi(x_n, x_n), f(x_n, y) + \varphi(x_n, y)\} = \varphi(x_n, x_n),$$

which gives a contradiction since x_n is a solution to problem (4.1) restricted to K_n. Consequently, $f(x_n, y) + \varphi(x_n, y) \geq \varphi(x_n, x_n)$, proving that x_n is in fact a solution to (4.1). □

In order to obtain a nonempty and bounded solution set we introduce the following assumption:

($\tilde{f}_7$) Any sequence $x_n \in K$ with $||x_n|| \to +\infty$ such that

$$f(x_n, y) + \varphi(x_n, y) \geq \varphi(x_n, x_n) \quad \forall\, y \in K$$

admits a subsequence (x_{n_k}) such that $\frac{x_{n_k}}{||x_{n_k}||}$ converges strongly.

This "compactness condition" reduces to the particular case considered in section 3.

THEOREM 4.4. *Assume that assumptions* $(\tilde{f}_i)$, $i = 0, \ldots, 5$, *and* $(\tilde{f}_7)$ *hold. If, in addition* $R_0 = \{0\}$ (*or* $R_1 = \{0\}$), *then the solution set to problem* (4.1) *is nonempty and bounded.*

Proof. Let us consider the sequence (x_n) constructed in the proof of Theorem 4.3. If $\sup_n ||x_n|| = +\infty$, we proceed exactly as in part (ii) of the same theorem to conclude that $\frac{x_n}{||x_n||} \rightharpoonup v \in R_0$. Assumption $(\tilde{f}_7)$ implies that $\frac{x_n}{||x_n||} \to v$, thus $v \neq 0$, which gives a contradiction since $R_0 = \{0\}$. Therefore, $\sup_n ||x_n|| < +\infty$. Thus, we are in part (i) of the proof of Theorem 4.3, and hence, a solution is obtained as a weak limit point of (x_n). □

Remark 4.5. One can recognize from our Theorem 4.3 a variant of Ky Fan's minimax theorem [F2] in case $f \equiv 0$; if $\varphi \equiv 0$, a variant of the Browder–Minty theorem for variational inequalities, or more generally, a variant of the problem considered in [BNS]. The particular situation $\varphi(x, x) = 0$ has been studied in [BO] under the assumption of upper semicontinuity of $\varphi(\cdot, y)$ contrary to the lower semicontinuity of $\varphi(x, \cdot)$ imposed here, and under the convexity assumption. Related results can be also found in [HS] and references therein.

Remark 4.6. We notice that in case $f \equiv 0$ and $\varphi(x, y) = i_{Q(x)}(y)$, where i_c denotes the indicator function of the set C, assumption $(\tilde{f}_5)$ reduces to the upper semicontinuity of the set-valued map $Q : K \to K$. Thus, in case the set K is bounded, we recover the Kakutani fixed-point theorem as expected.

We can apply our previous results to the case $f(x, y) = \langle F(x) - x^*, g(y) - g(x)\rangle$ and $\varphi(x, y) = i_{Q(x)}(y)$, with F and g satisfying suitable assumptions.

Some other models to which our results apply can be found in [Au].

Acknowledgment. The author is very grateful to the referees for valuable suggestions.

REFERENCES

[AGT] S. ADLY, D. GOELEVEN, AND M. THÉRA, *Recession mappings and noncoercive variational inequalities*, Nonlinear Anal., 26 (1996), pp. 1573–1603.

[Au] J.P. AUBIN, *Mathematical Methods of Game and Economic Theory*, North-Holland, Amsterdam, 1979.

[A1] A. AUSLENDER, *Noncoercive optimization problems*, Math. Oper. Res., 21 (1996), pp. 769–782.

[A2] A. AUSLENDER, *Asymptotic analysis for penalty and barrier methods in variational inequalities*, SIAM J. Control Optim., 37 (1999), pp. 653–671.

[AC] A. AUSLENDER AND P. COUTAT, *Closed convex sets without rays and asymptotes*, Set-Valued Anal., 2 (1994), pp. 19–33.

[BBGT] C. BAIOCCHI, G. BUTTAZZO, F. GASTALDI, AND F. TOMARELLI, *General existence theorems for unilateral problems in continuum mechanics*, Arch. Rational Mech. Anal., 100 (1988), pp. 149–189.

[BO] E. BLUM AND W. OETTLI, *From optimization and variational inequalities to equilibrium problems*, Math. Student, 63 (1994), pp. 123–145.

[BNS] H. Brezis, L. Niremberg, and G. Stampacchia, *A remark on Ky Fan's minimax principle*, Boll. Un. Mat. Ital. (4), 6 (1972), pp. 293–300.

[BT] G. Buttazzo and F. Tomarelli, *Compatibility conditions for nonlinear Neuman problems*, Adv. Math., 89 (1991), pp. 127–143.

[CCR] O. Chadli, Z. Chbani, and H. Riahi, *Recession methods for equilibrium problems and applications to variational and hemivariational inequalities*, Discrete Contin. Dynam. Systems, 5 (1999), pp. 185–196.

[C] J.P. Crouzeix, *Pseudomonotone variational inequality problems: Existence of solutions*, Math. Programming, 78 (1997), pp. 305–314.

[DH] A. Daniilidis and N. Hadjisavvas, *Coercivity conditions and variational inequalities*, Math. Programming, 86 (1999), pp. 433–438.

[F1] K. Fan, *A generalization of Tychonoff's fixed point theorem*, Math. Ann., 142 (1961), pp. 305–310.

[F2] K. Fan, *A minimax inequality and applications*, in Inequality III, O. Shisha, ed., Academic Press, New York, 1972, pp. 103–113.

[F] F. Flores-Bazán, *Ideal, Weakly Efficient Solutions for Convex Vector Optimization Problems*, Technical Report 99-26, Departamento de Ingeniería Matemática, Universidad de Concepción, Chile, 1999.

[FS] F. Flores-Bazán and W. Sosa, *Existence of solutions for noncoercive pseudomonotone equilibrium problems*, Math. Oper. Res., submitted.

[GT] F. Gastaldi and F. Tomarelli, *Some remarks on nonlinear and noncoercive variational inequalities*, Boll. Un. Mat. Ital. B (7), 1 (1987), pp. 143–165.

[GY] J.S. Guo and J.C. Yao, *Variational inequalities with nonmonotone operators*, J. Optim. Theory Appl., 80 (1994), pp. 63–74.

[HP] 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.

[HS] N. Hadjisavvas and S. Schaible, *Quasimonotonicity and pseudomonotonicity in variational inequalities and equilibrium problems*, in Generalized Convexity, Generalized Monotonicity: Recent Results, J.-P. Crouzeix, J.-E. Martinez-Legaz, and M. Volle, eds., Kluwer, Dordrecht, 1998, pp. 257–275.

[JM] J.L. Joly and U. Mosco, *A propos de l'existence et de la régularité des solutions de certaines inéquations quasivariationnelles*, J. Funct. Anal., 34 (1979), pp. 107–137.

[K1] S. Karamardian, *Duality in mathematical programming*, J. Math. Anal. Appl., 20 (1967), pp. 344–358.

[K2] S. Karamardian, *Generalized complementarity problems*, J. Optim. Theory Appl., 8 (1971), pp. 161–168.

[KS] D. Kinderlehrer and G. Stampacchia, *An Introduction to Variational Inequalities and Their Applications*, Classics in Appl. Math. 31, SIAM, Philadelphia, 2000.

[O] W. Oettli, *A remark on vector-valued equilibria and generalized monotonicity*, Acta Math. Vietnam., 22 (1997), pp. 213–221.

[R] R.T. Rockafellar, *Convex Analysis*, Princeton University Press, Princeton, NJ, 1972.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 691–715

A PRIMAL-DUAL METHOD FOR LARGE-SCALE IMAGE RECONSTRUCTION IN EMISSION TOMOGRAPHY*

CALVIN A. JOHNSON† AND ARIELA SOFER‡

Abstract. In emission tomography, images can be reconstructed from a set of measured projections using a maximum likelihood (ML) criterion. In this paper, we present a primal-dual algorithm for large-scale three-dimensional image reconstruction. The primal-dual method is specialized to the ML reconstruction problem. The reconstruction problem is extremely large; in several of our data sets the Hessian of the objective function is the product of a 1.4 million by 63 million matrix and its scaled transpose. As such, we consider only approaches that are suitable for large-scale parallel computation. We apply a stabilization technique to the system of equations for computing the primal direction and demonstrate the need for stabilization when approximately solving the system using an early-terminated conjugate gradient iteration.

We demonstrate that the primal-dual method for this problem converges faster than the logarithmic barrier method and considerably faster than the expectation maximization algorithm. The use of extrapolation in conjunction with the primal-dual method further reduces the overall computation required to achieve convergence.

Key words. tomography, estimation, large-scale problems, parallel computation, applications of nonlinear programming, primal-dual methods

AMS subject classifications. 92C55, 62G05, 65Y05, 90C06, 90C90, 90C30, 49M05

PII. S1052623497330148

1. Introduction.

In this paper we consider the image reconstruction problem in emission tomography. This problem is encountered in the field of nuclear medicine, which is concerned with the study of organ function through radioactively labeled "tracer" compounds. The quantity of interest in this problem is the spatial concentration of radioactive emissions within the object under study. The quality of the reconstructed image can depend upon a number of factors including the number of emission events (i.e., counts) collected by the scanner and the method used to reconstruct the image. In studies that are characterized by poor counting statistics (that is, few counts), statistical reconstruction methods that model the Poisson nature of the emission process have been shown to improve image quality over traditional, nonstatistical reconstruction methods [26, 35, 57]. The low-count problem has generated considerable interest in the medical imaging community because low radiotracer doses and short scanning durations are highly desirable.

The estimation of emission density in an organ is an inherently three-dimensional (3-D) process. Volume, or 3-D acquisition, improves the counting statistics compared with two-dimensional (2-D) acquisition (in which axially oblique coincidences are either physically or electronically blocked from detection) but increases the problem

*Received by the editors November 13, 1997; accepted for publication (in revised form) July 31, 2000; published electronically November 15, 2000. This work was performed by an employee of the U.S. Government or under U.S. Government contract. The U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U.S. Government purposes. Copyright is owned by SIAM to the extent not limited by these rights.

http://www.siam.org/journals/siopt/11-3/33014.html

†Center for Information Technology, National Institutes of Health, Bethesda, MD 20892-5624 (johnson@mail.nih.gov).

‡Department of Systems Engineering and Operations Research, George Mason University, Fairfax, VA 22030-4444 (asofer@gmu.edu). This author was partly supported by National Science Foundation grants DMI-9414355 and DMI-9800544.

size considerably. Since the 3-D problem may involve image and measurement vectors with millions of elements, the amount of computation required to perform 3-D statistical reconstructions can be quite substantial. In our computational studies, for example, the larger reconstructions consist of 1.4 million image variables which are reconstructed from a measurement vector with 63 million elements. As such, it is important to use reconstruction methods that converge rapidly. The statistical image reconstruction problem can be posed as a constrained nonlinear optimization problem. In this paper we present a primal-dual method for performing statistical 3-D reconstructions in emission tomography that has been specialized to the intricacies of the application. We demonstrate the rapid convergence of our primal-dual method in computational studies on low-count, 3-D positron emission tomography (PET) data.

This paper is organized as follows. In section 2 we present the statistical model and develop the objective function. Section 3 reviews the expectation maximization (EM) method for maximum likelihood (ML) reconstruction. In section 4 we develop a primal-dual method for ML reconstruction and discuss initialization, stabilization, and extrapolation enhancements. Computational tests comparing the primal-dual results to a logarithmic barrier approach and the EM method on small animal data are presented in section 5. Some concluding remarks are made in section 6.

2. Statistical model and objective function. We begin our discussion by forming a finite parameter space for the image estimates, as is customary [20]. Consider the situation depicted in Figure 2.1, where a grid of boxes or *voxels* has been imposed over the emitting object. (For simplicity, the figure is depicted in 2-D; the concept is readily extended to 3-D.) Given a set of measurements along lines of coincidence, we seek to estimate $x_i = E\{\xi_i\}$, $i = 1, \ldots, n$, the expected number of counts emitted from voxel i. Let X_i be the number of radioactive events emitted from voxel i. X_i are assumed to be independent Poisson-distributed random variables with mean x_i [53]. A *system matrix* $C \in \Re^{n \times N}$ is used to model a number of physical effects including spatially dependent resolution and attenuation. The elements $C_{i,j}$ of the system matrix represent the probability that an event emitted from voxel i will be detected by detector pair (coincidence line). The number of events emitted from voxel i and detected at coincidence line j is therefore $\mathbf{\Xi}_{i,j} = \xi_i C_{i,j}$, and $\mathbf{\Xi}_{i,j}$ are also independent Poisson variables. The measurements y_j are thus realizations of sums of independent Poisson variables $\mathbf{y}_j = \sum_i \mathbf{\Xi}_{i,j}$ with means $\hat{y}_j = E\{\mathbf{y}_j\} = \sum_i C_{i,j} x_i$. The above is a considerably simplified model of the actual measurement process; for further discussion on its validity to the present situation, see [24].

Given our simplified Poisson model, the likelihood may be written as

$$P\{y|x\} = \prod_j \frac{e^{-\hat{y}_j} \hat{y}_j^{y_j}}{y_j!} = \prod_j \frac{e^{-\sum_i C_{i,j} x_i} \left(\sum_i C_{i,j} x_i\right)^{y_j}}{y_j!}.$$

The ML objective function is formed by taking the log likelihood

$$\log P\{y|x\} = \sum_j \left(-\sum_i C_{i,j} x_i + y_j \log \sum_i C_{i,j} x_i - \log (y_j)! \right).$$

Ignoring the constant term, we define our objective function $f_{ML}(x)$ as

$$f_{ML}(x) = \sum_j \left(-\left(C^T x\right)_j + y_j \log \left(C^T x\right)_j \right) = -q^T x + \sum_j y_j \log \left(C^T x\right)_j, \tag{2.1}$$

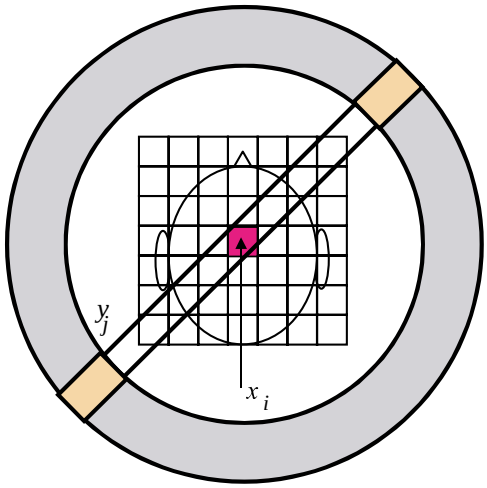


FIG. 2.1. *Relationship between estimate x_i and measurement y_j. Shown here is the case of PET, where emission-count measurements are taken along coincidence lines from pairs of detectors. A finite parameter space is formed by imposing a grid of voxels over the emitting region. The estimate of the expected emission intensity within voxel i is x_i.*

where $q = Ce_N \in \Re^n$ and $e_N \in \Re^N$ is a vector of 1's, so that q is the sum of the columns of C (which need not necessarily be 1). Defining

$$\hat{y} = C^T x$$

to be a *forward transformation*, we can write the gradient and Hessian of the objective function, respectively, as

$$\nabla f_{ML}(x) = -q + C\hat{Y}^{-1}y, \tag{2.2}$$

$$\nabla^2 f_{ML}(x) = -C\,Y\hat{Y}^{-2}C^T, \tag{2.3}$$

where $Y = \operatorname{diag}(y)$ and $\hat{Y} = \operatorname{diag}(\hat{y})$. The Hessian is negative semidefinite (since $y_j/\hat{y}_j \geq 0\ \forall j$), so the objective function (2.1) is concave. Thus, any local maximum will also be a global maximum.

Equation (2.2) sheds some insight into the computational costs associated with maximizing the objective function. Given a current solution estimate x^k, computing the gradient requires first computing a forward transformation $\hat{y}^k = C^T x^k$ and then computing a *backward transformation* $C\hat{Y}_k^{-1}y$ from the forward transformation. The costs of performing the forward transformation and backward transformation are similar and together dominate the computation associated with iterative reconstruction methods, especially in large scale. We shall revisit this computational structure, which is common to all iterative reconstruction methods.

Since the underlying activity distribution is nonnegative, the ML reconstruction problem is a constrained optimization problem with lower-bound constraints:

$$\begin{array}{ll} \text{maximize} & f_{ML}(x) \\ \text{subject to} & x \geq 0. \end{array} \tag{2.4}$$

The ML objective function has a finite maximum and compact level sets on $x \geq 0$ [36].

2.1. Maximum a posteriori reconstruction. Without regularity conditions on x, estimating the spatial emission distribution is a statistically ill-posed problem [7, 33]. The fully converged ML reconstruction, being dominated by noise and edge artifact, is not generally of biomedical interest [55]. Regularization can be included in the objective function by introducing a Bayesian formulation [20, 37]. Given prior probabilities $P\{x\}$ and $P\{y\}$ for the image and measurements, respectively, we define the posterior probability

$$P\{x|y\} = \frac{P\{y|x\}\, P\{x\}}{P\{y\}}.$$

The estimate of x is then obtained by maximizing the posterior probability $P\{x|y\}$.

A common choice for the image prior is the Gibbs distribution $P\{x\} = e^{-\gamma R(x)}$, although other priors (e.g., Gaussian, gamma) have been investigated [37, 39]. The popularity of Gibbs priors stems in part from their ability to capture the local correlation property of images [19]. The energy function R is defined as a sum of "potential" functions designed to discourage nonsmoothness in a neighborhood

$$R(x) = \frac{1}{2} \sum_i \sum_{l \in \mathcal{N}_i} V_{i,l}(x_i, x_l),$$

where $\mathcal{N}_i$ denotes the neighborhood of voxel i. In order to maintain concavity and twice continuous differentiability in the objective function, the potential function $V_{i,l}$ is chosen to be convex with continuous first and second derivatives. In our studies we have used the potential function $V_{i,l}(x_i, x_l) = V(x_i - x_l)$, where

$$V(z) = \delta^2 \left(\left| \frac{z}{\delta} \right| - \log\left(1 + \left| \frac{z}{\delta} \right| \right) \right) \tag{2.5}$$

and δ is a shaping constant that we typically set to 1 [38].

For maximum a posteriori (MAP) reconstructions, the objective function is the log-posterior likelihood $\log P\{x|y\}$. Ignoring a constant, our objective function becomes

$$f_{MAP}(x) = f_{ML}(x) - \gamma R(x). \tag{2.6}$$

The MAP reconstruction problem can also be posed as a constrained optimization problem

$$\begin{array}{ll} \text{maximize} & f_{MAP}(x) \\ \text{subject to} & x \geq 0. \end{array} \tag{2.7}$$

We note for future reference the following:

$$\begin{aligned} \nabla f_{MAP}(x) &= \nabla f_{ML}(x) - \gamma \nabla R(x) = -q + C\hat{Y}^{-1} y - \gamma \nabla R(x), \\ \nabla^2 f_{MAP}(x) &= \nabla^2 f_{ML}(x) - \gamma \nabla^2 R(x) = -C Y \hat{Y}^{-2} C^T - \gamma \nabla^2 R(x). \end{aligned}$$

Although the function R (with the potential function (2.5)) is concave it is not strictly concave. Since $v^T \nabla^2 R(x) v = 0$ only for vectors v that are a scalar multiple of the unit vector e_N, and since $e_N^T \nabla^2 f_{ML}(x) e_N < 0$, it follows that $\nabla^2 f_{MAP}(x)$ is negative definite and that f_{MAP} is strictly concave [38]. In addition, f_{MAP} has a finite maximum and bounded level sets on $x \geq 0$ [37].

2.2. The optimization problem. For convenience of notation, let us pose the reconstruction problem as a constrained *minimization* problem:

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & x \geq 0, \end{array} \tag{2.8}$$

where $f(x) = -f_{MAP}(x)$, with $\gamma \geq 0$. The case $\gamma = 0$ corresponds to the unregularized ML; in general we shall be more interested in the fully converged MAP where $\gamma > 0$.

The Karush–Kuhn–Tucker (KKT) first-order necessary conditions for optimality of (2.8) at a point x are existence of Lagrange multipliers λ so that

$$\nabla \ell(x, \lambda) = \nabla f(x) - \lambda = 0, \tag{2.9}$$

$$\lambda_i x_i = 0, \quad i = 1, \ldots, n, \tag{2.10}$$

$$x,\ \lambda \geq 0, \tag{2.11}$$

where $\ell(x, \lambda) = f(x) - \lambda^T x$ is the Lagrangian function. Due to the strict convexity of f, the second-order sufficiency conditions are satisfied, and x is the unique minimizer of f.

3. The EM algorithm. The EM method, as presented by Dempster, Laird, and Rubin [8] for ML estimation, is an iterative algorithm for computing ML estimates when the measurements are viewed as incomplete data. Shepp and Vardi [53] and Lange and Carson [36] applied the EM method to emission and transmission tomography problems, respectively. The EM algorithm has been proven to converge to an optimal solution of (2.4) [36, 56].

The EM algorithm for emission tomography can be derived [56, 27] from the optimality conditions for the reconstruction problem. For the unregularized problem, (2.9) can be written as

$$q - C\hat{Y}^{-1}y - \lambda = 0.$$

Let $X = \text{diag}(x)$, and $Q = \text{diag}(q)$. Premultiplication by X, and utilizing the complementary slackness condition, yields

$$XQe_N = XC\hat{Y}^{-1}y,$$

or since $QX = XQ$

$$x = XQ^{-1}C\hat{Y}^{-1}y \equiv \mathcal{M}(x).$$

Applying a fixed-point algorithm $x^{k+1} = \mathcal{M}\left(x^k\right)$ to the above equation yields the EM update equation

$$x^{k+1} = X_k Q^{-1} C \hat{Y}_k^{-1} y = x^k - X_k Q^{-1} \nabla f(x_k), \tag{3.1}$$

where x^k is the current image estimate, $X_k = \text{diag}(x^k)$, $\hat{y}^k = C^T x^k$, and $\hat{Y}_k = \text{diag}(\hat{y}^k)$. Given a positive initial solution $x^0 > 0$, the algorithm maintains nonnegativity at every iteration and converges to a fixed point $x^\infty = \mathcal{M}(x^\infty)$, which is an optimal solution of (2.4). The asymptotic rate of convergence is governed by the spectral radius of $\nabla\mathcal{M}(x^\infty)$, which is typically very close to unity. In one example using reasonable assumptions about the scanner geometry, the lower bound of the

spectral radius was calculated to be .99938 [17]. Indeed, EM has been observed to converge very slowly, especially close to the optimal solution. The slow convergence of the EM algorithm has limited its clinical applicability. The cost of one EM iteration is equivalent to the cost of one gradient calculation.

In MAP-EM, the presence of the regularizing term in (2.6) precludes a closed-form update equation such as (3.1) for ML-EM. We mention two algorithms that are commonly used for MAP-EM reconstructions: the "one step late" (OSL) algorithm and DePierro's algorithm. Green's OSL algorithm approximates $R(x)$ with the constant $R(x^k)$, thereby permitting a closed-form approximated update [16, 17]

$$x^{k+1} = X_k S^{-1} C \hat{Y}_k^{-1} y = x^k - X_k S_k^{-1} \nabla f(x_k), \tag{3.2}$$

where $S_k = \operatorname{diag}(q + \gamma \nabla(R(x^k)))$. OSL converges to the MAP solution provided that $\gamma \leq \bar{\gamma}$, where $\bar{\gamma}$ is an upper threshold for the prior strength. DePierro's algorithm is a "true" MAP-EM implementation that substitutes the convex function $R(x)$ with a separable, convex, and twice continuously differentiable function $R(x, x^k) \geq R(x)$, so that separable maximizations can be performed on the variables [9, 10]. Regularization improves the convergence rate of EM, with larger prior strengths resulting in lower spectral radii. However, for reasonable prior strengths (mild to moderate smoothing), the convergence rates of OSL and DePierro's algorithm are still quite close to unity.

The EM update formula on the right-hand side of (3.1) follows Kaufman [27], who was the first to pose the EM algorithm as an optimization algorithm (namely, a scaled steepest-ascent method). This representation allows for the inclusion of a line search [27, 28] to accelerate the method's performance. Likewise, the MAP update (3.2) can be enhanced by a line search.

Several other approaches for solving the ML estimation problem have been proposed. These include preconditioned conjugate gradient (CG) techniques [27, 28, 34, 42] or truncated-Newton methods [27, 28]. The nonnegativity constraints are maintained either by limiting the step length or by using a bending line search. The paper [44] explores active set methods, while [43] enforces nonnegativity via a quadratic penalty in the objective. In other work [29, 30, 31] a penalized least-squares objective is used instead of the ML. These problems are solved by a preconditioned CG and use specialized techniques to drive the complementary slackness to zero.

There is considerable debate within the PET community regarding the appropriate model for reconstruction. It has long been observed that the unregularized ML estimator gives grainy images. However if the EM algorithm is stopped early, the resulting solution often produces images of acceptable quality. For this reason some researchers argue that early termination is a form of smoothing and that no regularization is needed. Proponents of MAP argue that the approach allows the user to control the amount of regularization through the parameter, and that the regularized objective function is better conditioned. In either case, it has been observed that EM-type algorithms may lead to nonuniform convergence. In particular, the algorithms may converge slowly in "cold spots" (regions of low activity within regions of activity) and in areas of isolated activity within cold spots. The use of an interior-point algorithm offers the hope of more uniform convergence.

4. A primal-dual approach. The drawbacks of the EM algorithm motivate our investigation into interior-point approaches for the ML and MAP reconstruction problems. As is clear from (2.1), the objective function can be undefined outside the feasible region $x \geq 0$. Thus the ML and MAP reconstruction problems would appear to be "natural" candidates for interior-point algorithms. The reconstruction problem

is especially suited to an interior-point approach because its output is a grayscale image. Whether a particular value is exactly "zero" or just very close to zero is immaterial. Slight inaccuracies below the grayscale threshold are inconsequential; obtaining an image rapidly is a necessity.

Primal-dual methods have enjoyed considerable success in linear programming [18, 32, 40] and have recently been proposed for nonlinear programming [5, 13, 41]. Although they are closely related to the logarithmic barrier method [12, 58], primal-dual methods may pose some advantages. In the logarithmic barrier method, the Lagrange multiplier estimates may be inaccurate when the primal variables are not close to the barrier trajectory [11]. Primal-dual methods offer the potential of improved "centering" over barrier methods. Given the size of the current problem, the developments presented here must be suitable for large-scale parallel computation.

In a manner similar to classical barrier methods, primal-dual methods attempt to follow the "barrier trajectory," a smooth trajectory characterized by a *barrier parameter* $\mu > 0$ [12]. The points $(x(\mu), \lambda(\mu))$ along the trajectory satisfy a perturbed version of the KKT conditions:

$$\nabla f\left(x\left(\mu\right)\right) - \lambda\left(\mu\right) = 0, \tag{4.1}$$

$$\lambda_i\left(\mu\right) x_i\left(\mu\right) = \mu, \quad i = 1, \ldots, n, \tag{4.2}$$

$$x\left(\mu\right),\ \lambda\left(\mu\right) > 0. \tag{4.3}$$

Defining $X = \operatorname{diag}\{x_i, i = 1, \ldots, n\}$ and $\Lambda = \operatorname{diag}\{\lambda_i, i = 1, \ldots, n\}$, our method maintains (4.3) while attempting to solve (4.1), (4.2), that is,

$$\begin{bmatrix} \nabla f\left(x\right) - \lambda \\ \Lambda X e_n - \mu e_n \end{bmatrix} = 0. \tag{4.4}$$

Given the point (x^k, λ^k) and the barrier parameter $\mu = \mu_k$, the search direction $p = [p_x^T,\ p_\lambda^T]^T$ prescribed by Newton's method satisfies the "unsymmetric" primal-dual equations [41]

$$\begin{bmatrix} \nabla^2 f\left(x^k\right) & -I \\ \Lambda_k & X_k \end{bmatrix} \begin{bmatrix} p_x \\ p_\lambda \end{bmatrix} = - \begin{bmatrix} \nabla f\left(x^k\right) - \lambda^k \\ \Lambda_k X_k e_n - \mu_k e_n \end{bmatrix}. \tag{4.5}$$

Elimination of the $(1, 2)$ block of the matrix in (4.5) yields the *reduced system*

$$M_k p_x = -\nabla f(x^k) + \mu_k X_k^{-1} e_n, \tag{4.6}$$

$$p_\lambda = -\lambda^k - X_k^{-1} \Lambda_k p_x + \mu_k X_k^{-1} e_n, \tag{4.7}$$

where the "condensed" primal-dual matrix is given by

$$M_k = \nabla^2 f(x^k) + X_k^{-1} \Lambda_k. \tag{4.8}$$

We have implemented an algorithm in which the primal and dual variables are permitted to take separate step lengths:

$$\begin{bmatrix} x^{k+1} \\ \lambda^{k+1} \end{bmatrix} = \begin{bmatrix} x^k + \alpha_x p_x \\ \lambda^k + \alpha_\lambda p_\lambda \end{bmatrix}.$$

The primal step length α_x is chosen to ensure sufficient decrease in the merit function

$$F\left(x, \mu\right) = f\left(x\right) - \mu \sum_i \log x_i.$$

Observe that $F(x,\mu)$ is simply the logarithmic barrier function and that

$$\nabla_x F(x,\mu) = \nabla f(x) - \mu X^{-1} e_n$$

is identical to the right-hand side of (4.6) for $\mu = \mu_k$ and $x = x^k$. The unconstrained minimizer $x(\mu)$ of $F(x,\mu)$ satisfies the perturbed KKT conditions (4.1)–(4.3) with corresponding multiplier $\lambda_i(\mu) = \mu/x_i(\mu)$, $i = 1,\ldots,n$. Furthermore, the solution of the condensed primal-dual Newton equation (4.6) is guaranteed to be a descent direction of the merit function for $\mu > 0$, since

$$(\nabla_x F(x,\mu))^T p_x = p_x^T M p_x,$$

and M is positive definite. We shall discuss in further detail the computation of the primal search direction and step length.

The formula for the dual step length follows a suggestion by Conn, Gould, and Toint (CGT) [5]. If $\lambda^{(k)} + p_\lambda$ lies componentwise in the interval

$$\lambda^k + p_\lambda \in \left[\zeta^{-1}\min\left(e_n, \lambda^k, \mu_k X_{k+1}^{-1} e_n\right), \max\left(\zeta e_n, \lambda^k, \zeta\mu_k^{-1} e_n, \zeta\mu_k X_{k+1}^{-1} e_n\right)\right] \tag{4.9}$$

(where ζ is a constant parameter that we have set to 100), then $\lambda^{k+1} = \lambda^k + p_\lambda$; otherwise find $0 < \alpha_\lambda < 1$ such that $\lambda^{k+1} = \lambda^k + \alpha_\lambda p_\lambda$ minimizes

$$\left\|\Lambda_{k+1} X_{k+1} e_n - \mu_k e_n\right\|_\infty \tag{4.10}$$

subject to λ^{k+1} being in the interval (4.9). These conditions on the dual step might appear at first glance to be overly restrictive but are actually designed to give maximum flexibility in the choice of λ^{k+1}. CGT use these bounds on λ and nonsingularity of M to prove that, for any fixed parameter value $\bar{\mu}$, the minimization of $F(x,\bar{\mu})$ must be successful, that is, eventually a solution is found that satisfies the perturbed KKT conditions (4.1)–(4.3).

In general it is neither necessary nor desirable to reach full subproblem convergence. Rather, we have implemented a "short-step" algorithm in which only one primal-dual step is usually needed before adjusting μ. Setting the barrier parameter μ is an important consideration in primal-dual algorithms and has a strong influence on the convergence rate. A reduction in μ_k is performed whenever the "μ-criticality" conditions [5, 54] are satisfied:

$$\frac{\left(\lambda^{k+1}\right)^T x^{k+1}}{n} \le \vartheta^C \mu_k, \tag{4.11}$$

$$\left\|\nabla f\left(x^{k+1}\right) - \lambda^{k+1}\right\|_\infty \le \vartheta^{DF} \mu_k, \tag{4.12}$$

where ϑ^C and ϑ^{DF} are constant parameters. If the above conditions are satisfied, the barrier parameter is reduced according to

$$\mu_{k+1} = \frac{\left(\lambda^{k+1}\right)^T x^{k+1}}{n\rho}, \tag{4.13}$$

where ρ is a constant parameter such that

$$\frac{\vartheta^C}{\rho} < 1. \tag{4.14}$$

A consequence of (4.14) is that μ_k cannot increase. Furthermore, since the minimization of $F(x, \mu)$ must be successful, a μ-critical solution (a weaker requirement) must eventually be found. Thus it is impossible for μ_k to be nondecreasing. Using this argument, CGT prove that the algorithm must converge to a KKT solution [5].

In practice we find that both the primal and dual direction vectors are well scaled, and that α_x and α_λ are both typically close to 1. By far the most costly operations are computing the primal direction p_x and updating the gradient $\nabla F(x)$, as we shall explore. In contrast, the costs of the line search for the primal step length, the computation of the dual search direction (4.7), and the dual line search (4.9) are relatively insignificant. From empirical evidence in our computational studies, we have found that a "short-step" algorithm with gradual reduction in μ achieves the fastest convergence to the KKT conditions. Specifically, we define $\vartheta^C = 1.9$, $\rho = 2$, and $\vartheta^{DF} = 100$. These parameter values enable the μ-critical conditions to be met after only one primal-dual step for most subproblems.

4.1. Computing the primal direction. For large problems, factoring the condensed primal-dual matrix M or even forming the Hessian $\nabla^2 f(x)$ would be prohibitive due to the size of the matrix (376,000 $\times$ 376,000 for even the smaller reconstructions being considered in this paper) and the enormous amount of computation that would be required. Thus we must consider methods for approximating the Newton direction in (4.6). The approach we have successfully applied to this problem is motivated by the *truncated-Newton* [6] method of unconstrained optimization. The search direction is an approximate or *truncated* solution to the Newton equations [47, 49]

$$Mp_x \approx -\nabla F\left(x\right). \tag{4.15}$$

An early-terminated CG iteration [21] is used to obtain an approximate solution to (4.15).

An equivalent statement of (4.15) is that we seek to find the direction p_x that approximately minimizes the quadratic $Q(p_x) = \frac{1}{2} p_x^T M p_x + \nabla F(x) p_x$. A reasonable and effective truncation point for (4.15), based on the monotonicity of $Q(p_x)$, is proposed in [48]; the CG is terminated at subiteration l if

$$\frac{Q\left(p_x^l\right) - Q\left(p_x^{l-1}\right)}{Q\left(p_x^l\right)} \leq \frac{1}{2l}. \tag{4.16}$$

The CG termination rule (4.16) has been an important component of the reconstruction software in that it consistently yields a well-scaled primal direction vector as long as $\mu \geq \mu_s$, where μ_s is a threshold value below which stabilization is required. (We shall discuss the $\mu < \mu_s$ case in section 4.2.)

The CG method does not require storage of the Hessian or condensed primal-dual matrix but rather only application of matrix-vector products. From (2.3) we can write the first term of the matrix-vector product

$$\nabla^2 f_{ML}\left(x\right) v = -C\, Y \hat{Y}^{-2} C^T v \tag{4.17}$$

for an arbitrary vector $v \in \Re^n$. Computationally, (4.17) consists of a forward transformation ($C^T v$) followed by a diagonal scaling ($\hat{y}$ is already available from the computation of $\nabla f(x)$), followed by a backward transformation (premultiplication by C). To be explicit, recalling (4.8), we have

$$Mv = C\, Y \hat{Y}^{-2} C^T v + \gamma \nabla^2 R\left(x\right) v + X^{-1} \Lambda v,$$

where $\nabla^2 R(x)v$ can be computed exactly without incurring significant computational expense. The forward-and-back-transformation operation in (4.17) dominates the computational cost of a CG iteration. This operation is computationally similar to computing the gradient, or one EM iteration.

Some authors advocate solving simultaneously for p_x and p_λ, using the full unsymmetric primal-dual equations (4.5) or an equivalent symmetrized system [13, 14, 52, 61]. The unsymmetric primal-dual matrix in particular remains nonsingular, and its condition number remains bounded as $\mu \to 0$ [12, 41], when the standard conditions of a constraint qualification, strict complementarity, and the second-order sufficient conditions are satisfied at the solution. In our application, due to the size of our problem, we must use an iterative method. We believe that solving a symmetric system via a symmetric solver such as the CG would be more efficient than solving the full unsymmetric system via an unsymmetric iterative solver such as GMRES (even though our symmetric system is ill-conditioned), since the amount of work and storage required per iteration in GMRES increases linearly with the iteration count. An advantage of using the condensed system (4.6)–(4.7) is that although the primal search direction is computed inexactly, the equation for maintaining complementarity (4.7) is maintained. In practice we find that the resulting primal and dual direction vectors are both well scaled, and that α_x and α_λ are typically close to 1.

4.1.1. Preconditioning. The use of a preconditioner with the CG is essential for a competitive algorithm. Since every CG subiteration is as costly as a gradient evaluation or EM iteration, it is highly desirable to obtain a quality direction vector in as few CG iterations per subproblem as possible. We have investigated a number of preconditioners, including FFT-based preconditioners that model the approximately Toeplitz-block-Toeplitz nature of CC^T with a circulant-block-circulant approximation [2, 3], high-pass filter approximations to the FFT-based preconditioner [4], the EM preconditioner XQ^{-1} [34], the exact diagonal of M, and diagonal Hessian approximations [46].

Of the above preconditioners, by far the best performing was the exact diagonal of M, which can be computed at reasonable cost:

$$M_{i,i} = \sum_j \frac{C_{i,j}^2 y_j}{\hat{y}_j^2} + \gamma \frac{\partial^2}{\partial x_i^2} R(x) + \frac{\lambda_i}{x_i}. \tag{4.18}$$

Note that the first right-hand-side term in (4.18) is similar in form to a backward transformation, although a bit more expensive due to the squaring operations. We have found that the preconditioned CG method using an exact diagonal preconditioner in the form of (4.18) almost always requires using fewer than 10 iterations to achieve (4.16), regardless of the size of the problem. In many cases, only 3 or 4 CG iterations are required. Moreover, the directions produced using an exact diagonal preconditioner are well scaled (usually resulting in primal step sizes of near 1), and lead to rapid descent.

In contrast, the other preconditioners did not perform well. Already in the initial subproblems they tended to yield a poorly scaled search direction, which in turn resulted in small step lengths. Subsequent calls to the CG suffered further from this problem, and the algorithm made little progress. This behavior was particularly surprising for the block-circulant FFT-based preconditioners. These preconditioners perform very well in other reconstruction methods, especially in least-squares methods where the block-circulant approximation is well matched to the Hessian structure.

We were motivated to try them for our problem because the ML Hessian is almost block circulant. But because of the strong diagonal component in M and its spatially variant dependence on y, $\hat{y}$, x, and λ, shift-invariant Toeplitz models of M yield a poor approximation in our method.

4.1.2. Line search. For ML and MAP reconstructions, knowledge of the structure of the objective function can lead to a substantial reduction in the cost of implementing a line search over a more naive approach. Specifically, after the search direction p_x has been found, and once a forward transformation $\hat{w} = C^T p_x \in \Re^N$ has been computed, it is possible to compute the objective function and first and second directional derivative values at the trial points $(x^k + \alpha p_x)$ at nearly negligible cost. To see this, note that $\hat{y}^{k+1} = \hat{y}^k + \alpha\hat{w}$, and therefore [27, 28]

$$f\left(x^k + \alpha_x p_x\right) = q^T x^k + \alpha_x q^T p_x - \sum_j y_j \log\left(\hat{y}_j^k + \alpha_x \hat{w}_j\right) + \gamma R\left(x^k + \alpha_x p_x\right).$$

Similar expressions exist for the directional first and second derivatives [24].

After the initial forward transformation to compute $\hat{w}$, no further forward- or back-transformation operations are required during the line search at any of the trial points. The forward transformation $\hat{w}$ can be reused, so that only one backward transformation is subsequently required to update the gradient. The above observations and the well-behaved convex nature of the objective function have permitted us to implement a highly accurate but low-cost Newton line search. Due to the low cost of each step we have chosen a relatively strict tolerance of 0.05 on the Wolfe condition for termination of the line search. We find this line search technique to be highly effective and, in no small part, responsible for the positive results we report.

4.2. Stabilization. A well-known property of the Hessian of the primal barrier function is its increasingly ill-conditioned nature as $\mu \to 0$ [45]. Analogous results hold for the condensed primal-dual matrix: as the solution is approached the matrix becomes increasingly ill-conditioned. (For a detailed analysis, see the paper by Wright [60].)

In [50], Nash and Sofer developed an approximation to the Newton direction for the logarithmic barrier that avoids the structural ill-conditioning of the barrier Hessian and is suitable for large-scale problems. The direction is the sum of two vectors, one in the null space of the Jacobian of the active constraints, and the other orthogonal to it. The associated decoupling is based on a prediction of the binding set at the solution.

We have recently adapted this approximation to the condensed Newton equations arising in primal-dual methods. Although our derivation is valid for general nonlinear constraints, we present it here for the special case of bound constraints in the context of (4.6).

We will assume in the following that strict complementarity holds at the solution, that is, $\lambda_i^* > 0$ if $x_i^* > 0$. Define $\mathcal{J} = \{i : x_i^* = 0\}$ to be the index set of binding constraints at the solution, and $\hat{n}$ to be the number of binding constraints at the solution. We will assume that $0 < \hat{n} < n$, as is always the case in reconstructions of practical interest. Define $\mathcal{I} = \{i : x_i^* > 0\}$ the set of nonbinding constraints. Let $x^{\mathcal{I}}$ be the subvector of variables that are positive at the optimal solution, and $x^{\mathcal{J}}$ the subvector of variables that are zero at the optimal solution. Assume also that the

variables are ordered so that the positive variables are first, i.e.,

$$x = \begin{bmatrix} x^{\mathcal{I}} \\ x^{\mathcal{J}} \end{bmatrix}.$$

The Hessian of the objective function $H = \nabla^2 f(x)$ will then be similarly partitioned, as will the condensed primal-dual matrix

$$M = \begin{bmatrix} M_{\mathcal{I},\mathcal{I}} & M_{\mathcal{I},\mathcal{J}} \\ M_{\mathcal{I},\mathcal{J}}^T & M_{\mathcal{J},\mathcal{J}} \end{bmatrix} = \begin{bmatrix} H_{\mathcal{I},\mathcal{I}} + X_{\mathcal{I}}^{-1}\Lambda_{\mathcal{I}} & H_{\mathcal{I},\mathcal{J}} \\ H_{\mathcal{I},\mathcal{J}}^T & H_{\mathcal{J},\mathcal{J}} + X_{\mathcal{J}}^{-1}\Lambda_{\mathcal{J}} \end{bmatrix},$$

where $X_{\mathcal{I}}$, $X_{\mathcal{J}}$, $\Lambda_{\mathcal{I}}$, and $\Lambda_{\mathcal{J}}$ are the diagonal matrices of the associated components of x and λ.

We will assume that the sequence of iterates (x, λ) generated by the primal-dual satisfies the following properties, when μ is sufficiently small:

$$\begin{aligned} x_i &= \Theta(1), \quad \lambda_i = \Theta(\mu), \quad i \in \mathcal{I}, \\ x_i &= \Theta(\mu), \quad \lambda_i = \Theta(1), \quad i \in \mathcal{J}. \end{aligned}$$

Here we define $\tau = \Theta(\mu)$ if there exist constants $0 < \kappa_l < \kappa_u$ so that $\kappa_l \mu \le \|\tau\| \le \kappa_u \mu$ for all sufficiently small $\mu > 0$. We say that a vector or matrix is $\Theta(\mu)$ if its norm is $\Theta(\mu)$. We also define $\tau = O(\mu)$ if there exists some positive constant κ_u so that $\|\tau\| \le \kappa_u \mu$ for all sufficiently small $\mu > 0$.

We will also assume that near the solution the Hessian is reasonably well conditioned, so that $H = O(1)$. Now the diagonal terms of $M_{\mathcal{J},\mathcal{J}}$ are $O(1/\mu)$ and become unbounded as $\mu \to 0$. In contrast, the diagonal terms of $M_{\mathcal{I},\mathcal{I}}$ differ from those of the reduced Hessian $H_{\mathcal{I},\mathcal{I}}$ by $O(\mu)$, and the condition of $M_{\mathcal{I},\mathcal{I}}$ thus reflects that of the constrained problem. The condensed primal-dual matrix M can then be shown to have $\hat{n}$ "large" eigenvalues of magnitude $\Theta(1/\mu)$ and $n - \hat{n}$ "small" eigenvalues that differ from those of $H_{\mathcal{I},\mathcal{I}}$ by $O(\mu)$ and have magnitude $\Theta(1)$. The condensed primal-dual matrix thus suffers from the same structured ill-conditioning as the barrier Hessian.

For small values of μ we propose approximating the primal Newton direction p_x, by a direction $\tilde{p}_x$, whose null- and range-space components are computed as follows:

$$(M_{\mathcal{I},\mathcal{I}})\, \tilde{p}_x^{\mathcal{I}} = -\left(\nabla F^{\mathcal{I}} - M_{\mathcal{I},\mathcal{J}} X_{\mathcal{J}} \Lambda_{\mathcal{J}}^{-1} \nabla F^{\mathcal{J}}\right), \tag{4.19}$$

$$\tilde{p}_x^{\mathcal{J}} = -X_{\mathcal{J}} \Lambda_{\mathcal{J}}^{-1} \nabla F^{\mathcal{J}}. \tag{4.20}$$

The system for computing the component $\tilde{p}_x^{\mathcal{I}}$ involves the well-conditioned matrix $M_{\mathcal{I},\mathcal{I}}$ and can be solved exactly or inexactly via the CG method. The computation of $\tilde{p}_x^{\mathcal{J}}$ is straightforward. Thus, the ill-conditioning of the condensed primal-dual is avoided. We will show now that under the assumptions above, $\tilde{p}_x - p_x = O(\mu^2)$, so that the accuracy of the approximation increases as the solution is approached and the potential harm from ill-conditioning increases.

Using the well-known formula for the inverse of a partitioned matrix (see, e.g., [51, 61]), it follows that

$$\begin{aligned} \tilde{p}_x^{\mathcal{I}} - p_x^{\mathcal{I}} &= (M_{\mathcal{I},\mathcal{I}})^{-1} \left(M_{\mathcal{I},\mathcal{J}} G^{-1} M_{\mathcal{I},\mathcal{J}}^T M_{\mathcal{I},\mathcal{I}}^{-1} \nabla F^{\mathcal{I}} + M_{\mathcal{I},\mathcal{J}} (X_{\mathcal{J}} \Lambda_{\mathcal{J}}^{-1} - G^{-1}) \nabla F^{\mathcal{J}} \right), \\ \tilde{p}_x^{\mathcal{J}} - p_x^{\mathcal{J}} &= -G^{-1} M_{\mathcal{I},\mathcal{J}}^T M_{\mathcal{I},\mathcal{I}}^{-1} \nabla F^{\mathcal{I}} - (X_{\mathcal{J}} \Lambda_{\mathcal{J}}^{-1} - G^{-1}) \nabla F^{\mathcal{J}}, \end{aligned}$$

where

$$G = M_{\mathcal{J},\mathcal{J}} - M_{\mathcal{I},\mathcal{J}}^T M_{\mathcal{I},\mathcal{I}}^{-1} M_{\mathcal{I},\mathcal{J}}.$$

Now by definition

$$G = X_{\mathcal{J}}^{-1}\Lambda_{\mathcal{J}}\left(I + \Lambda_{\mathcal{J}}^{-1}X_{\mathcal{J}}O(1)\right) = X_{\mathcal{J}}^{-1}\Lambda_{\mathcal{J}}\left(I + O(\mu)\right),$$

so that $G^{-1} = O(\mu)$ and

$$G^{-1} - X_{\mathcal{J}}\Lambda_{\mathcal{J}}^{-1} = O(\mu^2).$$

Note further that

$$\nabla F^{\mathcal{I}} = (\nabla f(x))^{\mathcal{I}} - \mu X_{\mathcal{I}}^{-1}e = O(\mu) + \mu O(1) = O(\mu),$$

whereas

$$\nabla F^{\mathcal{J}} = (\nabla f(x))^{\mathcal{J}} - \mu X_{\mathcal{J}}^{-1}e = O(1) + \mu O(1/\mu) = O(1).$$

It follows that

$$\tilde{p}_x^{\mathcal{I}} - p_x^{\mathcal{I}} = O(\mu)O(\mu) + O(\mu^2) = O(\mu^2),$$

and

$$\tilde{p}_x^{\mathcal{J}} - p_x^{\mathcal{J}} = O(\mu)O(\mu) + O(\mu^2) = O(\mu^2),$$

so that $\tilde{p}_x - p_x = O(\mu^2)$.

In [50], Nash and Sofer prove (for the case of the Newton direction arising from the logarithmic barrier objective function) that, for sufficiently small μ, the vector computed using an approximation similar to (4.19) and (4.20) yields a descent direction with respect to the logarithmic barrier objective function. The proof is readily extended to the present primal-dual case; thus p_x is a descent direction for the merit function $F(x, \mu)$. We have found that, for the present problem, the above approximation to the Newton direction is useful for values of μ of order 10^{-4} or less.

Recently Wright [60] showed that the errors generated by backward-stable numerical methods (various Cholesky factorizations and Gaussian elimination with partial pivoting) for solving (4.6) are not magnified by the structured ill-conditioning. These methods are inappropriate for our large problems which involve potentially millions of variables. Instead we find an approximate solution using a CG iteration. When working in inexact arithmetic with large numbers of variables, the convergence rate of the CG method depends on the condition of M [15]. Thus the structural ill-conditioning in M can lead the CG iteration to spend an unnecessary amount of work in computing p_x. Further, as we have observed, the criterion for terminating the CG may be overly optimistic in an ill-conditioned system, so that the resulting direction is poorly scaled as $\mu \to 0$.

The potential effect of ill-conditioning is illustrated through an example in Table 4.1. This example was encountered during development and motivated the incorporation of stabilization into the algorithm. Starting at the subproblem $\mu = 1.49 \times 10^{-4}$, the primal step length, dual step length, and *ncg* (the number of CG iterations) are listed for both the nonstabilized and stabilized cases. This test was terminated at $\lambda^T x/n \leq 7.5 \times 10^{-5}$. Note that in the nonstabilized case, the number of CG iterations from the first subproblem in the test to termination is significantly lower in the stabilized test than the nonstabilized test. Note also that in many of the nonstabilized subproblems, either the primal or dual step length is small, indicating a poorly scaled direction or loss of accuracy.

TABLE 4.1

An example of the effect of stabilization. The number of CG iterations, ncg*, is counted from the beginning of the* $\mu = 1.40 \times 10^{-4}$ *subproblem. The termination condition in this example is* $\lambda^T x/n \leq 7.5 \times 10^{-5}$.

Nonstabilized				Stabilized			
μ	α_x	α_λ	ncg	μ	α_x	α_λ	ncg
1.49E-4	0.950	0.003	5	1.49E-4	0.942	1.000	12
1.26E-4	0.156	1.000	17	1.16E-4	0.923	1.000	17
1.01E-4	1.000	0.002	22	7.75E-5	0.159	0.8245	30
8.50E-5	0.143	1.000	34	6.47E-5	0.962	1.000	35
7.08E-5	0.392	1.000	46	3.25E-5			
5.83E-5	1.000	0.166	51				
4.77E-5	0.016	1.000	62				
3.75E-5							

There has been much recent interest in stabilization methods that do not require a prediction of the active set [13, 14, 59]. These approaches are based on factorization methods which are unsuitable for a problem as large as the present one. The argument against stabilization methods that require a prediction set is that the active set is unknown in interior-point methods. We argue that, close to the solution in the emission tomography reconstruction problem, an accurate prediction of the active set can be made. In our problem, the constraints have a simple interpretation. The positive variables correspond to those voxels containing at least some radioactive tracer, while the zero-valued variables correspond to those voxels that lack any tracer activity. Close to the solution, when μ becomes sufficiently small that stabilization is appropriate, the set of binding constraints is obvious and can be conservatively identified with a μ-dependent threshold.

4.3. Extrapolation. Fiacco and McCormick showed that the solutions $x(\mu)$ at the perturbed KKT solutions form a unique differentiable trajectory in μ [12]. The perturbed KKT conditions (4.1)–(4.3) define a "central path" as $\mu \to 0$. Thus, a successful algorithm may be able to move both "along" and "toward" the path. As discussed in [12], from the subproblem solutions $\{x(\mu_l),\ l = 1, \ldots, k\}$, the trajectory can be approximated as a polynomial

$$x(\mu) \simeq \sum_{l=k-r}^{k} c_l \mu^l, \tag{4.21}$$

where r is the degree of the approximating polynomial and $c_{k-r}, \ldots, c_k$ are $r+1$ vectors of coefficients. Using the approximation in (4.21), we find a direction Δx such that

$$\Delta x = \sum_{l=k-r}^{k} c_l \mu^l - x^k,$$

and set

$$\hat{x}(\mu_{k+1}) = x^k + \bar{\alpha}\ \Delta x \tag{4.22}$$

to be a prediction to the next subproblem's primal solution. Here x^k is the computed (approximate) subproblem solution for $\mu = \mu_k$. Primal feasibility is maintained by the step length $\bar{\alpha} = 0.98\alpha_{\max}$, where $\alpha_{\max}$ is the maximum step length that does not

violate nonnegativity in x. Then, in the manner of (4.7), we compute a dual direction vector according to

$$\Delta\lambda = -\lambda^k - X_k^{-1}\Lambda_k\left(\bar{\alpha}\triangle x\right) + \mu X_k^{-1}e_n. \tag{4.23}$$

The dual vector is then moved according to

$$\hat{\lambda}\left(\mu_{k+1}\right) = \lambda_k + \tilde{\alpha}_\lambda \triangle x, \tag{4.24}$$

which requires another dual line search to minimize (4.10). The resulting point $(\hat{x}(\mu_{k+1}), \hat{\lambda}(\mu_{k+1}))$ serves as a starting point for the $(k+1)$st subproblem, a prediction to the solution at μ_{k+1}. The extrapolated primal-dual method can be viewed as a predictor-corrector algorithm, with the extrapolation (4.22 and 4.24) serving as the "predictor" step, and the subproblem minimization serving as the centering or "corrector" step [23]. The degree r of the approximating polynomial is 1 when predicting the third subproblem, 2 for the fourth, and 3 for the fifth and beyond.

We have experimented with line searches in conjunction with (4.22), but often $\bar{\alpha} \ll 1$, and hence the line search just yields $\bar{\alpha}$. For this reason, we have found that (4.24) yields a more effective dual direction than does the equivalent of (4.7) in the context of extrapolation. Although the extrapolated search direction Δx can often be poorly scaled (i.e., $\bar{\alpha} \ll 1$), we have observed that the directions produced are always descent directions to the merit function and lead to a significant decrease in the objective function f. A number of reconstructions were performed in which $\Delta\lambda$ was computed by extrapolating the dual solution vector (rather than computing it via (4.24)); the discouraging nature of the results led us to abandon direct extrapolation of the dual vector in favor of (4.24) which is highly effective in comparison.

Following extrapolation, a gradient evaluation is required to update the vector $\nabla F(\hat{x}(\mu_{k+1}), \mu_{k+1})$. Since the primal-dual algorithm requires between 12 and 25 subproblems to perform a 3-D MAP reconstruction, extrapolation adds that many gradient evaluation operations to the computational cost. So extrapolation is only economical if it reduces the computational burden by at least as much as it adds. Our experience has been that for some data sets, the cost of extrapolation is well worthwhile, but for other data sets the benefits were only marginal. Extrapolation thus appears to serve as somewhat of a safeguard against difficult problems. In an extrapolated primal-dual reconstruction, the convergence measure $\max(\lambda_i x_i)$ does not decrease as monotonically as in a primal-dual reconstruction without extrapolation. Certain extrapolated steps seem to cause the algorithm to "get ahead of itself," but this effect is transient. On the studies we've performed, the algorithm does ultimately converge to an accurate solution with extrapolation.

4.4. Initialization. The choice of the initial barrier parameter may have a substantial effect on the algorithm. If the parameter is too small, the first subproblem may have extreme difficulty due to ill-conditioning; if the parameter is too large, then many (unnecessary) subproblems will be required to solve the problem. Proper initialization of the barrier parameter μ involves finding the most suitable point on the barrier trajectory based on the initial solution x^o and the measurement data y. Recalling the perturbed necessary conditions in (4.1), if the initial solution $\hat{x}^0$were to be *on* the central path, it would satisfy

$$\nabla F\left(\hat{x}^0, \mu_0\right) = q - C\hat{Y}^{-1}y + \gamma\nabla R\left(\hat{x}^0\right) - \mu_0\hat{X}_0^{-1}e_n = 0.$$

Premultiplying by $\left(\hat{x}^0\right)^T$ we arrive at

$$q^T\hat{x}^0 - y^T e_N + \gamma\nabla^T R\left(\hat{x}^0\right)\hat{x}^0 = n\mu_0.$$

This suggests the following rule for initialization, which we find quite effective:

$$\mu_0 = \frac{\left|q^T\hat{x}^0 - y^T e_N + \gamma\nabla^T R\left(\hat{x}^0\right)\hat{x}^0\right|}{n}. \tag{4.25}$$

Another, similar, initialization rule is motivated by the goal of finding an initial value μ_0 so that

$$\nabla f\left(\hat{x}^0\right) - \mu_0\hat{X}_0^{-1}e_n \approx 0. \tag{4.26}$$

While (4.26) cannot be solved exactly, we can try to find a μ_0 that results in a point $\hat{x}^0$ that is close to the barrier trajectory according to, say, the 2-norm. This motivation leads to an alternative initialization rule [51]

$$\mu_0 = \frac{\left\|\nabla f\left(\hat{x}^0\right)\right\|_2}{\left\|\hat{X}_0^{-1}e_n\right\|_2}. \tag{4.27}$$

During the course of development, both initialization rules were tried on certain data sets. Although both initialization rules performed well, reconstructions initialized with (4.25) usually reached the optimal solution in slightly less overall work than those initialized with (4.27).

The initial estimate for $\hat{x}^0$ and $\hat{\lambda}^0$ we used most frequently was in each case a positive uniform field. A discussion on the rationale of using a uniform field for $\hat{x}^0$ and on criteria for choosing the constant value of the primal initial solution may be found in [24]. Alternative choices for the initial dual vector may be preferable, and an investigation into this question may be worthwhile.

4.5. Termination. Given that subproblem termination is based on the μ-criticality conditions (4.11) and (4.12), the closeness of each subproblem solution can be measured by μ. If subproblems are solved exactly, $|f(x(\mu)) - f(x^*)| \leq n\mu$ [12]. The μ-criticality conditions, however, are designed for a "short-step" algorithm in which one truncated-Newton step should satisfy each subproblem for sufficiently small μ. To ensure the accuracy of the final solution, final termination is based on the following two requirements:

$$\frac{\lambda^T x}{n} \leq \varepsilon_1, \tag{4.28}$$

$$\frac{\left\|\nabla_x \ell\left(x,\lambda\right)\right\|_\infty}{1+\left|f\left(x\right)\right|} \leq \varepsilon_2. \tag{4.29}$$

We have found that reasonably accurate solutions are ensured when $\varepsilon_1 = 1.5 \times 10^{-4}$ and $\varepsilon_2 = 5 \times 10^{-9}$.

The traditional view in tomographic reconstruction is that a highly accurate solution is unnecessary. This view stems in part from the ill-posedness of the problem and the computational cost of taking a reconstruction to full convergence. From empirical evidence in our studies, the ability to perform certain imaging tasks such as "cold spot detectability" improves with accuracy of the solution. Although the termination criteria we propose above may not appear particularly strict, they are from a tomographic reconstruction perspective.

TABLE 5.1

Properties affecting computation, memory, and storage costs for two different-sized reconstruction problems. Gradient evaluation costs are based on a 2.5M-count study on 10 120MHz IBM RISC/6000 SP processors.

Size class	n	N	Elements in C	Density in C	Storage cost of C	Cost of gradient
Thick-slice	376,882	5.36×10^6	2.02×10^{12}	0.93%	390 MB	3.42 min.
Thin-slice	1.40×10^6	6.30×10^7	8.82×10^{13}	0.35%	1.42 GB	7.23 min.

5. Computational studies. To test our algorithm, we have performed a number of reconstructions on data acquired from a small animal scanner and on data generated by Monte Carlo simulations on the same animal scanner.

5.1. Size of the problem. Our studies involved two different-sized problems. Raw coincidence data from the scanner can be binned into either "thick-slice" or "thin-slice" measurement spaces, or both. "Thick-slice" reconstructions, in which $n = 376{,}000$ and $N = 5.35 \times 10^6$, require 3.4 minutes for a gradient evaluation using 10 IBM RISC/6000 SP processors (120 MHz) on a 2.5M-count study. For a "thin-slice" reconstruction with $n = 1.4 \times 10^6$ and $N = 6.3 \times 10^7$ on the same data and processors, a gradient evaluation requires 6.75 minutes. These properties are summarized in Table 5.1. The cost of storing the full $n \times N$ system matrix is prohibitive, even for thick-slice reconstructions. Extensive exploitation of the sparsity and symmetries inherent in the system matrix makes its storage and retrieval possible [24, 25].

The dominant computational operations of the reconstruction problems are the forward- and back-transformation operations that underlie EM iterations, gradient evaluations, Hessian-vector products, and diagonal Hessian calculations. These operations have been implemented in parallel via a data decomposition strategy that partitions the "measurement-space" vectors y and $\hat{y}$ across the processors. The "image-space" vectors such as x and λ are replicated over all processors. Our data decomposition is justifiable under the observation that $N \gg n$. On a data set with 2.5M counts, at most 47% of the elements of y will be nonzero in the thick-slice case, and at most 4% in the thin-slice case. (The thin-slice configuration has over 10 times as many lines of response as the thick-slice.) The dominant computational operations have been implemented in such a way to exploit sparsity in y and further conserve computation [24].

5.2. Cost metrics. We have devised metrics to measure the cost of an interior point reconstruction. Define the number of subproblems to be npr, the number of truncated-Newton iterations nit, the number of CG subiterations ncg. The cost of one CG iteration (dominated by the Hessian-vector product) is equivalent to the cost of one gradient calculation or EM iteration. One truncated-Newton iteration requires, in addition to the ncg operations, one diagonal Hessian evaluation plus one forward transformation and one backward transformation. The exact cost of these operations varies depending on the size of the problem and number of counts, but we shall approximate the cost of one truncated-Newton iteration to be the equivalent of two gradient calculations beyond the cost of the CGs.

Using this approximation, the total cost of unextrapolated interior-point reconstructions can be measured in units of equivalent number of gradient calculations (or EM iterations):

$$ngr = 2 \cdot nit + ncg.$$

TABLE 5.2

Summary of thick-slice primal-dual results and comparison with MAP-EM and LSEM. Extrapolation was not used, and in all cases $\rho = 2$.

Study	f^*	*npr*	*nit*	*ncg*	*ngr*	MAP-EM	LSEM
A	2,465,770	19	19	110	148	1000	344
B	2,397,197	23	23	164	210	>1000	634
C	2,269,180	22	22	126	170	990	482
D	2,752,484	20	21	169	211	>1000	>1000
E	2,536,110	26	26	131	183	770	292
F	3,296,013	23	23	141	187	>1000	>1000
G	3,660,344	24	24	127	175	>1000	724
Average *ngr*					183		

TABLE 5.3

Summary of thick-slice extrapolated primal-dual results and comparison with MAP-EM and LSEM; in all cases $\rho = 2$.

Study	f^*	*npr*	*nit*	*ncg*	*ngr*	MAP-EM	LSEM
A	2,465,772	17	17	94	145	960	332
B	2,397,232	16	16	91	139	>1000	435
C	2,269,190	17	17	94	145	850	418
D	2,752,502	14	16	119	165	>1000	>1000
E	2,536,112	20	20	106	166	750	279
F	3,296,029	18	18	115	169	>1000	855
G	3,660,384	20	20	100	160	>1000	430
Average *ngr*					156		

Extrapolation requires an additional gradient calculation following the extrapolation in order to update the gradient vector. With extrapolation we modify the formula to

$$ngr = npr + 2 \cdot nit + ncg.$$

5.3. Computational results. We have performed a number of 3-D reconstructions on data acquired from a small animal scanner and data generated by a Monte Carlo simulation of the same small animal scanner. Reconstructions of seven datasets were taken to full convergence, as defined by the termination criteria (4.28) and (4.29) with $\epsilon_1 = 1.5 \times 10^{-4}$ and $\epsilon_2 = 5 \times 10^{-9}$. The various datasets used in our computational studies represent a fairly diverse sample of the types of scans that might be encountered in practice. The number of counts in the datasets used in these studies ranged from 850K to 5.1M. The number of binding constraints at the optimal solution ranged from approximately 20% to 80%.

Our main results are summarized in Tables 5.2 and 5.3 for the nonextrapolated and extrapolated primal-dual cases, respectively. Studies A through D are reconstructions of data acquired from a small animal PET scanner, while studies E through G are reconstructions of Monte Carlo simulated data. These reconstructions were performed in "thick-slice" mode (376,832 variables) with the regularization parameter set at $\gamma = 3 \times 10^{-4}$. In these tables, the column "MAP-EM" indicates the number of DePierro MAP-EM iterations that were required to achieve the value of f^* in the same row. The column "LSEM" indicates the number of iterations required for an EM algorithm, where the search direction on the last term of (3.2) is enhanced by a line search. (To avoid excessive computation, the function values were only calculated every 10 MAP-EM iterations, and the final count was rounded down, to favor this method.) Since the cost of one gradient evaluation is equivalent to the cost of one EM

TABLE 5.4
Summary of thick-slice logarithmic barrier results and comparison with MAP-EM and LSEM. Extrapolation was used on all data sets, and in all cases $\rho = 10$.

Study	f^*	*npr*	*nit*	*ncg*	*ngr*	MAP-EM	LSEM
A	2,465,832	5	28	159	218	880	194
B	2,397,199	5	29	198	259	>1000	615
C	2,269,180	5	29	185	246	990	482
D	2,752,499	4	25	207	260	>1000	>1000
E	2,536,111	6	40	214	298	780	285
F	3,296,037	5	36	197	274	>1000	776
G	3,660,351	6	41	214	300	>1000	621
Average *ngr*					265		

iteration, the numbers in the columns *ngr* and MAP-EM and LSEM can be compared directly. We find that the primal-dual method consistently reaches convergence much more rapidly than either MAP-EM or LSEM.

Another interesting observation can be made in the comparison between Tables 5.2 and 5.3. Consider the number of EM iterations required to reach f^* for study C. In Table 5.2, the LSEM algorithm reached $f = 2{,}269{,}180$ in 482 iterations. In Table 5.3 on the same data set, the LSEM algorithm reached $f = 2{,}269{,}190$ in 418 iterations. Thus, the algorithm took 64 iterations to reduce the function value by only 10 units near the solution. MAP-EM did even worse, requiring 180 iterations to reduce the function value by 10. This is in fact a typical example of the slow limit behavior of the EM algorithm. In all studies, the EM method did not achieve the same convergence results obtained by the primal-dual method at termination. The Lagrangian gradient norm and complementary slackness values of the terminated MAP-EM and LSEM iterates were consistently much higher than those of the terminated primal-dual solution.

We have also performed these reconstructions using a stabilized logarithmic barrier algorithm based on the method presented in [50] and specialized to the present reconstruction problem. Many of the computational features of our logarithmic barrier implementation are identical to our primal-dual implementation, e.g., truncated Newton, line search, computation of the gradient, Hessian-vector product, etc. For a more detailed discussion, see [24]. The logarithmic barrier results are summarized and compared against MAP-EM in Table 5.4. Termination of the logarithmic barrier was defined by (4.29) and

$$\frac{\max(\lambda_i x_i)}{1 + |f(x)|} \leq 5 \times 10^{-10}.$$

These termination criteria for the logarithmic barrier correspond to roughly the same accuracy as (4.28) and (4.29) do for the primal-dual method. Being a "long-step" method, the logarithmic barrier gives the user less control over the exact stopping point than does the "short-step" primal-dual. All of the logarithmic barrier reconstructions in Table 5.4 used extrapolation. In all logarithmic barrier reconstructions, μ was reduced by a factor of 10 between subproblems.

The effect of extrapolation is illustrated in Figures 5.1 and 5.2. In Figure 5.1, the equivalent number of gradient evaluations (*ngr*) to reach termination is plotted against objective function "distance" $f - f^*$, the difference between the function value of the terminated solution and the lowest function value obtained for that reconstruction. In all seven test cases (those listed in Tables 5.2–5.4), the unextrapolated

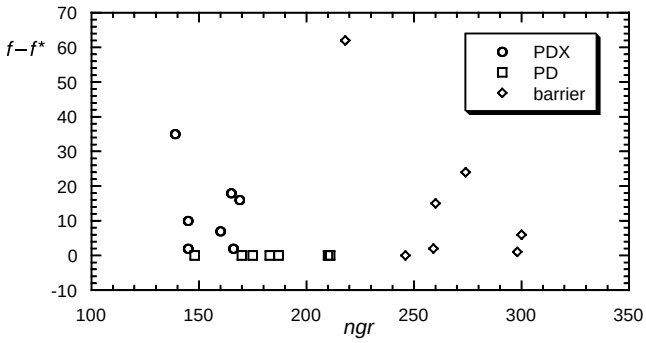


FIG. 5.1. *"Distance" from optimal solution at termination, as measured by difference in objective function $f - f^*$ (where f^* is here defined to be the lowest objective function obtained per study), versus work required to reach termination, as measured by ngr, the equivalent number of gradient evaluations. The studies included are those listed in Table* 5.2. *PD stands for nonextrapolated primal-dual, PDX for extrapolated primal-dual.*

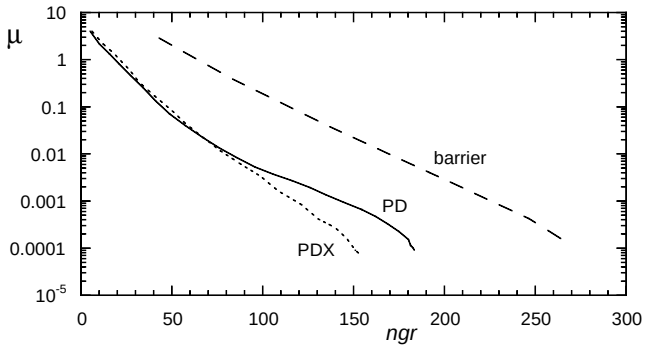


FIG. 5.2. *Average value of μ at subproblem termination versus average ngr (equivalent number of gradient evaluations) for the seven studies listed in Table* 5.2. *PD stands for nonextrapolated primal-dual, PDX for extrapolated primal-dual.*

primal-dual method achieved the lowest objective function value. Thus, $f - f^*$ is zero for all unextrapolated primal-dual (PD) results but greater than zero for the extrapolated primal-dual (PDX) and barrier results. The PDX results are clustered in a region of lower *ngr* than the PD results. This indicates that extrapolation lowers the computational expense to the solution at a slight deterioration in the final objective. Compared with the barrier method, either extrapolated or unextrapolated primal-dual produces equivalent or better accuracy with less computation required.

In Figure 5.2, the average number of equivalent gradient evaluations at subproblem termination is plotted against the average value of μ for each subproblem. Both averages (*ngr* and μ) were taken from the same seven test cases of Tables 5.2–5.4. Compared with either unextrapolated primal-dual (PD) or extrapolated primal-dual (PDX), the logarithmic barrier is clearly on a slower trajectory. The PD and PDX trajectories are quite similar until approximately $\mu = 0.01$, at which point the PD curve "swings out," while the PDX curve continues to descend log-linearly. This result confirms that the prediction (extrapolation) step becomes more accurate near the solution, resulting in more rapid convergence. However, a comparison of the objective functions indicates that the value of PDX μ is perhaps one step "ahead of itself," compared with the unextrapolated case.

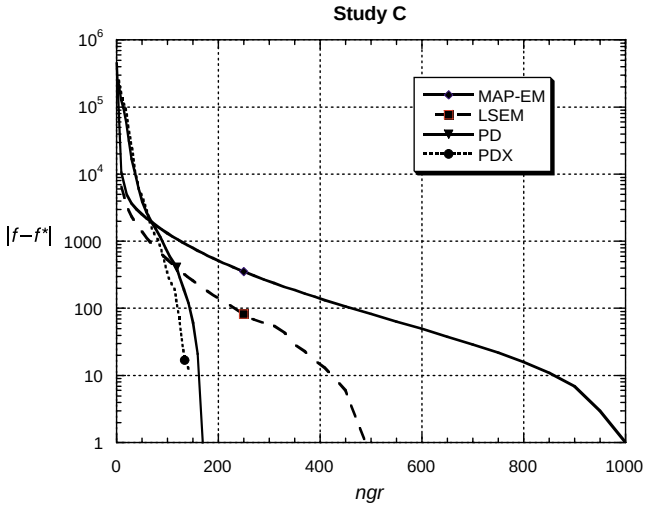


FIG. 5.3. *Improvement in objective function as a function of gradient evaluations, Study F. PDX denotes extrapolated primal-dual, PD denotes unextrapolated primal-dual, both using* $\rho = 2$.

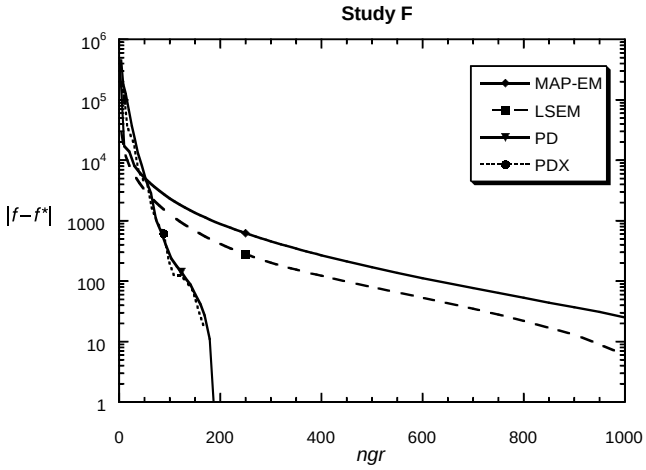


FIG. 5.4. *Improvement in objective function as a function of gradient evaluations, Study C. PDX denotes extrapolated primal-dual, PD denotes unextrapolated primal-dual, both using* $\rho = 2$.

The progress of the reconstruction on a study of a rat skull, Study C, is compared for the various algorithms in Figure 5.3. The measure used is $\|f - f^*\|$ (plotted on a logarithmic scale). In the initial iterations DePierro MAP-EM and LSEM progress rapidly and are ahead of the primal-dual method. However the interior-point methods rapidly reach the DePierro and LSEM objective values, and henceforth, surpass them. In the primal-dual methods depicted, the value of ρ is 2. The methods achieve faster initial progress using $\rho = 100$; however, the overall computational effort for full convergence with this parameter setting is greater. The progress of the reconstruction in another example, Study F, is compared in Figure 5.4.

We have also reconstructed a number of very large-scale "thin-slice" reconstructions involving 1.4×10^6 variables. Table 5.5 summarizes a number of properties of

TABLE 5.5

Summary of thin-slice extrapolated results, including convergence measures and computational costs to optimal solution.

Study	γ	f	$\frac{\|\nabla \ell\|}{1+\|f\|}$	$\frac{\max(\lambda_i x_i)}{1+\|f\|}$	*npr*	*nit*	*ncg*	*ngr*
B	8E-5	7,661,605	8.54E-11	1.77E-11	17	17	68	119
B	3E-5	7,658,720	9.77E-11	1.68E-11	17	17	72	123
B	1E-5	7,657,020	3.71E-10	2.14E-11	17	17	79	130
C	3E-5	5,826,032	9.08E-10	2.67E-11	15	16	56	103
F	3E-5	7,724,731	6.87E-10	7.96E-11	16	16	64	112
F	1E-5	7,721,001	1.29E-9	2.87E-11	14	14	71	113
H	3E-5	3,776,745	1.10E-9	3.89E-11	13	16	63	108
Average *ngr*								115

these extrapolated primal-dual reconstructions at the converged solution. A smaller group of datasets (the more visually "interesting" studies) were selected for the thin-slice work, and certain reconstructions were repeated with different values of the prior strength γ. Thin-slice reconstructions seem to require a lower prior strength than the corresponding thick-slice reconstructions. The most visually pleasing results were from reconstructions using $\gamma = 1 \times 10^{-5}$, which is 1/30 the prior strength that was generally found to be most satisfactory in thick-slice reconstructions. The total amount of work (as measured in *ngr*) required to reach termination in Table 5.5 is also quite pleasing. The number of variables in a thin-slice reconstruction is approximately 3.7 times the number in thick-slice. The number of nonzero-valued measurements in thin-slice mode is only marginally greater than in thick-slice mode, however, since the number of counts is the same in both cases. These thin-slice reconstructions may thus be better conditioned than their thick-slice counterparts.

In closing, we should comment that the tolerance we have used in our tests is stricter than that usually necessary. Indeed, less accurate solutions may still give acceptable images. When the EM method is applied to the (unregularized) ML objective, it is usually terminated after 50 or 100 iterations, and the images produced are often good. Thus EM-ML remains a practical method that can sometimes reach a solution of desirable image quality faster than an interior-point method. The difficulty with EM-ML is that its convergence is object-dependent [1]. Convergence in areas of high activity amidst low activity or vice versa is notoriously slow, and a fixed termination rule based on (say) 50 or 100 iterations cannot guarantee acceptable image quality. This has been observed in a number of reconstructions, including some of high biomedical interest. In contrast to ML-EM, the primal-dual algorithm has object-independent convergence characteristics. Furthermore, it is flexible and can be adapted to solve a problem efficiently both to the strict tolerance in the studies above by setting a modest rate of decrease for the barrier parameter, say, $\rho = 2$, and to a looser tolerance by setting a more aggressive reduction rate such as $\rho = 100$.

6. Conclusion. From the results of the previous section, it is clear that the primal-dual method can converge significantly faster than the EM algorithm for regularized ML reconstructions in emission tomography. The results also indicate that the primal-dual method converges faster than the logarithmic barrier method. The use of extrapolation in conjunction with the primal-dual method further reduces the amount of computation required to achieve convergence.

Given that the negative regularized ML objective function that we minimize is convex, approximately solving the reduced unsymmetric primal-dual Newton equa-

tions is appropriate. Symmetrizing the unsymmetric system, while potentially useful for nonconvex problems, would in this case require solving for $2n$ variables without avoiding the potential for ill-conditioning. Our stabilization technique avoids the structural ill-conditioning of the condensed primal-dual matrix, and therefore solving the reduced system poses no asymptotic difficulty as the barrier parameter approaches zero. The computational efficiency and relative simplicity of formation of the reduced system of equations pose such a strong advantage that our choice of primal-dual method almost seems obvious for this problem.

Since Newton's method converges quadratically near the solution, for a well-conditioned system in the limit as $\mu \to 0$, one truncated-Newton step per subproblem should yield an increasingly accurate and well-scaled direction to the subproblem solution for μ_k. As μ is decreased, the subproblem solutions should become "close" to each other for a convex problem [14]. Yet, the example in Table 4.1 illustrates that the direction produced by the early-terminated CG can in fact become less accurate for smaller μ due to the structured ill-conditioning in M. In practice, we do not require the accuracy of the test example in Table 4.1. Our termination conditions are defined to be near the point on the trajectory where the stabilization approximation becomes accurate enough to guarantee descent. These termination criteria are quite accurate by the standards of the tomography community. Thus, although most reconstruction problems are unlikely to be *severely* affected by ill-conditioning, the potential for slow convergence near the solution due to ill-conditioning does exist. Our experience has been that stabilization has been an effective safeguard against poor performance for small values of the barrier parameter.

Acknowledgments. The study utilized the high-performance computational capabilities of the IBM RISC/6000 SP system at the Center for Information Technology, National Institutes of Health, Bethesda, MD. We are grateful to Jürgen Seidel of the Department of Nuclear Medicine, National Institutes of Health, for kindly providing us with the small animal data and Monte Carlo simulation data. Our thanks go to two anonymous referees and the associate editor for their careful reading and helpful comments.

REFERENCES

[1] H.H. Barrett, D.W. Wilson, and B.M.W. Tsui, *Noise properties of the EM algorithm* I: *Theory*, Phys. Med. Biol., 39 (1994), pp. 833–846.

[2] R.H. Chan and M.K. Ng, *Conjugate gradient methods for Toeplitz systems*, SIAM Rev., 38 (1996), pp. 427–482.

[3] G. Chinn and S.-C. Huang, *A general class of preconditioners for statistical iterative reconstruction of emission computed tomography*, IEEE Trans. Med. Imag., 16 (1997), pp. 1–10.

[4] N.H. Clinthorne, T.-S. Pan, P-C. Chiao, W.L. Rogers, and J.A. Stamos, *Preconditioning methods for improved convergence rates in iterative reconstructions*, IEEE Trans. Med. Imag., 12 (1993), pp. 78–83.

[5] A.R. Conn, N. Gould, and Ph.L. Toint, *A primal-dual algorithm for minimizing a nonconvex function subject to bound and linear equality constraints*, in Nonlinear Optimization and Related Topics, Appl. Optim. 36, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000, pp. 15–49.

[6] R.S. Dembo and T. Steihaug, *Truncated-Newton algorithms for large-scale unconstrained optimization*, Math. Programming, 26 (1983), pp. 190–212.

[7] G. Demoment, *Image reconstruction and restoration: Overview of common estimation structures and problems*, IEEE Trans. Acoust. Speech Signal Process., 37 (1989), pp. 2024–2036.

[8] A.P. Dempster, N.M. Laird, and D.B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm*, J. Roy. Statist. Soc. Ser. B, 39 (1977), pp. 1–38.

[9] A.R. DePierro, *A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography*, IEEE Trans. Med. Imag., 14 (1995), pp. 132–137.
[10] A.R. DePierro, *On the convergence of an EM-type algorithm for penalized likelihood estimation in emission tomography*, IEEE Trans. Med. Imag., 14 (1995), pp. 762–765.
[11] J.-P. Dussault, *Numerical stability and efficiency of penalty algorithms*, SIAM J. Numer. Anal., 32 (1995), pp. 296–317.
[12] A.V. Fiacco and G.P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, John Wiley, New York, 1968.
[13] A. Forsgren and P.E. Gill, *Primal-dual interior methods for nonconvex nonlinear programming,* SIAM J. Optim., 8 (1998), pp. 1132–1152.
[14] 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.
[15] G. Golub and C. Van Loan, *Matrix Computations*, 2nd ed., The Johns Hopkins University Press, Baltimore, MD, 1989.
[16] P.J. Green, *On use of the EM algorithm for penalized likelihood estimation*, J. Roy. Statist. Soc. Ser. B., 52 (1990), pp. 443–452.
[17] P.J. Green, *Bayesian reconstructions from emission tomography data using a modified EM algorithm*, IEEE. Trans. Med. Imag., 9 (1990), pp. 84–93.
[18] C.C. Gonzaga, *Path-following methods for linear programming*, SIAM Rev., 34 (1992), pp. 167–224.
[19] T. Hebert and R. Leahy, *A generalized EM algorithm for* 3*-D Bayesian reconstruction from Poisson data using Gibbs priors*, IEEE Trans. Med. Imag., 8 (1989), pp. 194–202.
[20] G.T. Herman, *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*, Academic Press, New York, 1980.
[21] M.R. Hestenes and E. Steifel, *Methods of conjugate gradients for solving linear systems*, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.
[22] H.M. Hudson and R.S. Larkin, *Accelerated image reconstruction using ordered subsets of projection data*, IEEE Trans. Med. Imag., 13 (1994), pp. 601–609.
[23] F. Jarre and M.A. Saunders, *A practical interior-point method for convex programming*, SIAM J. Optim., 5 (1995), pp. 149–171.
[24] C.A. Johnson, *Nonlinear Optimization for Volume PET Reconstructions*, Ph.D. dissertation, Department of Operations Research and Engineering, George Mason University, Fairfax, VA, 1997.
[25] C.A. Johnson, Y. Yan, R.E. Carson, R.L. Martino, and M.E. Daube-Witherspoon, *A system for the* 3*D reconstruction of retracted-septa PET data using the EM algorithm*, IEEE Trans. Nucl. Sci., 42 (1995), pp. 1223–1227.
[26] C.A. Johnson, J. Seidel, R.E. Carson, W.R. Gandler, A. Sofer, M.V. Green, and M.E. Daube-Witherspoon, *Evaluation of* 3*D reconstruction algorithms for a small animal PET camera*, IEEE Trans Nucl. Sci., 44 (1997) pp. 1303–1308.
[27] L. Kaufman, *Implementing and accelerating the EM algorithm for positron emission tomography*, IEEE Trans. Med. Imag., 6 (1987), pp. 37–51.
[28] L. Kaufman, *Solving emission tomography problems on vector machines*, Ann. Oper. Res., 22 (1990), pp. 325–353.
[29] L. Kaufman, *Maximum likelihood, least squares and penalized least squares for PET*, IEEE Trans. Med. Imag., 12 (1993), pp. 200–214.
[30] L. Kaufman and A. Neumaier, *PET regularization for envelope guided conjugate gradients*, IEEE Trans. Med. Imag., 15 (1996), pp. 385–389.
[31] S. Kawata and L. Nalcioglu, *Constrained reconstruction by the conjugate gradient method*, IEEE Trans. Med. Imag., 4 (1985), pp. 65–71.
[32] 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.
[33] A. Kuruc, *Probability measure estimation using "weak" loss functions in positron emission tomography*, in Tomography, Impedance Imaging and Integral Geometry, Lectures in Appl. Math. 30, M. Cheney, P. Kuchment, and E.T. Quinto, eds., AMS, Providence, RI, 1994, pp. 125–142.
[34] D.S. Lalush and B.M.W. Tsui, *The importance of preconditioners in fast Poisson-based iterative reconstruction algorithms for SPECT*, in Proceedings of the 1995 IEEE Nuclear Science Symposium and Medical Imaging Conference, Conference Record 3, 1995, pp. 1326–1330.
[35] J. Liow and S. Strother, *Practical tradeoffs between noise, quantitation and number of iter-*

ations for maximum likelihood-based reconstructions, IEEE Trans. Med. Imag., 13 (1991), pp. 601–609.

[36] K. Lange and R. Carson, *EM reconstruction algorithms for emission and transmission tomography*, J. Comput. Assist. Tomogr., 8 (1984), pp. 306–316.

[37] K. Lange, M. Bahn, and R. Little, *A theoretical study of some maximum likelihood algorithms for emission and transmission tomography*, IEEE Trans. Med. Imag., 6 (1987), pp. 106–114.

[38] K. Lange, *Convergence of EM image reconstruction algorithms with Gibbs smoothing*, IEEE Trans. Med. Imag., 9 (1990), pp. 439–446.

[39] E. Levitan and G.T. Herman, *A maximum a posteriori probability expectation maximization algorithm for image reconstruction in emission tomography*, IEEE Trans. Med. Imag., 6 (1987), pp. 185–192.

[40] I. 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.

[41] G.P. McCormick, *The Superlinear Convergence of a Nonlinear Primal-Dual Algorithm*, Report T-550/91, Department of Operations Research, George Washington University, Washington, DC, 1991.

[42] E.U. Mumcuoglu and R. Leahy, *Gradient projection conjugate gradient algorithm in Bayesian PET reconstruction*, in Proceedings of the 1994 IEEE Nuclear Science Symposium and Medical Imaging Conference, Conference Record 3, 1995, pp. 1212–1216.

[43] E.U. Mumcuoglu, R.M. Leahy, S.R. Cherry, and Z. Zhou, *Fast gradient-based methods for Bayesian reconstruction of transmission and emission PET images*, IEEE Trans. Med. Imag., 13 (1994), pp. 687–701.

[44] E.U. Mumcuoglu, R.M. Leahy, and S.R. Cherry, *Bayesian reconstruction of PET images: Methodology and performance analysis*, Phys. Med. Biol., 41 (1996), pp. 1777–1807.

[45] W. Murray, *Analytic expressions for the eigenvalues and eigenvectors of the Hessian matrices of barrier and penalty functions*, J. Optim. Theory Appl., 7 (1971), pp. 189–196.

[46] S.G. Nash, *Preconditioning of truncated-Newton methods*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 599–616.

[47] S.G. Nash and A. Sofer, *Block truncated-Newton methods for parallel optimization*, Math. Programming, 45 (1989), pp. 529–546.

[48] S.G. Nash and A. Sofer, *Assessing a search direction within a truncated-Newton method*, Oper. Res. Lett., 9 (1990), pp. 219–221.

[49] S.G. Nash and A. Sofer, *A general-purpose parallel algorithm for unconstrained optimization*, SIAM J. Optim., 1 (1991), pp. 530–547.

[50] S.G. Nash and A. Sofer, *A barrier method for large-scale constrained optimizaton*, ORSA J. Comput., 5 (1993), pp. 40–53.

[51] S.G. Nash and A. Sofer, *Linear and Nonlinear Programming*, McGraw–Hill, New York, 1996.

[52] D.B. Ponceleon, *Barrier Methods for Large-Scale Quadratic Programming*, Report SOL 91-2, Ph.D. thesis, Department of Operations Research, Stanford University, Stanford, CA, 1991.

[53] L.A. Shepp and Y. Vardi, *Maximum likelihood reconstruction for emission tomography*, IEEE Trans Med. Imag., 1 (1982), pp. 113–122.

[54] E.M. Simantiraki and D.F. Shanno, *An infeasible-interior-point method for linear complementarity problems*, SIAM J. Optim., 7 (1997), pp. 620–640.

[55] D.L. Snyder, M.I. Miller, L.J. Thomas, Jr., and D.G. Politte, *Noise and edge artifacts in maximum-likelihood reconstructions for emission tomography*, IEEE Trans. Med. Imag., 6 (1987), pp. 228–238.

[56] Y. Vardi, L.A. Shepp, and L. Kaufman, *A statistical model for positron emission tomography*, J. Amer. Statist. Assoc., 80, (1985), pp. 8–37.

[57] D.W. Wilson and B.M.W. Tsui, *Noise properties of filtered-backprojection and ML-EM reconstructed emission tomography images*, IEEE Trans. Nucl. Sci., 40 (1993), pp. 1198–1203.

[58] M.H. Wright, *Interior methods for constrained optimization*, in Acta Numer., Cambridge University Press, Cambridge, UK, 1991, pp. 341–407.

[59] M.H. Wright, *Some properties of the Hessian of the logarithmic barrier function,* Math. Programming, 67 (1994), pp. 265–295.

[60] M.H. Wright, *Ill-conditioning and computational error in interior methods for nonlinear programming*, SIAM J. Optim., 9 (1999), pp. 84–111.

[61] S.J. Wright, *Stability of linear equations solvers in interior-point methods*, SIAM J. Matrix Anal. Appl., 16 (1995), pp. 1287–1307.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 716–747

A NEW SQP ALGORITHM FOR LARGE-SCALE NONLINEAR PROGRAMMING*

R. W. H. SARGENT† AND M. DING†

Abstract. An efficient new SQP algorithm capable of solving large-scale problems is described. It generates descent directions for an ℓ_1 plus log-barrier merit function and uses a line-search to obtain a sufficient decrease of this function. The unmodified exact Hessian matrix of the Lagrangian function is normally used in the QP subproblem, but this is set to zero if it fails to yield a descent direction for the merit function. The QP problem is solved by an interior-point method using an inexact Newton approach, iterating to an accuracy just sufficient to produce a descent direction in the early stages and tightening the accuracy as we approach a solution.

We prove finite termination of the algorithm, at an ϵ-optimal Fritz-John point if feasibility is attained. We also show that if any iterate is close enough to an isolated connected subset of local minimizers, then the iterates converge to this subset. The rate of convergence is Q-quadratic if the subset is an isolated minimizer which satisfies a second-order sufficiency condition, but Q-quadratic convergence to an ϵ-optimal point can still be achieved without any conditions beyond Lipschitz continuity of second-order derivatives.

The implementation SQPIPM is designed for problems with many degrees of freedom and is shown to perform well compared with other codes on a range of standard problems.

Key words. nonlinear programming, large-scale problems, SQP algorithm, interior-point methods, Q-quadratic convergence

AMS subject classifications. 65K05, 49D37

PII. S1052623496297012

1. Introduction.

In a recent review paper [17] one of the authors has described the development of SQP algorithms and the difficulties that arise in attempting to apply existing algorithms to large-scale problems.

In essence, SQP algorithms rely on the power of the Newton iteration and are based on strong optimality conditions which ensure a final superlinear convergence rate to a local optimum, while to achieve "global" convergence from arbitrary starting points they modify the QP subproblem so that its solution produces a "sufficient" decrease of a suitable merit function. It is accepted that this does not guarantee true global convergence, since the iterates from an infeasible starting point may converge to a region of attraction of a local nonzero minimum of the constraint violations, and to deal with this takes us into the realm of global optimization techniques.

The QP subproblem represents the local linearization of the problem Karush–Kuhn–Tucker (KKT) conditions, involving the Hessian matrix of the Lagrangian function. Trust-region methods add a step-length bound to this subproblem and seek a global minimizer for it, which involves decomposition into two subproblems, one in the null-space and the other in the range-space of the current set of active constraints. Line-search methods modify the "reduced Hessian matrix" in the null-space of these constraints, either directly or indirectly, to yield a strictly convex QP problem whose solution is a descent direction for the merit function. In both cases identification of the null-space and generation of an appropriate matrix is required, but unfortunately all methods proposed so far for achieving this decomposition involve the generation

*Received by the editors January 5, 1996; accepted for publication (in revised form) May 25, 2000; published electronically November 15, 2000.
http://www.siam.org/journals/siopt/11-3/29701.html
†Centre for Process Systems Engineering, Imperial College, London SW7 2BY, UK (r.w.h.sargent@ps.ic.ac.uk).

of relatively dense matrices with the dimension of the active set subspace, and this limits the applicability of the algorithms to problems with only a modest number of degrees of freedom.

One exception to this approach is the line-search algorithm of Betts and Frank [2]. By obtaining the inertia of the KKT matrix from its factorization, they check whether the reduced Hessian matrix is positive definite, and if it is not they add a positive multiple of the unit matrix to the full Hessian matrix.

In this paper we present an algorithm which uses a similar strategy to avoid the need for a range-space/null-space decomposition, though both the test and the corrective action are different. We use an interior-point method to solve the QP subproblem, which uses a barrier-function to eliminate inequality constraints and hence also the combinatorial problem of identifying the current active set, and we make further savings by use of an inexact Newton approach, solving the subproblem to an accuracy just sufficient to generate a descent direction for the merit function.

All SQP algorithms we are aware of assume that all local minimizers are KKT points, and proofs of a final superlinear convergence rate assume that the iterates converge to an isolated local minimizer at which a constraint qualification, a strict complementarity condition, and a second-order sufficiency condition hold. In fact, the algorithms can converge towards a Fritz-John point which does not satisfy a constraint qualification and fail due to the multipliers becoming unbounded. More generally these strong optimality conditions are often not satisfied by local minimizers in real-world problems.

In our algorithm we seek a Fritz-John point, and by exploiting the convexifying effect of the barrier-function we can also significantly weaken these conditions. We are able to prove finite termination of the algorithm, at an ϵ-optimal Fritz-John point if feasibility is attained, and if any iterate is sufficiently close to a connected subset of local minimizers, then the algorithm converges to this subset. The final rate of convergence is proved to be Q-quadratic if the subset is in fact an isolated local minimizer which satisfies a second-order sufficiency condition, though strict complementary is not required, and we show that we can still obtain Q-quadratic convergence to an ϵ-optimal point without *any* regularity conditions beyond Lipschitz continuity of second-order derivatives of the functions involved.

We describe the development of the algorithm in section 2 and give proofs of the convergence properties in section 3, then present the results of numerical testing in section 4, and conclude with some general comments in section 5.

2. Development of the algorithm. We consider the general nonlinear programming problem in the form

$$\min_{x \in X} \{ f^o(x) \mid f(x) = 0 \}, \tag{2.1}$$

where

$$\begin{aligned} &X = \{ x \in \Re^n \mid a \le x \le b \}, \\ &f^o : X \to \Re, \quad f : X \to \Re^m, \quad m \le n, \end{aligned}$$

and $f^o(x)$, $f(x)$ are twice continuously differentiable on X.

We also assume that a and b are finite vectors, a reasonable assumption for a practical algorithm which helps to avoid overflow.

Any solution of (2.1) must satisfy the Fritz-John conditions

$$\begin{aligned} & g^o(x)y^o + G(x)y - \underline{z} + \overline{z} = 0, \\ & f(x) = 0, \\ & x - a \geq 0, \quad \underline{z} \geq 0, \quad \underline{z}^T(x-a) = 0, \\ & b - x \geq 0, \quad \overline{z} \geq 0, \quad \overline{z}^T(b-x) = 0 \end{aligned} \tag{2.2}$$

for some multipliers $\eta = (y^o, y, \underline{z}, \overline{z})$ with $y^o \geq 0$, where we have written $g^o(x) \equiv [f_x^o(x)]^T$, $G(x) \equiv [f_x(x)]^T$.

We refer to the set of points $x \in \xi$ satisfying (2.2) for some η as "Fritz-John points" and denote by Ω the set of solutions $\zeta = (x, \eta)$ of (2.2). In fact we seek such a solution, rather than attempting to solve (2.1).

As in the classical SQP approach, we solve (2.2) by a damped Newton method, generating a sequence of iterates $\{x_k\}, k = 0, 1, 2, \ldots$, from an arbitrary starting point $x_o \in X$, using

$$x_{k+1} = x_k + \alpha_{k+1}\Delta x_{k+1}, \tag{2.3}$$

where $\alpha_{k+1} \in (0, 1]$ and Δx_{k+1} is obtained by solving the subproblem resulting from linearizing the functions $g^o(x), G(x)$, and $f(x)$ in (2.2) to yield

$$\begin{aligned} & g_k^o y^o + H_k(x - x_k) + G_k y - \underline{z} + \overline{z} = 0, \\ & f_k + G_k^T(x - x_k) = 0, \\ & x - a \geq 0,\ \underline{z} \geq 0,\ \underline{z}^T(x-a) = 0, \\ & b - x \geq 0,\ \overline{z} \geq 0,\ \overline{z}^T(b-x) = 0, \end{aligned} \tag{2.4}$$

where $f_k \equiv f(x_k), G_k \equiv G(x_k), H_k = \sum_{j=0}^m y^j f_{xx}^j(x_k)$.

However, only second-order errors are introduced if we replace y by y_k in the expression for H_k, and this has the advantage that H_k has a fixed value for the subproblem. Indeed, since we can compute the $f_{xx}^j(x_k)$ by automatic differentiation, H_k can be generated directly during this process, without storing the individual $f_{xx}^j(x_k)$.

Conditions (2.4) are then easily recognized to be the Fritz-John conditions for the QP:

$$\begin{aligned} \text{Minimize} \quad & (x - x_k)g_k^o + \tfrac{1}{2}(x - x_k)^T H_k(x - x_k), \\ \text{subject to} \quad & f_k + G_k^T(x - x_k) = 0, \\ & a \leq x \leq b, \end{aligned} \tag{2.5}$$

so we have an SQP algorithm.

For real-world problems users can usually specify a reasonable range for each variable, based on physical considerations. However, if the range is very wide it can be helpful to include a fixed step-length bound Δ, by modifying the variable bounds to

$$a_k \leq x \leq b_k, \tag{2.6}$$

where $a_k^i = \max\{a^i, x_k^i - \Delta^i\}$, $b_k^i = \min\{b^i, x_k^i + \Delta^i\}$.

We solve this QP by an interior-point method (cf. [15]), which starts by eliminating the bound inequalities using barrier functions to yield the following modified problem:

$$\begin{aligned} \text{Minimize} \quad & (x - x_k)g_k^o + \tfrac{1}{2}(x - x_k)^T H_k(x - x_k) - \mu L(x), \\ \text{subject to} \quad & f_k + G_k^T(x - x_k) = 0, \\ & a_k < x < b_k, \end{aligned} \tag{2.7}$$

where

$$L(x) = \sum_{i=1}^{n} \{\ln(x^i - a_k^i) + \ln(b_k^i - x^i)\}, \quad \mu > 0. \tag{2.8}$$

We then solve the Fritz-John conditions for this problem, again solving a sequence of linearized problems as in the damped Newton method, but also reducing μ at each step. Further details are given in section 2.1.

Having obtained a solution $\zeta = (x, y^o, y, \underline{z}, \overline{z})$ to (2.4), we set $\Delta x_{k+1} = x - x_k$, then choose α_{k+1} in (2.3) to obtain a sufficient decrease of the merit function

$$P(x, y^o, \tilde{y}, \mu) = y^o\{f^o(x) - \mu L(x)\} + \sum_{j=1}^{m} \tilde{y}^j |f^j(x)|. \tag{2.9}$$

This is similar to the Han merit function [10] for problem (2.1) using different weights $y^o, \tilde{y}^j$ for the objective function and individual constraints. It also includes the penalty term $\mu L(x)$, which may seem surprising since we always have $a_k < x < b_k$, but as we shall see this has a convexifying effect, which makes it easier to ensure that the solution Δx_{k+1} of (2.7) is a descent direction for the merit function.

In fact we do not solve problem (2.7) exactly, but use an inexact Newton approach, solving the subproblem to an accuracy just sufficient to obtain a descent direction for the merit function until we are close to a solution, then progressively tightening the accuracy in order to achieve quadratic convergence. Further details are given in section 2.2.

2.1. Solution of the QP subproblem. It is well known that if $f(x_k) \neq 0$, the linearized problem (2.5), and hence also (2.7), may have no feasible solution, so to circumvent this difficulty we use a "big-M" modification (see, for example, [16]) of problem (2.7):

$$\begin{array}{ll} \text{Minimize} & M_k x^o + (x - x_k) g_k^o + \frac{1}{2}(x - x_k)^T H_k (x - x_k) \\ & -\mu^o \ln x^o - \mu L(x), \\ \text{subject to} & f_k(1 - x^o) + G_k^T (x - x_k) = 0, \\ & x^o > 0,\ a_k < x < b_k,\ M_k > 0, \end{array} \tag{2.10}$$

whose Fritz-John conditions are

$$\begin{array}{l} y^o\{M_k - \mu^o / x^o\} - f_k^T y = 0, \\ y^o \left\{ g_k^o - \mu \sum_{i=1}^{n} ((x^i - a^i)^{-1} - (b^i - x^i)^{-1}) \right\} + H_k(x - x_k) + G(x) y = 0, \\ f_k(1 - x^o) + G_k^T (x - x_k) = 0, \\ x^o > 0,\ a_k < x < b_k, \end{array} \tag{2.11}$$

or, in "primal-dual" form,

$$M_k y^o - f_k^T y - z^o = 0, \tag{2.12a}$$

$$g_k^o y^o + H_k(x - x_k) + G_k y - \underline{z} + \overline{z} = 0, \tag{2.12b}$$

$$f_k(1 - x^o) + G_k^T (x - x_k) = 0, \tag{2.12c}$$

$$x^o z^o - y^o \mu^o = 0, \tag{2.12d}$$

$$\underline{X}_k \underline{z} - y^o \mu e = 0, \tag{2.12e}$$

$$\overline{X}_k \overline{z} - y^o \mu e = 0, \tag{2.12f}$$

where $\underline{X}_k = \mathrm{diag}[x_i - a_k^i]$, $\overline{X}_k = \mathrm{diag}[b_k^i - x_i]$.

Since $x^o = 1$, $x = x_k$ is a feasible point for problem (2.10), this always has a solution, and it follows that (2.12a)–(2.12f) always have a solution for any given $y^o > 0$, though if $[-f_k, G_k^T]$ does not have full rank there may be some redundant constraints. It is also well known (see [16]) that if some $x \in \overset{o}{X} = \{x \in \Re^n \mid a < x < b\}$ is feasible for (2.5), then for sufficiently large M_k, we have $x^o \to 0$ in any solution of (2.12a)–(2.12f) as $\mu^o \to 0, \mu \to 0$.

To solve (2.12a)–(2.12f) we generate a sequence of iterates $\chi_{kl} = (x_{kl}^o, x_{kl}, y_{kl}, z_{kl}^o, \underline{z}_{kl}, \overline{z}_{kl}), l = 0, 1, 2, \ldots$, using

$$\chi_{k,l+1} = \chi_{kl} + \alpha'_{k,l+1}\delta\chi_{k,l+1}, \tag{2.13}$$

where $\alpha'_{k,l+1} \in (0, 1]$ and $\delta\chi_{k,l+1}$ is the solution of the linearized problem

$$\begin{array}{l} r_{Dkl}^o - f_k^T\delta y_{k,l+1} = 0, \\ r_{Dkl} + H_k\delta x_{k,l+1} + G_k\delta y_{k,l+1} - \delta\underline{z}_{k,l+1} + \delta\overline{z}_{k,l+1} = 0, \\ r_{Pkl} - f_k\delta x_{k,l+1}^o + G_k^T\delta x_{k,l+1} = 0, \\ (r_{ckl}^o)' + z_{kl}^o\delta x_{k,l+1}^o + x_{kl}^o\delta z_{k,l+1}^o = 0, \\ (\underline{r}_{ckl})' + \underline{Z}_{kl}\delta x_{k,l+1} + \underline{X}_{kl}\delta\underline{z}_{k,l+1} = 0, \\ (\overline{r}_{ckl})' + \overline{Z}_{kl}\delta x_{k,l+1} + \overline{X}_{kl}\delta\overline{z}_{k,l+1} = 0, \end{array} \tag{2.14}$$

where

$$\begin{array}{l} r_{Dkl}^o = M_k y_k^o - f_k^T y_{kl} - z_{kl}^o, \\ r_{Dkl} = g_k^o y_k^o + H_k(x_{kl} - x_k) + G_k y_{kl} - \underline{z}_{kl} + \overline{z}_{kl}, \\ r_{Pkl} = f_k(1 - x_k^o) + G_k^T(x_{kl} - x_k), \\ (r_{ckl}^o)' = x_{kl}^o z_{kl}^o - y_k^o\gamma_{kl}^o\mu_{kl}^o, \\ (\underline{r}_{ckl})' = \underline{X}_{kl}\underline{z}_{kl} - y_k^o\gamma_{kl}\mu_{kl}e, \\ (\overline{r}_{ckl})' = \overline{X}_{kl}\overline{z}_{kl} - y_k^o\gamma_{kl}\mu_{kl}e, \\ \underline{Z}_{kl} = \mathrm{diag}[\underline{z}_{kl}^i], \quad \overline{Z}_{kl} = \mathrm{diag}[\overline{z}_{kl}^i]. \end{array} \tag{2.15}$$

In (2.15), γ_{kl}^o and γ_{kl} are target reduction factors for μ^o and μ, defined below, but as in (2.13) we scale the changes in μ^o and μ to give values

$$\begin{array}{rcl} \mu_{k,l+1}^o & = & \mu_{kl}^o(1 - \alpha'_{k,l+1}(1 - \gamma_{kl}^o)), \\ \mu_{k,l+1} & = & \mu_{kl}(1 - \alpha'_{k,l+1}(1 - \gamma_{kl})). \end{array} \tag{2.16}$$

To solve the linear system (2.14) we first eliminate $\delta z_{k,l+1}^o, \delta\underline{z}_{k,l+1}, \delta\overline{z}_{k,l+1}$ using

$$\begin{array}{rcl} \delta z_{k,l+1}^o & = & -(x_{kl}^o)^{-1}((r_{ckl}^o)' + z_{kl}^o\delta x_{k,l+1}^o), \\ \delta\underline{z}_{k,l+1} & = & -\underline{X}_{kl}^{-1}((\underline{r}_{ckl})' + \underline{Z}_{kl}\delta x_{k,l+1}), \\ \delta\overline{z}_{k,l+1} & = & -\overline{X}_{kl}^{-1}((\overline{r}_{ckl})' - \overline{Z}_{kl}\delta x_{k,l+1}); \end{array} \tag{2.17}$$

then to improve the conditioning we scale the resulting reduced system to yield

$$\begin{array}{l} \begin{bmatrix} 1 & 0 & -(D_{kl}^o)^{-\frac{1}{2}} f_k^T \\ 0 & D_{kl}^{-\frac{1}{2}} H_k D_{kl}^{-\frac{1}{2}} + I & D_{kl}^{-\frac{1}{2}} G_k \\ -f_k(D_{kl}^o)^{-\frac{1}{2}} & G_k^T D_{kl}^{-\frac{1}{2}} & 0 \end{bmatrix} \begin{bmatrix} (D_{kl}^o)^{\frac{1}{2}}\delta x_{k,l+1}^o \\ D_{kl}^{\frac{1}{2}}\delta x_{k,l+1} \\ \delta y_{k,l+1} \end{bmatrix} \\ = -\begin{bmatrix} (D_{kl}^o)^{-\frac{1}{2}}(r_{Dkl}^o + (x_{kl}^o)^{-1}(\underline{r}_{ckl}^o)') \\ D_{kl}^{-\frac{1}{2}}(r_{Dkl} + \underline{X}_{kl}^{-1}\underline{r}'_{ckl} - \overline{X}_{kl}^{-1}\overline{r}'_{ckl}) \\ r_{Pkl} \end{bmatrix}, \end{array} \tag{2.18}$$

where

$$D^o_{kl} \;=\; z^o_{kl}/x^o_{kl}, \quad D_{kl} \;=\; \underline{X}^{-1}_{kl}\underline{Z}_{kl} + \overline{X}^{-1}_{kl}\overline{Z}_{kl}.$$

The matrix in (2.18) is symmetric but indefinite, so it is appropriate to use a Bunch–Parlett factorization, which can be generated using frontal or multifrontal techniques, as described, for example, by Duff and Reid [7, 8]. This yields sparse factors, making it suitable for large problems.

The matrix can be singular, detected in practice by failure to complete the factorization due to lack of pivots above a specified threshold.

One reason could be rank deficiency of $[-f_k, G_k^T]$, but in this case the remaining rows and columns corresponding to linearly dependent equations are essentially null, and if $r_{Pkl} = 0$ the corresponding right-hand-side elements in (2.18) will also be null, so we still obtain a consistent solution of (2.18) in which the $\delta y^j_{k,l+1}$ corresponding to null columns can be chosen arbitrarily (they can be set to zero). It is therefore important to initialize the subproblem with $x^o_{k,o} = 1$ so that $r_{Pko} = 0$; the linearity of (2.12c) then ensures that all r_{Pkl} are zero.

Thus if the factorization is incomplete we first determine if the remaining residuals are sufficiently small. If so, the solution is acceptable, but otherwise the failure is due to indefiniteness of H_k and in this case we set $H_k = 0$ and re-solve the system.

Even if the factorization is completed, if (2.12) is not monotone $\|\delta\chi_{k,l+1}\|_\infty$ can become large compared with the distance to the bound, making $\alpha'_{k,l+1}$ correspondingly small. Hence if $\alpha'_{k,l+1} < \underline{\alpha}$ for some small $\underline{\alpha} > 0$ we also set $H_k = 0$. It is shown in [15] that with $H_k = 0$ the reduced matrix (after deletion of linearly dependent rows of $[-f_k, G_k^T]$ and corresponding columns) has a uniformly bounded inverse, so in this case we ignore the fixed pivot threshold.

Having obtained $\delta\chi_{k,l+1}$, we choose the step-length $\alpha'_{k,l+1}$ in (2.13) and (2.16) (see Appendix A for details) as the largest value in $[0, 1]$ for which

$$\begin{array}{rcl} \|r^o_{ck,l+1}\|_\infty & \le & \beta^o_{k,l+1} y^o_k \mu^o_{k,l+1}, \\ \|\underline{r}_{ck,l+1}\|_\infty & \le & \beta_{k,l+1} y^o_k \mu^o_{k,l+1}, \\ \|\overline{r}_{ck,l+1}\|_\infty & \le & \beta_{k,l+1} y^o_k \mu^o_{k,l+1}, \end{array} \tag{2.19}$$

where

$$\begin{array}{rcl} r^o_{ck,l+1} & = & x^o_{k,l+1} z^o_{k,l+1} - y^o_k \mu^o_{k,l+1}, \\ \underline{r}_{ck,l+1} & = & \underline{X}_{k,l+1}\underline{z}_{k,l+1} - y^o_k \mu_{k,l+1} e, \\ \overline{r}_{ck,l+1} & = & \overline{X}_{k,l+1}\overline{z}_{k,l+1} - y^o_k \mu_{k,l+1} e, \end{array} \tag{2.20}$$

and the $\beta^o_{kl}, \beta_{kl}, \gamma^o_{kl}$ are generated by a scheme similar to that described in [15]:

$$\beta^o_{kl} = \beta^o(1 - \delta_{kl}), \;\; \beta_{kl} = \beta(1 - \delta_{kl}). \tag{2.21}$$

$$\begin{array}{lll} \text{If } \; \alpha'_{k,l+1} \ge \tau \;\text{ or }\; \gamma^o_{kl} = \gamma, & \text{then set} & \gamma^o_{k,l+1} = \mu^o_{k,l+1}/\mu^o_{ko}, \\ & & \delta_{k,l+1} = \delta_{kl}\tau^{\frac{1}{2}}_{k,l+1} \\ & \text{else set} & \gamma^o_{k,l+1} = \gamma, \\ & & \delta_{k,l+1} = \delta_{kl}, \end{array} \tag{2.22}$$

where $\tau_{k,l+1} = \max\{1 - \tau, 1 - \alpha'_{k,l+1}(1 - \gamma^o_{k,l+1})\}$, $\beta^o \in (0,1), \beta \in (0,1)$, $\gamma^o_{ko} = \gamma \in (0,1), \delta_{ko} \in (0,1)$.

Since (2.12a), (2.12b) are linear, it also follows from (2.13) and (2.16) that

$$\|r_{Dk,l+1}\|_\infty = \|r_{Dkl}\|_\infty(1-\alpha'_{k,l+1}) \le \|r_{Dko}\|_\infty \mu^o_{k,l+1}/\mu^o_{k,o} \tag{2.23}$$

and similarly for $|r^o_{Dk,l+1}|$.

We show in section 3 that rules (2.21) and (2.22) ensure that $x_{k,l+1} \in \overset{o}{X}$, $z^o_{k,l+1} > 0, \underline{z}_{k,l+1} > 0, \overline{z}_{k,l+1} > 0$, and $\{\alpha'_{k,l+1}\}$ is bounded away from zero. Then from (2.16) and (2.22) we have $\{\mu^o_{kl}\} \to 0$, and hence from (2.19) and (2.23) all the residuals for (2.12a)–(2.12f) tend to zero. Thus by repeated iteration, (2.12a)–(2.12f) and hence (2.7) can be solved to arbitrarily high accuracy.

At first sight we could simply set $\mu_{kl} = \mu^o_{kl}$, hence also solving (2.5) to arbitrarily high accuracy, but as we shall see in section 3, because of its convexifying role it is important that μ_k does not tend to zero faster than Δx_{k+1}, and this is ensured by the following rule:

$$\begin{aligned} &\text{If} \quad y^o_k \mu_{k,l+1} \ge \|y^o_k, y_{k,l+1}\|_\infty \max\{\epsilon, \rho\|\Delta x_{k+1}\|^4\}, \\ &\quad \text{then set} \quad \gamma_{k,l+1} = \gamma^o_{k,l+1}, \\ &\quad \text{else set} \quad \gamma_{k,l+1} = 1, \end{aligned} \tag{2.24}$$

where $\epsilon \in (0,1)$, $\rho \in (0,1)$. Thus if the condition in (2.24) fails, μ is held constant until it is again satisfied, but otherwise μ is reduced by the same factor as μ^o.

The purpose of the subproblem is to generate a descent direction for the merit function, though the accuracy must also be progressively tightened as we approach a solution in order to obtain an ultimate quadratic convergence rate, and appropriate termination conditions are derived in the next section.

On termination we set

$$\begin{aligned} &\Delta x_{k+1} = x_{k,l+1} - x_k, \quad \mu_{k+1} = \mu_{k,l+1}, \\ &\eta_{k+1} = \|y^o_k, y_{k,l+1}\|^{-1}_\infty \{y^o_k, y_{k,l+1}, \underline{z}_{k,l+1}, \overline{z}_{k,l+1}\}, \end{aligned} \tag{2.25}$$

which ensures that $\|y^o_{k+1}, y_{k+1}\|_\infty = 1$ and $\mu_{k+1} \le \mu_k$.

To initialize the subproblem we set $x_{k,o} = x_k, \mu^o_{k,o} = \mu_{k,o} = \mu_k$, and normally $\eta_{k,o} = \eta_k$, but if $|\underline{r}^i_{cko}| > \beta_{ko} y^o_k \mu^o_{k,o}$ we reset $\underline{z}^i_{ko} = y^o_k \mu^o_{k,o}/\underline{X}^i_{k,o}$, $i = 1, 2, \ldots, n$, and similarly for $\overline{z}_{k,o}$.

We have already seen that we set $x^o_{k,o} = 1$ so that $r_{Pko} = 0$, and we also choose $z^o_{ko} = M_k y^o_k - f^T_k y_k$ so that $r^o_{Dko} = 0$. Again linearity of (2.12a) ensures that $r^o_{Dkl} = 0$ for all l.

We choose

$$M'_k = M_k y^o_k = \mu_o + \underline{M}\|f_k\|_1 \tag{2.26}$$

for some $\underline{M} > 0$. It follows that $r^o_{cko} > 0$, and we compute

$$\beta^o = \max\{\beta, r^o_{cko}/y^o_k \mu_k (1-\delta_{ko})\} \tag{2.27}$$

so that (2.19) is satisfied for $l = 0$.

We should like to choose $\underline{M}$ so that x^o_{kl} is always decreased, but an a priori choice to guarantee this is not possible. Hence, if on termination of the subproblem we have $x^o_{k,l+1} > 1 - \bar{\epsilon}$ for some small $\bar{\epsilon} \in (0,1)$, we set $g^o_{k+1} = 0$ and $H_{k+1} = 0$ in the next subproblem, which puts all the emphasis on attaining feasibility. If on termination we then obtain $x^o_{k,l+1} \le 1 - \bar{\epsilon}$, we can revert to normal subproblems, but we also increase $\underline{M}$ since this was clearly too small. However, if repeated failures eventually result in $\|\Delta x_k\| \le \epsilon$ we conclude that we are trapped in a neighborhood of a nonzero minimum of constraint violations and terminate; the only recourse is to try a different starting point.

2.2. The outer iteration. To initialize the algorithm we have a user-supplied estimate $x_o \in \overset{o}{X}$ and choose $y_o^o = 1, y_o = 0, \underline{z}_o = \mu_o \underline{X}_o^{-1} e, \overline{z}_o = \mu_o \overline{X}_o^{-1} e$. These initial choices then make all residuals except r_{Doo} zero for the first subproblem. There is no clear basis for choosing μ_o, so we leave this as a user-defined parameter, with a default value based on experimentation.

For succeeding iterations the iterates are defined by (2.3) and (2.25), where, as described earlier, we choose $\alpha_{k+1} \in (0, 1]$ to obtain a sufficient decrease of the merit function (2.9).

To obtain the form of this sufficient decrease condition, we first note that, since the $f^j(x), j = 0, 1, \ldots, m$, are continuously differentiable on X, we have

$$f_{k+1}^j = f_k^j + \alpha_{k+1} \Delta x_{k+1}^T g_k^j + O(\alpha_{k+1}^2),$$

and using (2.12c),

$$|f_{k+1}^j| = |f_k^j|(1 - \alpha_{k+1}(1 - x_{k,l+1}^o)) + O(\alpha_{k+1}^2),\ j = 1, 2, \ldots, m. \tag{2.28}$$

Also

$$L_{k+1} = L_k + \alpha_{k+1} y_k^o \mu_{k,l+1} \Delta x_{k+1}^T (\underline{X}_k^{-1} - \overline{X}_k^{-1}) e + O(\alpha_{k+1}^2),$$

and from (2.20),

$$\begin{aligned} \underline{X}_k^{-1} \underline{Z}_{k,l+1} (x_{k,l+1} - x_k) &= \underline{X}_k^{-1} (\underline{X}_{k,l+1} - \underline{X}_k) \underline{z}_{k,l+1} \\ &= \underline{X}_k^{-1} (\underline{r}_{ck,l+1} + y_k^o \mu_{k,l+1} e) - \underline{z}_{k,l+1}, \end{aligned}$$

and similarly for $\overline{X}_k^{-1} \overline{Z}_{k,l+1} (x_{k,l+1} - x_k)$, whence

$$\begin{aligned} & y_k^o \mu_{k,l+1} (\underline{X}_k^{-1} - \overline{X}_k^{-1}) e \\ &\quad = (\underline{X}_k^{-1} \underline{Z}_{k,l+1} + \overline{X}_k^{-1} \overline{Z}_{k,l+1})(x_{k,l+1} - x_k) \\ &\qquad + \underline{z}_{k,l+1} - \overline{z}_{k,l+1} - \underline{X}_k^{-1} \underline{r}_{ck,l+1} + \overline{X}_k^{-1} \overline{r}_{ck,l+1}. \end{aligned} \tag{2.29}$$

Thus, using (2.12), (2.28), and (2.29) we have

(2.30)

$$\begin{aligned} & \Delta P_{k+1} \\ & = P(x_k, y_{k+1}^o, \tilde{y}_{k+1}, \mu_{k+1}) - P(x_{k+1}, y_{k+1}^o, \tilde{y}_{k+1}, \mu_{k+1}) \\ & = y_{k+1}^o \{ f_k^o - f_{k+1}^o - \mu_{k+1} (L_k - L_{k+1}) \} \\ & \quad + \sum_{j=1}^m \tilde{y}_{k+1}^j (|f_k^j| - |f_{k+1}^j|) \\ & = \alpha_{k+1} y_{k+1}^o \Delta x_{k+1}^T \{ \mu_{k+1} (\underline{X}_k^{-1} - \overline{X}_k^{-1}) e - g_k^o \} \\ & \quad + \alpha_{k+1} (1 - x_{k+1}^o) \sum_{j=1}^m \tilde{y}_{k+1}^j |f_k^j| + O(\alpha_{k+1}^2) \\ & = \alpha_{k+1} (\psi_{k+1} - \| y_k^o, y_{k,l+1} \|_\infty^{-1} \Delta x_{k+1}^T (r_{D,k+1} + \underline{X}_k^{-1} \underline{r}_{c,k+1} - \underline{X}_k^{-1} \underline{r}_{c,k+1})) \\ & \quad + O(\alpha_{k+1}^2), \end{aligned}$$

where

$$
\begin{aligned}
\psi_{k+1} &= \Delta x_{k+1}^T \bar{H}_{k+1} \Delta x_{k+1} \\
&\quad + (1 - x^o_{k,l+1}) \sum_{j=1}^{m} (\tilde{y}^j_{k+1} |f^j_k| - y^j_{k+1} f^j_k), \\
\overline{H}_{k+1} &= \|y^o_k, y_{k,l+1}\|_\infty^{-1} H_k + \overline{D}_{k+1}, \\
\overline{D}_{k+1} &= \|y^o_k, y_{k,l+1}\|_\infty^{-1} (\underline{X}_k^{-1} \underline{Z}_{k,l+1} + \overline{X}_k^{-1} \overline{Z}_{k,l+1}) \\
&= \underline{X}_k^{-1} \underline{Z}_{k+1} + \overline{X}_k^{-1} \overline{Z}_{k+1}.
\end{aligned} \tag{2.31}
$$

We show in the next section that the algorithm terminates if at each step we satisfy the "descent condition"

$$
\psi_{k+1} \geq \underline{\rho} \Delta x_{k+1}^T \bar{D}_{k+1} \Delta x_{k+1}, \quad \underline{\rho} \in (0,1), \tag{2.32}
$$

and the "sufficient decrease condition"

$$
\Delta P_{k+1} \geq \delta \alpha_{k+1} \psi_{k+1} > 0, \quad \delta \in (0, 1/2), \tag{2.33}
$$

together with $(y^o_{k+1})^{-1} \tilde{y}_{k+1} \geq (y^o_k)^{-1} \tilde{y}_k$ if $y^o_{k+1} \mu_{k+1} \leq \max\{\epsilon, \rho \|\Delta x_{k+1}\|^4\}$.

Clearly (2.32) is satisfied if H_k is nonnegative definite and the second term of the expression for ψ_{k+1} in (2.31) is nonnegative, and we ensure the latter by using the following rule:

$$
\begin{aligned}
&\text{If } \; y^o_{k+1} \mu_{k+1} > \max\{\epsilon, \rho \|\Delta x_{k+1}\|^4\}, \text{ then set } \; \tilde{y}^j_{k+1} = \omega \text{ for all } j, \\
&\qquad \text{else set } \; \tilde{y}^j_{k+1} = \max\{\omega |y^j_{k+1}|, \tilde{y}^j_k y^o_{k+1} / y^o_k\},
\end{aligned} \tag{2.34}
$$

where $\omega > 1$. Then if condition (2.32) fails we again set $H_k = 0$ and re-solve the subproblem.

Then (2.32) ensures that $\psi_{k+1} > 0$, and as already noted we can make the residuals $r_{D,k+1}, \underline{r}_{c,k+1}, \overline{r}_{c,k+1}$ as small as desired by repeated iteration, so we iterate until

$$
\begin{aligned}
&\Delta x_{k+1}^T (r_{Dk,l+1} + \underline{X}_k^{-1} \underline{r}_{ck,l+1} - \overline{X}_k^{-1} \overline{r}_{ck,l+1}) \\
&\quad \leq \frac{1}{4} (1 - 2\delta) \psi_{k+1} \|y^o_k, y_{k,l+1}\|_\infty,
\end{aligned} \tag{2.35}
$$

which leaves a similar margin for the terms of $O(\alpha^2_{k+1})$ in (2.30).

To keep α_{k+1} bounded away from zero we could then use any of the standard reduction rules to satisfy (2.33), and we have used the simple rule of Armijo [1]: $\alpha_{k+1} = \theta^{i_{k+1}}$, where $\theta \in (0,1)$ and i_{k+1} is the smallest nonnegative integer for which (2.33) is satisfied. Each reduction then requires only re-evaluation of the functions $f^o(x), f(x)$, and $L(x)$.

This is enough to secure global convergence, but to achieve a final superlinear rate of convergence we need further conditions.

First we must ensure that H_k is not reset to zero when we are sufficiently close to a solution, and Lemma 1 of the next section shows that this is the case if we choose $\omega > 1$ (rather than $\omega = 1$) in (2.34) above.

Second, we must ensure that μ decreases sufficiently fast, and again we show in the next section that this is achieved by imposing the additional subproblem termination conditions

$$
\begin{aligned}
y^o_{k+1} \mu^o_{k+1} &\leq \max\{\epsilon^o, \sigma \|\Delta x_{k+1}\|^4\}, \\
y^o_{k+1} \mu_{k+1} &\leq \max\{\epsilon, \sigma \|\Delta x_{k+1}\|^4\}.
\end{aligned} \tag{2.36}
$$

Of course we must have $\sigma > \rho$, and by choosing σ suitably large we can ensure that these extra conditions become effective only close to a solution.

Finally, we need $\alpha_{k+1} = 1$ for all k sufficiently large, and as first pointed out by Maratos [13], this may not occur if α_{k+1} is chosen to reduce an l_1 merit function like (2.9).

To circumvent this problem, Mayne and Polak [14] proposed a "correction step," $\Delta\bar{x}_{k+1}$, obtained by orthogonal projection of $(x_k + \Delta x_{k+1})$ onto the current active set, followed by a curvilinear search,

$$x_{k+1} = x_k + \alpha_{k+1}\Delta x_{k+1} + \alpha_{k+1}^2 \Delta\bar{x}_{k+1}, \tag{2.37}$$

to satisfy the sufficient decrease condition.

Chamberlain et al. [5] proposed the so-called "watch-dog technique" as an alternative solution. In its simplest form a step with $\alpha_{k+1} = 1$ is tried, followed by up to $t \geq 1$ normal steps. If any of these produce a sufficient decrease of the merit function from its value at x_k, this iterate is accepted and the process repeated. Otherwise, after t failures these steps are rejected and a normal step is taken from x_k.

Either scheme produces the desired result. The correction step involves additional work, so Mayne and Polak used a test to limit its use to points close to a solution. The watch-dog technique eventually yields $\alpha_{k+1} = 1$ without extra work, but many steps may be rejected in earlier iterations.

In fact each normal step of the algorithm makes a projection onto the linearized constraints, so we use a simple "double-step" modification of Mayne and Polak's technique: If the step with $\alpha_{k+1} = 1$ fails to satisfy (2.33), we take a second step with $\alpha_{k+2} = 1$, then test the condition

$$\Delta P_{k+1} + \Delta P_{k+2} \geq \delta\psi_{k+1}. \tag{2.38}$$

If this is satisfied we set $x_{k+1} = x_k + \Delta x_{k+1} + \Delta x_{k+2}$ and continue. Otherwise we recompute

$$x_{k+1} = x_k + \alpha_{k+1}\Delta x_{k+1} + \alpha_{k+1}^2 \Delta x_{k+2}, \tag{2.39}$$

with α_{k+1} chosen to satisfy (2.33). Clearly this is always possible since the additional terms introduced are $O(\alpha_{k+1}^2)$.

This technique also allows larger steps in following a curved constraint if x_k is close to such a constraint at some distance from the solution, but on the other hand if α_{k+1} is small the second step is largely wasted. Thus we do not use a double-step systematically but try it only if $|f_k^j| \leq \bar{\epsilon}$, for some j, and then use (2.39) rather than (2.3) only if $\Delta P_{k+2} > 0$.

To terminate the algorithm, we use the failing condition $x_{k+1}^o > 1 - \bar{\epsilon}$, $\|\Delta x_{k+1}\| \leq \epsilon^o$, as discussed earlier, or the conditions

$$\begin{aligned} &\max\{\|r_{D,k+1}\|_\infty, \|f_{k+1}\|_\infty, \|\underline{r}_{c,k+1,o}\|, \|\overline{r}_{c,k+1,o}\|\} \leq \epsilon^o, \\ &y_{k+1}^o \mu_{k+1} \leq \epsilon, \end{aligned} \tag{2.40}$$

which indicates an ϵ-optimal solution.

This completes the description of the algorithm, but a detailed statement of the main algorithm is given in Appendix B and that of the subproblem algorithm in Appendix C.

3. Convergence. We consider finite termination in section 3.1 and rates of convergence in section 3.2.

3.1. Finite termination. In this section we give two theorems, one for finite termination of the subproblem and the other for that of the complete algorithm.

THEOREM 1. *The sequence* $\{\chi_{kl}\}, l = 0, 1, 2, \ldots,$ *is well defined, and for any* $\epsilon > 0$ *terminates at a point satisfying* $\|\Delta x_{kl}\|_\infty \leq \epsilon^o$ *with* $x^o_{kl} > 1 - \bar{\epsilon}$, *or* (2.40), *or* (2.35) *and* (2.36).

Proof. Suppose that χ_{kl} satisfies the interior point conditions

$$x_{kl} \in \overset{o}{X}, \quad x^o_{kl} > 0, \quad z^o_{kl} > 0, \quad \underline{z}_{kl} > 0, \quad \overline{z}_{kl} > 0, \quad \mu^o_{kl} > 0, \tag{3.1}$$

and also satisfies (2.19).

Then the solution of (2.17)–(2.18) is well defined, if necessary by dropping linearly dependent equations and/or setting $H_k = 0$, and Appendix A shows that there is always an $\alpha'_{k,l+1} \in (0, 1]$ which satisfies (2.19), so $\chi_{k,l+1}$ is well defined. Since $\gamma^o_{kl} \in (0, 1)$ it also follows from (2.16) that $\mu^o_{k,l+1} > 0$, and hence from (2.19), (2.20), and continuity that $\chi_{k,l+1}$ satisfies (3.1). Since $\chi_{k,o}$ satisfies (3.1) and (2.19) it follows by induction that the whole sequence $\{\chi_{kl}\}$ is well defined and satisfies these conditions.

If $\alpha'_{k,l+1} < \underline{\alpha}$ we reset $H_k = 0$, and it was shown in Sargent [15] that $\{\alpha'_{k,l+1}\}$ is then bounded away from zero, and hence from (2.16) and (2.22), $\mu^o_{kl} \to 0$ as $l \to \infty$. It then follows from (2.19) and (2.23) that the residuals of (2.12a)–(2.12f) tend to zero as $l \to \infty$.

Now suppose that $\{\|\Delta x_{kl}\|_\infty\}$ is bounded away from zero. Then we cannot have an infinite subsequence satisfying

$$y^o_k \mu_{kl'} > \|y^o_k, y_{kl'}\|_\infty \max(\epsilon, \rho \|\Delta x_{kl'}\|^4),$$

since this implies $\gamma_{kl'} = \gamma^o_{kl'}$ and $\{\mu_{kl'}\} \to 0$, producing a contradiction. Hence there is an $\bar{l} < \infty$ such that (2.36) is satisfied and $\mu_{kl} = \mu_{k\bar{l}} > 0$ for all $l \geq \bar{l}$. But from (2.31), (2.19), (2.20), and (2.21) we have

$$\begin{aligned} \bar{D}^i_{k+1} &\geq (\underline{X}^i_k \underline{X}^i_{k+1})^{-1} (\underline{X}^i_{k+1} \underline{Z}^i_{k+1}) \\ &\geq (1 - \beta) y^o_{k+1} \mu_{k+1} / (b^i - a^i)^2, \qquad i = 1, 2, \ldots, n, \end{aligned} \tag{3.2}$$

so from (2.32) the sequence $\{\psi_{kl}\}$ is bounded away from zero, and it follows that (2.35) must eventually be satisfied.

Otherwise there must be an infinite subsequence for which $\{\|\Delta x_{kl'}\|_\infty\} \to 0$, and from (2.24) it follows that $\{\mu_{kl}\} \to 0$, so eventually we satisfy $\|\Delta x_{kl}\|_\infty \leq \epsilon^o$ with $x^o_{kl} > 1 - \bar{\epsilon}$, or (2.40). □

If Ω is not empty, then for any $\epsilon > 0$ there is a neighborhood Ω_ϵ of Ω on which the system

$$\left\{ \begin{array}{lcl} r_D &=& g^o(x) y^o + G(x) y - \underline{z} + \overline{z}, \\ r_P &=& f(x), \\ x^i - a^i \geq 0, & z^i \geq 0, & z^i (x^i - a^i) = y^o \mu \\ b^i - x^i \geq 0, & \overline{z}^i \geq 0, & \overline{z}^i (b^i - x^i) = y^o \mu \end{array} \right\} i = 1, 2 \ldots, n \tag{3.3}$$

is satisfied with $\max\{\|r_D\|_\infty, \|r_P\|_\infty, y^o \mu\} \leq \epsilon$, and of course Ω_ϵ coincides with Ω if $\epsilon = 0$.

THEOREM 2. *If* $x_o \in \overset{o}{X}$, *the sequence* $\{\zeta_k\}$ *generated by the algorithm is well defined,* $\{x_k\}$ *remains in* $\overset{o}{X}$, *and the algorithm terminates for any* $\epsilon \geq 0, \epsilon^o > 0$.

If $x^o_k \leq 1 - \bar{\epsilon}$ *on termination, the termination point* ζ_k *tends to* Ω_ϵ *as* $\epsilon^o \to 0$.

Proof. Suppose that ζ_k satisfies (3.1). Then from Theorem 1, either the algorithm terminates with $\|\Delta x_{k+1}\| \le \epsilon^o, x^o_{k,l+1} > 1-\bar\epsilon$, or Δx_{k+1} and η_{k+1} are well defined and satisfy (2.35), while $\{x_k + \Delta x_{k+1}, \eta_{k+1}\}$ satisfies (3.1). We showed in the last section that $\alpha_{k+1} \in (0,1]$ is always well defined, and hence ζ_{k+1} is well defined and also satisfies (3.1). Since ζ_o satisfies (3.1) it follows that the whole sequence $\{\zeta_k\}$ is well defined, and $\{x_k\}$ remains in $\overset{o}{X}$.

Since $\gamma^o_{ko} = \gamma$ it follows from (2.16) that $\{y^o_k \mu^o_k\} \to 0$, and from (2.19) and (2.23) that $\|r_{Dk}\|, \|\underline{r}_{ck}\|, \|\overline{r}_{ck}\|$ also tend to zero.

Then, if there is an infinite subsequence such that

$$y^o_{k'}\mu_{k'} > \max\{\epsilon, \rho\|\Delta x_{k'}\|^4\},$$

it similarly follows that $\{y^o_k \mu_k\} \to 0$. If $\epsilon > 0$, this is a contradiction, so such a subsequence cannot occur, but if $\epsilon = 0$, it follows that $\{\|\Delta x_{k'}\|\} \to 0$, and if $x^o_{k',l+1} \le 1-\bar\epsilon$ for k' sufficiently large, $\{\|f_{k'}\|\} \to 0$ from (2.12c) and (2.40) is satisfied, so the algorithm terminates.

Otherwise there is a $\bar k < \infty$ such that $y^o_k \mu_k \le \max\{\epsilon, \rho\|\Delta x_k\|^4\}$ for all $k \ge \bar k$. Then from (2.24) and (2.16) we have $\mu_k = \mu_{\bar k}, k \ge \bar k$, and $\{y^o_k\}$ is bounded away from zero.

Now consider the function

$$\bar P_k = (y^o_k)^{-1} P(x_k, y^o_k, \tilde y_k, \mu_k) + \mu_k \bar L - (y^o_k)^{-1} \sum_{j=1}^m \tilde y^j_k \bar f^j, \tag{3.4}$$

where

$$\begin{aligned} \bar L &= \max\{L(x), x \in \overset{o}{X}\}, \\ \bar f^j &= \max\{|f^j(x)|, x \in X\}, \quad j = 1, 2, \ldots, m. \end{aligned}$$

Since X is compact and $L(x)$ is strictly concave on $\overset{o}{X}$, these quantities are well defined and $\{\bar P_k\}$ is uniformly bounded below. Further, from (2.16) $\{\mu_k\}$ is a nonincreasing sequence, and from (2.34) $\{(y^o_k)^{-1}\tilde y_k\}$ is a nondecreasing sequence, so from (2.9) and (3.4) we have

$$\begin{aligned} &\bar P_k - \bar P_{k+1} \\ &= (y^o_k)^{-1} P(x_k, y^o_k, \tilde y_k, \mu_k) - (y^o_{k+1})^{-1} P(x_k, y^o_{k+1}, \tilde y_{k+1}, \mu_{k+1}) \\ &\quad + (y^o_{k+1})^{-1}\{P(x_k, y^o_{k+1}, \tilde y_{k+1}, \mu_{k+1}) - P(x_{k+1}, y^o_{k+1}, \tilde y_{k+1}, \mu_{k+1})\} \\ &\quad + \bar L(\mu_k - \mu_{k+1}) + \sum_{j=1}^m \bar f^j \{(y^o_{k+1})^{-1}\tilde y^j_{k+1} - (y^o_k)^{-1}\tilde y^j_k\} \\ &= (y^o_{k+1})^{-1}\Delta P_{k+1} + (\bar L - L_k)(\mu_k - \mu_{k+1}) \\ &\quad + \sum_{j=1}^m (\bar f^j - |f^j_k|)\{(y^o_{k+1})^{-1}\tilde y^j_{k+1} - (y^o_k)^{-1}\tilde y^j_k\} \\ &\ge (y^o_{k+1})^{-1}\Delta P_{k+1} > 0. \end{aligned} \tag{3.5}$$

Hence $\{\Delta P_{k+1}\} \to 0$, and from (2.33), $\{\alpha_{k+1}\psi_{k+1}\} \to 0$ as $k \to \infty$.

Suppose that $\{\psi_{k+1}\}$ does not tend to zero. Then there is an infinite subsequence $\{\psi_{k'}\}$ such that $\psi_{k'} \ge \underline{\psi} > 0$ for all k', and hence there is a fixed $\bar\alpha \in (0,1]$, independent of k', such that (with $k' = k+1$) all the terms of $O(\alpha^2_{k+1})$ in (2.30) sum

to at most $\frac{1}{4}(1-2\delta)\alpha_{k+1}\underline{\psi}$ for any $\alpha_{k+1} \in [0, \bar{\alpha}]$. To satisfy (2.33) the Armijo rule therefore determines $\alpha_{k'} \geq \theta\bar{\alpha} > 0$ and otherwise $\alpha_{k+1} = 1$, so $\alpha_{k'}\psi_{k'} \geq \theta\bar{\alpha}\underline{\psi} > 0$, contradicting our hypothesis, and in fact $\{\psi_{k+1}\} \to 0$.

But then from (2.32) and (3.2) it follows that $\{\|\Delta x_k\|\} \to 0$. If $\epsilon = 0$, this is a contradiction, so this situation cannot occur, but if $\epsilon > 0$, it follows that $y_k^o \mu_k \leq \epsilon$ for all k sufficiently large.

Hence we have the following:

(a) If $\epsilon > 0, \zeta_k \in \Omega_\epsilon$ for all k sufficiently large and $\{\|\Delta x_k\|\} \to 0$.

(b) If $\epsilon = 0$, there is an infinite subsequence for which $\{\|\Delta x_{k'}\|\} \to 0$ and $\{\zeta_{k'}\}$ satisfies (2.2) in the limit as $k' \to \infty$. From continuity, the limit point $\bar{\zeta} \in \Omega$, so the termination point of the algorithm tends to Ω as $\epsilon^o \to 0$. □

3.2. Rates of convergence. In this section we give two theorems on rates of convergence, one to an ϵ-optimal point and the other to an exact solution. In both cases we assume that $\epsilon^o = 0$ and that (2.18) is solvable if its coefficient matrix is nonsingular.

We need some preliminary results, starting with a result on the following generalized linear complementarity problem:

$$
\left\{
\begin{array}{lcl}
r_D &=& g^o + Hx + Gy - \underline{z} + \overline{z} = 0, \\
r_P &=& f + G^T x = 0, \\
x - a \geq 0, \quad \underline{z} \geq 0, & & \underline{z}^T(x-a) = 0, \\
b - x \geq 0, \quad \overline{z} \geq 0, & & \overline{z}^T(b-x) = 0,
\end{array}
\right. \tag{3.6}
$$

with solution set S.

For each $\zeta \in S$ we can then define

$$
\begin{array}{lllllll}
\underline{B} &=& \{i \mid x^i > a^i\}, & & \overline{B} &=& \{i \mid x^i < b^i\}, \\
\underline{N} &=& \{i \mid \underline{z}^i > 0\}, & & \overline{N} &=& \{i \mid \overline{z}^i > 0\}, \\
\underline{J} &=& \{i \mid x^i = a^i, \underline{z}^i = 0\}, & & \overline{J} &=& \{i \mid x^i = b^i, \overline{z}^i = 0\},
\end{array}
$$

and also the "maximal complementary set" $\hat{S} \subset S$, for which $|\underline{B}(\zeta)| + |\overline{B}(\zeta)| + |\underline{N}(\zeta)| + |\overline{N}(\zeta)|$ is maximized.

Then (cf. [15]) all members of $\hat{S}$ have the same index sets $\hat{\underline{B}}, \hat{\overline{B}}, \hat{\underline{N}}, \hat{\overline{N}}, \hat{\underline{J}}, \hat{\overline{J}}$.

The matrix H is said to be *monotone* for (3.6) if $v^T H v \geq 0$ (*strictly monotone* if $v^T H v > 0$) for all v such that $\bar{G}^T v = 0$, where $\bar{G} = [G, I_{\underline{N}}, I_{\overline{N}}]$, $I_{\underline{N}}$ has columns $e_i, i \in \hat{\underline{N}}$, and $I_{\overline{N}}$ has columns $e_i, i \in \hat{\overline{N}}$.

It is well known that if H is monotone, the solution set S of (3.6) is not empty if there exists $\zeta = (x, y, \underline{z}, \overline{z})$ such that $r_D(\zeta) = 0, r_P(\zeta) = 0$, $a \leq x \leq b$, $\underline{z} \geq 0$, $\overline{z} \geq 0$.

The next lemma gives a result for this problem, analogous to that of Mangasarian and Shiau [12] for the standard monotone LCP.

LEMMA 1. *If the solution set of problem* (3.6) *is not empty, and H is monotone for* (3.6)*, there exists a fixed $\tau < \infty$ such that*

$$
\begin{aligned}
\|\zeta - \hat{\zeta}\| \leq \tau\{&\|r_D(\zeta)\| + \|r_P(\zeta)\| \\
&+ \|I_{\underline{B}}^T \underline{z}\| + \|I_{\overline{B}}^T \overline{z}\| + \|I_{\underline{N}}^T \underline{X}\| + \|I_{\overline{N}}^T \overline{X}\| + \sqrt{m}\}
\end{aligned} \tag{3.7}
$$

for all ζ such that $a \leq x \leq b, \underline{z} \geq 0, \overline{z} \geq 0$, where

$$
\begin{array}{lcl}
m &=& \underline{z}^T(x-a) + \overline{z}^T(b-x), \\
\hat{\zeta} &=& \arg\min_{\zeta' \in S} \|\zeta' - \zeta\|.
\end{array}
$$

If the problem has a strictly complementary solution, the term $\sqrt{m}$ *in* (3.7) *can be omitted.*

Proof. A proof for simple nonnegativity bounds on x is given in [15]. The extension to deal with general upper and lower bounds is obvious. ∎

LEMMA 2. *For any* ζ_k *sufficiently close to a local minimizer of problem* (2.1)*, the matrix in* (2.18) *is nonsingular and the descent condition* (2.32) *is satisfied.*

Proof. Since $\hat{f} = 0$ we have from (2.12a) and (2.26) that $z^o_{k+1} \geq \mu_o/2$ for all ζ_k close enough to $\hat{\zeta} \in \Omega$, and from (2.12d), (2.19), and (2.36), we have

$$x^o_{k,l+1} \leq 2(1+\beta^o) y^o_{k+1} \mu^o_{k+1}/\mu^o_o = O(\|\Delta x_{k+1}\|^4), \tag{3.8}$$

so $1 - x^o_{k+1} > \bar{\epsilon}$.

Now at any local minimizer $\hat{\zeta} \in \Omega$ we have $v^T \hat{H} v \geq 0$ for all $v \in \Re^n$ such that $\bar{G}^T v = 0$, where $\bar{G} = [\hat{G}, I_N, I_{\tilde{N}}]$. Also $\bar{D}^i_k > 0$ for all k, and $\bar{D}^i_k \to \infty$ as $\zeta_k \to \hat{\zeta}$ for any $i \in I_{\underline{N}}$ or $i \in I_{\overline{N}}$, so we have

$$v^T \bar{H}_{k+1} v \geq \underline{\rho} v^T \bar{D}_{k+1} v \tag{3.9}$$

for any ζ_k sufficiently close to $\hat{\zeta}$ and all $v \in \Re^n$ such that $G^T_k v = 0$.

From Lemma 3 in [15] it follows that the matrix in (2.18) has a uniformly bounded inverse, and $\Delta\chi_{k,l+1}$ is well defined. We can also choose $\lambda > 0$ large enough to ensure that

$$v^T(\bar{H}_{k+1} + \lambda G_k G^T_k) v \geq v^T \bar{D}_{k+1} v \ \text{ for all } \ v \in \Re^n. \tag{3.10}$$

Now from (2.31) and (2.12c) we have

$$\begin{aligned} \psi_{k+1} = {} & \Delta x^T_{k+1}(\bar{H}_{k+1} + \lambda G_k G^T_k)\Delta x_{k+1} \\ & -\lambda(1 - x^o_{k+1})^2 \|f_k\|^2 \\ & +(1 - x^o_{k+1}) \sum_{j=1}^{m} (\tilde{y}^j_{k+1} |f^j_k| - y^j_{k+1} f^j_k). \end{aligned} \tag{3.11}$$

But from (2.34),

$$\sum_{j=1}^{m} (\tilde{y}^j_{k+1} |f^j_k| - y^j_{k+1} f^j_k) \geq \sum_{j=1}^{m} \tilde{y}^j_{k+1} |f^j_k| (\omega - 1)/\omega, \tag{3.12}$$

and $\{\tilde{y}^j_{k+1}\}$ is bounded away from zero for each j, so the last term in (3.11) exceeds the second term for all ζ_k sufficiently close to $\hat{\zeta}$, and it follows from (3.10) that (2.32) is then satisfied. ∎

LEMMA 3. *If* $\psi_{k+1} = \Omega(\|\Delta x_{k+1}\|^2), x^o_{k,l+1} = o(\|\Delta x_{k+1}\|), \|\Delta x_{k+2}\| = o(\|\Delta x_{k+1}\|)$ *and* H_k *is not reset to zero, then* $\alpha_{k+1} = 1$ *for all* k *sufficiently large.*

Proof. Suppose that $\alpha_{k+1} = 1$.

Then from (2.12c) we have $f_{k+1} = O(\|\Delta x_{k+1}\|^2)$, and from (2.3), (2.9), (2.9), (2.28), and (2.29) we have

$$
\begin{aligned}
\Delta P &= y^o_{k+1}(f^o_k - f^o_{k+1}) - y^o_{k+1}\mu_{k+1}(L_k - L_{k+1}) \\
&\quad + \sum_{j=1}^{m} \tilde{y}^j_{k+1}(|f^j_k| - |f^j_{k+1}|) \\
&= \sum_{j=1}^{m} (\tilde{y}^j_{k+1}|f^j_k| - y^j_{k+1}f^j_k) - \sum_{j=1}^{m} (\tilde{y}^j_{k+1}|f^j_{k+1}| - y^j_{k+1}f^j_{k+1}) \\
&\quad + \frac{1}{2}\Delta x^T_{k+1}\bar{H}_k \Delta x_{k+1} \\
&\quad - \Delta x^T_{k+1}(r_{Dk,l+1} + \underline{X}_k^{-1}\underline{r}_{ck,l+1} - \overline{X}_k^{-1}\overline{r}_{ck,l+1}) \\
&\quad + o(\|\Delta x_{k+1}\|^2) \\
&\geq \left(\delta + \frac{1}{4}(1-2\delta)\right)\psi_{k+1} + \frac{1}{2}\sum_{j=1}^{m}(\tilde{y}^j_{k+1}|f^j_k| - y^j_{k+1}|f^j_k|) \\
&\quad - \sum_{j=1}^{m}(\tilde{y}^j_{k+1}|f^j_{k+1}| - y^j_{k+1}f^j_{k+1}) + o(\|\Delta x_{k+1}\|^2),
\end{aligned} \tag{3.13}
$$

whence

$$
\Delta P_{k+1} + \Delta P_{k+2} \geq \left(\delta + \frac{1}{4}(1-2\delta)\right)\psi_{k+1} + o(\|\Delta x_{k+1}\|^2), \tag{3.14}
$$

and (2.38) is satisfied for any $\delta \in (0, \frac{1}{2})$ and all k sufficiently large. Since eventually $\|f_k\| \leq \bar{\epsilon}$, a double-step is eventually always used and the result follows. ∎

For the next theorem we define the ϵ-neighborhood $\tilde{N}_\epsilon$ of $(\tilde{\Omega}, 0)$ for any $\epsilon > 0$ and any nonempty isolated connected subset $\tilde{\Omega} \subset \Omega$ as the set of all (ζ, μ) satisfying

$$
\begin{aligned}
&g^o(x)y^o + G(x)y - \underline{z} + \overline{z} = 0, \\
&\underline{X}\underline{z} = \overline{X}\overline{z} = y^o\mu e, \quad \|\eta\|_\infty = 1, \\
&\|f(x)\| \leq \epsilon, \quad y^o\mu \leq \epsilon, \quad y^o \geq 0, \quad \mu \geq 0.
\end{aligned} \tag{3.15}
$$

It follows from Theorem 10 of Fiacco and McCormick [9] that $(\tilde{N}_\epsilon - \tilde{\Omega})$ is not empty if $\tilde{\Omega}$ is a set of local minimizers of problem (2.1).

THEOREM 3. *If $\epsilon^o = 0$ and for some $\bar{k} < \infty$, $\zeta_{\bar{k}}$ is sufficiently close to an isolated connected subset $\tilde{\Omega} \subset \Omega$ of local minimizers of problem* (2.1)*, then for any $\epsilon > 0$ sufficiently small, either the sequence $\{\zeta_k\}$ terminates at some $(\tilde{\zeta}, \tilde{\mu}) \in \underline{N}_\epsilon$ or $\mu_k = \tilde{\mu} > 0$ for all k sufficiently large and $\{\zeta_k\} \to \tilde{\zeta}, \{x_k\} \to \tilde{x}$, both at a Q-superlinear rate.*

If in addition there is a neighborhood of $\tilde{x}$ on which

$$
\|f^j_{xx}(x) - f^j_{xx}(\tilde{x})\| \leq K\|x - \tilde{x}\|^\nu, \quad j = 0, 1, \ldots, m, \tag{3.16}
$$

for some $\nu \in (0, 1]$ and $K < \infty$, then the Q-order of convergence of $\{\zeta_k\}$ is at least $(1 + \nu)$, while that of $\{x_k\}$ is at least $\min\{1 + \nu, 1 + \sqrt{5}/2\}$.

Proof. For ζ_k sufficiently close to $\tilde{\Omega}$, from (2.14), (2.19), (2.23), (2.36), and (3.8)

with $\alpha_{k+1} = 1$ we have

$$\begin{aligned} r_{D,k+1,o} &= r_{D,k,l+1} + \sum_{j=0}^{m} y_{k+1}^j (g_{k+1}^j - g_k^j - f_{xx}^j(x_k)\Delta x_{k+1}) \\ &\quad + \sum_{j=0}^{m} (y_{k+1}^j - y_k^j) f_{xx}^j(x_k)\Delta x_{k+1} \\ &= o(\|\Delta x_{k+1}\|), \\ f_{k+1} &= f_k(1 - x_{k+1}^o) + G_k^T \Delta x_{k+1} + O(\|\Delta x_{k+1}\|^2) \\ &= O(\|\Delta x_{k+1}\|^2), \\ \underline{r}_{c,k+1,o} &= O(\|\Delta x_{k+1}\|^4), \\ \overline{r}_{c,k+1,o} &= O(\|\Delta x_{k+1}\|^4), \end{aligned} \tag{3.17}$$

whence

$$\|(f_{k+1}, r_{D,k+1,o}, \underline{r}_{c,k+1,o}, \overline{r}_{c,k+1,o})\| = o(\|\Delta x_{k+1}\|). \tag{3.17a}$$

From Theorem 2, for any $\epsilon > 0$ and all k sufficiently large we have $\mu_k = \tilde{\mu}$ for some fixed $\tilde{\mu} > 0$ with $y_k^o \mu_k^o \le y_k^o \tilde{\mu} \le \epsilon$, and $\{y_k^o\}$ is bounded away from zero.

From (2.21), (2.19), and (2.20) we have

$$D_{kl}^i \ge (\underline{X}_{kl}^i)^{-2}(\underline{X}_{kl}^i \underline{Z}_{kl}^i) \ge (1-\beta) y_k^o \mu_{kl}/(b^i - a^i)^2, \quad i = 1, 2, \ldots, n, \tag{3.18}$$

so D_{kl}^{-1} is uniformly bounded, and if ζ_k is close enough to $\tilde{\Omega}$ for Lemma 2 to hold, it follows from (3.9) that the inverse of the Jacobian matrix for (2.14) is uniformly bounded, and we can assume that the pivot threshold is small enough for H_k not to be reset to zero.

Defining $\tilde{\zeta}_k = \arg\min_\zeta \{\|\zeta - \zeta_k\| \mid (\zeta, \tilde{\mu}) \in \tilde{N}_\epsilon\}$, we have from (2.14) and continuity of the $f_{xx}^j(x)$ on X that

$$\left\{ \begin{aligned} r_{Dko} &= g_k^o y_k^o + G_k y_k - \underline{z}_k + \overline{z}_k \\ &= (g_k^o - \tilde{g}_k^o) y_k^o + (G_k - \tilde{G}_k) y_k + \tilde{g}_k^o (y_k^o - \tilde{y}_k^o) \\ &\quad + \tilde{G}_k (y_k - \tilde{y}_k) - (\underline{z}_k - \tilde{\underline{z}}_k) + (\overline{z}_k - \tilde{\overline{z}}_k) \\ &= \sum_{j=0}^{m} y_k^j f_{xx}^j(x_k)(x_k - \tilde{x}_k) + \tilde{g}_k^o (y_k^o - \tilde{y}_k^o) \\ &\quad + \tilde{G}_k (y_k - \tilde{y}_k) - (\underline{z}_k - \tilde{\underline{z}}_k) + (\overline{z}_k - \tilde{\overline{z}}_k) \\ &\quad + o(\|x_k - \tilde{x}_k\|), \\ f_k &= \tilde{f}_k + \tilde{G}_k^T (x_k - \tilde{x}_k) + O(\|x_k - \tilde{x}_k\|^2), \\ \underline{r}_{cko} &= \tilde{\underline{X}}_k (\underline{z}_k - \tilde{\underline{z}}_k) + \tilde{\underline{Z}}_k (x_k - \tilde{x}_k) \\ &\quad + (\underline{X}_k - \tilde{\underline{X}}_k)(\underline{z}_k - \tilde{\underline{z}}_k) - (y_k^o - \tilde{y}_k^o)\tilde{\mu} e, \\ \overline{r}_{cko} &= \tilde{\overline{X}}_k (\overline{z}_k - \tilde{\overline{z}}_k) + \tilde{\overline{Z}}_k (x_k - \tilde{x}_k) \\ &\quad + (\overline{X}_k - \tilde{\overline{X}}_k)(\overline{z}_k - \tilde{\overline{z}}_k) - (y_k^o - \tilde{y}_k^o)\tilde{\mu} e. \end{aligned} \right. \tag{3.19}$$

Again, if ζ_k is sufficiently close to $\tilde{\Omega}$ the Jacobian matrix of this system has a uniformly bounded inverse, and it follows that

$$\|(f_k, r_{Dko}, \underline{r}_{cko}, \overline{r}_{cko})\| \approx \|\zeta_k - \tilde{\zeta}_k\|. \tag{3.19a}$$

Again setting $\alpha_{k+1} = 1$, we also have from (2.14) that

$$
\left\{\begin{array}{rcl}
r_{Dk,l+1} &=& r_{Dko} + g_k^o(y_{k+1}^o - y_k^o) + H_k(x_{k+1} - x_k) \\
&& +G_k(y_{k+1} - y_k) - (\underline{z}_{k+1} - \underline{z}_k) + (\overline{z}_{k+1} - \overline{z}_k), \\
0 &=& f_k(1 - x_{k+1}^o) + G_k^T(x_{k+1} - x_k), \\
\underline{r}_{ck,l+1} &=& \underline{r}_{cko} + \underline{X}_k(\underline{z}_{k+1} - \underline{z}_k) + \underline{Z}_k(x_{k+1} - x_k) \\
&& +(\underline{X}_{k+1} - \underline{X}_k)(\underline{z}_{k+1} - \underline{z}_k) - (y_{k+1}^o - y_k^o)\tilde{\mu}e, \\
\overline{r}_{ck,l+1} &=& \overline{r}_{cko} + \overline{X}_k(\overline{z}_{k+1} - \overline{z}_k) + \overline{Z}_k(x_{k+1} - x_k) \\
&& +(\overline{X}_{k+1} - \overline{X}_k)(\overline{z}_{k+1} - \overline{z}_k) - (y_{k+1}^o - y_k^o)\tilde{\mu}e,
\end{array}\right. \tag{3.20}
$$

and using (2.19), (2.23), (2.36), and (3.8) we have

$$
\|(f_k, r_{Dko}, \underline{r}_{cko}, \overline{r}_{cko})\| \approx \|\zeta_{k+1} - \zeta_k\|. \tag{3.20a}
$$

Then from (3.17a) and (3.20a) we have

$$
\begin{array}{rcl}
\|\Delta x_{k+1}\| &\leq& \|\Delta\zeta_{k+1}\| \approx \|(f_k, r_{Dko}, \underline{r}_{cko}, \overline{r}_{cko})\| \\
&=& o(\|\Delta x_k\|) = o(\|\Delta\zeta_k\|).
\end{array} \tag{3.21}
$$

Since $\|f_k\| \to 0$, a double-step is tried systematically for all k sufficiently large, and it follows from (3.21) and Lemma 3 that we shall indeed have $\alpha_{k+1} = 1$, as assumed above, for all k sufficiently large. Hence $\{\zeta_k\}$ is a Cauchy sequence and converges to some $\tilde{\zeta}$.

But from (3.17a), (3.19a), and (3.20a) we have

$$
\begin{array}{rcl}
\|\zeta_{k+1} - \tilde{\zeta}_{k+1}\| &\approx& \|(f_{k+1}, r_{D,k+1,o}, \underline{r}_{c,k+1,o}, \overline{r}_{c,k+1,o})\| \\
&=& o(\|\Delta x_{k+1}\|) = o(\|\Delta\zeta_{k+1}\|) \\
&=& o\|(f_k, r_{D,k,o}, \underline{r}_{c,k,o}, \overline{r}_{c,k,o})\| = o\|\zeta_k - \tilde{\zeta}_k\|,
\end{array} \tag{3.22}
$$

so $\{\zeta_k\}$ converges Q-superlinearly to some $\tilde{\zeta}$ such that $(\tilde{\zeta}, \tilde{\mu}) \in \tilde{N}_\epsilon$.

Then we have

$$
\|\Delta x_{k+1}\| \leq \|x_{k+1} - \tilde{x}\| + \|x_k - \tilde{x}\|, \tag{3.23}
$$

and from (3.22),

$$
\|x_{k+1} - \tilde{x}\| \leq \|\zeta_{k+1} - \tilde{\zeta}\| = o(\|\Delta x_{k+1}\|) = o(\|x_k - \tilde{x}\|), \tag{3.24}
$$

so $\{x_k\}$ also converges Q-superlinearly.

Now if (3.16) holds, we have from (3.17) and (3.21) that

$$
\begin{array}{rcl}
r_{D,k+1,o} &=& r_{Dk,l+1} + O\{\|\Delta x_{k+1}\| \max(\|x_k - \tilde{x}\|^\nu, \|x_{k+1} - \tilde{x}\|^\nu)\} \\
&& +O(\|\Delta y_{k+1}\| \cdot \|\Delta x_{k+1}\|),
\end{array}
$$

whence

$$
\begin{array}{l}
\|(f_{k+1}, r_{D,k+1,o}, \underline{r}_{c,k+1,o}, \overline{r}_{c,k+1,o})\| \\
= O(\|\zeta_k - \tilde{\zeta}\|^{1+\nu}) \\
= O(\|x_k - \tilde{x}\|^{1+\nu}) + O(\|x_{k-1} - \tilde{x}\| \cdot \|x_k - \tilde{x}\|).
\end{array} \tag{3.25}
$$

Then using (3.19a) we have

$$\begin{aligned} &\|\zeta_{k+1}-\tilde{\zeta}\| = O(\|\zeta_k-\tilde{\zeta}\|^{\nu}),\\ &\|x_{k+1}-\tilde{x}\| = O(\|x_k-\tilde{x}\|^{1+\nu}) + O(\|x_{k-1}-\tilde{x}\|\cdot\|x_k-\tilde{x}\|), \end{aligned} \tag{3.26}$$

from which the final results follow. ∎

Although we can make $\tilde{\zeta}$ arbitrarily close to $\tilde{\Omega}$ by making ϵ sufficiently small, the proof does not hold for $\epsilon = 0$ unless the Jacobian matrix of (2.14) is nonsingular for $\zeta \in \Omega$. It is easily shown that this requires strict complementarity, and strict monotonicity of $\tilde{H}$, which in turn implies that $\tilde{\Omega}$ corresponds to an isolated local minimizer. However, by making use of Lemma 1 we can weaken these conditions to those implying only that the approximating QP at the solution has a unique solution. This requires that either the linearized constraints define a unique feasible point, or the classical McCormick second-order sufficiency conditions apply. Under these conditions we have the following result.

THEOREM 4. *If $\epsilon^o = \epsilon = 0$ and for some $\bar{k} < \infty$, $x_{\bar{k}}$ is sufficiently close to an isolated local minimizer $\tilde{x}$ of problem* (2.1) *satisfying the above conditions, then $\{x_k\} \to \tilde{x}$ and $\{\zeta_k\} \to \tilde{\zeta} \in \Omega$, both at a Q-superlinear rate.*

If in addition there is a neighborhood of $\tilde{x}$ on which (3.16) *holds, then the Q-order of convergence of $\{\zeta_k\}$ is at least $(1+\nu)$, and that of $\{x_k\}$ is at least $\min\{1+\nu, 1+\frac{\sqrt{5}}{2}\}$.*

Proof. From the stated conditions $\tilde{\zeta}$ is also the unique solution of the system

$$\begin{cases} \tilde{g}^o\tilde{y}^o + \tilde{H}(x-\tilde{x}) + \tilde{G}y - \underline{z} + \overline{z} = 0,\\ \tilde{G}^T(x-\tilde{x}) = 0,\\ x-a \;\geq\; 0, \quad \underline{z} \geq \;0, \quad \underline{z}^T(x-a) \;=\; 0,\\ b-x \;\geq\; 0, \quad \overline{z} \geq \;0, \quad \overline{z}^T(b-x) \;=\; 0, \end{cases} \tag{3.27}$$

with $\tilde{y}^o > 0$ and $\|\tilde{\eta}\|_\infty = 1$.

We can therefore rescale both $\tilde{\eta}$ and η_k so that $y_k^o = \tilde{y}^o = 1$, and $\{\eta_k\}$ will be uniformly bounded.

Then from Lemma 1 we have

$$\begin{aligned} \|\zeta_k-\tilde{\zeta}\| \leq \tau\{&\|\tilde{r}_{Dk}\| + \|\tilde{r}_{Pk}\| + \|I_{\underline{B}}^T\underline{z}_k\| + \|I_{\overline{B}}^T\overline{z}_k\|\\ &+\|I_{\underline{N}}^T\underline{X}_k\| + \|I_{\overline{N}}^T\overline{X}_k\| + \sqrt{m_k}\,\}, \end{aligned} \tag{3.28}$$

where

$$\begin{aligned} m_k &= \underline{z}_k^T(x_k-a) + \overline{z}_k^T(b-x_k),\\ \tilde{r}_{Dk} &= \tilde{g}^o + \tilde{H}(x_k-\tilde{x}) + \tilde{G}y_k - \underline{z}_k + \overline{z}_k,\\ \tilde{r}_{Pk} &= \tilde{G}^T(x_k-\tilde{x}). \end{aligned} \tag{3.29}$$

Again we can assume that x_k is close enough to $\tilde{x}$ for Lemma 1 to hold for some $\tilde{\zeta} \in \Omega$, so that H_k is not reset to zero and (3.17) holds. It follows that

$$\begin{aligned} \tilde{r}_{Dk} &= r_{Dko} - \sum_{j=0}^{m} y_k^j\{g_k^j - \tilde{g}^j - f_{xx}^j(\tilde{x})(x_k-\tilde{x})\}\\ &\quad -\sum_{j=0}^{m}(y_k^j - \tilde{y}^j) f_{xx}^j(\tilde{x})(x_k-\tilde{x})\\ &= o(\|\Delta x_k\|) + o(\|y_k-\tilde{y}\|\cdot\|x_k-\tilde{x}\|),\\ \tilde{r}_{Pk} &= f_k + O(\|x_k-\tilde{x}\|^2) = O(\|\Delta x_k\|^2). \end{aligned} \tag{3.30}$$

We can also assume that x_k is close enough to $\tilde{x}$ to have

$$\underline{X}_k^i \geq \tfrac{1}{2}\underline{\tilde{X}}^i, i \in \underline{B}; \quad \overline{X}_k^i \geq \tfrac{1}{2}\overline{\tilde{X}}^i, i \in \overline{B};$$
$$\underline{z}_k^i \geq \tfrac{1}{2}\underline{\tilde{z}}^i, i \in \underline{N}; \overline{z}_k^i \geq \tfrac{1}{2}\overline{\tilde{z}}^i, i \in \overline{N}.$$

Then from (2.14), (2.19), and (2.36) we have

$$\begin{aligned} &\|I_{\underline{B}}^T \underline{z}_k\|, \|I_{\overline{B}}^T \overline{z}_k\|, \|I_{\underline{N}}^T \underline{X}_k\|, \|I_{\overline{N}}^T \overline{X}_k\| = O(\|\Delta x_k\|^4) \ , \\ &\underline{z}_k^T(x_k - a) + \overline{z}_k^T(b - x_k) \\ &\quad = \ e^T(\underline{r}_{ck} + y_k^o \mu_k e) + e^T(\overline{r}_{ck} + y_k^o \mu_k e) \\ &\quad = \ O(\|\Delta x_k\|^4). \end{aligned} \tag{3.31}$$

Then from (3.28), (3.30), (3.31), and (3.23),

$$\begin{aligned} \|x_{k+1} - \tilde{x}\| &\leq \|\zeta_{k+1} - \tilde{\zeta}\| \\ &= o(\|x_k - \tilde{x}\|) = o(\|\zeta_k - \tilde{\zeta}\|). \end{aligned} \tag{3.32}$$

Thus both $\{\zeta_k\}$ and $\{x_k\}$ converge Q-superlinearly to $\tilde{\zeta}$ and $\tilde{x}$, respectively.

If (3.16) also holds, we have

$$\tilde{r}_{D,k+1} = O(\|x_k - \tilde{x}\|^{\min\{1+\nu, 1+\frac{\sqrt{5}}{2}\}})$$

and it follows that $\{\zeta_k\}$ and $\{x_k\}$ both converge with Q-order at least $\min\{1+\nu, 1+\frac{\sqrt{5}}{2}\}$. □

4. Numerical results. As already noted, our algorithm requires upper and lower bounds on all variables, and an interior starting point, which should not be too close to a bound. In real-world problems users can usually set sensible bounds based on physical considerations and also decide on appropriate action if the solution happens to be on such an "artificial" bound. However standard test-sets of problems may well omit some bounds, and we therefore use default values of $\pm 10^5$ for any missing bounds.

For slack variables we set $s_s^i = f^i(x_s)$, where x_s is the specified starting value; then for the initial point we use the rule

$$x_o^i = \max\{a^i + \Delta_s^i, \min\{b^i - \Delta_s^i, x_s^i\}\}, \tag{4.1}$$

where

$$\Delta_s^i = \min\{1, 0.1(b^i - a^i)\}.$$

We also incorporated step-length bounds as in (2.6), with

$$\Delta^i = 10 \max\{1, |x^i|\}. \tag{4.2}$$

The algorithm contains the parameters $\underline{\alpha}, \beta, \gamma, \delta, \delta_{k,o}, \bar{\epsilon}, \theta, \underline{\rho}, \rho, \sigma, \ \tau, \omega, \underline{M}, \underline{m}$, as well as the desired tolerances ϵ^o, ϵ, and our first task was to determine reasonable default values for these parameters. For this purpose we carried out comparative tests on a small subset of the problems published by Hock and Schittkowski [11].

For $\underline{M} = 1000, \underline{m} = 10, \bar{\epsilon} = 10^{-7}$, no failures with $x^o_{kl} > 1 - \bar{\epsilon}$ were encountered, so other values were not tested. For other parameters the best all-round values obtained with $\epsilon^o = \epsilon = 10^{-7}$ were found to be

$$\begin{array}{llll} \delta = 10^{-4}, & \theta = 0.5, & \omega = 1.5, & \\ \underline{\rho} = 0.1, & \rho = 10^{-4}, & \sigma = 10^3, & \\ \bar{\beta}^0 = 0.9, & \beta = 0.9, & \gamma = 0.3, & \delta_{k,o} = 0.45, \\ \tau = 0.8, & \underline{\alpha} = 0.01, & \mu_o = 100. & \end{array}$$

Tables 1, 2, 3, and 4 show the results for our algorithm (SQPIPM) for varying values of β, γ, σ, and μ (in each case with all other parameters at the above values). Those for other parameters showed much less sensitivity over quite a wide range.

From Table 1 there is evidence of instability if β is too close to unity, probably because some iterates approach the bounds too closely before μ is small. A large γ also provides "centering," but this must be balanced against the smaller reduction factor for μ^o and μ.

Smaller values of σ do force increased accuracy in solving the subproblems, which tends to result in fewer outer iterations at the expense of more inner iterations, but the effect is not as large as expected.

The most sensitive parameter is μ_o, but no regular pattern is discernible. Since μ is rapidly reduced, we opted for robustness and chose a default value of 100.

In fact the Hock–Schittkowski subset had reasonable upper and lower bounds and interior starting points, so for Tables 1, 2, 3 we did not incorporate a step-length bound, and we also used a heuristic for choosing μ_o, subsequently abandoned, which accounts for the differences between these tables and Table 4.

We then applied the algorithm with the above default parameters to the full Hock–Schittkowski test-set and a selection of large problems from the CUTE collection [3]. The results are given in Tables 5, 6, and 7, together with comparative results for LANCELOT (see Conn, Gould, and Toint [6]) and NITRO (see [4]), where $\mathrm{N_o}$ denotes the number of outer iterations, $\mathrm{N_f}$ denotes the number of function evaluations, $\mathrm{N_i}$ denotes the number of inner iterations, $\mathrm{N_d}$ denotes the number of derivative evaluations, NF1 denotes the number of failures in QP due to $\alpha'_{k,l+1} < \underline{\alpha}$ (with $\underline{\alpha} = 0.01$), and NF2 denotes the number of failures in QP due to $\psi < \underline{\rho} \Delta x^T D \Delta x$ (with $\underline{\rho} = 0.1$). We note that SQPIPM makes one derivative evaluation on each outer iteration.

In these tables, results for NITRO are as reported in [4] and were obtained using a SPARCstation 20. Results for LANCELOT (LAN) were obtained using a SPARCstation 10. For SQPIPM the runs were performed on a Pentium II PC, using a FORTRAN 77 compiler and double precision. All the results used second derivatives and default values for all parameters. For SQPIPM the linear systems (2.18) were solved using MA47 [8].

On the problems tested by NITRO, SQPIPM had four failures and NITRO had three, but SQPIPM performed consistently better on problems solved by both algorithms, taking only 46% of the total number of derivative evaluations.

Similarly LANCELOT had ten failures on the full set, compared with nine for SQPIPM, and again the latter required only 42% of the total number of derivative evaluations for the problems solved by both.

For SQPIPM the total numbers of inner iterations and function evaluations for each problem are also satisfactorily small, averaging only 1.71 function evaluations and 1.76 inner iterations per outer iteration, with maxima of 9.0 (for HS91) and 6.5 (HS55), respectively.

TABLE 1
Numerical tests on the value of β.

Problem	Outer iterations : function evaluations : inner iterations (NF1[a] : NF2[b])				
	$\beta = 0.99$	$\beta = 0.9$	$\beta = 0.8$	$\beta = 0.7$	$\beta = 0.5$
HS32	12:12:17	12:12:17	13:13:18	10:10:16	11:11:22
HS39	17:61:22 (1:0)	18:69:23 (1:0)	18:69:23 (1:0)	18:69:23 (1:0)	17:61:22 (1:0)
HS53	4:4:10	4:4:10	4:4:10	4:4:10	4:4:10
HS63	8:13:15	8:11:15	7:7:14	8:8:15	8:9:14
HS64	24:26:27	24:24:26	24:24:27	25:25:28	29:29:32
HS70	40:128:51 (0:9)	13:17:13	14:15:14	14:14:14	16:16:16
HS72	19:19:20	18:18:22	20:20:20	22:22:22	29:29:29
HS73	11:11:15	11:11:15	11:11:15	12:12:15	13:13:16
HS77	10:11:16	10:11:16	10:11:16	10:11:16	10:11:16
HS78	5:5:13	5:5:13	5:5:13	5:5:13	5:5:13
HS79	5:5:11	5:5:11	5:5:11	5:5:11	5:5:11
HS80	10:15:17	10:15:17	8:8:16	7:7:15	8:9:16
HS81	8:8:13	9:11:13	9:9:15	9:9:15	9:10:14
HS93	9:9:15 (0:1)	9:9:15 (0:1)	9:9:15 (0:1)	10:10:16 (0:1)	10:10:15 (0:1)
HS100	38:67:45 (2:0)	32:47:38	31:50:37	34:64:42	36:48:41

[a]No. of failures in QP due to $\alpha'_{k,l+1} < \underline{\alpha}$ (with $\underline{\alpha} = 0.01$).
[b]No. of failures in QP due to $\psi < \underline{\rho}\Delta x^T D \Delta x$ (with $\underline{\rho} = 0.1$).

We give figures for (NF1:NF2) only where these are nonzero.

It is further encouraging that the performance is in general much better on the larger, more realistic CUTE problems (only SVANBERG and COSHFUN required any α reduction), and for the "scalable" problems in Table 6 the performance does not significantly change as the problem size increases.

Many of these problems are nonconvex, so it is not surprising that there are failures to generate a descent direction, and the lack of convexity can also require a small α to satisfy (2.19). However, resetting $H_k = 0$ seems to be an effective remedy, and performance is not significantly affected. Again the larger CUTE problems seem less susceptible to these failures.

It is of interest that H_k was never reset as a result of failure to complete the factorization, and on only one problem (HS61) did the algorithm converge to an infeasible point.

In fact there seems to be no generic deficiency in the algorithm, and the causes of serious failure seem to be different in each case.

TABLE 2
Numerical tests on the value of γ.

Problem	Outer iterations : function evaluations : inner iterations (NF1[a] : NF2[b])				
	$\gamma = 0.01$	$\gamma = 0.1$	$\gamma = 0.3$	$\gamma = 0.5$	$\gamma = 0.7$
HS32	21:21:33	21:21:26	12:12:17	12:12:22	12:12:32
HS39	18:69:19 (1:0)	18:69:19 (1:0)	18:69:23 (1:0)	18:69:28 (1:0)	18:69:32 (1:0)
HS53	5:5:7	4:4:8	4:4:10	4:4:13	4:4:16
HS63	7:7:9	7:7:10	11:8:15	10:10:19 (0:1)	28:16:33 (0:3)
HS64	24:24:24	23:23:24	24:24:26	26:28:30	33:38:42
HS70	12:12:12	11:11:11	13:17:13	26:48:41 (1:6)	18:27:33 (0:1)
HS72	21:21:22	17:17:22	18:18:22	19:19:23	24:24:30
HS73	12:13:16	12:17:15	11:11:15	17:17:22	27:27:37
HS77	10:11:10	10:11:12	10:11:16	10:11:20	10:11:23
HS78	5:5:6	5:5:9	5:5:13	5:5:15	5:5:19
HS79	5:5:6	5:5:9	5:5:11	5:5:15	5:5:18
HS80	16:33:18	12:21:14	10:15:17	12:13:24	13:17:30
HS81	7:7:9	8:8:10	9:11:13	12:18:18	12:13:26
HS93	7:7:14 (0:1)	7:7:11 (0:1)	9:9:15 (0:1)	17:23:29 (0:3)	20:23:34 (0:1)
HS100	35:88:76 (5:0)	33:64:39 (3:0)	32:47:38	35:54:42	42:60:55

[a]No. of failures in QP due to $\alpha'_{k,l+1} < \underline{\alpha}$ (with $\underline{\alpha} = 0.01$).
[b]No. of failures in QP due to $\psi < \underline{\rho}\Delta x^T D \Delta x$ (with $\underline{\rho} = 0.1$).

We give figures for (NF1:NF2) only where these are nonzero.

5. General comments.

5.1. Algorithm performance. The results given in the last section show that the algorithm is reasonably robust, and the performance is quite uniform over the range of problems tested. It has also performed well on a number of application problems in chemical process synthesis and design. Of course these encouraging preliminary results need confirmation over a wide range of large problems, but a more important priority is to identify, and if possible eliminate, the causes of failure.

The problems tested are only of moderate size in today's terms, but the only limitation in extending the size indefinitely is the use of a direct method for solving the linear systems. In process problems $G(x)$ is usually very sparse, and $H(x)$ is much sparser, but for really large problems it may be necessary to envisage the use of iterative techniques, which do not require storage of the KKT matrix in (2.18). Unfortunately, in our experience methods currently available do not seem to have adequate robustness or efficiency. We note that about 80% of the computation time

TABLE 3
Numerical tests on the value of σ.

Problem	Outer iterations : function evaluations : inner iterations (NF1[a] : NF2[b])					
	$\sigma = 10^5$	$\sigma = 10^4$	$\sigma = 10^3$	$\sigma = 10^2$	$\sigma = 10$	$\sigma = 1$
HS32	15:15:16	14:14:16	12:12:17	7:7:17	6:6:20	5:5:21
HS39	18:69:23 (1:0)	18:69:24 (1:0)	18:69:23 (1:0)	18:69:23 (1:0)	18:69:25 (1:0)	18:69:25 (1:0)
HS53	5:5:11	5:5:10	4:4:10	4:4:9	4:4:10	4:4:10
HS63	9:9:14	8:8:14	8:11:15	8:11:15	7:7:14	7:7:16 (1:0)
HS64	25:25:27	24:24:27	24:24:26	23:23:26	23:23:26	22:22:28
HS70	13:17:13	13:17:13	13:17:13	13:17:15	13:17:15	13:17:16
HS72	18:18:22	18:18:22	18:18:22	18:18:22	18:18:22	20:20:22
HS73	12:12:15	12:12:16	11:11:15	11:11:14	10:10:14	8:8:14
HS77	10:11:15	10:11:15	10:11:16	10:11:17	10:11:16	10:11:15
HS78	5:5:11	5:5:13	5:5:13	5:5:11	5:5:12	5:5:10
HS79	5:5:11	5:5:10	5:5:11	5:5:11	5:5:13	5:5:12
HS80	22:57:22	10:10:17	10:15:17	7:7:15	7:7:16	7:7:14
HS81	10:10:14	10:10:14	9:11:13	15:21:22	14:20:22	16:22:23
HS93	10:10:16 (0:1)	10:10:16 (0:1)	9:9:15 (0:1)	9:9:15 (0:1)	8:8:15 (0:1)	8:8:14 (0:1)
HS100	34:49:38	32:47:37	32:47:38	32:47:37	31:46:36	34:64:42

[a]No. of failures in QP due to $\alpha'_{k,l+1} < \underline{\alpha}$ (with $\underline{\alpha} = 0.01$).
[b]No. of failures in QP due to $\psi < \underline{\rho}\Delta x^T D \Delta x$ (with $\underline{\rho} = 0.1$).

We give figures for (NF1:NF2) only where these are nonzero.

is spent in MA47, so improvement of the linear system solver does offer the greatest scope for speeding up the algorithm, but for direct methods it is difficult to see what can be done beyond using parallelization.

Although some of the provisions to secure robustness may seem crude, the small number of times they are involved in practice gives little incentive to make them more sophisticated.

Within the subproblem, step-lengths are computed by solving single-variable quadratic equations, and more elaborate techniques give little improvement. In fact, since so few iterations are required we could envisage simplifying the algorithm by using fixed β and γ.

Similarly there is little incentive for replacing the Armijo rule in the outer iteration by a more sophisticated rule requiring gradient evaluations or for adopting a trust-region approach.

Although we have incorporated a step-length bound to avoid very large steps, this

TABLE 4
Numerical tests on the value of μ_o.

Problem	Outer iterations : function evaluations : inner iterations (NF1[a] : NF2[b])					
	$\mu_o = 0.01$	$\mu_o = 0.1$	$\mu_o = 1$	$\mu_o = 10$	$\mu_o = 100$	$\mu_o = 1000$
HS32	10:10:15	10:10:15	9:9:15	11:11:17	15:16:18	17:18:20
HS39	19:70:23 (1:0)	18:55:23 (1:0)	15:30:19	11:15:14	9:9:14	9:9:17
HS53	4:4:7	4:4:8	4:4:8	4:4:8	4:4:10	4:4:11
HS63	25:30:67 (7:4)	15:16:109 (4:4)	9:9:15 (0:1)	7:7:14	8:8:13	13:16:22 (0:1)
HS64	* * *[c]	* * *	22:22:26	24:24:26	24:24:26	25:25:29
HS70	13:13:13	14:15:14	19:33:26 (1:1)	20:24:23 (0:3)	20:24:23 (0:2)	21:24:27 (0:3)
HS72	17:18:26	17:17:25	18:18:22	21:21:21	22:22:23	27:27:28
HS73	33:33:77 (6:0)	23:23:27	13:13:17	12:12:16	15:15:18	17:17:22
HS77	10:11:12 (6:0)	10:11:13	10:11:13	10:11:14	11:13:17	10:11:17
HS78	5:5:8	5:5:9	5:5:10	5:5:12	5:5:13	5:5:13
HS79	5:5:8	5:5:9	5:5:10	5:5:10	5:5:11	5:5:13
HS80	7:7:11	6:6:9	6:6:12	9:10:16	8:10:18	12:26:18
HS81	10:22:14 (1:0)	14:22:17	9:11:12	10:11:16	9:27:16	14:34:21
HS93	17:28:46 (2:0)	18:20:23 (0:1)	14:23:21 (1:0)	11:16:16 (0:1)	15:22:25 (1:0)	14:20:19
HS100	11:15:16	11:15:16	11:15:15	12:16:17	12:14:16	15:16:19

[a]No. of failures in QP due to $\alpha'_{k,l+1} < \underline{\alpha}$ (with $\underline{\alpha} = 0.01$).
[b]No. of failures in QP due to $\psi < \underline{\rho} \Delta x^T D \Delta x$ (with $\underline{\rho} = 0.1$).

We give figures for (NF1:NF2) only where these are nonzero.
[c]Failure to converge in 1000 inner iterations.

can slow progress and the choice of bound is likely to be very problem dependent. We believe it is difficult for users to provide a satisfactory bound and it is much easier for them to set, and if necessary adjust, reasonable variable bounds. In our experience such an approach is more efficient than any generalized rules for setting and adjusting step-length bounds based on past algorithm behavior.

However, a claim often made for trust-region methods is that they ensure convergence to a local minimum, not simply a Fritz-John point. In fact this can also be achieved with a line-search method which uses directions of negative curvature, as discussed in [17], but unfortunately both approaches seem to require range-space/null-space decomposition of the constraints, with the resulting generation of dense matrices which limits problem size. Until we learn how to exploit the properties of the Hessian matrix without the use of such techniques, it seems that for large problems we must be content with a less ambitious goal.

TABLE 5
Numerical tests on the Hock–Schittkowski set: Part I.

Problem	Performance			Problem	Performance		
	SQPIPM $N_o : N_f : N_i$ (NF1:NF2)[a]	LAN N_d	NITRO N_d		SQPIPM $N_o : N_f : N_i$ (NF1:NF2)	LAN N_d	NITRO N_d
HS1	24:29:25	34		HS18	50:115:53	92	
HS2	8:18:18 (0:1)	7	18	HS19	15:15:35	35	47
HS3	12:12:17	5	12	HS20	7:7:11	24	18
HS4	7:7:14	3	11	HS21	18:18:20	2	
HS5	7:7:13	5		HS22	10:10:15	10	15
HS6	16:68:27 (0:7)	57		HS23	23:23:27	44	
HS7	9:14:16 (0:1)	19	33	HS24	11:11:18 (0:2)	8	19
HS8	6:6:10	12		HS25	28:34:33 (0:3)	1	
HS9	5:5:12	5		HS26	20:20:25	40	13
HS10	12:12:16	18	17	HS27	30:110:33	17	
HS11	9:9:14	16	14	HS28	5:5:10	4	3
HS12	11:12:15	24		HS29	27:111:48 (0:17)	31	
HS13	$***$[b]	59	123	HS30	11:17:17	8	
HS14	10:10:15	13	14	HS31	9:9:14	14	13
HS15	119:916:423 (102:0)	47		HS32	15:16:18	6	19
HS16	16:32:32 (2:2)	17	15	HS33	24:29:25 (0:1)	13	28
HS17	10:10:26	20	27	HS34	16:19:20	20	

[a] We give figures for (NF1:NF2) only where these are nonzero.
[b] Failure to converge in 1000 inner iterations.

5.2. Theoretical results. Of course the main purpose of the convergence proofs is to establish what features are necessary to obtain good performance of the algorithm, as well as providing a guarantee of this performance.

Thus Theorem 2 shows the importance of rule (2.24), which prevents $\{\mu_k\}$ going to zero faster than $\{\|\Delta x_k\|\}$, for achieving global convergence, while Theorem 4 shows that in order to achieve a final Q-quadratic convergence rate the subproblem must be solved to an accuracy of $O(\|\Delta x_k\|^4)$.

Closer examination of the proof of Theorem 3 shows in contrast that, for Q-quadratic convergence to an ϵ-optimal point, a subproblem accuracy of $O(\|\Delta x_k\|^2)$ would suffice (cf. (3.17)). The apparent anomaly is resolved by noting that Theorem 3 is concerned with the solution of an equality-constrained problem, for which strict complementarity is irrelevant, while Theorem 4 must allow for the possible lack of

TABLE 5
Numerical tests on the Hock–Schittkowski set: Part II.

Problem	Performance			Problem	Performance		
	SQPIPM $N_o : N_f : N_i$ (NF1:NF2)[a]	LAN N_d	NITRO N_d		SQPIPM $N_o : N_f : N_i$ (NF1:NF2)	LAN N_d	NITRO N_d
HS35	8:8:14	7		HS51	4:4:10	3	3
HS36	7:7:13 (0:1)	12		HS52	4:4:10	7	8
HS37	6:6:12 (0:1)	17		HS53	4:4:10	7	8
HS38	$***$[b]	59		HS54	68:68:77	3	
HS39	9:9:14	21	117	HS55	8:8:52 (3:0)	7	
HS40	5:5:13	11		HS56	25:39:35 (0:4)	20	
HS41	7:7:16	7		HS57	$***$	2	
HS42	5:5:11	13		HS59	118:479:140 (0:12)	336	
HS43	11:14:16	23		HS60	6:6:13	16	
HS44	11:11:22 (0:4)	7		HS61	Fail[c]	20	
HS45	9:10:21 (0:2)	3		HS62	5:6:14	35	
HS46	22:23:24	25	20	HS63	8:8:13	15	$---$[d]
HS47	19:29:26 (0:2)	20		HS64	24:24:26	51	43
HS48	3:3:8	4		HS65	15:15:19	29	20
HS49	18:18:18	16		HS66	13:13:18	10	
HS50	9:9:14	13		HS67	39:113:44	57	

[a]We give figures for (NF1:NF2) only where these are nonzero.
[b]Failure to converge in 1000 inner iterations.
[c]Program failed due to $|x^o| > 1 - \bar{\epsilon}$ and $\|\Delta x\| < \epsilon$.
[d]Failure to converge after 1000 derivative evaluations.

strict complementarity. If in fact convergence is to a strictly complementary solution in the latter case, Lemma 1 shows that the $\sqrt{m}$ term is not required, and again a subproblem accuracy of $O(\|\Delta x_k\|^2)$ would suffice.

Apart from pointing to the need for these practical provisions in the algorithm, the theoretical results are of some interest in themselves.

Theorem 1 proves finite termination of an interior-point algorithm for a generalized LCP of the form of (2.12), of course with $\mu^o = \mu = 0$, where H_k need not be monotone, or even symmetric (although in this case a general linear solver is required for (2.18)).

TABLE 5
Numerical tests on the Hock-Schittkowski set: Part III.

Problem	Performance			Problem	Performance		
	SQPIPM $N_o : N_f : N_i$ (NF1:NF2)[a]	LAN N_d	NITRO N_d		SQPIPM $N_o : N_f : N_i$ (NF1:NF2)	LAN N_d	NITRO N_d
HS68	39:58:54 (0:14)	93		HS95	24:38:28	8	156
HS69	8:8:15 (0:1)	44		HS96	24:38:28	8	224
HS70	20:24:23 (0:2)	31	35	HS97	20:23:24	21	45
HS71	12:13:16	16	22	HS98	19:20:23	18	53
HS72	22:22:23	90	44	HS99	6:6:9	− − −	
HS73	15:15:18	16	29	HS100	12:14:16	56	20
HS74	27:27:34 (3:0)	28	17	HS101	420:2840:439 (5:0)	− − −	
HS75	36:47:41 (3:0)	143	108	HS102	420:2918:424 (0:1)	− − −	
HS76	9:9:15	7		HS103	347:2409:349 (1:0)	− − −	
HS77	11:13:17	25	18	HS104	15:16:18	57	
HS78	5:5:13	12	36	HS105	47:138:87 (0:32)	13	34
HS79	5:5:11	10	7	HS106	* * *	− − −	221
HS80	8:10:18	16	74	HS107	7:7:12	31	16
HS81	9:27:16	18	95	HS108	* * *	34	49
HS83	14:14:19	23		HS109	* * *	− − −	− − −
HS84	13:13:29 (3:1)	74		HS110	5:5:11	7	
HS85	* * *[b]	− − −[c]		HS111	13:16:17 (0:1)	48	− − −
HS86	13:13:16	18	16	HS112	11:11:18	44	14
HS87	* * *	123		HS113	20:20:23	73	17
HS88	11:12:14	54		HS114	45:53:49	757	42
HS89	22:42:25 (0:2)	65		HS116	* * *	− − −	
HS90	14:16:15	58		HS117	19:19:23	67	40
HS91	84:755:242 (52:8)	69		HS118	17:17:21	17	
HS92	14:16:17	56		HS119	14:14:18	30	31
HS93	15:22:25 (1:0)	− − −	14				

[a] We give figures for (NF1:NF2) only where these are non-zero.
[b] Failure to converge in 1000 inner iterations.
[c] Failure to converge after 1000 derivative evaluations.

TABLE 6
Numerical results of large-scale test problems.

Problem	n	m	Performance	
			SQPIPM[a]	LANCELOT[b]
HAGER4-50	101	50	18:18:19	15
HAGER4-100	201	100	18:18:19	15
HAGER4-500	1001	500	20:20:20	13
HAGER4-1000	2001	1000	23:23:23	14
HAGER4-2000	4001	2000	26:26:26	14
OPTCNTRL30	92	60	12:12:17	47
OPTCNTRL100	302	200	26:26:26	128
OPTCNTRL300	902	600	27:27:27	135
OPTCNTRL500	1502	1000	26:26:27	138
OPTCNTRL700	2102	1400	27:27:27	137
OPTCNTRL1000	3002	2000	26:26:27	161
SVANBERG500	1000	500	25:41:29	87
SVANBERG1000	2000	1000	26:42:30	105
SVANBERG2000	4000	2000	26:39:30	101

[a]Outer iterations : Function evaluations : Inner iterations (NF1 : NF2).
We give figures for (NF1:NF2) only where these are nonzero.
[b]Derivative evaluations.
Note that SQPIPM makes one derivative evaluation on each outer iteration.

The proof of Theorem 3 essentially follows the classical approach and demonstrates the convexifying role of the barrier function, which eliminates the need for a second-order sufficiency condition and the consequential restriction of the result to convergence to a regular *isolated* minimizer. As already noted, the use of the barrier function eliminates the need for strict complementarity, and the proof also relaxes the need for exact solution of the subproblem. Elimination of the need for a constraint qualification results partly from the use of a barrier function to eliminate inequality constraints and partly from use of the big-M technique to deal with linear dependence of the equality constraints.

Q-quadratic convergence to an exact solution (Theorem 4) still requires a second-order sufficiency condition, unless the linearized constraints at the solution define a unique point, which implies that the solution in question must be an isolated minimizer satisfying a constraint qualification, but again strict complementarity is not required. The use of the Mangasarian–Shiau-type bound from Lemma 1 makes possible a simple short proof of this result.

TABLE 7
Numerical results and comparisons.

Problem	n	m	Performance		
			SQPIPM[a]	LANCELOT[b]	NITRO[c]
COSHFUN	61	20	70:174:70	608	40
DIXCHLNV	100	50	18:18:23	$---^{d}$	18
GAUSSELM	14	11	7:7:15	27	78
HAGER4	2001	1000	23:23:23	14	18
HIMMELBK	24	14	$***^{e}$	133	51
OPTMASS	1210	1005	7:7:14	$---$	39
SVANBERG	1000	500	25:41:29	87	35

[a]Outer iterations : Function evaluations : Inner iterations (NF1 : NF2).
We give figures for (NF1:NF2) only where these are nonzero.
[b]Derivative evaluations.
[c]Derivative evaluations.
[d]The algorithm did not meet the stopping test after 1000 derivative evaluations.
[e]The algorithm did not meet the stopping test after 1000 inner iterations.

Appendix A. Computing $\alpha'_{k,l+1}$. From (2.13), (2.14), and (2.15) we have

$$\begin{aligned}\underline{X}_{k,l+1}\underline{z}_{k,l+1} &= (\underline{X}_{kl} + \alpha'_{k,l+1}\delta\underline{X}_{k,l+1})(\underline{z}_{kl} + \alpha'_{k,l+1}\delta\underline{z}_{k,l+1}) \\ &= \underline{X}_{kl}\underline{z}_{kl} + \alpha'_{k,l+1}(\underline{Z}_{kl}\delta x_{k,l+1} + \underline{X}_{kl}\delta\underline{z}_{k,l+1}) \\ &\quad +(\alpha'_{k,l+1})^2\delta\underline{X}_{k,l+1}\delta\underline{z}_{k,l+1} \\ &= \underline{X}_{kl}\underline{z}_{kl}(1-\alpha'_{k,l+1}) + \alpha'_{k,l+1}y^o_k\gamma_{kl}\mu_{kl}e \\ &\quad +(\alpha'_{k,l+1})^2\delta\underline{X}_{k,l+1}\delta\underline{z}_{k,l+1},\end{aligned} \tag{A.1}$$

and using (2.20),

$$\begin{aligned}\underline{r}_{ck,l+1} &= \underline{X}_{k,l+1}\underline{z}_{k,l+1} - y^o_{k,l+1}\mu_{k,l+1}e \\ &= \underline{r}_{ckl}(1-\alpha'_{k,l+1}) + (\alpha'_{k,l+1})^2\delta\underline{X}_{k,l+1}\delta\underline{z}_{k,l+1}.\end{aligned} \tag{A.2}$$

To satisfy $\|\underline{r}_{ck,l+1}\|_\infty \le \beta_{k,l+1}y^o_k\mu^o_{k,l+1}$, we require

$$\left\{\begin{array}{rcl}\bar{\phi}^i &=& \beta_{k,l+1}y^o_k\mu^o_{k,l+1} - \underline{r}^i_{ck,l+1} \ge 0, \\ \underline{\phi}^i &=& \beta_{k,l+1}y^o_k\mu^o_{k,l+1} + \underline{r}^i_{ck,l+1} \ge 0,\end{array}\right. \tag{A.3}$$

and again using (2.16) and (A.2) we may write

$$\left\{\begin{array}{rcl}\bar{\phi}^i &=& \bar{a}^i_o + \bar{a}^i_1\alpha'_{k,l+1} + \bar{a}^i_2(\alpha'_{k,l+1})^2 \ge 0, \\ \underline{\phi}^i &=& \underline{a}^i_o + \underline{a}^i_1\alpha'_{k,l+1} + \underline{a}^i_2(\alpha'_{k,l+1})^2 \ge 0,\end{array}\right. \tag{A.4}$$

where

$$\begin{aligned}\bar{a}_o^i &= \beta_{k,l+1} y_{k,l}^o \mu_{k,l}^o - \underline{r}_{ckl}^i,\\ \bar{a}_1^i &= \beta_{k,l+1} y_{k,l}^o \mu_{k,l}^o - \bar{a}_o^i,\\ \bar{a}_2^i &= -\delta x_{k,l+1}^i \delta z_{k,l+1}^i,\end{aligned}$$

$$\begin{aligned}\underline{a}_o^i &= \beta_{k,l+1} y_{k,l}^o \mu_{k,l}^o + \underline{r}_{ckl}^i,\\ \underline{a}_1^i &= \beta_{k,l+1} y_{k,l}^o \mu_{k,l}^o - \bar{a}_o^i,\\ \underline{a}_2^i &= \delta x_{k,l+1}^i \delta z_{k,l+1}^i.\end{aligned}$$

Then, e.g., for $\bar{\phi}^i \geq 0$, we have the rule

$$\text{(A.5)} \qquad \begin{cases} \text{If} \quad \bar{a}_1^i \geq 0, \bar{a}_2^i \geq 0, \text{or } \ 4\bar{a}_o^i \bar{a}_2^i \geq (\bar{a}_1^i)^2, \ \text{ then } \ \alpha'_{k,l+1} = 1, \\ \quad \text{else } \ \alpha'_{k,l+1} = 2\bar{a}_o^i (\sqrt{(\bar{a}_1^i)^2 - 4\bar{a}_o^i \bar{a}_2^i} - \bar{a}_1^i)^{-1}. \end{cases}$$

Note, however, that $\bar{\phi}^i \geq 0$ for all $\alpha'_{k,l+1} \in [0,1]$ if $\delta x_{k,l+1}^i \delta z_{k,l+1}^i \leq 0$, while $\underline{\phi}^i \geq 0$ for all $\alpha'_{k,l+1} \in [0,1]$ if $\delta x_{k,l+1}^i \delta z_{k,l+1}^i \geq 0$, so we need only solve one quadratic equation per element to determine the limiting value of $\alpha'_{k,l+1}$.

Similar results hold for $r_{ck,l+1}^o$ and $\overline{r}_{ck,l+1}$.

Appendix B. The main algorithm.

Given: $a, b, x, \epsilon, \epsilon^o$;
$\underline{\alpha}, \beta, \gamma, \delta, \delta_{k,o}, \bar{\epsilon}, \theta, \underline{\rho}, \rho, \sigma, \tau, \omega, \underline{M}, \underline{m}$

Initialization

1. a) Set $x_o = x, y^o = 1, y = 0, \tilde{y}^j = \omega$ for all j, $\gamma_o = \gamma$, FEAS = TRUE,
   b) Compute $f^o(x), f(x), L(x), X, \overline{X}, \mu, z, \overline{z}$.

General Iteration

2. Solve the subproblem:
   a) Compute $G(x)$.
      If FEAS=TRUE then compute $g^o(x), H(x, y, y^o)$
      else set $g^o = 0, H = 0$.
   b) Solve the subproblem (2.12) to obtain: $x^o, \zeta, \Delta x, \tilde{y}, \psi, \mu$.
   c) If $x^o \geq 1 - \bar{\epsilon}$ and $\|\Delta x\| \leq \epsilon^o$ then STOP (algorithm fails).
   d) If $x^o \geq 1 - \bar{\epsilon}$ then set FEAS=FALSE
      else if FEAS=FALSE then set $\underline{M} := \underline{m} \cdot \underline{M}$
      set FEAS=TRUE
3. a) Compute $f^o(x), f(x), L(x), P_o = P(x_o, y^o, \tilde{y}, \mu)$,
      $\Delta P_1 = P_o - P(x, y^o, \tilde{y}, \mu)$.
   b) If $\Delta P_1 \geq \delta\psi$ then set $x_o = x$ and repeat from step 2a.
4. a) Set $\Delta x_1 = \Delta x, \ \alpha = 1$.
   b) If $|f^j| \geq \bar{\epsilon}$ for all j, then set $\Delta x = 0$ and go to step 7.
   c) Set $\zeta_1 = \zeta, \tilde{y}_1 = \tilde{y}, \mu_1 = \mu, \psi_1 = \psi, \gamma_1 = \gamma_o$.
5. Solve the subproblem:
   a) Compute $G(x)$.
      If FEAS=TRUE then compute $g^o(x), H(x, y, y^o)$
      else set $g^o = 0, H = 0$.
   b) Solve the subproblem (2.12) to obtain $x^o, \zeta, \Delta x, \tilde{y}, \mu, \psi$.
   c) If $x^o \geq 1 - \bar{\epsilon}$ and $\|\Delta x\| \leq \epsilon^o$ then STOP (algorithm fails).

d) If $x^o \geq 1-\bar{\epsilon}$ then set FEAS=FALSE
else if FEAS=FALSE then set $\underline{M} := \underline{m} \cdot \underline{M}$
set FEAS=TRUE

6. a) Compute $f^o(x), f(x), L(x), \Delta P_2 = P(x_1, y^o, \tilde{y}, \mu) - P(x, y^o, \tilde{y}, \mu)$.
b) If $\Delta P_1 + \Delta P_2 \geq \delta\psi_1$ then set $x_o = x$ and repeat from step 2a.
c) If $\Delta P_2 \leq 0$ then set $\Delta x = 0$.
d) Set $\eta = \eta_1, \tilde{y} = \tilde{y}_1, \mu = \mu_1, \psi = \psi_1, \gamma_o = \gamma_1$.
7. Repeat until $P_o - P \geq \delta\alpha\psi$:
a) Set $\alpha := \theta\alpha,\ x = x_o + \alpha(\Delta x_1 + \alpha\Delta x)$.
b) Compute $f^o(x), f(x), L(x), P = (x, y^o, \tilde{y}, \mu)$.
8. Repeat from step 2a.

Appendix C. The subproblem algorithm.

Given: $f^o, f, g^o, G, H, \zeta, \mu, \gamma_o, \tilde{y}$,
$a, b, \epsilon, \epsilon^o$,
$\underline{\alpha}, \beta, \gamma, \delta, \delta_{k,o}, \underline{\rho}, \rho, \sigma, \tau, \omega, \underline{M}$.

Initialization

1. a) Set $\zeta_o = \zeta, \mu_o = \mu, \tilde{\gamma} = \gamma_o, \tilde{y}_o = \tilde{y}$.
b) Set $x^o = 1, \mu^o = \mu, \tilde{\gamma}^o = \gamma, \tilde{\delta} = \sqrt{1-\tau}$.
c) Compute $z^o, z, \overline{z}, r_D, r_c^o, \underline{r}_c, \overline{r}_c, \beta^o, \beta^{o\prime}, \beta'$.
d) If $y^o\mu \leq \epsilon$ and $\max\{\|r_D\|_\infty, \|f\|_\infty, \|\underline{r}_c\|_\infty, \|\overline{r}_c\|_\infty\} \leq \epsilon^o$
then STOP (ζ is a solution).

General Iteration

2. a) Solve (2.18) to obtain $\delta x^o, \delta x, \delta y$.
If the system is inconsistent then set $H = 0$
$\zeta = \zeta_o, \mu^o = \mu = \mu_o, \tilde{\gamma} = \gamma_o$
repeat from step 1b.
b) Compute $\delta z^o, \delta\underline{z}, \delta\overline{z}$ from (2.17).
3. a) Compute α' to satisfy (2.19), updating $\chi, \mu^o, \mu, r_D, r_c^o, \underline{r}_c, \overline{r}_c, \Delta x$.
b) If $\alpha' < \underline{\alpha}$ and $H \neq 0$ then set $H = 0$
$\zeta = \zeta_o, \mu^o = \mu = \mu_o, \tilde{\gamma} = \gamma_o$
repeat from step 1b.
4. a) Compute $s = \|y^o, y\|_\infty$.
b) If $\alpha' \geq \tau$ or $\tilde{\gamma}^o = \gamma$ then set $\tilde{\gamma}^o = \mu^o/\mu_o$,
$\tilde{\delta} = \tilde{\delta}(\max\{1-\tau, 1-\alpha'(1-\tilde{\gamma}^o)\})^{\frac{1}{2}}$,
else set $\tilde{\gamma}^o = \gamma$.
c) If $y^o\mu \leq s \cdot \max(\epsilon, \rho\|\Delta x\|^4)$ then set $\tilde{y}^j = \max(\tilde{y}_o^j, \omega|y^j|)$, all j
$\tilde{\gamma} = 1$
else set $\tilde{y}^j = s\omega$, all j
$\tilde{\gamma} = \tilde{\gamma}^o$
d) Compute $\bar{D} = \underline{X}_o^{-1}Z + \overline{X}_o^{-1}\overline{Z}$

$$\psi = \Delta x^T(\bar{H} + \bar{D})\Delta x + (1-x^o)\sum_{j=1}^{m}(\tilde{y}^j|f^j| - y^j f^j)$$

e) If $\psi < \underline{\rho}\Delta x^T\bar{D}\Delta x$ then set $H = 0, \zeta = \zeta_o, \mu^o = \mu = \mu_o, \tilde{\gamma} = \gamma_o$
repeat from step 1b.
5. If $\Delta x^T(r_D + \underline{X}_o^{-1}\underline{r}_c - \tilde{X}_o^{-1}\overline{r}_c) > \frac{1}{4}(1-2\delta)\psi$ or $y^o\mu^o > s \cdot \max(\epsilon^o, \sigma\|\Delta x\|^4)$

or $y^o\mu > s \cdot \max(\epsilon, \sigma\|\Delta x\|^4)$ then
  a) Compute $\beta^{o\prime}, \beta'$.
  b) Return to step 2a.

6. a) Rescale $\eta := s^{-1}\eta,\ \psi := s^{-1}\psi,\ \tilde{y} := s^{-1}\tilde{y}$.
   b) RETURN.

REFERENCES

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

[2] J. T. Betts and P. D. Frank, *A sparse nonlinear optimization algorithm*, J. Optim. Theory Appl., 3 (1994), pp. 519–541.

[3] 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.

[4] R. H. Byrd, M. E. Hribar, and J. Nocedal, *An interior point algorithm for large scale nonlinear programming*, SIAM J. Optim., 9 (1999), pp. 877–900.

[5] R. M. Chamberlain, M. J. D. Powell, C. Lemarechal, and H. C. Pedersen, *The watchdog technique for forcing convergence in algorithms for constrained optimization*, Math. Prog. Study, 17 (1982), pp. 1–17.

[6] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *LANCELOT: A Fortran Package for Nonlinear Optimization* (*Release A*), Springer Ser. Comput. Math. 17, Springer-Verlag, Berlin, 1992.

[7] I. S. Duff and J. K. Reid, *The multifrontal solution of indefinite sparse symmetric linear systems*, ACM Trans. Math. Software, 9 (1983), pp. 302–325.

[8] I. S. Duff and J. K. Reid, *MA47, A Fortran Code for Direct Solution of Indefinite Sparse Symmetric Linear Systems*, RAL Report Ral-95-001, Rutherford Appleton Laboratory, Didcot, UK, 1995.

[9] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, SIAM, Philadelphia, 1990.

[10] S. P. Han, *A globally convergent method for nonlinear programming*, J. Optim. Theory Appl., 22 (1977), pp. 297–309.

[11] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econom. and Math. Systems 187, Springer-Verlag, New York, 1981.

[12] O. L. Mangasarian and T. H. Shiau, *Error bounds for monotone linear complementarity problems*, Math. Programming, 36 (1986), pp. 81–89.

[13] N. Maratos, *Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization Problems*, Ph.D. thesis, Imperial College, University of London, 1978.

[14] D. Q. Mayne and E. Polak, *A superlinearly convergent algorithm for constrained optimization problems*, Math. Prog. Study, 16 (1982), pp. 45–61.

[15] R. W. H. Sargent, *Infeasible-Interior-Point Methods for Generalized Monotone Linear Complementarity Problems*, Report C94-12, Centre for Process Systems Engineering, Imperial College, London, 1994.

[16] R. W. H. Sargent, *A homework exercise—the "big-M" problem*, in Algorithms for Continuous Optimization—The State of the Art, E. Spedicato, ed., NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci. 434, Kluwer Academic Publishers, Dordrecht, 1994, pp. 475–479.

[17] R. W. H. Sargent, *The development of the SQP algorithm for nonlinear programming*, in Large-Scale Optimization with Applications, Part II: Optimal Design and Control, L. T. Biegler, T. F. Coleman, A. R. Conn, and F. N. Santosa, eds., IMA Vol. Math. Appl. 93, Springer-Verlag, New York, 1997, pp. 1–19.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 748–760

REGULARIZATION OF $\mathbf{P}_0$-FUNCTIONS IN BOX VARIATIONAL INEQUALITY PROBLEMS*

G. RAVINDRAN† AND M. SEETHARAMA GOWDA‡

Abstract. Two recent papers [F. Facchinei, *Math. Oper. Res.*, 23 (1998), pp. 735–745 and F. Facchinei and C. Kanzow, *SIAM J. Control Optim.*, 37 (1999), pp. 1150–1161] have shown that for a continuously differentiable $\mathbf{P}_0$-function f, the nonlinear complementarity problem NCP(f_ε) corresponding to the regularization $f_\varepsilon(x) := f(x) + \varepsilon x$ has a unique solution for every $\varepsilon > 0$, that dist $(x(\varepsilon), \mathrm{SOL}(f)) \to 0$ as $\varepsilon \to 0$ when the solution set SOL(f) of NCP(f) is nonempty and bounded, and NCP(f) is stable if and only if the solution set is nonempty and bounded. These results are proved via the Fischer function and the mountain pass theorem. In this paper, we generalize these nonlinear complementarity results to a box variational inequality problem corresponding to a continuous $\mathbf{P}_0$-function where the regularization is described by an integral. We also describe an upper semicontinuity property of the inverse of a weakly univalent function and study its consequences.

Key words. complementarity problem, box variational inequality problem, regularization, weakly univalent function

AMS subject classifications. 90C33, 90C30, 65H10

PII. S1052623497329567

1. Introduction.

Consider a continuous function $f : R^n \to R^n$ and a rectangular box K in R^n. Then the *box variational inequality problem*, denoted by BVI(f, K), is to find an $x^* \in K$ such that

$$\langle f(x^*), x - x^* \rangle \geq 0 \quad \text{for all} \quad x \in K. \tag{1.1}$$

When $K = R^n_+$, this problem reduces to the nonlinear complementarity problem NCP(f): Find $x^* \in R^n$ such that

$$x^* \geq 0, \quad f(x^*) \geq 0, \quad \text{and} \quad \langle f(x^*), x^* \rangle = 0. \tag{1.2}$$

Both the NCP and BVI have been extensively studied in the literature; see [4], [5], [6], [10], [11], [12], [15], [20], [21], [23], and the references therein.

We say that f is a $\mathbf{P}_0$ ($\mathbf{P}$)-function if for every pair (x, y) with $x \neq y$,

$$\max_{x_i \neq y_i} (x - y)_i [f_i(x) - f_i(y)] \geq 0 \quad (> 0). \tag{1.3}$$

Generalizing earlier results for monotone functions, Facchinei [8] and Facchinei and Kanzow [9] have shown the following in the NCP setting: Consider a continuously differentiable $\mathbf{P}_0$-function f and its Tikhonov regularization $f_\varepsilon(x) := f(x) + \varepsilon x$. Then

(a) NCP(f_ε) has a unique solution $x(\varepsilon)$ for each $\varepsilon > 0$.

(b) When the solution set SOL(f) of NCP(f) is nonempty and bounded,

$$\text{dist } (x(\varepsilon), \mathrm{SOL}(f)) \to 0 \quad \text{as} \quad \varepsilon \to 0.$$

*Received by the editors November 3, 1997; accepted for publication (in revised form) July 14, 2000; published electronically November 15, 2000.

http://www.siam.org/journals/siopt/11-3/32956.html

†Indian Statistical Institute, 8th Mile, Mysore Road, Bangalore 560 059, India (ravi@isibang.ac.in).

‡Department of Mathematics and Statistics, University of Maryland–Baltimore County, Baltimore, MD 21250 (gowda@math.umbc.edu, http://www.math.umbc.edu/∼gowda). The work of this author was based on research supported by National Science Foundation grant CCR-9307685.

(c) SOL(f) is stable if and only if it is nonempty and bounded.

Although item (a) follows from a result of Megiddo and Kojima [19, Thm. 3.4], the method of proving these results in the cited papers via the Fischer function and the mountain pass theorem is quite interesting and novel. In a related paper, Sun [23] carries out an algorithmic analysis of a continuously differentiable $\mathbf{P}_0$ complementarity problem via regularization techniques.

In this paper, we generalize the results of Facchinei and Kanzow in several ways. We consider a BVI instead of a NCP, assume only continuous $\mathbf{P}_0$-property of f, and deal with integral regularizations of the fixed point map of BVI (1.1) of the form

$$\widehat{F}_\varepsilon(x) := \int_R \{x - \Pi_K(x - f(x) - \varepsilon x - \varepsilon s\,\mathbf{e})\} d\mu(s),$$

where $\mathbf{e}$ is the vector of ones in R^n, and μ is a Borel measure on R. Our analysis is based on degree theory and the classical result that a coercive local homeomorphism of R^n is a global homeomorphism of R^n. In contrast to our theoretical analysis, Qi [20] makes an algorithmic study of a BVI with a continuously differentiable $\mathbf{P}_0$-function via the mountain pass theorem and the normal map.

2. Preliminary results. Throughout this paper, K denotes a rectangular box in R^n, i.e.,

$$K = K_1 \times K_2 \times \cdots \times K_n,$$

where each K_i is a closed interval in R. It is well known that BVI(f, K) is equivalent to finding a zero of the (fixed point map) $\widehat{f}$ defined by

$$\widehat{f}(x) := x - \Pi_K(x - f(x)), \tag{2.1}$$

where Π_K denotes the (orthogonal) projection onto K. We note that when $K = R^n_+$ (the nonnegative orthant),

$$\widehat{f}(x) = x \wedge f(x),$$

where "$\wedge$" denotes the componentwise minimum of vectors involved.

Given a continuous function $f : R^n \to R^n$, $\varepsilon > 0$, and a (positive) Borel measure μ on R [22] with

$$\mu(R) = 1 \text{ and } \Delta := \int_R |s| d\mu(s) < \infty,$$

we define the following regularizations of $\widehat{f}$:

$$\widehat{f}_\varepsilon(x) := \int_R \{x - \Pi_K(x - f(x) - \varepsilon s\mathbf{e})\} d\mu(s) \tag{2.2}$$

and

$$\widehat{F}_\varepsilon(x) := \int_R \{x - \Pi_K(x - f(x) - \varepsilon x - \varepsilon s\mathbf{e})\} d\mu(s), \tag{2.3}$$

where the integration is performed componentwise. As in various regularization methods, the objective here is to study the zero set of $\widehat{f}$ (i.e., the solution set of BVI(f, K))

via the zero sets of $\widehat{f}_\varepsilon$ and $\widehat{F}_\varepsilon$. The "transforms" (2.2) and (2.3) appear in smoothing methods for NCP and BVI; see [4], [5], [6], [12], and the references therein. Using the nonexpansive property of the projection map $x \mapsto \Pi_K(x)$ and writing $||x||$ for the Euclidean norm of x in R^n, we easily get the following inequalities:

$$||\widehat{f}_\varepsilon(x) - \widehat{f}(x)|| \leq \varepsilon\sqrt{n}\Delta \quad (x \in R^n)$$

and for any compact set E, there is a constant C such that

$$||\widehat{F}_\varepsilon(x) - \widehat{f}(x)|| \leq \varepsilon C \quad (x \in E). \tag{2.4}$$

These inequalities show that $\widehat{f}_\varepsilon$ and $\widehat{F}_\varepsilon$ are approximations of $\widehat{f}$ for small ε.

To illustrate the above regularizations, we consider the following examples.

Example 1. We let $K = R^n_+$ and μ be the point mass at the origin so that $\int_R g\, d\mu = g(0)$ for every continuous g on R. Then

$$\widehat{f}_\varepsilon(x) = x \wedge f(x) \quad \text{and} \quad \widehat{F}_\varepsilon(x) = x \wedge [f(x) + \varepsilon x].$$

We note that the zero set of $\widehat{F}_\varepsilon$ is precisely the solution set of NCP(f_ε) where $f_\varepsilon(x) = f(x) + \varepsilon x$ is the Tikhonov regularization of f.

Other examples are obtained by putting $d\mu = \rho(s)ds$ where ρ is a density function on R [4], [5], [12]. Specifically, when $K = R^n_+$, using the terminology of [5] and writing x_+ for $\Pi_{R^n_+}(x)$, we let

$$\widehat{p}(x, \varepsilon) := \int_R (x - t\mathbf{e})_+ \frac{1}{\varepsilon}\rho\left(\frac{t}{\varepsilon}\right) dt = \int_R (x - \varepsilon s\mathbf{e})_+ \rho(s)ds$$

so that

$$\widehat{f}_\varepsilon(x) = x - \widehat{p}(x - f(x), \varepsilon)$$

and

$$\widehat{F}_\varepsilon(x) = x - \widehat{p}(x - f(x) - \varepsilon x, \varepsilon).$$

In the following two examples, we specify ρ and $\widehat{p}(x, \varepsilon)$; in each case, the corresponding $\widehat{f}_\varepsilon$ and $\widehat{F}_\varepsilon$ are given by the above expressions. For further examples, we refer to [5].

Example 2 (see [5]). We let

$$\rho(s) := \frac{e^{-s}}{(1 + e^{-s})^2},$$

which is the so-called neural networks smooth plus function. Then for $x \in R^n$,

$$\widehat{p}(x, \varepsilon) = x + \varepsilon \log(\mathbf{e} + e^{-\frac{x}{\varepsilon}}),$$

where we recall that $\mathbf{e}$ is the vector of ones in R^n.

Example 3 (see [5]). Here we let

$$\rho(s) := \frac{2}{(s^2 + 4)^{\frac{3}{2}}},$$

which is the so-called Chen–Harker–Kanzow–Smale smooth plus function. Then for $x \in R^n$,

$$\widehat{p}(x, \varepsilon) = \frac{x + \sqrt{x^2 + 4\varepsilon^2\mathbf{e}}}{2}.$$

2.1. P and P_0-properties. In this subsection, we establish the **P** and $\mathbf{P}_0$-properties of $\widehat{f}_\varepsilon$ and $\widehat{F}_\varepsilon$ given by (2.2) and (2.3).

PROPOSITION 2.1. *For a continuous function $f : R^n \to R^n$, let*

$$\theta(x) := x - \Pi_K(x - f(x)),$$
$$\theta(x, \varepsilon, s) := x - \Pi_K(x - f(x) - \varepsilon s\mathbf{e}) \quad (s \in R,\ \varepsilon > 0),$$

and

$$\phi(x, \varepsilon, s) := x - \Pi_K(x - f(x) - \varepsilon x - \varepsilon s\mathbf{e}) \quad (s \in R,\ \varepsilon > 0).$$

Then

(a) *$\theta(x)$ and $\theta(x, \epsilon, s)$ are **P**-functions ($\mathbf{P}_0$-functions) in x (for fixed s and ε) whenever f is a **P**-function ($\mathbf{P}_0$-function);*

(b) *$\phi(x, \varepsilon, s)$ is a **P**-function in x (for fixed s and ε) when f is a $\mathbf{P}_0$-function.*

Proof. (a) Assume that f is a **P**-function. Let $x \neq y$ and pick an i such that

$$(x_i - y_i)[f_i(x) - f_i(y)] > 0.$$

Without loss of generality, let $x_i > y_i$ so that $f_i(x) > f_i(y)$. We show that θ is a **P**-function by showing that $(x_i - y_i)[\theta_i(x) - \theta_i(y)] > 0$. If the inequality were not true, then $\theta_i(x) \leq \theta_i(y)$ which means that

$$x_i - \Pi_{K_i}(x_i - f_i(x)) \ \leq \ y_i - \Pi_{K_i}(y_i - f_i(y)).$$

By considering all possible values of the quantities involved in the above expression, we can check (see the appendix) that the above inequality cannot hold. We conclude that θ is a **P**-function. A similar argument shows that θ is a $\mathbf{P}_0$-function when f is a $\mathbf{P}_0$-function. Since f is a **P** ($\mathbf{P}_0$)-function if and only if $f(x) + \varepsilon\mathbf{e}s$ is a **P** ($\mathbf{P}_0$)-function, we get the stated assertion about $\theta(x, \varepsilon, s)$. (b) follows easily from (a) since $f(x) + \varepsilon x$ is a **P**-function when f is a $\mathbf{P}_0$-function. □

Remark 1. The proof of the above proposition actually shows the following: Suppose f is a **P**-function ($\mathbf{P}_0$-function) and $x \neq y$. If $x_i \neq y_i$ and

$$(x_i - y_i)[f_i(x) - f_i(y)] > 0 \qquad (\geq 0),$$

then for the same index i,

$$(x_i - y_i)[\theta_i(x, \varepsilon, s) - \theta_i(y, \varepsilon, s)] > 0 \ (\geq 0) \quad \text{for all } s \in R,\ \varepsilon > 0,$$

and

$$(x_i - y_i)[\phi_i(x, \varepsilon, s) - \phi_i(y, \varepsilon, s)] > 0 \quad \text{for all } s \in R,\ \varepsilon > 0.$$

PROPOSITION 2.2. *Given f and μ, let $\widehat{f}_\varepsilon$ and $\widehat{F}_\varepsilon$ be as defined in* (2.2) *and* (2.3). *Then the following statements hold:*

(a) *If f is a $\mathbf{P}_0$ (**P**)-function, then $\widehat{f}_\varepsilon$ is a $\mathbf{P}_0$ (**P**)-function.*

(b) *If f is a $\mathbf{P}_0$-function, then $\widehat{F}_\varepsilon$ is a **P**-function and hence one-to-one.*

Proof. Let f be a $\mathbf{P}_0$ (**P**)-function. Fix $x \neq y$ in R^n. Then there exists an index i such that $x_i \neq y_i$ and $(x_i - y_i)[f_i(x) - f_i(y)] \geq 0$ (> 0). From Remark 1, we see that

$$(x_i - y_i)[\theta_i(x, \varepsilon, s) - \theta_i(y, \varepsilon, s)] \geq 0 \qquad (> 0) \quad \text{for all } s \in R.$$

Since $\mu(R) > 0$, integration leads to

$$(x_i - y_i)[\widehat{f}_\varepsilon(x) - \widehat{f}_\varepsilon(y)]_i \geq 0 \qquad (> 0).$$

Thus we have (a). Item (b) is proved by applying (a) to the **P**-function $f(x) + \varepsilon x$. The one-to-one assertion follows from the **P**-property. □

In the result below, we identify a condition under which $\widehat{f}_\varepsilon$ is a **P**-function.

PROPOSITION 2.3. *Suppose f is a $\mathbf{P}_0$-function and for each i, the closed interval K_i is bounded either below or above, and μ does not vanish on any infinite interval. Then $\widehat{f}_\varepsilon$ is a* **P***-function.*

Proof. Let $x \neq y$. From Remark 1 and the previous proposition we know that for some index i, $x_i \neq y_i$,

$$(x_i - y_i)[\theta_i(x, \varepsilon, s) - \theta_i(y, \varepsilon, s)] \geq 0 \quad \text{for all } s \in R$$

and

$$(x_i - y_i)[\widehat{f}_\varepsilon(x) - \widehat{f}_\varepsilon(y)]_i \geq 0.$$

We claim that the latter inequality is strict. Assume the contrary and let, without loss of generality, $x_i > y_i$ and $[\widehat{f}_\varepsilon(x) - \widehat{f}_\varepsilon(y)]_i = 0$. It follows that

$$\theta_i(x, \varepsilon, s) = \theta_i(y, \varepsilon, s) \quad \text{a.e. } \mu, \tag{2.5}$$

that is, the set of s where the above equality fails to hold has μ measure zero. Assume that K_i is bounded below by $l_i > -\infty$ (the case of K_i being bounded above is similar). Then for all s in some interval $[\delta, \infty)$, $x_i - f_i(x) - \varepsilon s \leq l_i$ and $y_i - f_i(y) - \varepsilon s \leq l_i$; hence for all such s, $\theta_i(x, \varepsilon, s) = x_i - l_i$ and $\theta_i(y, \varepsilon, s) = y_i - l_i$. Since $x_i - l_i \neq y_i - l_i$, we see from (2.5) that $\mu[\delta, \infty) = 0$, contradicting the assumption on μ. □

COROLLARY 2.4. *Suppose f is a $\mathbf{P}_0$-function and $K = R^n_+$. Then*

$$\widehat{f}(x) = x \wedge f(x)$$

and

$$\widehat{f}_\varepsilon(x) = \int_R x \wedge (f(x) + \varepsilon \mathbf{e} s) d\mu(s)$$

are $\mathbf{P}_0$-functions. Moreover, $\widehat{f}_\varepsilon$ will be a **P***-function when one of the following conditions is satisfied:*

(i) *f is a* **P***-function;*

(ii) *μ does not vanish on any infinite interval of the form $[\delta, \infty)$.*

Proof. The stated property of $\widehat{f}$ follows from Proposition 2.2(a) by taking μ to be the point mass at the origin. For the stated properties of $\widehat{f}_\varepsilon$ see Proposition 2.2(a) and the proof of Proposition 2.3. □

Remark 2. Corollary 2.4 says that the composition of a $\mathbf{P}_0$ (**P**)-function and the min function is a $\mathbf{P}_0$ (**P**)-function. Similar things can be said about the Fischer function. Recall that the ith component of the Fischer function $\psi : R^n \times R^n \to R^n$ is given by

$$\psi_i(a, b) = (a_i + b_i) - \sqrt{a_i^2 + b_i^2}.$$

We claim that *if f is a $\mathbf{P}$-function, then so is $\Psi(x) := \psi(x, f(x))$.*

To see this, let $x \neq y$. Then there exists an index i such that $(x_i - y_i)(f_i(x) - f_i(y)) > 0$. We show that for the same index i, $(x_i - y_i)(\Psi_i(x) - \Psi_i(y)) > 0$. Without loss of generality, we assume that $x_i > y_i$ and show that $\Psi_i(x) - \Psi_i(y) > 0$. Assume the contrary, $\Psi_i(x) \leq \Psi_i(y)$, and let, for simplicity, $\alpha = x_i$, $\beta = y_i$, $\gamma = f_i(x)$, $\delta = f_i(y)$. We have $\alpha > \beta$ and $\gamma > \delta$. Now $\Psi_i(x) \leq \Psi_i(y)$ leads to $(\alpha - \beta) + (\gamma - \delta) \leq \sqrt{\alpha^2 + \gamma^2} - \sqrt{\beta^2 + \delta^2}$. Squaring, simplifying, and using the inequality $(\alpha - \beta)(\gamma - \delta) > 0$, we get

$$\sqrt{\alpha^2 + \gamma^2}\sqrt{\beta^2 + \delta^2} < \alpha\beta + \gamma\delta,$$

which upon squaring leads to $(\alpha\delta - \beta\gamma)^2 < 0$. We conclude that Ψ is a $\mathbf{P}$-function. The $\mathbf{P}_0$-property is similarly established. Note that the zeros of Ψ are precisely the solutions of NCP(f).

Remark 3. Let f be a $\mathbf{P}_0$-function. Let $\eta(x) : R^n \to R^n$ be a function whose ith component function η_i is a function of x_i only and that it is strictly increasing in that variable. Then it is easily verified that for any $\varepsilon > 0$, $f(x) + \varepsilon\eta(x)$ is a $\mathbf{P}$-function. In particular $f(x) + \varepsilon\eta(x)$ is a $\mathbf{P}$-function where η is given, for some (disjoint) index sets I, J, and an $\overline{x} \in R^n$, by

$$\eta_i(x) = \begin{cases} -e^{-x_i} & \text{if } i \in I, \\ e^{x_i} & \text{if } i \in J, \\ (x_i - \overline{x}_i) & \text{if } i \notin I \cup J. \end{cases}$$

2.2. An upper semicontinuity property. In this section, we digress a little bit from our main theme to describe an upper semicontinuity property of the (multivalued) inverse of a weakly univalent function. Weakly univalent functions were introduced in [13]; they are generalizations of $\mathbf{P}_0$-functions. The results of this section, which are also of independent interest, are crucial to the proof of the main result (Theorem 3.3) of the paper.

THEOREM 2.5. *Let $g : R^n \to R^n$ be weakly univalent, that is, g is continuous and there exist one-to-one continuous functions $g_k : R^n \to R^n$ such that $g_k \to g$ uniformly on every bounded subset of R^n. Suppose that $q^* \in R^n$ such that $g^{-1}(q^*)$ is nonempty and compact. Then for any given $\varepsilon > 0$, there exists a $\delta > 0$ such that for any weakly univalent function h and for any q with*

$$\sup_{\overline{\Omega}} ||h(x) - g(x)|| < \delta, \quad ||q - q^*|| < \delta, \tag{2.6}$$

we have

$$\emptyset \neq h^{-1}(q) \subseteq g^{-1}(q^*) + \varepsilon\mathcal{B}, \tag{2.7}$$

where $\mathcal{B}$ denotes the open unit ball in R^n and $\Omega := g^{-1}(q^*) + \varepsilon\mathcal{B}$. In particular, $h^{-1}(q)$ and $g^{-1}(q)$ are nonempty, connected, and uniformly bounded for q in a neighborhood of q^*.

Proof. Let $\varepsilon > 0$ and $\Omega := g^{-1}(q^*) + \varepsilon\mathcal{B}$. Under the stated assumptions on g, it follows from Remark (2) in section 3 of [13] (together with the excision property of the degree) that

$$deg\,(g, \Omega, q^*) = \pm 1. \tag{2.8}$$

Let $\delta := \frac{1}{2}\,\mathrm{dist}\,(q^*, g(\partial\Omega))$, where $\partial\Omega$ denotes the boundary of Ω. Then for h and q satisfying (2.6),

$$\sup_{\overline{\Omega}} ||(h(x) - q) - (g(x) - q^*)|| < \mathrm{dist}\,(q^*, g(\partial\Omega)),$$

and hence by the nearness property of the degree [17, Thm. 2.1.2],

$$deg\,(h, \Omega, q) = \pm 1.$$

It follows that $h(x) = q$ will have a solution in Ω and no solutions on $\partial\Omega$. We now claim that $h^{-1}(q) \subseteq \Omega$. The decomposition

$$h^{-1}(q) = [h^{-1}(q) \cap \Omega] \cup [h^{-1}(q) \cap (\overline{\Omega})^c]$$

shows that $h^{-1}(q) \cap \Omega$ is a nonempty, bounded, closed, and open subset of $h^{-1}(q)$. It follows from Theorem 2 in [13] that $h^{-1}(q) = h^{-1}(q) \cap \Omega$, proving (2.7). The same theorem proves the connectedness of $h^{-1}(q)$. Finally putting $h = g$, we get the stated assertion about g. This completes the proof. □

The above theorem has a number of important consequences.

COROLLARY 2.6. *Let g be weakly univalent, Y be a closed convex subset of R^n such that $g^{-1}(Y)$ is bounded, and for some $q^* \in Y$, $g^{-1}(q^*) \neq \emptyset$. Then for each $q \in Y$, $g^{-1}(q) \neq \emptyset$ and $g^{-1}(Y)$ is connected.*

Proof. For any fixed $q \in Y$, we consider the homotopy $H(x,t) := g(x) - [tq^* + (1-t)q]$. By (2.8) and the homotopy invariance of the degree, we conclude that the degree of $g(x) - q$ over a suitable bounded open set is nonzero proving the nonemptyness of $g^{-1}(q)$. It follows from Theorem 2.5 that g^{-1} is upper semicontinuous at each point of Y. Since Y is connected, $g^{-1}(Y)$ is closed, $g^{-1}(q)$ is connected for all $q \in Y$, and g^{-1} is upper semicontinuous at each point of Y, it follows that $g^{-1}(Y)$ is also connected. □

Remark 4. It follows from Corollary 2.6 that if a weakly univalent function $g : R^n \to R^n$ is proper (that is, the inverse image of any compact set is compact), then it is onto.

To see our next consequence, consider a $\mathbf{P}_0$-function f. Then by Proposition 2.2(b), $\widehat{F}_\varepsilon$ is a univalent function. Letting $\varepsilon \to 0$ in (2.4), we see that $\widehat{f}$ is a weakly univalent function. Theorem 2.5 now gives the following.

COROLLARY 2.7. *Let f be a continuous $\mathbf{P}_0$-function and let $\widehat{f}$ be given by* (2.1). *Suppose that $(\widehat{f})^{-1}(q^*)$ is nonempty and compact. Then for each $\varepsilon > 0$, there exists a $\delta > 0$ such that*

$$\emptyset \neq (\widehat{f})^{-1}(q) \subseteq (\widehat{f})^{-1}(q^*) + \varepsilon\mathcal{B} \tag{2.9}$$

for all q with $||q - q^*|| < \delta$. In particular, $(\widehat{f})^{-1}(q)$ is nonempty and (uniformly) bounded for all q in a neighborhood of q^*.

Remark 5. Let f be as in Corollary 2.7. Suppose that $(\widehat{f})^{-1}(0)$ is nonempty and bounded. Then $(\widehat{f})^{-1}(q)$ is uniformly bounded for $||q||$ small. This can be described equivalently by means of level sets: For all small positive numbers α, the level sets

$$\{x : ||\widehat{f}(x)|| \leq \alpha\}$$

are bounded. Such a boundedness result has been used to analyze convergence in various iterative schemes; see, e.g., [6] and [24].

Remark 6. As yet another illustration of Theorem 2.5, suppose f is a $\mathbf{P}_0$-function, and consider $\Psi(x) = \psi(x, f(x))$ where ψ is the Fischer function mentioned in Remark 2. We know from Remark 2 that Ψ is a $\mathbf{P}_0$-function and $(\Psi)^{-1}(0)$ is the solution set of the nonlinear complementarity problem NCP(f). By applying Theorem 2.5, one can state a result similar to Corollary 2.7 with Ψ in place of $\widehat{f}$. A modification of Remark 5 for Ψ gives the following result: For all small positive numbers α, the level sets

$$\{x : ||\Psi(x)|| \leq \alpha\}$$

are bounded. This result generalizes Lemma 4.3 in [8] proved for a continuously differentiable $\mathbf{P}_0$-function via the mountain pass theorem.

We now state an upper semicontinuity property of the solution set of a BVI.

COROLLARY 2.8. *Let f be a continuous $\mathbf{P}_0$-function and $q \in R^n$. Let* BVI(f, K, q) *and* SOL(f, K, q) *denote, respectively, the BVI problem corresponding to the function $f(x) + q$ on K and its solution set. Suppose that for $q^* \in R^n$,* SOL(f, K, q^*) *is nonempty and bounded. Then for each $\varepsilon > 0$, there exists a $\delta > 0$ such that*

$$\emptyset \neq \text{SOL}(f, K, q) \subseteq \text{SOL}(f, K, q^*) + \varepsilon\mathcal{B} \tag{2.10}$$

for all q with $||q - q^*|| < \delta$. In particular, SOL(f, K, q) *is nonempty, connected, and (uniformly) bounded for all q in a neighborhood of q^*.*

Proof. The result follows from Theorem 2.5 by putting

$$g(x) = x - \Pi_K(x - f(x) - q^*), \quad \text{and} \quad h(x) = x - \Pi_K(x - f(x) - q). \qquad \square$$

When Corollary 2.8 is specialized to $K = R^n_+$, we get the following.

COROLLARY 2.9. *If the nonlinear complementarity problem* NCP(f) *corresponding to a continuous $\mathbf{P}_0$-function has a nonempty bounded solution set, then the problem is strictly feasible, that is, there exists an x^* such that*

$$x^* > 0 \quad \textit{and} \quad f(x^*) > 0.$$

Proof. In Corollary 2.8, we put $K = R^n_+$, $q^* = 0$, and take $q < 0$ sufficiently close to zero. Then SOL(f, R^n_+, q) is nonempty and every solution u in SOL(f, R^n_+, q) satisfies $u \geq 0$ and $f(u) \geq -q > 0$. By continuity we produce an x^* satisfying the properties listed above. $\square$

Remark 7. In Corollary 2.9 we considered an NCP. The BVI version reads as follows. Suppose K is a rectangular box with 0^+K denoting the recession cone of K and $(0^+K)^*$ denoting the dual of 0^+K. If $(0^+K)^*$ has a nonempty interior and f is a continuous $\mathbf{P}_0$-function with SOL$(f, K, 0)$ nonempty and bounded, then there exists an x^* such that

$$x^* \in K \quad \text{and} \quad f(x^*) \in \text{int}\,(0^+K)^*.$$

This can be seen by taking $q \in -\text{int}\,(0^+K)^*$ that is close to zero and applying (2.10) to get a solution x^* of SOL(f, K, q). The inequality (1.1) shows that $f(x^*) + q \in (0^+K)^*$ from which we get the stated properties of x^*.

Now consider a continuous monotone f so that for all x, y,

$$\langle f(x) - f(y), x - y\rangle \geq 0.$$

In this setting, it is well known that NCP(f) is solvable with a bounded solution set whenever it has a strictly feasible solution (see [16, Thm. 4.1] or [15, Thm. 3.4]).

Corollary 2.9 proves the converse of this result. While we have used degree theoretic considerations to prove this converse, McLinden has proved this converse in the setting of maximal monotone operators [18, Thm. 4] and Chen, Chen, and Kanzow [3] prove it (for a continuously differentiable f) using the Fischer function and the mountain pass theorem. Easy examples of affine $\mathbf{P}_0$-functions show that the converse statement in the above corollary need not hold.

2.3. A coercivity property. We saw in Proposition 2.2 that for a continuous $\mathbf{P}_0$-function f, $\widehat{F}_\varepsilon$ is univalent. Toward establishing the homeomorphism property of $\widehat{F}_\varepsilon$, we prove the following result.

PROPOSITION 2.10. *Suppose f is a $\mathbf{P}_0$-function and define $\widehat{F}_\varepsilon$ as in* (2.3). *Then $\widehat{F}_\varepsilon$ is coercive on R^n, that is, for any sequence $\{x^k\}$ with $||x^k|| \to \infty$, we have $||\widehat{F}_\varepsilon(x^k)|| \to \infty$.*

Proof. Fix ε. To show that $\widehat{F}_\varepsilon$ is coercive, we follow an argument given in the proof of Proposition 3.4 in [8]. Let $\{x^k\}$ be any sequence such that $||x^k|| \to \infty$. Passing through a subsequence, if necessary, we may suppose that there exists an index set J such that for each $i \in J$, $|x_i^k| \to \infty$ as $k \to \infty$ and for $i \notin J$, $\{x_i^k\}$ is bounded. Define a bounded sequence $\{y^k\}$ as follows:

$$y_i^k := \begin{cases} 0 & \text{if } i \in J, \\ x_i^k & \text{if } i \notin J. \end{cases}$$

Since $x^k \neq y^k$ for all large k, we can use the $\mathbf{P}_0$-property of f to get an index $i \in J$ so that without loss of generality,

$$x_i^k[f_i(x^k) - f_i(y^k)] = (x_i^k - y_i^k)[f_i(x^k) - f_i(y^k)] \geq 0$$

for all k. For simplicity we may take $i = 1$ and note that $|x_1^k| \to \infty$. We assume without loss of generality, x_1^k converges either to ∞ or to $-\infty$.

Suppose that x_1^k goes to ∞. Then the above inequality shows (assuming $x_1^k > 0$ for all k) that $f_1(x^k)$ is bounded below by $\alpha := \inf_k f_1(y^k)$. Now consider

$$(\widehat{F}_\varepsilon)_1(x^k) = \int_R \{x_1^k - \Pi_{K_1}[x_1^k - f_1(x^k) - \varepsilon x_1^k - \varepsilon s]\} d\mu(s). \tag{2.11}$$

This integral can be written as the sum of integrals over A_k, B_k, and C_k where

$$A_k := \{s : x_1^k - f_1(x^k) - \varepsilon x_1^k - \varepsilon s < l_1\}$$

when K_1 is bounded below with $l_1 = \inf K_1$,

$$B_k := \{s : x_1^k - f_1(x^k) - \varepsilon x_1^k - \varepsilon s \in K_1\},$$

and

$$C_k := \{s : x_1^k - f_1(x^k) - \varepsilon x_1^k - \varepsilon s > u_1\}$$

when K_1 is bounded above with $u_1 = \sup K_1$.

Note that some of these sets may be empty. When K_1 is bounded, in view of $\mu(R) = 1$, the integral (2.11) behaves like x_1^k and hence goes to infinity as $k \to \infty$. When $K_1 = R$, the integral reduces to $f_1(x^k) + \varepsilon x_1^k + \varepsilon C$ where C is a constant. Since $f_1(x^k)$ is bounded below, even in this case also the integral goes to infinity. Now

consider the case when K_1 is bounded below but not above. Then C_k is empty, the integral in (2.11) reduces to

$$\int_{A_k}(x_1^k - l_1)d\mu(s) + \int_{B_k}[f_1(x^k) + \varepsilon x_1^k + \varepsilon s]d\mu(s) = x_1^k[\mu(A_k) + \varepsilon\mu(B_k)] + \gamma_k,$$

where

$$\gamma_k = -l_1\mu(A_k) + f_1(x^k)\mu(B_k) + \varepsilon\int_{B_k} s d\mu(s) \geq \alpha\mu(B_k) - l_1 - \varepsilon\Delta,$$

where we recall that $\alpha \leq f_1(x^k)$, $\Delta := \int_R |s|d\mu(s)$, and $\mu(R) = 1$. Since $\mu(A_k) + \mu(B_k) = 1$, there exists a positive number δ such that $\mu(A_k) + \varepsilon\mu(B_k) \geq \delta$ for all k large. Hence the integral in (2.11) exceeds $x_1^k\delta +$ constant. It follows that the integral goes to infinity as $k \to \infty$. Similar arguments can be used when K_1 is unbounded below but bounded above. Thus we have shown that as $x_1^k \to \infty$, $(\widehat{F}_\varepsilon)_1(x^k) \to \infty$. The proof that $(\widehat{F}_\varepsilon)_1(x^k) \to -\infty$ as $x_1^k \to -\infty$ is similar; we omit the details. Thus we have shown that for any sequence $\{x^k\}$ going to infinity in the norm, $\{\widehat{F}_\varepsilon(x^k)\}$ goes to infinity in the norm through a subsequence. This proves that for any such sequence $\{x^k\}$, $||\widehat{F}_\varepsilon(x^k)|| \to \infty$ as $k \to \infty$. This completes the proof. ∎

3. The main result. We now consider BVI(f, K) and denote its solution set by SOL(f, K).

For our main theorem below, we introduce the following stability concepts. The first one appears in [8], though with a different name.

DEFINITION 3.1. *We say that* BVI(f, K) *is linearly stable if for every* $\varepsilon > 0$, *there exists a* $\delta^* > 0$ *such that for any continuous function* g *with*

$$||g(x) - f(x)|| \leq \delta^*(1 + ||x||) \quad \textit{for all } x \in \mathrm{SOL}(f, K) + \varepsilon\mathcal{B},$$

BVI(g, K) *has a solution in* SOL$(f, K) + \varepsilon\mathcal{B}$.

DEFINITION 3.2. *We say that* BVI(f, K) *is directionally stable if for every* $\varepsilon > 0$ *and every continuous function* h *there exists a* $\overline{\delta} > 0$ *such that for* $0 \leq \delta \leq \overline{\delta}$, BVI$(f + \delta h, K)$ *has a solution in* SOL$(f, K) + \varepsilon\mathcal{B}$.

We are now ready for our main result.

THEOREM 3.3. *Let* $f : R^n \to R^n$ *be a continuous* $\mathbf{P}_0$*-function and let* $\widehat{F}_\varepsilon$ *be as in* (2.3). *Then the following statements hold:*

(a) *For each* $\varepsilon > 0$, *the equation* $\widehat{F}_\varepsilon(x) = 0$ *has a unique solution* $x(\varepsilon)$. *Moreover, the mapping* $\varepsilon \mapsto x(\varepsilon)$ *from* $(0, \infty)$ *to* R^n *is continuous.*

(b) *If* SOL(f, K) *is nonempty and bounded, then* $\mathrm{dist}(x(\varepsilon), \mathrm{SOL}(f, K)) \to 0$ *as* $\varepsilon \to 0$.

(c) (i) $\Longleftrightarrow$ (ii) $\Longrightarrow$ (iii) *where*

(i) BVI(f, K) *is directionally stable;*

(ii) SOL(f, K) *is nonempty and bounded;*

(iii) BVI(f, K) *is linearly stable.*

Moreover, when K_i *is bounded below for each* i, (iii)$\Longrightarrow$ (ii).

Proof. Statement (a). For $\varepsilon > 0$, Propositions 2.2 and 2.10 show that $\widehat{F}_\varepsilon$ is univalent and coercive. Since coercivity in R^n is the same as properness (that is, the inverse image of any compact set is compact), by a classical result of Banach and Mazur ([1] or [2, Thm. 5.1.4]), it follows that $\widehat{F}_\varepsilon$ is a homeomorphism of R^n and hence $\widehat{F}_\varepsilon(x) = 0$ will have a unique solution. An application of Theorem 2.5 with $q = q^* = 0$, $g := \widehat{F}_{\varepsilon^*}$, and $h := \widehat{F}_\varepsilon$ proves the continuity of $x(\varepsilon)$ at any $\varepsilon^* > 0$.

Statement (b). Now let $\mathrm{SOL}(f,K)$ be nonempty and bounded. For any $\zeta > 0$, let $D = \mathrm{SOL}(f,K) + \zeta\mathcal{B}$. Then for all sufficiently small ε, $\widehat{F}_\varepsilon$ is close to $\widehat{f}$ on $\overline{D}$, and hence by Theorem 2.5, $(\widehat{F}_\varepsilon)^{-1}(0) \subseteq (\widehat{f})^{-1}(0) + \zeta\mathcal{B}$. Since $x(\varepsilon) = (\widehat{F}_\varepsilon)^{-1}(0)$ and $(\widehat{f})^{-1}(0) = \mathrm{SOL}(f,K)$, we have $x(\varepsilon) \in \mathrm{SOL}(f,K) + \zeta B$. Hence for all such ε we have

$$\mathrm{dist}(x(\varepsilon), \mathrm{SOL}(f,K)) \leq \zeta.$$

This implies statement (b).

Statement (c). (i)$\Longrightarrow$ (ii): We suppose (i) holds. Then by definition, $\mathrm{SOL}(f,K)$ is nonempty. To see (ii), that is, to see the boundedness of $\mathrm{SOL}(f,K)$, we follow an argument given in the proof of Theorem 4.4 in [8]. Assume that $\mathrm{SOL}(f,K)$ is unbounded. We produce small perturbations $f(x)+\delta\eta(x)$ for which the corresponding BVI has no solution. Let $\{x^k\}$ be an unbounded sequence of solutions in $\mathrm{SOL}(f,K)$. Without loss of generality, we may take a nonempty index set $I \cup J$ and a vector $\bar{x}$ such that $x_i^k \to \infty\,(i \in I)$, $x_i^k \to -\infty\,(i \in J)$, and $x_i^k \to \bar{x}_i\,(i \notin I \cup J)$. Let η be a function defined by

$$\eta_i(x) := \begin{cases} -e^{-x_i} & \text{if } i \in I, \\ e^{x_i} & \text{if } i \in J, \\ (x_i - \bar{x}_i) & \text{if } i \notin I \cup J. \end{cases}$$

Since $f(x) + \delta\eta(x)$ is a $\mathbf{P}$-function (for $\delta > 0$), by (i), $\mathrm{SOL}(f + \delta\eta, K)$ is singleton for all small $\delta > 0$. For any such δ, by Corollary 2.7 applied to the function $x \mapsto x - \Pi_K[x - f(x) - \delta\eta(x)]$, for all q sufficiently close to zero, the sets

$$\{x :\, x - \Pi_K[x - f(x) - \delta\eta(x)] = q\}$$

are uniformly bounded. We show that this is false by showing

$$x^k - \Pi_K[x^k - f(x^k) - \delta\eta(x^k)] \to 0 \tag{3.1}$$

as $k \to 0$. Now (3.1) follows easily from $x^k - \Pi_K[x^k - f(x^k)] = 0$, $\eta(x^k) \to 0$, and the inequality

$$||\{x^k - \Pi_K[x^k - f(x^k) - \delta\eta(x^k)]\} - \{x^k - \Pi_K[x^k - f(x^k)]\}|| \leq \delta||\eta(x^k)||.$$

Thus we reach a contradiction. Hence (i)$\Longrightarrow$ (ii).

(ii)$\Longrightarrow$ (iii), (ii)$\Longrightarrow$ (i): Assume that $\mathrm{SOL}(f,K)$ is nonempty and bounded and let $\varepsilon > 0$. Then by Remark 2 in section 3 of [13], $deg(\widehat{f}, \Omega, 0) = \pm 1$ for $\Omega := \mathrm{SOL}(f,K) + \varepsilon\mathcal{B}$. It follows from the nearness property of degree [17, Thm. 2.1.2] that for any continuous function g with

$$\sup_{\Omega} ||g(x) - f(x)|| < \widehat{\delta} := \mathrm{dist}(0, \widehat{f}(\partial\Omega)),$$

$deg(\widehat{g}, \Omega, 0) = \pm 1$ where $\widehat{g}(x) = x - \Pi_K(x - g(x))$. Hence $\widehat{g}$ will have a zero in Ω. By taking $\delta^* = \widehat{\delta}(1 + \sup_\Omega ||x||)^{-1}$, we verify the linear stability of $\mathrm{BVI}(f,K)$.

For a given continuous h and $\varepsilon > 0$, we take a $\overline{\delta} > 0$ such that $\overline{\delta}(\sup_\Omega ||h(x)||) < \widehat{\delta}$ and verify the directional stability of $\mathrm{BVI}(f,K)$.

Now suppose that each interval K_i is bounded below. Suppose we have (iii) and that the solution set is unbounded. We proceed as in the proof of (i)$\Longrightarrow$ (ii). Since in this setting the index set J is empty, we see that resulting function $f(x) + \delta\eta(x)$

satisfies the linear stability condition for small δ. As before we get a contradiction when $\mathrm{BVI}(f+\delta\eta, K)$ has a solution for small δ. This completes the proof. □

By specializing K and μ, one could get various special cases of Theorem 3.3. In particular, by taking $K = R^n_+$, and μ as the point mass at the origin, we get the following generalization of Facchinei and Kanzow results (mentioned in the introduction) for continuous $\mathbf{P}_0$-functions in the NCP setting.

COROLLARY 3.4. *Consider* $\mathrm{NCP}(f)$ *where* f *is a continuous* $\mathbf{P}_0$*-function. Let* $f_\varepsilon(x) = f(x) + \varepsilon x$. *Then the following hold.*

(a) *For each* $\varepsilon > 0$, $\mathrm{NCP}(f_\varepsilon)$ *has a unique solution* $x(\varepsilon)$ *and, moreover, the mapping* $\varepsilon \mapsto x(\varepsilon)$ *is continuous on* $(0, \infty)$.
(b) *If* $\mathrm{SOL}(f)$ *is nonempty and bounded, then* $\mathrm{dist}(x(\varepsilon), \mathrm{SOL}(f)) \to 0$ *as* $\varepsilon \to 0$.
(c) $\mathrm{NCP}(f)$ *is linearly stable if and only if* $\mathrm{SOL}(f)$ *is nonempty and bounded.*

4. Concluding remarks. In this paper, based on a result of Banach and Mazur and on degree theory, we have generalized some results of Facchinei and Kanzow. These generalizations deal with integral regularizations of the fixed point map corresponding to a BVI problem. The ideas of the paper can be used in other contexts as well. For example, one could study regularizations based on the normal map and on smoothing (see [14]).

Appendix.

LEMMA A.1. *Consider four real numbers* $\alpha,\ \beta,\ \gamma,\ \delta$ *with* $\alpha > \beta$ *and* $\gamma > \delta$. *Let* L *be a closed interval in* R. *Then*

$$\alpha - \Pi_L(\alpha - \gamma) > \beta - \Pi_L(\beta - \delta).$$

Note. With a limiting argument, one can see that $\alpha > \beta, \gamma \geq \delta \Longrightarrow \alpha - \Pi_L(\alpha - \gamma) \geq \beta - \Pi_L(\beta - \delta)$.

Proof. Suppose, if possible, $\alpha - \Pi_L(\alpha - \gamma) \leq \beta - \Pi_L(\beta - \delta)$. We consider all possible cases and show that in each case, the inequality fails. The possible values of $\alpha - \Pi_L(\alpha - \gamma)$ are

(1) $\alpha - l$ if $\alpha - \gamma \leq l$ where $l = \inf L > -\infty$;
(2) γ if $l \leq \alpha - \gamma \leq u$ where $l = \inf L > -\infty$ and $u = \sup L < \infty$;
(3) $\alpha - u$ if $u \leq \alpha - \gamma$ where $u = \sup L < \infty$;

and the possible values of $\beta - \Pi_L(\beta - \delta)$ are

(a) $\beta - l$ if $\beta - \delta \leq l$ where $l = \inf K > -\infty$;
(b) δ if $l \leq \beta - \delta \leq u$ where $l = \inf L > -\infty$ and $u = \sup L < \infty$;
(c) $\beta - u$ if $u \leq \beta - \delta$ where $u = \sup L < \infty$.

We look at the following cases under the assumption that $\alpha - \Pi_L(\alpha - \gamma) \leq \beta - \Pi_L(\beta - \delta)$.

(i) (1) and (a) hold: This is not possible since $\alpha > \beta$.
(ii) (2) and (b) hold: This is not possible since $\gamma > \delta$.
(iii) (3) and (c) hold: This is not possible since $\alpha > \beta$.
(iv) (1) and (b) hold: Then $\alpha - l \leq \delta$. Since $\alpha - l > \beta - l$, this implies $\beta - l < \delta$ which contradicts statement (b).
(v) (1) and (c) hold: Then $\alpha - u \leq \alpha - l \leq \beta - u$ contradicts $\alpha > \beta$.
(vi) (2) and (a) hold: Then $\gamma \leq \beta - l \leq \delta$ contradicts $\gamma > \delta$.
(vii) (2) and (c) hold: Then $\gamma \leq \beta - u < \alpha - u$ contradicts (2).
(viii) (3) and (a) hold: Then $\gamma \leq \alpha - u \leq \beta - l \leq \delta$ contradicts $\gamma > \delta$.
(ix) (3) and (b) hold: Then $\alpha - u \leq \delta < \gamma$ contradicts (3).

This completes the proof of the lemma. □

REFERENCES

[1] S. Banach and S. Mazur, *Über mehrdeutige stetige Abbildungen*, Studia Math., 5 (1934), pp. 174–178.

[2] M.S. Berger, *Nonlinearity and Functional Analysis*, Academic Press, New York, 1977.

[3] B. Chen, X. Chen, and C. Kanzow, *A penalized Fischer–Burmeister NCP-function*, Math. Program., 88 (2000), pp. 211–216.

[4] C. Chen and O.L. Mangasarian, *Smoothing methods for convex inequalities and linear complementarity problems*, Math. Programming, 71 (1995), pp. 51–69.

[5] 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.

[6] X. Chen and Y. Ye, *On homotopy-smoothing methods for box-constrained variational inequalities*, SIAM J. Control Optim., 37 (1999), pp. 589–616.

[7] R.W. Cottle, J.-S. Pang, and R. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, MA, 1992.

[8] F. Facchinei, *Structural and stability properties of $\mathbf{P}_0$ 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, Eds., *Complementarity and Variational Problems: State of the Art*, SIAM, Philadelphia, PA, 1997.

[11] S. Gabriel, *A hybrid smoothing method for mixed complementarity problems*, Comput. Optim. Appl., 9 (1998), pp. 153–173.

[12] S.A. Gabriel and J.J. Moré, *Smoothing of mixed complementarity problems*, in Complementarity and Variational Problems: State of the Art, SIAM, Philadelphia, PA, 1997, pp. 105–116.

[13] M.S. Gowda and R. Sznajder, *Weak univalence and the connectedness of inverse images of continuous functions*, Math. Oper. Res., 24 (1999), pp. 255–261.

[14] M.S. Gowda and M.A. Tawhid, *Existence and limiting behavior of trajectories associated with $\mathbf{P}_0$-equations*, Comput. Optim. Appl., 12 (1999), pp. 229–251.

[15] 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.

[16] S. Karamardian, *Complementarity problems over cones with monotone and pseudomonotone maps*, J. Optim. Theory Appl., 18 (1976), pp. 445–454.

[17] N.G. Lloyd, *Degree Theory*, Cambridge University Press, Cambridge, UK, 1978.

[18] L. McLinden, *Stable monotone variational inequalities*, Math. Programming, 48 (1990), pp. 303–338.

[19] N. Megiddo and M. Kojima, *On the existence and uniqueness of solutions in nonlinear complementarity theory*, Math. Programming, 12 (1977), pp. 110–130.

[20] H.-D. Qi, *A regularized smoothing Newton method for box constrained variational inequality problems with $\mathbf{P}_0$-functions*, SIAM J. Optim., 10 (2000), pp. 315–330.

[21] L. Qi, D. Sun, and G. Zhou, *A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities*, Math. Program., 87 (2000), pp. 1–35.

[22] W. Rudin, *Real and Complex Analysis*, McGraw–Hill Book Company, New York, 1987.

[23] D. Sun, *A regularization Newton method for solving nonlinear complementarity problems*, Appl. Math. Optim., 40 (1999), pp. 315–339.

[24] P. Tseng, *Analysis of an Infeasible Interior Path-Following Method for Complementarity Problems*, Research Report, Department of Mathematics, University of Washington, Seattle, WA, 1997.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 761–780

THEORY OF GLOBALLY CONVERGENT PROBABILITY-ONE HOMOTOPIES FOR NONLINEAR PROGRAMMING*

LAYNE T. WATSON†

Abstract. For many years globally convergent probability-one homotopy methods have been remarkably successful on difficult realistic engineering optimization problems, most of which were attacked by homotopy methods because other optimization algorithms failed or were ineffective. Convergence theory has been derived for a few particular problems, and considerable fixed point theory exists, but generally convergence theory for the homotopy maps used in practice for nonlinear constrained optimization has been lacking. This paper derives some probability-one homotopy convergence theorems for unconstrained and inequality constrained optimization, for linear and nonlinear inequality constraints, and with and without convexity. Some insight is provided into why the homotopies used in engineering practice are so successful, and why this success is more than dumb luck. By presenting the theory as variations on a prototype probability-one homotopy convergence theorem, the essence of such convergence theory is elucidated.

Key words. constrained optimization, globally convergent, homotopy algorithm, nonlinear equations, probability-one homotopy

AMS subject classifications. 65F10, 65F50, 65H10, 65K10

PII. S105262349936121X

1. Introduction.

Continuation methods for optimization, as for nonlinear systems of equations, have been around for a long time and studied extensively. This paper concerns only a recent variant known as *globally convergent probability-one homotopy* methods. The words "continuation" and "homotopy" are often used interchangeably, but subtle and fundamental distinctions can be drawn between continuation, homotopy, and probability-one homotopy methods. These distinctions have been discussed numerous times in the literature [5], [8], [25], [28], [33]. The purpose of this paper is to help close a gap in the convergence theory for globally convergent probability-one homotopy methods applied to nonlinear programming, and to offer some theoretical justification for the observed success of homotopies in engineering practice.

From a high-level perspective, all the fundamental convergence theory was done by Chow, Mallet-Paret, and Yorke [5] and Watson [23], and all that remains is to verify that a particular homotopy map has the right properties. Alas, the devil is in the details, which are indeed often nontrivial. It is appropriate to sketch out here what is well understood and where gaps remain.

Much of the early work on computational homotopy algorithms was motivated by Brouwer fixed point problems: given a continuous function f from a compact, convex subset of finite dimensional Euclidean space into itself, find a fixed point $x = f(x)$. The algorithms and theory are elegant and well understood for both simplicial [3], [6], [7] and continuous [3], [5], [20] approaches.

*Received by the editors September 3, 1999; accepted for publication (in revised form) August 15, 2000; published electronically December 7, 2000. This work was supported in part by National Science Foundation grant DMS-9625968 and Air Force Office of Scientific Research grant F49620-96-1-0104.

http://www.siam.org/journals/siopt/11-3/36121.html

†Departments of Computer Science and Mathematics, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061-0106 (ltw@cs.vt.edu).

For nonlinear systems of equations $F(x) = 0$ not derived from Brouwer fixed point problems, the theory [3], [5], [25], [26] and algorithms [28], [33] are well developed in terms of properties of F. Special cases, such as when F is a polynomial system, have a deep and rich supporting theory [13], and special, highly sophisticated algorithms have been devised to exploit the structure of F [14], [15], [33]. However, except in rare instances that usually result in polynomial systems, a physical model does not directly result in a finite dimensional nonlinear system of equations $F(x) = 0$. Rather, $F(x) = 0$ results from a discretization, approximation, or iteration step of another mathematical model of the physical phenomenon. The catch is that abstract conditions on F (for a homotopy algorithm to converge) do not easily translate into meaningful or verifiable conditions on the physical model or on the discretization/approximation/iteration process. The gap here is considerable: not many homotopy convergence theorems are stated at the level of physical modeling or the high-level processes that spawn the nonlinear systems $F(x) = 0$ to be solved.

One notable exception is the solution of nonlinear two-point boundary value problems (BVPs). Conditions on the original two-point BVP itself for which an approximation $F(x) = 0$ is solvable by a globally convergent homotopy algorithm have been derived in a series of papers. Convergence theorems directly addressing the nonlinear two-point BVP exist for approximation processes based on shooting [21], finite differences [23], collocation [31], and finite elements [32]. This is significant because many physical models reduce to two-point BVPs, and thus convergence theory exists for a large class of problems of interest.

For nonlinear constrained optimization the gap has been large. Global convergence theorems, stated in terms of conditions on the objective function and constraints, for homotopy algorithms have been an elusive quarry. Some attempts include [16], [22], [24], [27], [29], [30]. Recently Lin et al. [9], [10], using a particular classical homotopy map (not a probability-one map) and constraint aggregation, have obtained convergence results for general nonlinear programming problems with a strong "normal cone condition" assumption. The convergence theory presented here has comparatively weak assumptions, applies to homotopy maps actually used in practice, and does not use constraint aggregation, which is numerically ill-conditioned in practice [30]. Probability-one homotopy algorithms have been enormously successful in engineering practice, notwithstanding the lack of theory. The goal of this work is to narrow the gap by providing such theorems for inequality constraints, and to help explain theoretically the observed success in practice [26]. Extending the theory for nonlinear equality constraints seems to require a homotopy existence theory for underdetermined nonlinear systems, and would at least involve a nontrivial extension of the proofs here. Nonlinear equality constraints are undeniably important, which mandates future work on homotopy theory for them.

There is a variant of probability-one homotopy theory for piecewise smooth functions [1], [2], and this might seem like a more natural tool for constrained optimization. Recent work along these lines includes [4], [18], and [19]. Despite the appeal of these nonsmooth formulations, they are not yet seriously competitive with the existing sophisticated numerical implementations for smooth formulations on realistic large-scale problems [4].

After some background in section 2, the theory is presented as a series of refinements applied to successively more general optimization problems. Sections 3 and 4 summarize some known results, but all the results after Theorem 4.1 are new. The progression is from unconstrained (section 3) to nonnegative constraints (section 4)

to linear constraints (section 5) to nonlinear convex constraints (section 6) to general nonlinear constraints (section 7). First, convexity is assumed and then finally dropped in section 7. Section 7 also provides some insight into why the homotopy maps used in engineering practice might (or might not) work.

2. Background and notation. Let E^n denote n-dimensional Euclidean space, $E^{m\times n}$ the set of real $m \times n$ matrices, and x_i the ith component of a vector $x \in E^n$. The ith row of a matrix $A \in E^{m\times n}$ is denoted by $A_{i\cdot}$ and the jth column by $A_{\cdot j}$. For sets of indices M and N, A_{MN} is the submatrix of A with rows indexed by M and columns indexed by N. Similarly, x_M is the subvector of the vector x corresponding to the indices in M. No distinction is made between row vectors and column vectors, except when matrix arithmetic is involved. Following Mangasarian's notation for $x \in E^n$, $x > 0$ means all $x_i > 0$, $x \geqq 0$ means all $x_i \geqq 0$, and $x \geq 0$ means $x \geqq 0$ but $x \neq 0$. $\|\cdot\|$ is the 2-norm unless otherwise indicated.

The *gradient* of a differentiable function $f : E^n \to E$ is the row vector $\nabla f(x) = \left(\frac{\partial f}{\partial x_1}(x), \ldots, \frac{\partial f}{\partial x_n}(x)\right)$. The *Jacobian matrix* of $F : E^n \to E^m$ is

$$DF(x) = \nabla F(x) = \begin{pmatrix} \nabla F_1(x) \\ \vdots \\ \nabla F_m(x) \end{pmatrix}.$$

The *Hessian matrix* of the C^2 function $f : E^n \to E$ is

$$\nabla^2 f(x) = D(\nabla f(x)) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2}(x) & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_1}(x) \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_1 \partial x_n}(x) & \cdots & \frac{\partial^2 f}{\partial x_n^2}(x) \end{pmatrix}.$$

For open $U \subset E^n$, open $V \subset E^m$, $n > m$, a C^2 map $\rho : U \to V$ is said to be *transversal to zero* if $D\rho$ has full rank on $\rho^{-1}(0)$. Note that in the trivial case where $\rho^{-1}(0)$ is empty, ρ is trivially transversal to zero.

The theoretical foundation of probability-one homotopies (referred to in early work as the Chow–Yorke algorithm) was laid by Chow, Mallet-Paret, and Yorke [5], and the algorithm was immediately recast as a practical computational procedure by Watson [20]. The intent here is not to summarize or survey probability-one homotopy developments—see the survey papers [25] (early history), [26] (applications), and [33] for the latest numerical algorithms.

Depending on the context and intended use, the supporting theory is presented differently. The best formulation for the work here is contained in Lemmas 2.1–2.3 from [22], which are restated here for convenience.

LEMMA 2.1. *Let $\rho : E^m \times [0, 1) \times E^n \to E^n$ be a C^2 map which is transversal to zero, and define*

$$\rho_a(\lambda, z) = \rho(a, \lambda, z).$$

Then for almost all $a \in E^m$, the map ρ_a is also transversal to zero.

Lemma 2.1 is known as a parametrized Sard's theorem, and its significance is partially given by the following.

LEMMA 2.2. *In addition to the hypotheses of Lemma* 2.1*, suppose that for each $a \in E^m$ the system $\rho_a(0, z) = 0$ has a unique nonsingular solution $z^{(0)}$. Then for*

almost all $a \in E^m$ *there is a smooth zero curve* $\gamma \subset [0,1) \times E^n$ *of* $\rho_a(\lambda, z)$*, emanating from* $(0, z^{(0)})$*, along which the Jacobian matrix* $D\rho_a(\lambda, z)$ *has* rank n*.* γ *does not intersect itself or any other zero curves of* ρ_a*, does not bifurcate, has finite arc length in any compact subset of* $[0,1) \times E^n$*, and either goes to infinity or reaches (has an accumulation point in) the hyperplane* $\lambda = 1$*.*

LEMMA 2.3. *Under the hypotheses of Lemma* 2.2*, if the zero curve* γ *is bounded, then it has an accumulation point* $(1, \bar{z})$*. Furthermore, if* rank $D\rho_a(1, \bar{z}) = n$*, then* γ *has finite arc length.*

Conceptually, how all this relates to optimization is as follows: (1) Convert an optimization problem to a nonlinear system of n equations in n unknowns, $F(x) = 0$. (2) Construct a homotopy map $\rho_a(\lambda, x)$ satisfying the hypotheses of the above lemmas, and with $\rho_a(1, x) = F(x)$. (3) Track the zero curve γ of ρ_a from the known point $(0, z^{(0)})$ to a point $(1, \bar{x})$. $\bar{x}$ then solves the original optimization problem. Each of these steps can be fraught with theoretical and computational difficulties, and homotopy algorithms are often considered (with some truth) more art than science. The third step, homotopy zero curve tracking, is close to routine, with robust, numerically stable mathematical software [33] being available. The homotopy construction step is definitely an art, but good maps ρ_a are known for large classes of problems, and several books exist on the topic [13], [3]. The first step, conversion of an optimization problem to a nonlinear system, is perhaps the least understood and most debatable. Why convert a difficult optimization problem into a (possibly even more) difficult nonlinear system? There are enough examples of such counterintuitive conversions being successful (e.g., Karmarkar's algorithm converts a linear program into a series of nonlinear programs) to keep the question open.

3. Unconstrained convex optimization. The simplest possible case, convex unconstrained optimization, is worth mentioning because it shows how everything should work in the ideal case. While a homotopy algorithm is not advocated for convex unconstrained optimization, it is nevertheless reassuring that the theory does cover this case elegantly.

THEOREM 3.1. *Let* $f : E^n \to E$ *be a* C^3 *convex map with a minimum at* $\tilde{x}$*,* $\|\tilde{x}\| \leqq M$*. Then for almost all* a*,* $\|a\| < M$*, there is a zero curve* γ *of the homotopy map*

$$\rho_a(\lambda, x) = \lambda \nabla f(x) + (1 - \lambda)(x - a),$$

along which the Jacobian matrix $D\rho_a(\lambda, x)$ *has full rank, emanating from* $(0, a)$ *and having an accumulation point* $(1, \bar{x})$*, where* $\bar{x}$ *solves*

$$\min_x f(x).$$

If the Hessian matrix $\nabla^2 f(\bar{x})$ *is nonsingular, then* γ *has finite arc length.*

Theorem 3.1 is proved in [24], but a sketch of the proof is repeated here for several reasons. First, it illustrates that a simple proof suffices for the unconstrained case. Second, this proof is a prototype for many homotopy convergence proofs. Often, the essence of a homotopy convergence theorem proof is to construct a map $\rho_a(\lambda, x)$ to which this proof applies, or to generalize the prototype proof to apply to a particular ρ_a.

Let (λ, x) be any point with $0 \leqq \lambda < 1$ and $\|x\| = 3M$. Now $\|a\| < M$ and $\|\tilde{x}\| \leqq M$ give

$$(x - \tilde{x})(x - a) > 0,$$

and the convexity of f at the minimum $\tilde{x}$ gives

$$(x-\tilde{x})\nabla f(x) = (x-\tilde{x})\big(\nabla f(x) - \nabla f(\tilde{x})\big) \geqq 0.$$

Combining these inequalities yields

$$(x-\tilde{x})\big[\lambda\,\nabla f(x) + (1-\lambda)(x-a)\big] > 0,$$

which means that $\rho_a(\lambda, x) \neq 0$ for $0 \leqq \lambda < 1$ and $\|x\| = 3M$. Hence γ is bounded, being contained in the solid cylinder $[0,1] \times \{x \mid \|x\| \leqq 3M\}$. The conclusion follows from Lemma 2.3. □

The essence of the above proof is that the zero curve γ of $\rho_a(\lambda, x)$ emanating from the trivially found starting point $(0, a)$ does not pierce the surface

$$[0,1) \times \{x \mid \|x\| = r\}$$

of some sufficiently large (solid) cylinder containing $(0, a)$. Then γ must be contained inside the solid cylinder, hence bounded, and must therefore pierce (or at least accumulate at) the hyperplane $\lambda = 1$ at a point $(1, \bar{x})$. This prototype convergence proof reveals a fundamental difference between continuation, homotopy, and probability-one homotopy algorithms. For the former two, a convergence theorem would have to address the *existence* and *connectivity* of γ for $0 \leqq \lambda \leqq 1$, requiring assumptions beyond the mere boundedness of γ. In contrast, a probability-one homotopy convergence proof essentially amounts to proving the connected component of $\rho_a^{-1}(0)$ containing $\big(0, z^{(0)}\big)$ is bounded. The other requirements—transversality of ρ, $\rho_a(0, z)$ being a trivial map, $\rho_a(1, z) = F(z)$ —are normally trivially satisfied by the construction of ρ. Finally, note that continuation and homotopy algorithms must typically explicitly deal with singularities along γ, whereas a well-constructed probability-one zero curve γ has no singularities, theoretical or numerical.

4. Nonnegatively constrained convex optimization. Let $f : E^n \to E$ be a C^3 convex function, and say that f is *uniformly convex* if there exists $\nu > 0$ such that $x\big[\nabla^2 f(z)\big]x \geqq \nu\,\|x\|^2$ for all $x, z \in E^n$. Consider next the constrained optimization problem

$$\min f(x) \quad \text{such that} \quad x \geqq 0. \tag{4.1}$$

Since f is convex and Slater's constraint qualification is satisfied, the Kuhn–Tucker optimality conditions are both necessary and sufficient. Hence (4.1) is equivalent to the *nonlinear complementarity problem*

$$x \geqq 0, \qquad F(x) \geqq 0, \qquad x\,F(x) = 0, \tag{4.2}$$

where $F(x) = \nabla f(x)$. There are numerous ways to rewrite (4.2) as a nonsmooth [4] or smooth nonlinear system of equations, but the simplest way (meeting the C^2 requirement for smooth probability-one homotopies), due to Mangasarian [12], is as

$$K(x) = 0, \tag{4.3}$$

where

$$K_i(x) = -\big|F_i(x) - x_i\big|^3 + \big(F_i(x)\big)^3 + x_i^3. \tag{4.4}$$

This choice for $K(x)$ permits the use of the canonical homotopy map

$$\rho_a(\lambda, x) = \lambda\, K(x) + (1-\lambda)(x-a). \tag{4.5}$$

Since (4.1), (4.2), and (4.3) are equivalent, $\rho_a(1, \bar{x}) = 0$ gives a solution $\bar{x}$ to (4.1).

A convergence theorem for (4.5) uses the following general existence result for the nonlinear complementarity problem from [24].

LEMMA 4.1. *Suppose every zero of $K(x)$ lies in the ball $\|x\| < r$, where r is such that $x \geqq 0$ and $\|x\| \geqq r$ imply $x_k > 0$ and $F_k(x) \geqq 0$ for some index k. Then there exists $\delta > 0$ such that for almost all $a \geqq 0$ with $\|a\| < \delta$ there is a zero curve γ of $\rho_a(\lambda, x)$, along which $D\rho_a(\lambda, x)$ has full rank, connecting $(0, a)$ to $(1, \bar{x})$, where $\bar{x}$ is a zero of $K(x)$.*

This lemma directly gives the following theorem (from [24]).

THEOREM 4.1. *Let $f : E^n \to E$ be a C^3 uniformly convex map. Then there exists $\delta > 0$ such that for almost all $a \geqq 0$ with $\|a\| < \delta$ there is a zero curve γ of*

$$\rho_a(\lambda, x) = \lambda\, K(x) + (1-\lambda)(x-a),$$

where

$$K_i(x) = -\left|\frac{\partial f(x)}{\partial x_i} - x_i\right|^3 + \left(\frac{\partial f(x)}{\partial x_i}\right)^3 + x_i^3,$$

along which $D\rho_a(\lambda, x)$ has full rank, connecting $(0, a)$ to a point $(1, \bar{x})$, where $\bar{x}$ solves the constrained optimization problem (4.1).

Note that homotopy convergence theorems are often also *existence* theorems, as is the case with Theorem 4.1, and consequently the assumptions certainly cannot be weaker than are required for existence of a solution. The uniform convexity assumption of Theorem 4.1 is one way to guarantee the existence of a solution to (4.1). If one *assumes* that (4.1) has a solution, then a theorem like the following is possible.

THEOREM 4.2. *Let $f : E^n \to E$ be a C^3 convex map, and assume that* (4.1) *has a solution $\tilde{x}$, and that the level sets of f are bounded. Then there exists $\delta > 0$ such that for almost all $a \geqq 0$ with $\|a\| < \delta$ there is a zero curve γ of*

$$\rho_a(\lambda, x) = \lambda\, K(x) + (1-\lambda)(x-a),$$

where

$$K_i(x) = -\left|\frac{\partial f(x)}{\partial x_i} - x_i\right|^3 + \left(\frac{\partial f(x)}{\partial x_i}\right)^3 + x_i^3,$$

along which $D\rho_a(\lambda, x)$ has full rank, emanating from $(0, a)$ and reaching a point $(1, \bar{x})$, where $\bar{x}$ solves (4.1).

Proof. Since $K(x) = 0$ is equivalent to (4.2), which is equivalent to (4.1), it suffices to verify the hypotheses of Lemma 4.1 for the nonlinear complementarity problem with $F(x) = \nabla f(x)$. First note that by assumption the solutions of (4.1) are bounded, and therefore all the zeros of $K(x)$ lie in some open ball $B(M) = \{\, x \in E^n \mid \|x\| < M \,\}$. That is, every solution $\tilde{x}$ of (4.1) satisfies $\|\tilde{x}\| < M$.

Observe that it suffices to consider only points (λ, x) with $0 \leqq \lambda < 1$ and $x \geqq 0$, since $x_i < 0$, $a_i \geqq 0$ imply $K_i(x) < 0$ and $x_i - a_i < 0$, which then imply $\left[\rho_a(\lambda, x)\right]_i < 0$; hence $x \geqq 0$ along the zero curve γ of ρ_a. $f(x)$ has a maximum at some point $\hat{x}$ on the compact set

$$S_1 = \{\, x \in E^n \mid x \geqq 0, \quad \|x\| = M \,\}.$$

By assumption, the level set

$$S_2 = \left\{ y \in E^n \mid y \geqq 0, \quad f(y) \leqq f(\hat{x}) \right\}$$

is contained in some closed ball $\left\{ x \in E^n \mid \|x\| \leqq r/2 \right\}$. Since $\hat{x} \in S_1 \cap S_2$, $0 < M \leqq r/2 < r$. Now consider any $z \geqq 0$ with $\|z\| = r$. It follows that

$$f(z) > f(\hat{x}) \geqq f\left(\frac{M}{r} z\right) > f(\tilde{x}),$$

and from the convexity of f,

$$\begin{aligned} & f\left(\frac{M}{r} z\right) \geqq f(z) + \nabla f(z)\left(\frac{M}{r} z - z\right) \\ & \Rightarrow \left(1 - \frac{M}{r}\right) z \nabla f(z) \geqq f(z) - f\left(\frac{M}{r} z\right) > 0 \\ & \Rightarrow z \nabla f(z) > 0 \end{aligned}$$

$\Rightarrow z_k > 0$ and $\left(\nabla f(z)\right)_k > 0$ for some index k, the requirement of Lemma 4.1. ∎

Using concepts like recession cones and indicator functions from convex analysis [17], very short proofs can be given for the next two theorems. The essential fact from [17] is that if one nonempty level set is bounded, then all the level sets are bounded. In the interest of maintaining an elementary exposition, short direct proofs are given here. A variant of Theorem 4.2 can be obtained without reference to level sets. One such possibility is the next theorem.

THEOREM 4.3. *Let $f : E^n \to E$ be a C^3 convex map, and assume that* (4.1) *has a solution $\tilde{x}$ at which f is strictly convex. Then the conclusion of Theorem* 4.2 *holds.*

Proof. $\tilde{x}$ is the unique minimum point from strict convexity, and hence the zeros of $K(x)$ are bounded. The proof of Theorem 4.2 applies if it can be shown that the level set S_2 is bounded. Suppose not. Then there exists a sequence $y^{(k)} \in S_2$ with $\left\|y^{(k)}\right\| \to \infty$. The vectors $y^{(k)} / \left\|y^{(k)}\right\|$ lie on the compact unit sphere, and therefore have a convergent subsequence $y^{(k_i)} / \left\|y^{(k_i)}\right\| \to y \geqq 0$. Reduce to this subsequence. For each k, choose $0 < t_k < 1$ such that

$$\left\| (1 - t_k)\tilde{x} + t_k y^{(k)} \right\| = M. \tag{4.6}$$

Now by the strict convexity of f at $\tilde{x}$,

$$f\left((1 - t_k)\tilde{x} + t_k y^{(k)}\right) < (1 - t_k) f(\tilde{x}) + t_k f\left(y^{(k)}\right) \leqq (1 - t_k) f(\tilde{x}) + t_k f(\hat{x}). \tag{4.7}$$

Taking the limit as $k \to \infty$ ($\left\|y^{(k)}\right\| \to \infty$ and (4.6) give $t_k \to 0$) yields

$$f(\tilde{x} + \alpha y) \leqq f(\tilde{x}), \tag{4.8}$$

where $\tilde{x} + \alpha y \geqq 0$, $\|\tilde{x} + \alpha y\| = M > \|\tilde{x}\|$, which contradicts the strict convexity of f at the minimum point $\tilde{x}$. ∎

The most general version of the homotopy convergence theorem for (4.1), whose proof is a refinement of the previous proof, is given last. Theorems 4.2 and 4.3 could have been dispensed with, but presenting the proofs as a series of refinements is instructive.

THEOREM 4.4. *Let $f : E^n \to E$ be a C^3 convex map, assume that* (4.1) *has a solution $\tilde{x}$, and that every solution $\tilde{x}$ satisfies $\|\tilde{x}\| < M$. Then the conclusion of Theorem* 4.2 *holds.*

Proof. By assumption the zeros of $K(x)$ are bounded. The proof of Theorem 4.3, after the first sentence, applies verbatim, with the following changes. Without strict convexity, the strict inequality in (4.7) becomes inequality ($\leqq$), but this doesn't matter in the limit. Equation (4.8) still obtains, but now the contradiction is that $\tilde{x} + \alpha y$ is also a solution of (4.1), which does not satisfy $\|\tilde{x} + \alpha y\| < M$. ∎

5. Linearly constrained convex optimization. Let $f : E^n \to E$ be a C^3 convex function, and let $A \in E^{m\times n}$, $b \in E^m$. First consider the problem

$$\min f(x) \qquad \text{subject to} \qquad g(x) = Ax - b \leqq 0. \tag{5.1}$$

Since both f and g are convex and g satisfies the Arrow–Hurwicz–Uzawa constraint qualification, (5.1) is equivalent to the Kuhn–Tucker problem

$$\big(\nabla f(x)\big)^t + A^t u = 0, \tag{5.2}$$

$$Ax - b \leqq 0, \tag{5.3}$$

$$u \geqq 0, \tag{5.4}$$

$$u^t(Ax - b) = 0. \tag{5.5}$$

As before, the complementarity conditions (5.3)–(5.5) can be replaced by a nonlinear system $K(x, u) = 0$, defined by

$$K_i(x, u) = -|b_i - A_{i\cdot}x - u_i|^3 + (b_i - A_{i\cdot}x)^3 + u_i^3.$$

One possible homotopy map, which has been successful in practice on some difficult engineering optimization problems [30], is

$$\rho_a(\lambda, x, u) = \lambda \begin{pmatrix} \big(\nabla f(x)\big)^t + A^t u \\ K(x, u) \end{pmatrix} + (1 - \lambda) \begin{pmatrix} x - x^0 \\ u - u^0 \end{pmatrix}, \tag{5.6}$$

where $a = (x^0, u^0)$ is the random probability-one homotopy parameter vector, and $u^0 > 0$. This is the direct generalization of what was done for simple nonnegativity constraints $x \geqq 0$, and one would expect it to work. For instance, it is known that a quadratic programming problem with general linear inequality constraints is equivalent to a quadratic programming problem with only nonnegativity constraints. Unfortunately, the homotopy map (5.6) does not suffice. Qualitatively, it worked before because (with convex f) $K(x, u)$ and $u - u^0$ had the same sign for large arguments; thus $\rho_a(\lambda, x, u)$ could not be zero outside some large ball. This meant the zero curve γ of ρ_a could not penetrate the surface of that ball, and hence had to reach a solution of the original problem. In (5.6), x and u can play off against each other, permitting $\rho_a^{-1}(0)$ to be unbounded.

As a simple example, consider $f(x) = (1/2)x^2$ and $g(x) = 1 - x \leqq 0$, and take $x^0 = -1$, $u^0 = 0.1$. The zero curve γ of the homotopy map ρ_a in (5.6) is unbounded, as shown in Figure 1. Of course a lucky guess for (x^0, u^0) may still work. For $(x^0, u^0) = (0, -1)$, γ has several turning points but still reaches $\lambda = 1$ in finite arc length.

The idea behind the repair of (5.6) is to replace (5.3) with

$$Ax - b - (1 - \lambda)b^0 \leqq 0, \tag{5.7}$$

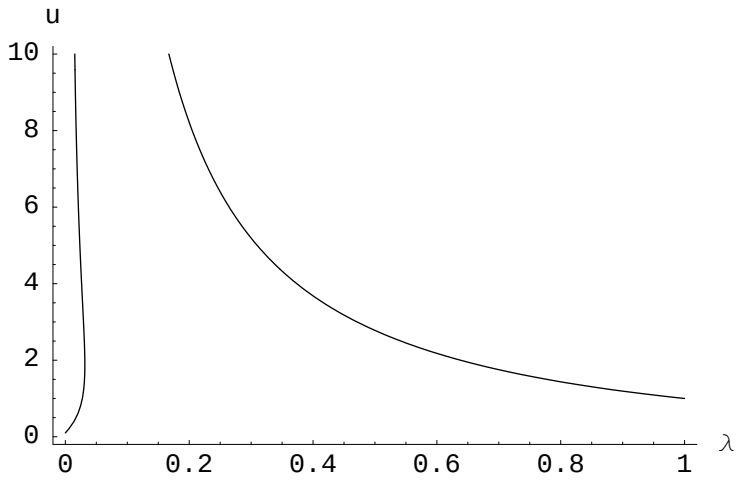


FIG. 1. *Example of unbounded homotopy zero curve.*

and because of the technical necessity to preserve transversality, replace $\lambda K(x,u) + (1-\lambda)(u-u^0)$ with

$$K(\lambda,x,u) - (1-\lambda)c^0. \tag{5.8}$$

Assume that the feasible set $\{x \mid Ax - b \leqq 0\}$ is nonempty and bounded (this is not an obstacle in practice, because variable bounds can always be added). For some arbitrary initial guess $x^0 \in E^n$, choose $b^0 \in E^m$, $b^0 > 0$, such that $Ax^0 - b - b^0 < 0$. Also choose $c^0 \in E^m$ such that $c^0 > 0$. Define $S_\lambda = \{x \mid Ax - b - (1-\lambda)b^0 \leqq 0\}$, and observe that

$$S_0 \supset S_{\lambda_1} \supset S_{\lambda_2} \supset S_1 \neq \emptyset \quad \text{for } 0 \leqq \lambda_1 < \lambda_2 \leqq 1. \tag{5.9}$$

It would be desirable if complementarity could be automatically enforced by defining K with something like

(5.10)
$$\begin{aligned} K_i(\lambda,x,u,b^0,c^0) = &-\left|(1-\lambda)b_i^0 + b_i - A_{i\cdot}x - \left(u_i - (1-\lambda)c_i^0\right)\right|^3 \\ &+ \left((1-\lambda)b_i^0 + b_i - A_{i\cdot}x\right)^3 + \left(u_i - (1-\lambda)c_i^0\right)^3, \qquad i = 1,\ldots,m, \end{aligned}$$

and then

$$\rho(x^0,b^0,c^0,\lambda,x,u) = \begin{pmatrix} \lambda\left[\left(\nabla f(x)\right)^t + A^t u\right] + (1-\lambda)(x-x^0) \\ K(\lambda,x,u,b^0,c^0) \end{pmatrix}. \tag{5.11}$$

As always for probability-one homotopies, technically $0 \leqq \lambda < 1$. Unfortunately this K results in ρ in (5.11) not being transversal to zero, due to inherent cancellation in the structure of K in (5.10). Something a bit more complicated is required, such as

(5.12)
$$\begin{aligned} K_i(\lambda,x,u,b^0,c^0) = &-\left|(1-\lambda)b_i^0 + b_i - A_{i\cdot}x - u_i\right|^3 + \left((1-\lambda)b_i^0 + b_i - A_{i\cdot}x\right)^3 \\ &+ u_i^3 - (1-\lambda)c_i^0, \qquad i = 1,\ldots,m. \end{aligned}$$

The complication is that given $c^0 > 0$ and $(1-\lambda)b^0+b-Ax^0 > 0$, some work is required to find the starting point at $\lambda = 0$: the value u^0 for u such that $K(0, x^0, u^0, b^0, c^0) = 0$. However, it is easily verified that K_i is a strictly monotone increasing function of u_i, and thus u^0 can always be uniquely determined.

Let $a = (x^0, b^0, c^0)$, and define $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$, using (5.11) and (5.12).

THEOREM 5.1. *Let $f : E^n \to E$ be a C^3 convex function, let $A \in E^{m\times n}$, $b \in E^m$, and assume that $S_1 = \{x \mid Ax - b \leqq 0\}$ is nonempty and bounded. Let $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$ be defined from* (5.11) *and* (5.12). *Then for almost all $x^0 \in E^n$, almost all $b^0 \in E^m$ such that $b^0 > 0$ and $Ax^0 - b - b^0 < 0$, and almost all $c^0 \in E^m$ with $c^0 > 0$, there exists a zero curve γ of $\rho_a(\lambda, x, u)$ emanating from $(0, x^0, u^0)$, along which the Jacobian matrix $D\rho_a(\lambda, x, u)$ has* rank $n+m$, *and reaching a point $(1, \bar{x}, \bar{u})$, where $\bar{x}$ solves* $\min_{x\in S_1} f(x)$. *If* rank $D\rho_a(1, \bar{x}, \bar{u}) = n + m$, *then γ has finite arc length.*

Proof. Several facts need to be verified first.

(i) S_λ is nonempty and bounded for $0 \leqq \lambda \leqq 1$. This follows from the assumption that $S_1 \neq \emptyset$, (5.9), and the fact that boundedness is unrelated to the constant term $b + (1 - \lambda)b^0$: S_λ is bounded if and only if $\{x \mid Ax \leqq 0\} = \{0\}$. Furthermore, observe that int $S_\lambda \neq \emptyset$ for $0 \leqq \lambda < 1$.

(ii) ρ defined from (5.11) and (5.12) is transversal to zero, for $0 \leqq \lambda < 1$, and $c^0 > 0$. It is easily verified that full rank $(n + m)$ comes from the D_{x^0} and D_{c^0} columns. This is a good illustration of the fact that the dimension of the probability-one homotopy parameter vector $a = (x^0, b^0, c^0)$ need not equal the dimension of the homotopy map ρ, and of how this flexibility can be used to advantage.

(iii) There is a unique point (x^0, u^0) such that $\rho_a(0, x, u) = 0$. For $\lambda = 0$, clearly $x = x^0$ from (5.11). Now given x^0, b^0 such that $b^0 + b - Ax^0 > 0$, and $c^0 > 0$, it can be verified that $K = 0$ from (5.12) has a unique solution $u^0 > 0$.

By Lemma 2.1, $\rho_a(\lambda, x, u)$ is also transversal to zero for almost all $a = (x^0, b^0, c^0) \in E^n \times E^m \times (0, \infty)^m$. The statement of the theorem restricts b^0 to depend on x^0, but this is immaterial to the transversality of ρ_a, since the full rank of $D\rho$ does not depend on D_{b^0}. The set of all (x^0, b^0, c^0) described in the theorem is open, and ρ_a is transversal to zero for almost all points a in this set.

From (iii), there is exactly one solution (x^0, u^0) to $\rho_a = 0$ at $\lambda = 0$. Therefore Lemma 2.2 applies, and the existence (for almost all the prescribed points (x^0, b^0, c^0)) of a zero curve γ and the full rank of $D\rho_a$ along γ follow. γ emanates from the point $(0, x^0, u^0)$, and either reaches a solution point $(1, \bar{x}, \bar{u})$ or wanders off to infinity. By Lemma 2.3, it suffices to prove that γ is bounded. The finite arc length statement about γ also follows from Lemma 2.3.

(iv) Consider an arbitrary point (λ, x, u) on γ for $0 < \lambda < 1$. A careful examination of the signs of the terms in K_i in (5.12) reveals that $K_i < 0$ if $u_i < 0$ or $(1 - \lambda)b_i^0 + b_i - A_{i.}x < 0$. Therefore everywhere along γ, $u \geqq 0$ and $x \in S_\lambda \subset S_0$ is bounded from (i) and (5.9). Furthermore, $u_i > 0$ and $(1 - \lambda)b_i^0 + b_i - A_{i.}x > 0$ along γ.

Suppose that γ is not bounded, and let $(\lambda^{(k)}, x^{(k)}, u^{(k)}) \to \infty$ be a sequence of points on γ. Since $[0, 1] \times S_0$ is compact, $\left\{(\lambda^{(k)}, x^{(k)})\right\}_{k=1}^{\infty}$ has a convergent subsequence $\left(\lambda^{(k_i)}, x^{(k_i)}\right) \to (\hat{\lambda}, \hat{x})$. Now from (5.11), this means that $u^{(k_i)} \geqq 0$, $\left\|u^{(k_i)}\right\| \to \infty$, and

$$\hat{\lambda}\Big[\big(\nabla f(\hat{x})\big)^t + A^t u^{(k_i)}\Big] + (1 - \hat{\lambda})(\hat{x} - x^0) \to 0, \tag{5.13}$$

$$K(\hat{\lambda}, \hat{x}, u^{(k_i)}, b^0, c^0) \to 0. \tag{5.14}$$

If $\hat{\lambda} = 1$, then $\hat{x}$ is a solution to (5.1), and $\bar{u}$ corresponding to $\bar{x} = \hat{x}$ can be constructed from $u^{(k_i)}$. In a degenerate case, γ converges to a (possibly unbounded) manifold of solution points $(1, \bar{x}, \bar{u})$. So now consider $0 \leqq \hat{\lambda} < 1$ and two cases.

Case 1. For some j, $(1 - \hat{\lambda})b_j^0 + b_j - A_{j\cdot}\hat{x} > 0$ and $\limsup_{k_i \to \infty} u_j^{(k_i)} = \infty$. As observed earlier, K_j is a strictly monotone increasing function of u_j. Therefore $\|u^{(k_i)}\| \to \infty$ implies $\|K(\hat{\lambda}, \hat{x}, u^{(k_i)}, b^0, c^0)\|_\infty$ is increasing, which contradicts (5.14). Therefore γ is bounded and the theorem follows.

Case 2. $(1 - \hat{\lambda})b_j^0 + b_j - A_{j\cdot}\hat{x} = 0$ for every j with $\limsup_{k_i \to \infty} u_j^{(k_i)} = \infty$; denote this set of indices by J. Suppose first that $\hat{\lambda} = 0$. Then from (5.13), $A^t\lambda^{(k_i)}u^{(k_i)} \to x^0 - \hat{x}$. A subsequence argument yields a vector w such that $\big(A_{J\cdot}\big)^t w = x^0 - \hat{x}$, $w \geqq 0$. Combining the relations $Ax^0 - b - b^0 < 0$ and $\big(b^0 + b - A\hat{x}\big)_J = 0$ gives $A_{J\cdot}(x^0 - \hat{x}) < 0$. Now all these relations result in

$$0 \geqq \big(x^0 - \hat{x}\big)^t \big(A_{J\cdot}\big)^t w = \big(x^0 - \hat{x}\big)^t \big(x^0 - \hat{x}\big) > 0,$$

a contradiction. Therefore $\hat{\lambda} \neq 0$. As observed in item (i), $\{x \mid Ax \leqq 0\} = \{0\}$, which is equivalent to the positive cone $\mathcal{C}(A^t) = \{A^t y \mid y \geqq 0\} = E^n$. Therefore there exists w such that

$$A^t w = -\big(\nabla f(\hat{x})\big)^t - (1 - \hat{\lambda})(\hat{x} - x^0)/\hat{\lambda}, \qquad w \geqq 0.$$

Writing $u^{(k_i)} = w + v^{(k_i)}$ then gives $A^t v^{(k_i)} \to 0$ and $\big\|v^{(k_i)}\big\| \to \infty$. Recall that $u^{(k_i)} = w + v^{(k_i)} \geqq 0$, which means that any negative components of $v^{(k_i)}$ must be bounded by $\|w\|_\infty$, and therefore negative components of $v^{(k_i)} / \big\|v^{(k_i)}\big\|_\infty$ converge to zero as $\big\|v^{(k_i)}\big\| \to \infty$. The bounded sequence $\big\{v^{(k_i)} / \big\|v^{(k_i)}\big\|_\infty\big\}_{i=1}^\infty$ has a subsequence converging to some point $v \in E^m$ with $\|v\|_\infty = 1$, and $v \geqq 0$ by the preceding remark. Finally, $A^t v^{(k_i)} \to 0$ and $\big\|v^{(k_i)}\big\| \to \infty$ imply $A^t\big(v^{(k_i)} / \big\|v^{(k_i)}\big\|_\infty\big) \to 0$ yielding $A^t v = 0$, $v \geqq 0$, $\|v\|_\infty = 1$, or $\big(A_{J\cdot}\big)^t v_J = 0$, $v_J \geq 0$. By Gordan's Theorem of the Alternative [11], $A_{J\cdot} z > 0$ has no solution. However, since $\operatorname{int} S_{\hat{\lambda}} \neq \emptyset$, there is an interior feasible point x so that combining the relations $Ax - b - (1 - \hat{\lambda})b^0 < 0$ and $\big((1 - \hat{\lambda})b^0 + b - A\hat{x}\big)_J = 0$ yields $A_{J\cdot}(x - \hat{x}) < 0$. This contradiction proves that γ is bounded, and the theorem follows. □

COROLLARY 5.1. *Suppose that the zero curve γ of $\rho_a(\lambda, x, u)$ defined from* (5.11)–(5.12) *has the property that $\|x\|$ is bounded but $\|u\| \to \infty$ along γ, for $0 \leqq \lambda < 1$. Then there exists $v \in E^m$ such that $A^t v = 0$, $v \geq 0$, and every index i for which $v_i \neq 0$ has $|u_i| \to \infty$.*

It is instructive to consider why the proof of Theorem 5.1 worked, and how it could have gone wrong. After all, homotopy methods don't always work, and the conclusion of Corollary 5.1 is a plausible situation. The structure of ρ in (5.11)–(5.12) is important for transversality (Lemma 2.2) and the fact that γ (for this particular ρ) cannot return to $\lambda = 0$. Yet many other choices for ρ could possess these properties equally well. The boundedness of x along γ is especially opportune, being a direct consequence of the relaxation (5.7) and the "interior" map (5.8), which forces points (λ, x, u) along γ to be strictly feasible (interior) for *different* constraints (from the original ones). Effectively λ and x are under control (not so if S_λ were unbounded), and $\|u\| \to \infty$ is the only potential problem. Controlling $\|u\|$ is delicate, and was achieved here by having $\operatorname{int} S_\lambda \neq \emptyset$ for $0 \leqq \lambda < 1$ (trivially true for the $g(x)$ in (5.1)

but not so trivial in more general contexts). $\|u\|$ was controlled in earlier sections by a global monotonicity property, a much stronger condition than S_λ bounded with nonempty interior for $0 \leqq \lambda < 1$.

6. General nonlinear convex optimization. The optimization problem (5.1) can be generalized in several different directions. If the convexity assumption is dropped for either f or g, then (5.2)–(5.5) become only necessary conditions. Certainly many optimization algorithms are based on necessary optimality conditions, in which case only convergence to a stationary point is guaranteed. This particular direction of generalization will be pursued in a later section. Here in this section the intent is to preserve the optimality conditions being both necessary and sufficient, and thus $g(x)$ in (5.1) will be generalized.

Let $f : E^n \to E$ and $g : E^n \to E^m$ be C^3 convex functions, and assume that g satisfies the Arrow–Hurwicz–Uzawa constraint qualification at every solution of

$$\min f(x) \qquad \text{subject to} \qquad g(x) \leqq 0. \tag{6.1}$$

Under these assumptions (6.1) is equivalent to the Kuhn–Tucker problem

$$\big(\nabla f(x)\big)^t + \big(\nabla g(x)\big)^t u = 0, \tag{6.2}$$

$$g(x) \leqq 0, \tag{6.3}$$

$$u \geqq 0, \tag{6.4}$$

$$u^t g(x) = 0. \tag{6.5}$$

Given the discussion in the last section, it seems reasonable to try the direct generalization of (5.11) for the homotopy map

$$\rho(x^0, b^0, c^0, \lambda, x, u) = \begin{pmatrix} \lambda\Big[\big(\nabla f(x)\big)^t + \big(\nabla g(x)\big)^t u\Big] + (1-\lambda)(x - x^0) \\ K(\lambda, x, u, b^0, c^0) \end{pmatrix}, \tag{6.6}$$

where

$$\begin{aligned} K_i(\lambda, x, u, b^0, c^0) = &-\big|(1-\lambda)b_i^0 - g_i(x) - u_i\big|^3 + \big((1-\lambda)b_i^0 - g_i(x)\big)^3 \\ &+ u_i^3 - (1-\lambda)c_i^0, \qquad i = 1, \ldots, m, \end{aligned} \tag{6.7}$$

is the direct generalization of (5.12). The question is how changing from linear constraints $Ax - b \leqq 0$ to nonlinear convex constraints $g(x) \leqq 0$ affects the proof of Theorem 5.1. The crux of the proof seems to be the sets

$$S_\lambda = \big\{x \in E^n \mid g(x) - (1-\lambda)b^0 \leqq 0\big\}, \tag{6.8}$$

which need to satisfy (5.9), int $S_\lambda \neq \emptyset$ for $0 \leqq \lambda < 1$, and to be bounded for $0 \leqq \lambda \leqq 1$. Given some arbitrary initial guess $x^0 \in E^n$, choose $b^0 \in E^m$, $b^0 > 0$, such that $g(x^0) - b^0 < 0$. As before, choose $c^0 \in E^m$ such that $c^0 > 0$. Assuming that the feasible set $S_1 = \{x \in E^n \mid g(x) \leqq 0\}$ is nonempty and bounded is not a severe restriction, since for any practical problem variable bounds can always be added. The boundedness of the sets S_λ for $0 \leqq \lambda \leqq 1$ follows from Corollary 8.3.3, Theorem 8.4, and Theorem 8.7 of [17]. For completeness a short direct proof follows.

LEMMA 6.1. *Let $g : E^n \to E^m$ be a C^3 convex function, and let $x^0 \in E^n$, $b^0 \in E^m$, $\delta \in E$ be such that $b^0 \geqq \delta e > 0$ and $g(x^0) - b^0 < 0$. Define*

$$S_\lambda = \big\{x \in E^n \mid g(x) - (1-\lambda)b^0 \leqq 0\big\}.$$

If S_1 is nonempty and bounded, then S_λ is nonempty and bounded for $0 \leqq \lambda \leqq 1$, and int $S_\lambda \neq \emptyset$ *for* $0 \leqq \lambda < 1$.

Proof. Since $S_1 \neq \emptyset$,

$$S_0 \supset S_{\lambda_1} \supset S_{\lambda_2} \supset S_1 \neq \emptyset \quad \text{for } 0 \leqq \lambda_1 < \lambda_2 \leqq 1.$$

Suppose that S_{λ_1} is unbounded while S_{λ_2} is bounded, say, $S_{\lambda_2} \subset B(r/2) = \{x \in E^n \mid \|x\| < r/2\}$. Pick any point $\tilde{x} \in S_{\lambda_2}$ (hence $\|\tilde{x}\| < r/2$). Now there exists a sequence $y^{(k)} \in S_{\lambda_1}$ with $r < \left\|y^{(k)}\right\| \to \infty$. Reduce to a convergent subsequence $y^{(k_i)} / \left\|y^{(k_i)}\right\| \to y$, and for each subsequence index k choose $0 < t_k < 1$ such that

$$\left\|(1-t_k)\tilde{x} + t_k y^{(k)}\right\| = r. \tag{6.9}$$

Now by the convexity of g,

$$\begin{aligned} g\left((1-t_k)\tilde{x} + t_k y^{(k)}\right) &\leqq (1-t_k)g(\tilde{x}) + t_k g\left(y^{(k)}\right) \\ &\leqq (1-t_k)(1-\lambda_2)b^0 + t_k(1-\lambda_1)b^0. \end{aligned}$$

Taking the limit as $k \to \infty$ ($\left\|y^{(k)}\right\| \to \infty$ and (6.9) give $t_k \to 0$) yields

$$g(\tilde{x} + \alpha y) \leqq (1-\lambda_2)b^0 \tag{6.10}$$

for $\|\tilde{x} + \alpha y\| = r$. Now (6.10) $\Longrightarrow \tilde{x} + \alpha y \in S_{\lambda_2} \Longrightarrow r = \|\tilde{x} + \alpha y\| < r/2$, a contradiction. Therefore S_λ must be nonempty and bounded for all $0 \leqq \lambda \leqq 1$. The statement about int S_λ follows easily by continuity. ∎

Let $a = (x^0, b^0, c^0)$, and define $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$, according to (6.6) and (6.7). As before, u^0 is uniquely defined by $K(0, x^0, u^0, b^0, c^0) = 0$.

THEOREM 6.1. *Let $f : E^n \to E$ and $g : E^n \to E^m$ be C^3 convex functions, let g satisfy the Arrow–Hurwicz–Uzawa constraint qualification at every solution of* (6.1)*, and assume that $S_1 = \{x \in E^n \mid g(x) \leqq 0\}$ is nonempty and bounded. Let $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$ be defined from* (6.6) *and* (6.7)*. Then for almost all $x^0 \in E^n$, almost all $b^0 \in E^m$ such that $b^0 > 0$ and $g(x^0) - b^0 < 0$, and almost all $c^0 \in E^m$ with $c^0 > 0$, there exists a zero curve γ of $\rho_a(\lambda, x, u)$ emanating from $(0, x^0, u^0)$, along which the Jacobian matrix $D\rho_a(\lambda, x, u)$ has* rank $n + m$*, reaching a point $(1, \bar{x}, \bar{u})$, where $\bar{x}$ solves* $\min_{x \in S_1} f(x)$*. If* rank $D\rho_a(1, \bar{x}, \bar{u}) = n + m$*, then γ has finite arc length.*

Proof. By the convexity and constraint qualification assumptions, (6.1) is equivalent to (6.2)–(6.5), which are equivalent to $\rho(x^0, b^0, c^0, 1, x, u) = 0$. A careful examination of the proof of Theorem 5.1 reveals that it is valid if (a) $Ax - b$ is replaced with $g(x)$, (b) ρ and K from (5.11)–(5.12) are replaced with ρ and K from (6.6)–(6.7), (c) the sets S_λ from (6.8) are nonempty and bounded for $0 \leqq \lambda \leqq 1$ and have nonempty interiors for $0 \leqq \lambda < 1$, and (d) the conclusion of Corollary 5.1 also leads to a contradiction in the present more general situation. Consider each item in turn.

(a) Replacing the function $Ax - b$ with the function $g(x)$ affects nothing in the proof of Theorem 5.1. The appearance of $\nabla g(x)$ rather than (constant) A in ρ does have some effect. The argument ruling out the possibility $\hat{\lambda} = 0$ becomes $\nabla g_J(\hat{x})\left(x^0 - \hat{x}\right) \leqq g_J(x^0) - g_J(\hat{x}) < 0$, using the convexity of g. Arguments involving (5.13) and $\mathcal{C}(A^t)$ are valid with $\nabla g(\hat{x})$ replacing A. The final contradiction in Case 2 is addressed below in item (d).

(b) The transversality and other fundamental properties of ρ and K from (6.6)–(6.7) are easily verified.

(c) Lemma 6.1 provides these crucial facts about the sets S_λ.

(d) The question here is, given $\big(\nabla g_J(\hat{x})\big)^t v_J = 0$, $v_J \geq 0$, and int $S_{\hat{\lambda}} \neq \emptyset$, does the same contradiction ensue by finding a vector z such that $\big(\nabla g_J(\hat{x})\big)z < 0$? The answer is yes since $x \in$ int $S_{\hat{\lambda}}$ gives $g(x) - (1-\hat{\lambda})b^0 < 0$, and then subtracting $\big(g(\hat{x}) - (1-\hat{\lambda})b^0\big)_J = 0$ gives

$$\nabla g_J(\hat{x})\big(x - \hat{x}\big) \leqq \big(g(x) - g(\hat{x})\big)_J < 0. \qquad \square \tag{6.11}$$

A result similar to Corollary 5.1 could be derived, but the conclusion—$\big(\nabla g(\hat{x})\big)^t v = 0$, $v \geq 0$ has a solution—is not interesting since the point $\hat{x}$ has no special significance.

The postmortem comments on the proof of Theorem 5.1 apply also to the proof of Theorem 6.1, with the latter being technically (but not conceptually) more difficult. Properties of the sets S_λ required proof, and γ again could not return to $\lambda = 0$. Convexity of g was crucial in converting constraint values into a gradient inequality as in (6.11), both for obtaining $\hat{\lambda} \neq 0$ and for the final contradiction (6.11). Neither quasiconvexity nor pseudoconvexity suffice for g.

7. Nonconvex programs. The convergence theory in the preceding sections might, at first glance, seem trivial and contrived (with the assumptions dictated more by the exigencies of the proof rather than by practical applications), and not to address the homotopy maps actually used on practical engineering problem. Indeed, homotopy maps like (4.5), (5.11), and (6.6), although they work, are rarely used in practice. There are two significant questions to be answered: (1) How important is convexity, which has figured prominently in the discussion so far? (2) How important is it that homotopy maps as in (4.5), (5.11), or (6.6) be used? To both questions, the answer turns out to be: not very!

Convexity simplifies proofs, but is really only needed to make the Kuhn–Tucker conditions sufficient for optimality. Without convexity, convergence only to a stationary point can be guaranteed. The proofs in sections 3 and 4 used convexity, but only because those results were for the canonical map $\lambda F(x) + (1-\lambda)(x-a)$, and were done using a nonlinear complementarity result that depended on convexity (pseudoconvexity). Note, for instance, that the homotopy map (6.6) is not of the canonical form $\lambda F(x) + (1-\lambda)(x-a)$, which is the map used for Brouwer fixed point problems $x = f(x)$ (where $F(x) = x - f(x)$) [5]. This particular (canonical) map, which unfortunately is often thought of as "the" homotopy map, is only appropriate when F comes from a fixed point map or has some sort of global monotonicity property like $x\,F(x) \geqq 0$ for all $\|x\| \geqq r$ for some sufficiently large $r > 0$. Convexity (pseudoconvexity) is sufficient, but not necessary, for such global monotonicity, and hence is a natural assumption when using the map $\lambda F(x)+(1-\lambda)(x-a)$. The theory in sections 3 and 4 could be generalized to assume something like "$f(x)$ acts like a pseudoconvex function for $\|x\| \geqq r$ sufficiently large," but it hardly seems worth the trouble, since (as will be shown) homotopy maps like (6.6) obviate the need for convexity (when abandoning *sufficient* conditions for optimality!). Sections 5 and 6 used convexity to derive properties like γ cannot return to $\lambda = 0$, $\nabla g_J(\hat{x})z < 0$ has a solution z, and the boundedness of the sets S_λ in (6.8) for $0 \leqq \lambda \leqq 1$. Convexity is overkill, though, and these much weaker properties can be explicitly assumed.

Let $f : E^n \to E$ and $g : E^n \to E^m$ be C^3 functions, and assume that g satisfies

the Arrow–Hurwicz–Uzawa constraint qualification at every local solution of

$$\min f(x) \qquad \text{subject to} \qquad g(x) \leqq 0. \tag{7.1}$$

If $\bar{x}$ solves (7.1) locally, then there exists $\bar{u} \in E^m$ such that $(\bar{x}, \bar{u})$ solves the Kuhn–Tucker problem

$$\big(\nabla f(x)\big)^t + \big(\nabla g(x)\big)^t u = 0, \tag{7.2}$$

$$g(x) \leqq 0, \tag{7.3}$$

$$u \geqq 0, \tag{7.4}$$

$$u^t g(x) = 0. \tag{7.5}$$

Let $F : E^n \times [0,1] \to E$ and $G : E^n \times [0,1] \to E^m$ be C^3 functions such that

$$F(x,1) = f(x), \qquad G(x,1) = g(x), \tag{7.6}$$

and the optimization problem

$$\min F(x,0) \qquad \text{subject to} \qquad G(x,0) \leqq 0 \tag{7.7}$$

has an easily obtained (local) solution x^0. In practice, $F(x,\lambda)$, $G(x,\lambda)$ represent a family of optimization problems

$$\min F(x,\lambda) \qquad \text{subject to} \qquad G(x,\lambda) \leqq 0, \tag{7.8}$$

where λ is embedded deeply and nonlinearly in the objective function $F(x,\lambda)$ and constraints $G(x,\lambda)$. This embedding often embodies considerable physical insight into the problem (7.1), and (7.7) is a version of (7.1) with simplified physics and/or geometry. A good choice for (7.8) may take years to develop, and generally requires considerable problem-specific knowledge and the intimate involvement of an engineer or scientist. The payoff will be a robust, globally convergent algorithm that is more efficient than applying an "off-the-shelf" algorithm, and that avoids spurious solutions (e.g., unstable equilibria in mechanics or unstable circuit operating points can be expressly avoided).

One could naively solve (7.8) with continuation varying λ from 0 to 1, but this is precisely the point at which the probability-one theory can make a significant improvement over simple continuation in λ (and also over arc length continuation). A probability-one homotopy for (7.8) guarantees the existence of a zero curve γ with good numerical properties, the importance of which for practical computation cannot be overstated. The homotopy map (6.6) is generalized to

$$\rho(x^0, b^0, c^0, \lambda, x, u) = \begin{pmatrix} \lambda\Big[\big(\nabla_x F(x,\lambda)\big)^t + \big(\nabla_x G(x,\lambda)\big)^t u\Big] + (1-\lambda)(x - x^0) \\ K(\lambda, x, u, b^0, c^0) \end{pmatrix}, \tag{7.9}$$

where

$$\begin{aligned} K_i(\lambda, x, u, b^0, c^0) = &-\big|(1-\lambda)b_i^0 - G_i(x,\lambda) - u_i\big|^3 + \big((1-\lambda)b_i^0 - G_i(x,\lambda)\big)^3 \\ &+ u_i^3 - (1-\lambda)c_i^0, \qquad i = 1, \ldots, m, \end{aligned} \tag{7.10}$$

is the direct generalization of (6.7). The map (7.9), or some minor variation thereof, is what is typically used in practice, and has been extremely successful on industrial optimization problems.

The key observation in the proofs in sections 5 and 6 is that *what matters most is not the structure of the homotopy map ρ, but the nature of the sets S_λ.* (Of course, ρ still has to satisfy the hypotheses of Lemma 2.3, and some technical conditions along γ are required.) The general convergence theory is now developed.

Given some arbitrary initial guess $x^0 \in E^n$, choose $b^0 \in E^m$ such that $b^0 > 0$ and $G(x^0, 0) - b^0 < 0$. Choose $c^0 \in E^m$ such that $c^0 > 0$. Consider the sets

$$S_\lambda = \{x \in E^n \mid G(x,\lambda) - (1-\lambda)b^0 \leqq 0\}, \qquad 0 \leqq \lambda \leqq 1. \tag{7.11}$$

Note that $S_\lambda \neq \emptyset$ for small λ since $x^0 \in \operatorname{int} S_0$. However, since the constraints $G(x,\lambda)$ now can change with λ, (5.9) need not hold; i.e., the sets S_λ do not necessarily form a chain $S_0 \supset S_{\lambda_1} \supset S_{\lambda_2}$ for $0 < \lambda_1 < \lambda_2$. The question is, exactly what properties must S_λ have in order for the proofs of Theorems 5.1 and 6.1 to extend to the general nonconvex problem (7.1)? It is not necessary for the sets S_λ to form a chain as in (5.9) or even to satisfy $\bigcap_{0\leqq\lambda\leqq1} S_\lambda \neq \emptyset$. Certainly each S_λ must be nonempty; otherwise K from (7.10) cannot possibly be zero: $S_\lambda = \emptyset$ implies for each $x \in E^n$ there is an index i such that $(1-\lambda)b_i^0 - G_i(x,\lambda) < 0$, which means for all x some $K_i(\lambda, x, u, b^0, c^0) < 0$ for all u, and thus $\rho_a(\lambda, x, u) \neq 0$ for any x, u.

A point (λ, x, u) on the zero curve γ of $\rho_a(\lambda, x, u)$ must have $x \in S_\lambda$ and $u \geqq 0$ (otherwise $K \neq 0$), but S_λ bounded for $0 \leqq \lambda \leqq 1$ (the conclusion of Lemma 6.1) does not imply x along γ is bounded. The weakest assumption to keep x along γ bounded would then seem to be $\bigcup_{0\leqq\lambda\leqq1} S_\lambda$ is bounded. This condition is a bit subtle, though, as S_λ depends indirectly on x^0, and x^0 is supposed to be generic. Precisely, the requirement is as follows. Let $X^0 \subset E^n$, $B^0 \subset E^m$ be open nonempty sets such that for each point $x^0 \in X^0$, there exists $b^0 \in B^0$ such that $b^0 > 0$, $G(x^0, 0) - b^0 < 0$. Then $\bigcup_{0\leqq\lambda\leqq1} S_\lambda$ must be bounded for each $x^0 \in X^0$, $b^0 \in B^0$ satisfying $b^0 > 0$, $G(x^0, 0) - b^0 < 0$.

The above discussion is summarized in the hypotheses of the following theorem. Let $a = (x^0, b^0, c^0)$, and define $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$, according to (7.9) and (7.10). As always, u^0 is uniquely defined by $K(0, x^0, u^0, b^0, c^0) = 0$.

THEOREM 7.1. *Let $f : E^n \to E$ and $g : E^n \to E^m$ be C^3 functions, let g satisfy the Arrow–Hurwicz–Uzawa constraint qualification at every local solution of* (7.1)*, let $X^0 \subset E^n$ and $B^0 \subset \{b \in E^m \mid b > 0\}$ be open and nonempty, and for $b^0 \in B^0$ and $0 \leqq \lambda \leqq 1$ define*

$$S_\lambda(b^0) = \{x \in E^n \mid G(x,\lambda) - (1-\lambda)b^0 \leqq 0\}.$$

For each $x^0 \in X^0$ assume there exists $b^0 \in B^0$ such that $G(x^0, 0) - b^0 < 0$. For each $x^0 \in X^0$ and $b^0 \in B^0$ satisfying $G(x^0, 0) - b^0 < 0$, further assume that $S_\lambda(b^0)$ is nonempty for $0 \leqq \lambda \leqq 1$ and that $\bigcup_{0\leqq\lambda\leqq1} S_\lambda(b^0)$ is bounded. Let $\rho_a(\lambda, x, u) = \rho(x^0, b^0, c^0, \lambda, x, u)$ be defined from (7.9) *and* (7.10)*. Then for almost all $x^0 \in X^0$, almost all $b^0 \in B^0$ such that $G(x^0, 0) - b^0 < 0$, and almost all $c^0 \in E^m$ with $c^0 > 0$, there exists a zero curve γ of $\rho_a(\lambda, x, u)$ emanating from $(0, x^0, u^0)$, along which the Jacobian matrix $D\rho_a(\lambda, x, u)$ has* rank $n + m$*. If in addition there exists $\kappa > 0$ such that for any point (λ, x, u) on γ,*

$$\|(\lambda, x, u) - (0, x^0, u^0)\| > 1 \Longrightarrow \lambda \geqq \kappa,$$

and for any accumulation point $(\hat{\lambda}, \hat{x})$ *of* (λ, x) *along* γ

$$\left[\nabla_x G_J(\hat{x}, \hat{\lambda})\right] z > 0 \text{ has a solution } z,$$

where $J = \{j \mid G_j(\hat{x}, \hat{\lambda}) - (1-\hat{\lambda}) b_j^0 = 0\}$, *then* γ *reaches a point* $(1, \bar{x}, \bar{u})$, *where* $(\bar{x}, \bar{u})$ *solves the Kuhn–Tucker problem* (7.2)–(7.5). *If* rank $D\rho_a(1, \bar{x}, \bar{u}) = n + m$, *then* γ *has finite arc length.*

Proof. The homotopy map ρ defined from (7.9)–(7.10), similar to ρ and K from (5.11)–(5.12) and (6.6)–(6.7), satisfies the hypotheses of Lemma 2.2. Thus a homotopy zero curve γ exists and it only remains to show γ is bounded. If γ reaches a point $(1, \bar{x}, \bar{u})$, since $\rho_a(1, \bar{x}, \bar{u}) = 0$ is equivalent to the necessary optimality conditions (7.2)–(7.5), $(\bar{x}, \bar{u})$ will be a stationary point for the original optimization problem (7.1).

As before, by the nature of K and the boundedness of $\bigcup S_\lambda$, (λ, x) is bounded along γ. Suppose that γ is unbounded, and let $(\lambda^{(k)}, x^{(k)}, u^{(k)}) \to \infty$ be a sequence of points along γ. As before, there is a subsequence $\left(\lambda^{(k_i)}, x^{(k_i)}\right) \to (\hat{\lambda}, \hat{x})$, with $u^{(k_i)} \geqq 0$ and $\left\|u^{(k_i)}\right\| \to \infty$. By assumption, $\hat{\lambda} \geqq \kappa > 0$ and thus $\hat{\lambda} \neq 0$. The argument for the case $\hat{\lambda} = 1$ is identical to that in the proof of Theorem 5.1. Now consider only $0 < \hat{\lambda} < 1$. The argument for Case 1, where for some j,

$$(1-\hat{\lambda}) b_j^0 - G_j(\hat{x}, \hat{\lambda}) > 0 \qquad \text{and} \qquad \overline{\lim_{k_i \to \infty}} u_j^{(k_i)} = \infty,$$

is identical to that for Theorem 5.1.

Case 2, where for every j,

$$\overline{\lim_{k_i \to \infty}} u_j^{(k_i)} = \infty \Longrightarrow (1-\hat{\lambda}) b_j^0 - G_j(\hat{x}, \hat{\lambda}) = 0,$$

leads to the system

$$\left(\nabla_x G_J(\hat{x}, \hat{\lambda})\right)^t v_J = 0, \qquad v_J \geq 0,$$

having a solution v_J, where $J \subset \{j \mid G_j(\hat{x}, \hat{\lambda}) - (1-\hat{\lambda}) b_j^0 = 0\}$. By Gordan's Theorem of the Alternative,

$$\left(\nabla_x G_J(\hat{x}, \hat{\lambda})\right) z > 0$$

has no solution z. This contradicts the explicit hypothesis about $\nabla_x G_J$ along γ. Therefore γ is bounded, and the theorem follows. □

COROLLARY 7.1. *If the assumption in Theorem* 7.1 *about* $\lambda \geqq \kappa > 0$ *for points on* γ *far from* $(0, x^0, u^0)$ *is replaced by*

$$\text{rank } D_{(x,u)} \rho_a(\lambda, x, u) = n + m$$

along γ, *then the conclusions of Theorem* 7.1 *hold.*

Proof. The rank assumption implies that γ has no turning points, a much stronger assumption than simply $\lambda \geqq \kappa > 0$ eventually; i.e., as arc length s increases, γ does not asymptotically approach the hyperplane $\lambda = 0$. □

On many realistic engineering applications, γ does in fact have several turning points, and if the convergence theory could not accommodate turning points, it would not accurately reflect practice. Theorems 5.1, 6.1, and 7.1 have been presented as a

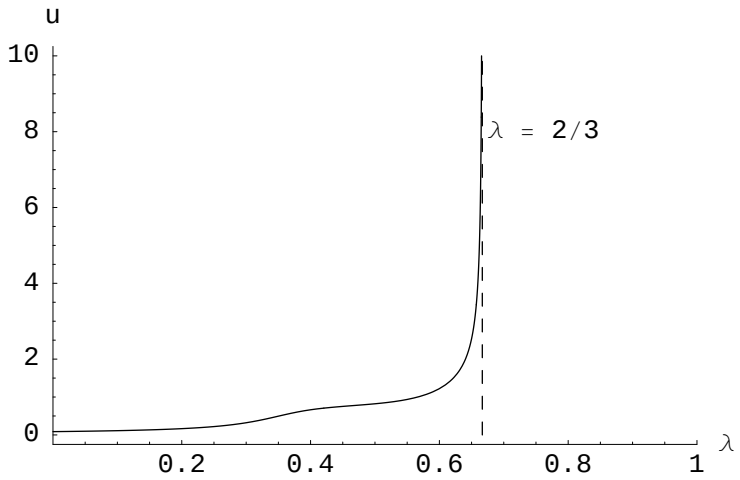


FIG. 2. *Example of homotopy zero curve when some S_λ is empty.*

series of generalizations with proof refinements, and so the postmortem comments on Theorems 5.1 and 6.1 in essence apply here also. Proving that γ is bounded amounts to controlling, in some fashion, each of λ, x, u along γ.

λ is controlled by preventing $\liminf_{s\to\infty} \lambda(s) = 0$. This frequently happens when the homotopy map is poorly chosen. For instance, $\lambda(s) \to 0$ as $u(s) \to \infty$ in Figure 1 for the homotopy map (5.6). The theory here shows that with the right homotopy map, $0 < \liminf_{s\to\infty} \lambda(s) = \hat{\lambda} < 1$ cannot happen except in rare degenerate cases involving the active constraint gradients $\nabla_x G_J(\hat{x}, \hat{\lambda})$.

x is controlled by the property that $x \in S_\lambda(b^0)$, and by assumption $\bigcup_{0\leqq\lambda\leqq 1} S_\lambda(b^0)$ is bounded. What happens if $S_\lambda(b^0) = \emptyset$ for some $0 < \lambda < 1$? For complicated problems, it is easy to unwittingly construct a family (7.8) for which some $S_\lambda(b^0)$ is empty. Consider the problem

$$\min_x F(x, \lambda) = x \qquad \text{subject to} \qquad G(x, \lambda) = x^2 - 1 + 2\lambda \leqq 0$$

and take $x^0 = 0$, $b^0 = 1$, $c^0 = 1$. $S_{2/3}(1) = \{0\}$, $S_\lambda(1) = \emptyset$ for $\lambda > 2/3$ (i.e., there is no solution at $\lambda = 1$) so something has to fail. Figure 2 shows what happens to γ for the map (7.9).

u is controlled by the property (in the convex case) or the assumption (in the nonconvex case) that $\big(\nabla_x G_J(\hat{x}, \hat{\lambda})\big) z > 0$ has a solution z, where J is related to active constraints at an accumulation point $(\hat{\lambda}, \hat{x})$ of $((\lambda, x)$ along$)$ γ. This condition can be interpreted as a "constraint qualification for homotopy maps." Since its failure to hold represents a degenerate situation, it can be achieved (in principle) by generically perturbing the map $G(x, \lambda)$.

As mentioned earlier, the map (7.9) closely resembles those used in practice, and thus Theorem 7.1 reflects practice. Generally, for each λ, the problem (7.8) is physically meaningful with S_λ being nonempty and bounded. The constraint qualification involving $\nabla_x G_J(x, \lambda)$ holds generically, and thus is not normally a concern. $\liminf_{s\to\infty} \lambda(s) > 0$ must be assumed, and this is the fly in the ointment. This condition is achieved by some sort of global monotonicity property (which often *does* hold for practical problems, related to energy considerations), the rank condition of

Corollary 7.1 (extremely hard to verify for a complicated problem), or by the clever construction of (7.8). There is no silver bullet!

Acknowledgments. The author is indebted to Gene Allgower, Kurt Georg, Alexander Morgan, and many mentors, colleagues, and engineers who have shared their problems and insights. Thanks are due also to a referee whose insightful observations helped strengthen the theorems and proofs.

REFERENCES

[1] J. C. Alexander, *The topological theory of an imbedding method*, in Continuation Methods, H. G. Wacker, ed., Academic Press, New York, 1978, pp. 37–68.

[2] J. C. Alexander, T.-Y. Li, and J. A. Yorke, *Piecewise smooth homotopies*, in Homotopy Methods and Global Convergence, B. C. Eaves, F. J. Gould, H.-O. Peitgen, and M. J. Todd, eds., Plenum, New York, 1983, pp. 1–14.

[3] E. L. Allgower and K. Georg, *Numerical Continuation Methods*, Springer-Verlag, Berlin, 1990.

[4] S. C. Billups, A. L. Speight, and L. T. Watson, *Nonmonotone path following methods for nonsmooth equations and complementarity problems*, in Applications and Algorithms of Complementarity, M. C. Ferris, O. L. Mangasarian, and J.-S. Pang, eds., Kluwer, Norwell, MA, to appear.

[5] S. N. Chow, J. Mallet-Paret, and J. A. Yorke, *Finding zeros of maps: Homotopy methods that are constructive with probability one*, Math. Comp., 32 (1978), pp. 887–899.

[6] B. C. Eaves, *Homotopies for computation of fixed points*, Math. Programming, 3 (1972), pp. 1–22.

[7] B. C. Eaves and R. Saigal, *Homotopies for computation of fixed points on unbounded regions*, Math. Programming, 3 (1972), pp. 225–237.

[8] Y. Ge, L. T. Watson, E. G. Collins, Jr., and D. S. Bernstein, *Probability-one homotopy algorithms for full and reduced order H^2/H^∞ controller synthesis*, Optimal Control Appl. Methods, 17 (1996), pp. 187–208.

[9] Z. Lin, Y. Li, and B. Yu, *A combined homotopy interior point method for general nonlinear programming problems*, Appl. Math. Comp., 80 (1996), pp. 209–224.

[10] Z. Lin, B. Yu, and G. Feng, *A combined homotopy interior point method for convex nonlinear programming*, Appl. Math. Comp., 84 (1997), pp. 193–211.

[11] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[12] O. L. Mangasarian, *Equivalence of the complementarity problem to a system of nonlinear equations*, SIAM J. Appl. Math., 31 (1976), pp. 89–92.

[13] A. P. Morgan, *Solving Polynomial Systems Using Continuation for Scientific and Engineering Problems*, Prentice-Hall, Englewood Cliffs, NJ, 1987.

[14] A. P. Morgan and A. J. Sommese, *A homotopy for solving general polynomial systems that respects m-homogeneous structures*, Appl. Math. Comput., 24 (1987), pp. 101–113.

[15] A. P. Morgan and A. J. Sommese, *Computing all solutions to polynomial systems using homotopy continuation*, Appl. Math. Comput., 24 (1987), pp. 115–138.

[16] A. B. Poore and Q. Al-Hassan, *The expanded Lagrangian system for constrained optimization problems*, SIAM J. Control Optim., 26 (1988), pp. 417–427.

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

[18] H. Sellami, *A homotopy continuation method for solving normal equations*, Math. Programming, 82 (1998), pp. 317–337.

[19] H. Sellami and S. M. Robinson, *Implementation of a continuation method for normal maps*, Math. Programming, 76 (1997), pp. 563–578.

[20] L. T. Watson, *A globally convergent algorithm for computing fixed points of C^2 maps*, Appl. Math. Comput., 5 (1979), pp. 297–311.

[21] L. T. Watson, *An algorithm that is globally convergent with probability one for a class of nonlinear two-point boundary value problems*, SIAM J. Numer. Anal., 16 (1979), pp. 394–401.

[22] L. T. Watson, *Solving the nonlinear complementarity problem by a homotopy method*, SIAM J. Control Optim., 17 (1979), pp. 36–46.

[23] L. T. Watson, *Solving finite difference approximations to nonlinear two-point boundary value problems by a homotopy method*, SIAM J. Sci. Stat. Comput., 1 (1980), pp. 467–480.

[24] L. T. Watson, *Computational experience with the Chow-Yorke algorithm*, Math. Programming, 19 (1980), pp. 92–101.
[25] L. T. Watson, *Numerical linear algebra aspects of globally convergent homotopy methods*, SIAM Rev., 28 (1986), pp. 529–545.
[26] L. T. Watson, *Globally convergent homotopy algorithms for nonlinear systems of equations*, Nonlinear Dynam., 1 (1990), pp. 143–191.
[27] L. T. Watson, *A survey of probability-one homotopy methods for engineering optimization*, Arabian J. Sci. Engrg., 16 (1991), pp. 297–323.
[28] L. T. Watson, S. C. Billups, and A. P. Morgan, *HOMPACK: A suite of codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software, 13 (1987), pp. 281–310.
[29] L. T. Watson, J. P. Bixler, and A. B. Poore, *Continuous homotopies for the linear complementarity problem*, SIAM J. Matrix Anal. Appl., 10 (1989), pp. 259–277.
[30] L. T. Watson and R. T. Haftka, *Modern homotopy methods in optimization*, Comput. Methods Appl. Mech. Engrg., 74 (1989), pp. 289–305.
[31] L. T. Watson and M. R. Scott, *Solving spline collocation approximations to nonlinear two-point boundary value problems by a homotopy method*, Appl. Math. Comput., 24 (1987), pp. 333–357.
[32] L. T. Watson and L. R. Scott, *Solving Galerkin approximations to nonlinear two-point boundary value problems by a globally convergent homotopy method*, SIAM J. Sci. Statist. Comput., 8 (1987), pp. 768–789.
[33] L. T. Watson, M. Sosonkina, R. C. Melville, A. P. Morgan, and H. F. Walker, *Algorithm* 777*: HOMPACK*90*: A suite of Fortran* 90 *codes for globally convergent homotopy algorithms*, ACM Trans. Math. Software, 23 (1997), pp. 514–549.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 781–795

CHAOTIC BEHAVIOR OF THE AFFINE SCALING ALGORITHM FOR LINEAR PROGRAMMING*

ILEANA CASTILLO† AND EARL R. BARNES‡

Abstract. The affine scaling algorithm for linear programming involves a step-size parameter t that must be chosen in the interval $(0, 1)$. It is known that the algorithm converges to an optimal solution for values of $t \leq 2/3$. In this paper we examine the behavior of the algorithm for values of $t > 2/3$. We show that for certain values of t in this range the algorithm can exhibit chaotic behavior.

Key words. affine scaling, step-size parameter, chaos

AMS subject classifications. 49M40, 65K05, 90C20, 90C25

PII. S1052623496314070

1. Introduction.

Shortly after Karmarkar proposed his interior point algorithm for linear programming in 1984 [8] several authors proposed a conceptually simpler interior point algorithm, which has become known as the affine scaling algorithm. See [1] and [15]. The standard linear programming problem requires a linear form to be minimized over a polyhedron $\mathbf{P}$. The affine scaling algorithm accomplishes this as follows. Let x^0 be an interior point of $\mathbf{P}$ and construct an ellipsoid E_0 centered at x^0 and contained completely inside $\mathbf{P}$. Let ξ_0 denote the point where the linear form achieves its minimum value over E_0. Let x^1 be a point obtained by moving from x^0 in the direction $\xi_0 - x^0$ a fraction t of the distance from x^0 to the boundary of $\mathbf{P}$. This step is now repeated using x^1 as the starting point. t remains fixed throughout the process. Our geometric intuition suggests that the sequence $\{x^k\}$ converges to the optimal solution of our linear programming problem for any value of t in the interval $(0, 1)$. However, this is not the case. A counterexample has been reported in [9]. But many questions about the convergence behavior of the algorithm remain open. It seems an anomaly that so simple an algorithm has resisted a complete analysis for so long. Actually, the analysis of the algorithm began long before Karmarkar's work [8]. The affine scaling algorithm was first proposed by the Soviet mathematician Dikin in 1967 [4]. Dikin did not offer a convergence proof until seven years later in [5]. His proof assumed the linear programming problem to be primal nondegenerate. In those days there was no sizable effort to find an alternative to the simplex algorithm, so Dikin's work went largely unnoticed, especially in the West. Then in 1984, Karmarkar proposed an interior point algorithm that outperformed the simplex algorithm on certain classes of problems. This led to the rapid discovery of several interior point linear programming algorithms, including the rediscovery of Dikin's algorithm. Karmarkar's algorithm is substantially more difficult to motivate, and to describe, than Dikin's. Yet it is easier to analyze. In fact, Karmarkar showed that his algorithm converges in polynomial time, and his proof makes no nondegeneracy

*Received by the editors December 23, 1996; accepted for publication (in revised form) November 5, 1999; published electronically December 7, 2000. This work was supported in part by NSF grant DDM-9014823. It was taken from the first author's Ph.D. thesis [2].
http://www.siam.org/journals/siopt/11-3/31407.html
†Department of Industrial Engineering, Instituto Tecnológico y de Estudios Superiores de Monterrey, Toluca, Mexico.
‡School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0205 (ebarnes@isye.gatech.edu).

assumptions. The convergence proofs for Dikin's algorithm that appeared in [1] and [15] assume the linear programming problem is both primal and dual nondegenerate. These proofs are therefore less general than the one Dikin had given more than 10 years earlier. We are now beginning to understand why the convergence question was so hard to settle. In some cases researchers were asking the wrong question. They were trying to prove convergence for all values of t in the interval $(0, 1)$, based on geometric intuition. Actually, the primal variables do converge to some limit, but, as we see in [9], it may not be the solution of the linear programming problem. The problem with our geometric interpretation of the affine scaling algorithm is that it ignores the dual variables, and these are the ones that exhibit unexpected behavior. In order to be more specific we require a mathematical description of the affine scaling algorithm. We assume we are given a linear programming problem in the following standard form:

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax = b \\ & x \geq 0, \end{array} \tag{1.1}$$

where c and x are n-dimensional column vectors, A is an $m \times n$ matrix of rank m, and b is a given m-dimensional column vector. We assume that this problem has a solution and that the polytope $\mathbf{P} = \{x | Ax = b, x \geq 0\}$ has a nonempty interior. We also assume that the linear function $c^T x$ is not constant on $\mathbf{P}$. This means that c does not lie in the row space of A.

The dual of (1.1) is

$$\begin{array}{ll} \max & b^T \lambda \\ \text{subject to} & A^T \lambda \leq c, \end{array} \tag{1.2}$$

where λ is an m-dimensional column vector.

The affine scaling algorithm chooses a step-size t satisfying $0 < t < 1$ and constructs primal and dual sequences, $\{x^k\}$ and $\{\lambda^k\}$, as follows.

- Start with $x^0 > 0$ satisfying $Ax^0 = b$.
- Given $x^k > 0$, $k \geq 0$, define $D_k = \operatorname{diag}(x_1^k, \ldots, x_n^k)$ and compute the following vectors:

$$\begin{aligned} \lambda^k &= (AD_k^2 A^T)^{-1} AD_k^2 c, \\ s^k &= c - A^T \lambda^k, \\ x^{k+1} &= x^k - t \frac{D_k^2 s^k}{\phi(D_k s^k)}, \end{aligned} \tag{1.3}$$

where $\phi(s)$, for any vector $s = (s_1, s_2, \ldots, s_n) \in \Re^n$, is defined by $\phi(s) = \max_i s_i$.

There is a version of this algorithm where the above definition of ϕ is replaced by the Euclidean noun of s, and we write $\phi(s) = \|s\|$. Since $\max s_i \leq \|s\|$, this version is known as the short-step affine scaling algorithm. The version described in (1.3) is known as the long-step version. The algorithm proposed by Dikin in 1967 was the short-step version.

As we have said, the affine scaling algorithm, while simple to describe, resisted analysis for a long time. But a lot of progress has been made recently. In 1989 Tsuchiya [12] proved, assuming only dual nondegeneracy, that for $0 < t \leq \frac{1}{8}$ the sequences $\{x^k\}$ and $\{\lambda^k\}$ generated by the short-step affine scaling algorithm converge to solutions of the respective problems (1.1) and (1.2). In 1990 Tsuchiya [11] showed

that this nondegeneracy assumption can be removed from the analysis. Then in 1991 Dikin [6] gave a convergence proof for the long-step method for $0 < t \leq \frac{1}{4}$ if $b = 0$ and $c > 0$ in (1.1). In 1992 Tsuchiya and Muramatsu [14] gave a considerable generalization of this result. They proved that for the long-step method, with step-size satisfying $0 < t \leq \frac{2}{3}$, the sequence x^k converges to an interior point of the optimal face of solutions of (1.1) and the sequence $\{\lambda^k\}$ converges to the analytic center of the face of optimal solutions of (1.2). Shortly after this Hall and Vanderbei [7] constructed an example where the sequence $\{\lambda^k\}$ generated by the long-step method fails to converge for any step-size $t > \frac{2}{3}$. By contrast, in 1997 Dikin and Roos [16] proved that for the short-step method both of the sequences $\{x^k\}$ and $\{\lambda^k\}$ converge to solutions of (1.1) and (1.2), respectively, for any step-size t satisfying $0 < t \leq 1$ if $b = 0$ and $c > 0$. This result was extended to a class of convex programming problems by Monteiro and Tsuchiya [17]. Their analysis shows that the result in [16] for homogeneous problems can be extended to the general problem (1.1).

2. An example. The following simple example demonstrates how complicated the behavior of the affine scaling algorithm can be:

$$\begin{array}{ll} \min & 10x_1 + 10x_2 + 5x_3 + x_4 - x_5 \\ \text{subject to} & x_1 + 2x_2 - 3x_3 - 2x_4 - x_5 = 0 \\ & -x_1 + 2x_2 - x_3 - x_4 - x_5 = 0 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0. \end{array} \tag{2.1}$$

The dual of this problem is to

$$\begin{array}{ll} \max & 0 \\ \text{subject to} & \lambda_1 - \lambda_2 \leq 10, \\ & 2\lambda_1 + 2\lambda_2 \leq 10, \\ & -3\lambda_1 - \lambda_2 \leq 5, \\ & -2\lambda_1 - \lambda_2 \leq 1, \\ & -\lambda_1 - \lambda_2 \leq -1. \end{array} \tag{2.2}$$

Thus the dual problem is just to find a feasible point for the inequalities (2.2).

The point $(1, 1)$ is feasible for the dual. Therefore, $x = 0$ is a solution for the primal problem (2.1). Clearly this solution is unique, for the conditions

$$\begin{aligned} 10x_1 + 10x_2 + 5x_3 + x_4 - x_5 &= 0, \\ -x_1 + 2x_2 - x_3 - x_4 - x_5 &= 0, \end{aligned}$$

$x_j \geq 0$ imply that $x_j = 0$, $j = 1, \ldots, 5$, as can be seen by eliminating x_5 from one of these equations. The difficulty of analyzing the convergence behavior of (1.3) applied to (2.1) becomes apparent if we run (1.3) for a few values of t. If we take $t = 0.5$ and $x^0 = (2, 4, 1, 2, 3)^t$, we obtain, after 31 iterations,

$$x^{31} = \begin{pmatrix} 45043 \\ 225241 \\ 48477 \\ 93132 \\ 27383 \end{pmatrix} \times 10^{-14}.$$

The dual sequence $\{\lambda^k\}$ converges much faster. The first six terms in this sequence are

$$\lambda^1 = \begin{pmatrix} 3.04073319755601 \\ 0.44806517311609 \end{pmatrix}, \qquad \lambda^2 = \begin{pmatrix} 3.04978042919414 \\ 0.53564835204215 \end{pmatrix},$$

$$\lambda^3 = \begin{pmatrix} 3.05123780355246 \\ 0.55152026110043 \end{pmatrix}, \qquad \lambda^4 = \begin{pmatrix} 3.05129969865096 \\ 0.55216405532858 \end{pmatrix},$$
$$\lambda^5 = \begin{pmatrix} 3.05129983620499 \\ 0.55216549196332 \end{pmatrix}, \qquad \lambda^6 = \begin{pmatrix} 3.05129983620574 \\ 0.55266549197123 \end{pmatrix}.$$

The last term is correct to 12 decimal places.

Suppose now that we take $t = 0.84$. Starting again with $x^0 = (2, 4, 1, 2, 3)^T$ we obtain after 22 iterations

$$x^{22} = \begin{pmatrix} 462 \\ 946 \\ 235 \\ 454 \\ 739 \end{pmatrix} \times 10^{-14}.$$

Thus after 22 iterations the sequence $\{x^k\}$ is very close to the solution of (2.1). On the other hand, the sequence $\{\lambda^k\}$ does not appear to be converging. For example, we have

$$\lambda^{17} = \begin{pmatrix} 3.0432890 \\ 0.4674003 \end{pmatrix}, \qquad \lambda^{18} = \begin{pmatrix} 3.0440522 \\ 0.4754970 \end{pmatrix},$$
$$\lambda^{19} = \begin{pmatrix} 3.0433001 \\ 0.4675194 \end{pmatrix}, \qquad \lambda^{20} = \begin{pmatrix} 3.0440708 \\ 0.4757432 \end{pmatrix},$$
$$\lambda^{21} = \begin{pmatrix} 3.0432944 \\ 0.4674515 \end{pmatrix}, \qquad \lambda^{22} = \begin{pmatrix} 3.0440598 \\ 0.4755515 \end{pmatrix}.$$

Later on we will do a finer analysis that will show that the subsequence $\{\lambda^1, \lambda^3, \ldots\}$ of odd indices is converging to the point

$$\xi = \begin{pmatrix} 3.0432968 \\ 0.46748284 \end{pmatrix} \tag{2.3}$$

and the subsequence $\{\lambda^2, \lambda^4, \ldots\}$ of even terms is converging to the limit

$$\eta = \begin{pmatrix} 3.0440618 \\ 0.4756198 \end{pmatrix}. \tag{2.4}$$

For $t = 0.89$ and $x^0 = (2, 4, 1, 2, 3)^T$ we have the following results:

$$x^{20} = \begin{pmatrix} 0.0896 \\ 0.1655 \\ 0.0442 \\ 0.09087 \\ 0.1064 \end{pmatrix} \times 10^{-10},$$

$$\lambda^{13} = \begin{pmatrix} 3.0318 \\ 0.3659 \end{pmatrix}, \quad \lambda^{14} = \begin{pmatrix} 2.9891 \\ 0.5019 \end{pmatrix}, \quad \lambda^{15} = \begin{pmatrix} 3.0314 \\ 0.3726 \end{pmatrix}, \quad \lambda^{16} = \begin{pmatrix} 2.9650 \\ 0.5469 \end{pmatrix},$$
$$\lambda^{17} = \begin{pmatrix} 3.0318 \\ 0.3659 \end{pmatrix}, \quad \lambda^{18} = \begin{pmatrix} 2.9891 \\ 0.5019 \end{pmatrix}, \quad \lambda^{19} = \begin{pmatrix} 3.0314 \\ 0.3725 \end{pmatrix}, \quad \lambda^{20} = \begin{pmatrix} 2.9651 \\ 0.5467 \end{pmatrix}.$$

(2.5)

From this it is clear that the subsequence $\{\lambda^1, \lambda^3, \ldots\}$ of odd indices is no longer converging but has split into two convergent subsequences

$$\{\lambda^1, \lambda^5, \lambda^9, \lambda^{13}, \ldots\} \quad \text{and} \quad \{\lambda^3, \lambda^7, \lambda^{11}, \lambda^{15}, \ldots\}.$$

Similarly, the subsequence $\{\lambda^2, \lambda^4, \ldots\}$ of terms with even indices is now diverging but has two convergent subsequences

$$\{\lambda^2, \lambda^6, \lambda^{10}, \lambda^{14}, \ldots\} \quad \text{and} \quad \{\lambda^4, \lambda^8, \lambda^{12}, \lambda^{16}, \ldots\}.$$

There is another way to describe the observations we have just made. For $t = 0.5$ the sequence $\{\lambda^k\}$ converges to the analytic center of the polytope (2.2) as required by the theorem in [14]. For $t = 0.84$ the sequence converges to a periodic orbit of period 2 on the points (2.3) and (2.4). For $t = 0.89$ the sequence $\{\lambda^k\}$ converges to a periodic orbit of period 4 as can be seen by examining the terms in (2.5). Thus it appears that the sequence $\{\lambda^k\}$ always converges to a periodic orbit for certain values of $t > 2/3$. The period of the orbit seems to double as t passes through certain values. As t approaches a certain critical value t_c the period doublings occur with increasing rapidity. Thus as $t \uparrow t_c$ the lengths of the periods approach ∞. For t slightly larger than t_c the sequence $\{\lambda_k\}$ seems to wander about randomly. It is helpful to think of it as converging to a periodic orbit with infinite period. Such an orbit of course never closes, and the sequence $\{\lambda_k\}$ therefore does not behave with any discernible regularity. We say it is in a state of chaos. This is precisely the kind of behavior that has been observed in certain dynamical systems of the form $z_{k+1} = F(t, z_k)$. See, for example, [3]. We will demonstrate this experimentally in section 3 below. The demonstration will require us to compute hundreds of terms of the sequence $\{\lambda^k\}$ due to the slow convergence of this sequence. This cannot be done directly, for we have seen that the sequence $\{x^k\}$ converges very rapidly to 0 as $k \to \infty$. This makes the computation of the inverse $(AD_k^2A^T)^{-1}$ impractical for values of k only slightly larger than 30. Fortunately we can compute very large numbers of terms of the sequence $\{\lambda^k\}$ by properly scaling the variable x^k in (1.3). We will explain how this can be done next. Incidentally, one reason the chaotic behavior we are going to show has probably not been observed before is that normally the emphasis is on the primal variables, and these converge after a relatively small number of iterations. But recently the dual variables generated by interior point methods have begun to be used in column generation procedures for solving very large linear programming problems. See [2] for examples of this. Some related uses of the analytic center are given in [18] and [19]. So it is becoming increasingly important to understand the convergence properties of these dual sequences generated by interior point algorithms.

To emphasize this point we give here a small example showing that the multipliers generated by the affine scaling algorithm are superior to those generated by the simplex algorithm for use in certain column generation procedures. In large problems the effect we will demostrate is much more dramatic.

Consider a cutting-stock problem where 11-ft boards are to be cut to satisfy the following requirements: two 8-ft boards, three 4-ft boards, five 3-ft boards, and three 2-ft boards. By a cutting pattern we mean a column vector $p = (\alpha, \beta, \gamma, \delta)^T$, where α, β, γ, and δ are, respectively, the number of 8-ft, 4-ft, 3-ft, and 2-ft boards contained in the pattern. Cutting patterns are in a 1-1 correspondence with nonnegative integers $\alpha, \beta, \gamma, \delta$ satisfying $8\alpha + 4\beta + 3\gamma + 2\delta \leq 11$. For example, $p_1 = (1, 0, 1, 0)^T$, $p_2 = (0, 2, 0, 1)^T$, $p_3 = (0, 1, 1, 1)^T$, and $p_4 = (1, 0, 0, 1)^T$ are cutting patterns. Suppose we

want to get the cuts we need from these patterns. Let x_j denote the number of times pattern p_j is used. We must then solve

$$\begin{array}{ll} \min & x_1 + x_2 + x_3 + x_4 \\ \text{subject to} & p_1x_1 + p_2x_2 + p_3x_3 + p_4x_4 \geq b, \\ & x_1, x_2, x_3, x_4 \geq 0, \end{array} \tag{2.6}$$

where $b = (2, 3, 5, 3)^t$ is the requirements vector.

The solution of this problem is

$$x_1 = 2, \quad x_2 = 0, \quad x_3 = 3, \quad x_4 = 0.$$

It is degenerate. The solution of the corresponding dual problem is not unique. Any set of variables $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ satisfying

$$\begin{aligned} \lambda_1 + \lambda_3 &= 1, \\ 2\lambda_2 + \lambda_4 &\leq 1, \\ \lambda_2 + \lambda_3 + \lambda_4 &= 1, \\ \lambda_1 + \lambda_4 &\leq 1, \\ \lambda_1, \lambda_2, \lambda_3, \lambda_4 &\geq 0 \end{aligned} \tag{2.7}$$

is optimal for the dual. For our purposes we need a description of the polytope defined by (2.7). The first and third equation in (2.7) imply that

$$\lambda_1 = 1 - \lambda_3 \geq 0 \quad \text{and} \quad \lambda_2 = 1 - \lambda_3 - \lambda_4 \geq 0. \tag{2.8}$$

Substituting these expressions for λ_1 and λ_2 in the inequalities in (2.7) gives the inequalities

$$2\lambda_3 + \lambda_4 \geq 1, \qquad \lambda_3 - \lambda_4 \geq 0 \tag{2.9}$$

for λ_3 and λ_4. Thus the polytope defined by (2.7) can be described in terms of λ_3 and λ_4 by the inequalities (2.8) and (2.9), together with $\lambda_3 \geq 0$, $\lambda_4 \geq 0$.

These inequalities describe the shaded region in Figure 1.

When we solved (2.6) by the simplex algorithm we obtained the following optimal dual variables:

$$\lambda_1^* = \frac{2}{3}, \quad \lambda_2^* = \frac{1}{3}, \quad \lambda_3^* = \frac{1}{3}, \quad \lambda_4^* = \frac{1}{3}.$$

This solution corresponds to the vertex λ^* in Figure 1. The simplex algorithm always gives a vertex solution. Suppose we use λ^* to generate a new pattern to improve the solution we obtained by solving (2.6). If such a pattern exists it corresponds to nonnegative integers $\alpha, \beta, \gamma, \delta$ satisfying

$$\frac{2}{3}\alpha + \frac{1}{3}\beta + \frac{1}{3}\gamma + \frac{1}{3}\delta > 1$$

and

$$8\alpha + 4\beta + 3\gamma + 2\delta \leq 11. \tag{2.10}$$

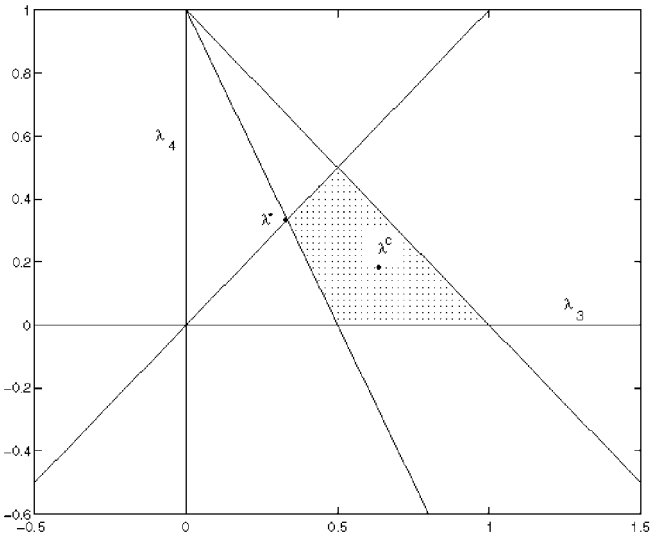


FIG. 1.

Usually the existence of such a pattern is checked by solving the following knapsack problem:

$$\begin{aligned} \max \quad & \frac{2}{3}\alpha + \frac{1}{3}\beta + \frac{1}{3}\gamma + \frac{1}{3}\delta \\ \text{subject to} \quad & 8\alpha + 4\beta + 3\gamma + 2\delta \le 11, \\ & \alpha, \beta, \gamma, \delta \ \text{ are integers } \ge 0. \end{aligned}$$

This problem has two solutions: $\alpha = \beta = \gamma = 0, \delta = 5$, and $\alpha = \beta = 0, \gamma = 1, \delta = 4$.

These solutions satisfy (2.10) so it appears that adding the patterns $p_5 = (0,0,0,5)^T$ and $p_6 = (0,0,1,4)^T$ to (2.6) will improve our current solution. However, adding these patterns to (2.6) adds the constraints $5\lambda_4 \le 1$, $\lambda_3 + 4\lambda_4 \le 1$ to the dual constraints (2.7). When these constraints are added to Figure 1 some of the dual variables that were optimal for the dual of (2.6) remain optimal for the dual of the problem obtained by adding p_5 and p_6 to (2.6). Thus adding the new patterns will not improve our solution.

Now consider solving (2.6) by the affine scaling algorithm. We first convert the constraints in (2.6) to equalities by introducing surplus variables. A starting solution x^0 for the algorithm (1.3) can be obtained by giving x_1, x_2, x_3, x_4 sufficiently large values that all surplus variables are positive. For a step-size $t \le 2/3$ the affine scaling algorithm converges to a solution x^* with

$$x_1^* = 2, \quad x_2^* = 0, \quad x_3^* = 3, \quad x_4^* = 0.$$

The corresponding sequence $\{\lambda^k\}$ converges to the optimal dual solution

$$\lambda_1^c = .3653, \quad \lambda_2^c = .1827, \quad \lambda_3^c = .6347, \quad \lambda_4^c = .1827 \tag{2.11}$$

corresponding to the point λ^c in Figure 1. λ^c corresponds to the analytic center of the polytope defined by (2.7). An inequality of the form

$$\alpha\lambda_1 + \beta\lambda_2 + \gamma\lambda_3 + \delta\lambda_4 \le 1$$

with $\alpha, \beta, \gamma, \delta \geq 0$, which is required to cut away the point λ^c in Figure 1 is clearly more likely to cut away the entire polytope than one that is merely required to cut away the vertex λ^*. This is why the multipliers at the analytic center of the polytope (2.7) are better for column generation than those at the vertices and generated by the simplex algorithm. The simplex multipliers tend to be high on the polytope because they provide a maximum to the objective $b^T\lambda$.

To see how all this works out in our present example consider generating a column using the multipliers (2.11). We must solve the problem

$$\begin{array}{ll} \max & .3653\alpha + .1827\beta + .6347\gamma + .1827\delta \\ \text{subject to} & 8\alpha + 4\beta + 3\gamma + 2\delta \leq 11, \\ & \alpha, \beta, \gamma, \delta \text{ are integers } \geq 0. \end{array}$$

The solution is $\alpha = \beta = 0, \gamma = 3, \delta = 1$. Adding the pattern $p_7 = (0, 0, 3, 1)^T$ to (2.6) adds the constraint $3\lambda_3 + \lambda_4 \leq 1$ to (2.7) and clearly cuts away the entire polytope in Figure 1. Thus adding the pattern p_7 to (2.6) will lead to an improvement in our objective value.

This shows that for our small problem the column generated by the dual prices λ^c is superior to the columns generated by the simplex dual prices λ^*. In large problems it generally happens that many iterations of the simplex algorithm are required to produce a column that improves the current solution. Interior point algorithms, all of which include an affine scaling component, tend to require far fewer steps. This motivates our interest in the dual variables generated by the affine scaling algorithm.

Consider the sequences $\{x^k\}$ and $\{\lambda^k\}$ generated by the algorithm (1.3) applied to (2.1). We are going to compute an explicit expression for λ^k. By definition, λ^k satisfies the equation

$$(AD_k^2A^T)\lambda^k = AD_k^2c.$$

Let M_1 denote the matrix obtained by replacing the first column of $AD_k^2A^T$ with AD_k^2c, and let M_2 denote the matrix obtained by replacing the second column of $AD_k^2A^t$ with AD_k^2c. Then by Cramer's rule we have

$$\lambda_1^k = \frac{\det M_1}{\det AD_k^2A^T} \quad \text{and} \quad \lambda_2^k = \frac{\det M_2}{\det AD_k^2A^T}.$$

For $i < j$ let

$$\begin{aligned} \alpha_{ij} &= a_{1i}a_{2j} - a_{2i}a_{1j}, \\ \beta_{ij} &= c_ia_{2j} - c_ja_{2i}, \\ \gamma_{ij} &= c_ja_{1i} - c_ia_{1j}. \end{aligned} \tag{2.12}$$

By the Cauchy–Binet theorem we have

$$\begin{aligned} \det AD_k^2A^T &= \Sigma_{i<j}\alpha_{ij}^2(x_i^kx_j^k)^2, \\ \det M_1 &= \Sigma_{i<j}\alpha_{ij}\beta_{ij}(x_i^kx_j^k)^2, \\ \det M_2 &= \Sigma_{i<j}\alpha_{ij}\gamma_{ij}(x_i^kx_j^k)^2. \end{aligned}$$

Therefore,

$$\lambda_1^k = \frac{\Sigma_{i<j}\alpha_{ij}\beta_{ij}(x_i^kx_j^k)^2}{\Sigma_{i<j}\alpha_{ij}^2(x_i^kx_j^k)^2} \quad \text{and} \quad \lambda_2^k = \frac{\Sigma_{i<j}\alpha_{ij}\gamma_{ij}(x_i^kx_j^k)^2}{\Sigma_{i<j}\alpha_{ij}^2(x_i^kx_j^k)^2}. \tag{2.13}$$

Dikin [5] used this representation of λ^k to show that the sequence $\{\lambda^k\}$ is always bounded. However, the sequence $\{x^k\}$ is converging to zero. Therefore, for large values of k the determinants $\det AD_k^2A^T$, $\det M_1$, $\det M_2$ are practically zero, and large errors are encountered if one attempts to evaluate λ^k using the formulas (2.13). In fact, for degenerate problems the scheme (1.3), as it is written, is very unstable and cannot be used to compute λ^k and x^k accurately for large values of k. However, there is a way to modify the scheme so that the sequence $\{\lambda^k\}$ can be computed accurately. The expression for λ^k is not changed if the values x_j^k are scaled. For our purposes it is convenient to express λ^k in terms of the variables

$$z^k = \frac{x_2^k}{x_1^k}, \quad y^k = \frac{x_3^k}{x_1^k}, \quad w^k = \frac{x_4^k}{x_1^k}, \quad v^k = \frac{x_5^k}{x_1^k}.$$

After computing the constants in (2.13) we can write λ^k in terms of these variables as

$$\lambda_1 = \frac{120z^2 + 20y^2 + 27w^2 + 22v^2 - 80z^2y^2 - 24z^2w^2 - 4y^2w^2 - 12y^2v^2 - 2w^2v^2}{16z^2 + 16y^2 + 9w^2 + 4v^2 + 16z^2y^2 + 4z^2w^2 + y^2w^2 + 4y^2v^2 + w^2v^2},$$

$$\lambda_2 = \frac{-40z^2 - 140y^2 - 63w^2 - 18v^2 + 160z^2y^2 + 44z^2w^2 + 7y^2w^2 + 16y^2v^2 + 3w^2v^2}{16z^2 + 16y^2 + 9w^2 + 4v^2 + 16z^2y^2 + 4z^2w^2 + y^2w^2 + 4y^2v^2 + w^2v^2}, \tag{2.14}$$

where we have suppressed the superscript k on all variables.

The conditions (2.1) imply that

$$\begin{aligned} 1 + 2z - 3y - 2w - v &= 0, \\ -1 + 2z - y - w - v &= 0. \end{aligned} \tag{2.15}$$

These equations can be solved for y and z in terms of w and v giving

$$\begin{aligned} y &= \frac{2 - w}{2}, \\ z &= \frac{w + 2v + 4}{4}. \end{aligned} \tag{2.16}$$

Since y and z must remain positive while iterating the scheme (1.3), we must restrict w and v to the region

$$0 < w < 2, \qquad v > 0. \tag{2.17}$$

We must discover how the variables y, z, w, v are propagated under the scheme (1.3). Consider the variable $s = c - A^T\lambda$. A brief calculation shows that

$$s = \frac{1}{\Delta(w,v)} \begin{pmatrix} 400z^2y^2 + 108z^2w^2 + 21y^2w^2 + 68y^2v^2 + 15w^2v^2 \\ 400y^2 + 162w^2 + 32v^2 + 4y^2w^2 + 32y^2v^2 + 8w^2v^2 \\ 400z^2 + 63w^2 + 68v^2 - 8z^2w^2 + 2w^2v^2 \\ 216z^2 - 84y^2 + 30v^2 + 16z^2y^2 - 4y^2v^2 \\ 64z^2 - 136y^2 - 45w^2 + 64z^2y^2 + 16z^2w^2 + 2y^2w^2 \end{pmatrix},$$

where

$$\Delta(w,v) = 16z^2 + 16y^2 + 9w^2 + 4v^2 + 16z^2 + 4z^2w^2 + y^2w^2 + 4y^2v^2 + w^2v^2$$

is the determinant in the denominator of (2.14), with y and z given in terms of w and v by (2.16).

By definition we have

$$D = \operatorname{diag}(x_1, x_2, x_3, x_4, x_5) = x_1 \operatorname{diag}(1, z, y, w, v).$$

Therefore,

$$Ds = \frac{x_1}{\Delta(w,v)} \begin{pmatrix} P_1(w,v) \\ P_2(w,v) \\ P_3(w,v) \\ P_4(w,v) \\ P_5(w,v) \end{pmatrix},$$

where

$$P_1(w,v) = 400z^2y^2 + 108z^2w^2 + 21y^2w^2 + 68y^2v^2 + 15w^2v^2,$$

$$P_2(w,v) = 400zy^2 + 162zw^2 + 32zv^2 + 4zy^2w^2 + 32zy^2v^2 + 8zw^2v^2,$$

$$P_3(w,v) = 400z^2y + 63w^2y + 68v^2y - 8z^2w^2y + 2w^2v^2y,$$

$$P_4(w,v) = 216z^2w - 84y^2w + 30v^2w + 16z^2y^2w - 4y^2v^2w,$$

$$P_5(w,v) = 64z^2v - 136y^2v - 45w^2v + 64z^2y^2v + 16z^2w^2v + 2y^2w^2v.$$

Let P denote the vector $(P_1, P_2, P_3, P_4, P_5)$. We will write P^k to denote the vector P with $(w,v) = (w^k, v^k)$.

We can now write the formula for x^{k+1} in (1.3) as

$$x_i^{k+1} = x_i^k \left\{ 1 - t \frac{P_i(w^k, v^k)}{\phi(P^k)} \right\}.$$

From this it follows that

$$w^{k+1} = \frac{x_4^{k+1}}{x_1^{k+1}} = \frac{x_4^k \left\{ 1 - t \frac{P_4(w^k,v^k)}{\phi(P^k)} \right\}}{x_1^k \left\{ 1 - t \frac{P_1(w^k,v^k)}{\phi(P^k)} \right\}}$$

$$= w^k \frac{\phi(P^k) - tP_4(w^k, v^k)}{\phi(P^k) - tP_1(w^k, v^k)}. \tag{2.18}$$

Similarly

$$v^{k+1} = v^k \frac{\phi(P^k) - tP_5(w^k, v^k)}{\phi(P^k) - tP_1(w^k, v^k)}, \tag{2.19}$$

$$y^{k+1} = y^k \frac{\phi(P^k) - tP_3(w^k, v^k)}{\phi(P^k) - tP_1(w^k, v^k)}, \tag{2.20}$$

and

$$z^{k+1} = z^k \frac{\phi(P^k) - tP_2(w^k, v^k)}{\phi(P^k) - tP_1(w^k, v^k)}. \tag{2.21}$$

Formulas (2.18) and (2.19) can be used to compute w^k and v^k accurately for very large values of k. Given w^k and v^k we can compute y^k and z^k from the formulas (2.16). Finally, λ_1^k and λ_2^k can be computed by substituting in (2.14), and the values obtained are accurate since the denominator in (2.14) is bounded away from zero due to the inhomogeneity of the equations (2.15). Equation (2.14) was used to compute the limits (2.3) and (2.4).

3. Numerical experiments for the case $\frac{2}{3} < t < 1$. We are now ready to analyze the behavior of the sequence $\{\lambda^k\}$ generated by (1.3) for problem (2.1). Recall that λ_1^k and λ_2^k are given in terms of the variables w^k, v^k, y^k, z^k, by the equations (2.14). We will update these variables using the formulas (2.18)–(2.21). Since each of these variables is a function of w^k and v^k, it suffices to analyze the behavior of the sequence $\{(w^k, v^k)\}$. This sequence has a very complicated behavior. We will only describe the essence of its behavior.

For $0 < t \leq \frac{2}{3}$, and for any starting point (w^0, v^0) in the region (2.17), the sequence $\{(w^k, v^k)\}$ converges to a unique point (w^*, v^*) as required by the convergence result in [14]. However, as t passes through $\frac{2}{3}$, the limit point (w^*, v^*) sprouts into six branches. For a fixed t, slightly larger than $\frac{2}{3}$, the points on these branches separate into three pairs, which we will denote by $\{(w_1^+, v_1^+), (w_1^-, v_1^-)\}, \{(w_2^+, v_2^+), (w_2^-, v_2^-)\}$, and $\{(w_3^+, v_3^+), (w_3^-, v_3^-)\}$, respectively. The region (2.17) is divided into three subregions R_1, R_2, and R_3 such that, if (w^0, v^0) is chosen in R_i, the sequence (w^k, v^k) remains in R_i and oscillates between (w_i^+, v_i^+) and (w_i^-, v_i^-) as $k \to \infty$. Each of the regions R_i is irregularly shaped, connected, and global in (2.17). They are described in detail in [2]. As t increases through certain critical values above $\frac{2}{3}$, some of the points $(w_i^{\pm}, v_i^{\pm})$ bifurcate and the period of oscillation of the sequence (w^k, v^k) doubles.

To show some of the complicated behavior the affine scaling algorithm can exhibit we will compute the sequence $\{(w^k, v^k)\}$ for various values of t, always starting with $w^0 = .7$ and $v^0 = 2$. For the sake of simplicity, we will just show the behavior of $\{v^k\}$. The oscillatory behavior of $\{w^k\}$ is similar to that of $\{v^k\}$. For each t in the interval [.65,1] we will generate the sequences $\{w^k\}$ and $\{v^k\}$ using formulas (2.18)–(2.21) with $w^0 = .7$ and $v^0 = 2$. For each value of t let $V(t)$ denote the set of limit points of the sequence $\{v^k\}$. In Figure 2 we have plotted the points (t, v) for each $v \in V(t)$. Such plots are referred to in [3] as Feigenbaum diagrams in honor of the physicist who used them to study the behavior of complicated dynamical systems. For $t \leq \frac{2}{3}$ the sequence $\{v^k\}$ converges. Thus $V(t)$ contains a single point. For $\frac{2}{3} < t < .8700$ $V(t)$ contains two points. For $.8700 < t < .8915$ $V(t)$ contains four points. For $.8915 < t < .8960$ $V(t)$ contains eight points. Clearly the number of points in $V(t)$ doubles at certain critical times. The first six of these times are given, approximately, by

$$t_1 = .66667, \quad t_2 = .8700, \quad t_3 = .8915, \quad t_4 = .8960, \quad t_5 = .90011, \quad t_6 = .90025.$$

There seems to be an infinite sequence of points at which the graph of $V(t)$ bifurcates. These bifurcations are strongly related to a result by Tsuchiya and Monteiro [13]. These authors show that for a homogeneous problem with unique solution $x^* = 0$, the sequence $\{x^k\}$ generated by the affine scaling algorithm has at least two directions of

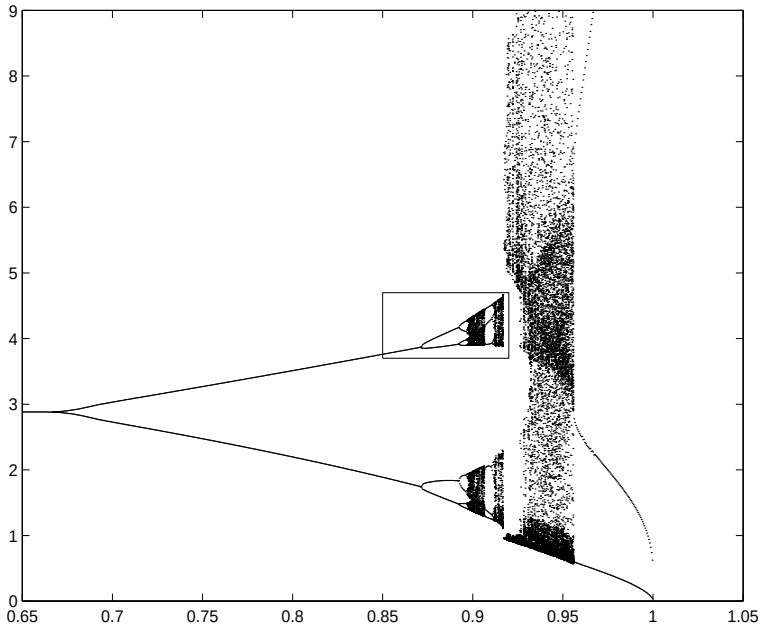


FIG. 2. *Feigenbaum diagram for the sequence* $\{v^k\}$.

approach to 0. Now consider the sequences $\{w^k\}$, $\{v^k\}$, $\{y^k\}$, $\{z^k\}$ defined by (2.18)–(2.21) with $w^0 = .7$ and $v^0 = 2$. For $t_1 < t < t_2$ each of these sequences converges to a 2-cycle. Let's denote the points in this cycle by $\{w^-, w^+\}$, $\{v^-, v^+\}$, $\{y^-, y^+\}$, and $\{z^-, z^+\}$, respectively. Then for large values of k we have

$$w^{2k} = \frac{x_4^{2k}}{x_1^{2k}} \approx w^-, \qquad w^{2k+1} = \frac{x_4^{2k+1}}{x_1^{2k+1}} \approx w^+,$$

and similar equations hold for the terms of the sequences $\{v^k\}$, $\{y^k\}$, $\{z^k\}$. In vector notation this means that for large values of k,

$$x^{2k} \approx x_1^{2k}(1, z^-, y^-, w^-, v^-)^T$$

and

$$x^{2k+1} \approx x_1^{2k+1}(1, z^+, y^+, w^+, v^+)^T.$$

Since the sequence $\{x_1^k\}$ converges to 0, it follows that the sequence $\{x^k\}$ has exactly two directions of approach to 0 for $t_1 < t < t_2$. We have seen that the orbits to which the sequences $\{w^k\}$, $\{v^k\}$, $\{y^k\}$, $\{z^k\}$ converge depend on their initial values. Thus the direction of approach to 0 for $\{x^k\}$ has three choices, depending on how x^0 is chosen. Similarly, for $t_2 < t < t_3$ the sequence $\{x^k\}$ has exactly four directions of approach to 0, and they depend on x^0. Continuing in this way it would appear that the sequence $\{x^k\}$ always has 2^p directions of approach to 0 for some integer p. However, we will see that there are values of t in $(2/3, 1)$ for which the sequences $\{w^k\}$, $\{v^k\}$, $\{y^k\}$, $\{z^k\}$ converge to periodic orbits with periods 3 and 6. Thus the number of directions of approach of $\{x^k\}$ to 0 need not be a power of 2. To better understand

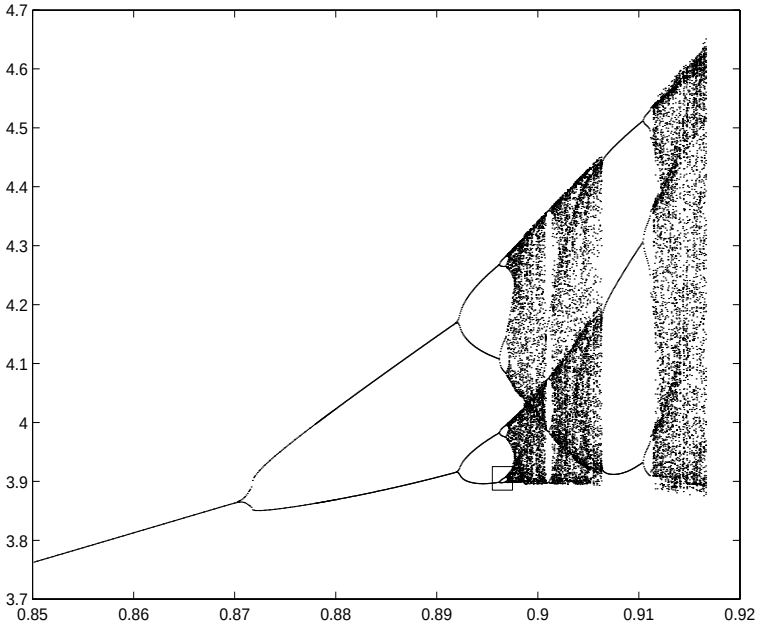


FIG. 3. *Inset from Figure* 2.

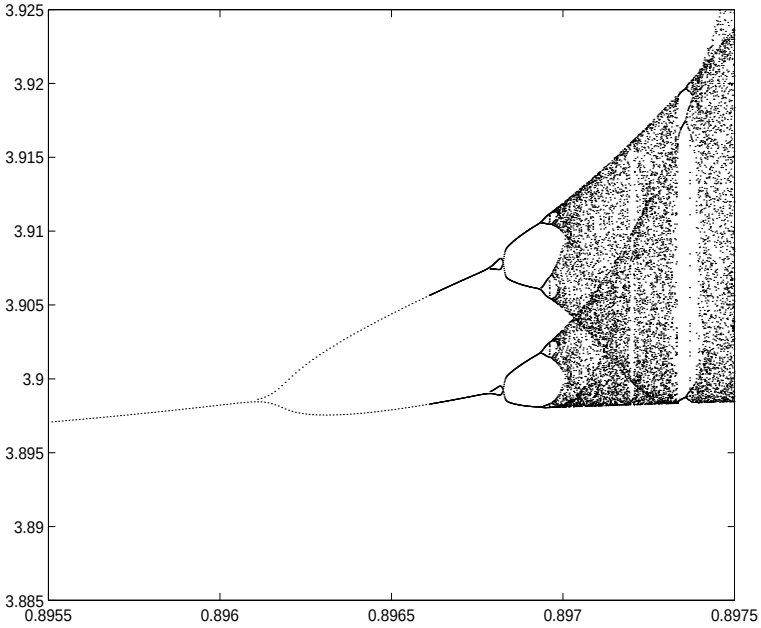


FIG. 4. *Inset from Figure* 5.

the bifurcation points for the function $V(t)$ we have enlarged the portion of Figure 2 inside the small rectangle. This magnification is shown in Figure 3. Similarly, the region inside the small rectangle in Figure 3 is shown in Figure 4. Notice the strong

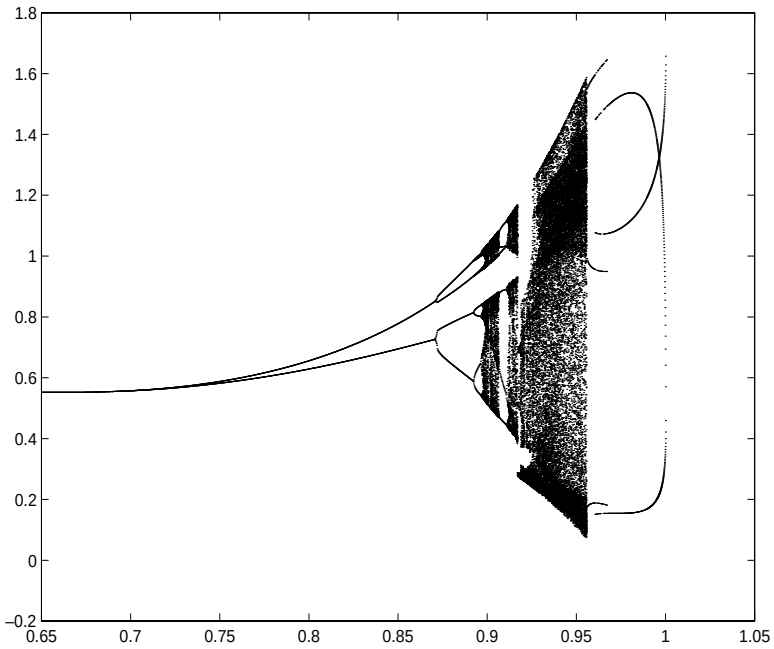


FIG. 5. *Feigenbaum diagram for the sequence* $\{\lambda_2^k\}$.

similarity between Figures 3 and 4. This indicates the fractal nature of Figure 3. It also suggests that the bifurcation gaps $t_{k+1} - t_k$ decrease like a geometric progression. The convergence properties of the sequence $\{(w^k, v^k)\}$ is an interesting topic in itself. However, our main concern is the convergence of the sequence $\{\lambda^k\}$ generated by (1.3). For the current example this sequence is given in terms of the sequence $\{(w^k, v^k)\}$ by the formulas (2.14). The bifurcations that occur in the sequence $\{(w^k, v^k)\}$ are transferred to the sequence $\{\lambda^k\}$ through these equations. In Figure 5 we have plotted the accumulation points of $\{\lambda_2^k\}$ against t for each t in the interval $.65 \leq t < 1$. The accumulation points of $\{\lambda_1^k\}$ form a similar plot so we do not show them. Note that for t only slightly larger than $2/3$ the two accumulation points of $\{\lambda_2^k\}$ are very close together. In fact for t very close to $2/3$ the divergence of the sequence $\{\lambda^k\}$ is hard to detect. This is why in the numerical experiments given at the beginning of section 2 we started with $t = .84$.

In order to generate the pictures in Figures 2, 3, 4, and 5 we had to iterate the affine scaling algorithm long enough to identify the accumulation points of the sequence $\{(w^k, v^k)\}$ for each t in the range $2/3 < t < 1$. Typically this required 200 iterations. In practice one would never run the affine scaling algorithm this long. Still there is a lesson in what we have shown. There are ranges of values of t for which the sequence $\{(w^k, v^k)\}$ has random behavior. And this random behavior can be seen even for small values of k. For our example this generally happens for $t > .9$. However, within this range there are intervals where the sequence $\{(w^k, v^k)\}$ converges to a periodic orbit. For example, for $t = .91$ $\{(w^k, v^k)\}$ converges to a periodic orbit of period 6. From Figure 5 we see that the corresponding sequence $\{\lambda^k\}$ converges to a periodic orbit of period 6.

What is more interesting is the fact that for t sufficiently close to 1 the sequence $\{\lambda^k\}$ converges to a periodic orbit of period 3. Thus for certain values of $t > 2/3$

the sequence $\{\lambda^k\}$ converges to a periodic orbit, while for other values the sequence $\{\lambda^k\}$ has no regular behavior. It appears to wander around randomly. It exhibits chaotic behavior. We wish to emphasize that this chaotic behavior is not restricted to specially constructed examples. It can happen in meaningful linear programming models. For instance the sequence $\{\lambda^k\}$ generated by the affine scaling algorithm for the cutting-stock problem in section 2 is chaotic for t sufficiently large.

REFERENCES

[1] E. R. Barnes, *A variation on Karmarkar's algorithm for solving linear programming problems*, Math. Programming, 36 (1986), pp. 174–182.

[2] I. Castillo, *Some Properties of the Affine Scaling Algorithm*, Ph.D. thesis, Georgia Institute of Technology, Atlanta, GA, 1996.

[3] R. L. Devaney, *An Introduction to Chaotic Dynamical Systems*, The Benjamin/Cummings Publishing Co., Menlo Park, CA, 1989.

[4] I. I. Dikin, *Iterative solution of problems of linear and quadratic programming*, Dokl. Akad. Nauk SSSR, 174 (1967), pp. 747–748. (Translation: Sov. Math. Dokl. 8 (1967), pp. 674–675.)

[5] I. I. Dikin, *On the Convergence of an Iterative Process*, Upravlyaemye Sistemi, 12 (1974), pp. 54–60 (in Russian).

[6] I. I. Dikin, *The Convergence of Dual Variables*, Tech. report, Siberian Energy Institute, Irkutsk, Russia, 1991.

[7] L. A. Hall and R. J. Vanderbei, *Two-thirds is sharp for affine scaling*, Oper. Res. Lett., 13 (1993), pp. 197–201.

[8] N. K. Karmarkar, *A new polynomial-time algorithm for linear programming*, Combinatorica, 4 (1984), pp. 373–395.

[9] W. F. Mascarenhas, *The Affine Scaling Algorithm Fails for* $\lambda = 0.999$, Tech. report, Universidade Estadual de Campinas, Campinas S.P., Brazil, 1993.

[10] R. D. C. Monteiro, T. Tsuchiya, and Y. Wang, *A simplified global convergence proof of the affine scaling algorithm*, Ann. Oper. Res., 47 (1993), pp. 443–482.

[11] T. Tsuchiya, *Global convergence of the affine-scaling methods for degenerate linear programming problems*, Math. Programming, 52 (1991), pp. 377–404.

[12] T. Tsuchiya, *Global convergence property of the affine scaling method for primal degenerate linear programming problems*, Math. Oper. Res., 17 (1992), pp. 527–557.

[13] T. Tsuchiya and R. D. C. Monteiro, *Superlinear convergence of the affine scaling algorithm*, Math. Programming, 75 (1996), pp. 77–110.

[14] T. Tsuchiya and M. Muramatsu, *Global convergence of a long-step affine scaling algorithm for degenerate linear programming problems*, SIAM J. Optim., 5 (1995), pp. 525–551.

[15] R. J. Vanderbei, M. S. Meketon, and B. A. Freedman, *A modification of Karmarkar's linear programming algorithm*, Algorithmica, 1 (1986), pp. 395–407.

[16] I. I. Dikin and C. Roos, *Convergence of the dual variables for the primal affine scaling method with unit steps in the homogeneous case*, J. Optim. Theory Appl., 95 (1997), pp. 305–321.

[17] R. D. C. Monteiro and T. Tsuchiya, *Global convergence of the affine scaling algorithm for convex quadratic programming*, SIAM J. Optim., 8 (1998), pp. 26–58.

[18] J.-L. Goffin and J. P. Vial, *Shallow, deep and very deep cuts in the analytic center cutting plane method*, Math. Programming, 84 (1999), pp. 89–103.

[19] J.-L. Goffin, J. Gondzio, R. Sarkissian, and J.-P. Vial, *Solving nonlinear multicommodity flows problems by the analytic center cutting plane method*, Math. Programming, 76 (1997), pp. 131–154.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 796–817

GLOBAL OPTIMIZATION WITH POLYNOMIALS AND THE PROBLEM OF MOMENTS*

JEAN B. LASSERRE†

Abstract. We consider the problem of finding the unconstrained global minimum of a real-valued polynomial $p(x) : \mathbb{R}^n \to \mathbb{R}$, as well as the global minimum of $p(x)$, in a compact set K defined by polynomial inequalities. It is shown that this problem reduces to solving an (often finite) sequence of convex linear matrix inequality (LMI) problems. A notion of Karush–Kuhn–Tucker polynomials is introduced in a global optimality condition. Some illustrative examples are provided.

Key words. global optimization, theory of moments and positive polynomials, semidefinite programming

AMS subject classifications. 90C22, 90C25

PII. S1052623400366802

1. Introduction. Given a real-valued polynomial $p(x) : \mathbb{R}^n \to \mathbb{R}$, we are interested in solving the problem

$$\mathbb{P} \mapsto p^* := \min_{x \in \mathbb{R}^n} p(x), \tag{1.1}$$

that is, finding the *global minimum* p^* of $p(x)$ and, if possible, a global minimizer x^*. We are also interested in solving

$$\mathbb{P}_K \mapsto p_K^* := \min_{x \in K} p(x), \tag{1.2}$$

where K is a (not necessarily convex) compact set defined by polynomial inequalities $g_i(x) \geq 0$, $i = 1, \ldots, r$, which includes many applications of interest and standard problems like quadratic, linear, and 0-1 programming as particular cases.

In the one-dimensional case, that is, when $n = 1$, Shor [17] first showed that (1.1) reduces to a convex problem. Next, Nesterov [13], invoking a well-known representation of nonnegative polynomials as a sum of squares of polynomials, provided a self-concordant barrier for the cone K_{2n} of nonnegative univariate polynomials so that efficient interior point algorithms are available to compute a global minimum.

However, the multivariate case radically differs from the one-dimensional case, for not every nonnegative polynomial can be written as a sum of squares of polynomials. Even more, as mentioned in Nesterov [13], the global unconstrained minimization of a 4-degree polynomial is an NP-hard problem. Via successive changes of variables, Shor [18] (see also Ferrier [5]) proposed to transform (1.2) into a quadratic, quadratically constrained optimization problem and then solve a standard convex linear matrix inequality (LMI) relaxation to obtain good lower bounds. By adding redundant quadratic constraints one may improve the lower bound and sometimes obtain the optimal value.

In this paper, we will show that the global unconstrained minimization (1.1) of a polynomial can be approximated as closely as desired (and often can be obtained exactly) by solving a finite sequence of convex LMI optimization problems of the

*Received by the editors January 28, 2000; accepted for publication (in revised form) August 28, 2000; published electronically January 19, 2001.
http://www.siam.org/journals/siopt/11-3/36680.html
†LAAS-CNRS, 7 Avenue du Colonel Roche, 31077 Toulouse Cédex 4, France (lasserre@laas.fr).

same flavor as in the one-dimensional case. A similar conclusion also holds for the constrained optimization problem $\mathbb{P}_K$ in (1.2), when K is a compact set, not necessarily convex, defined by polynomial inequalities. The difference between nonnegative and strictly positive polynomials is the reason why, in some cases, only an asymptotic result is possible. Indeed, for the latter, several representations in terms of weighted sums of squares are always possible, whereas few results are known for the former. However, from a numerical point of view, the distinction is irrelevant. In the constrained case, the nonnegative squared polynomials in the representation of the polynomial $p(x)-p_K^*$ can be interpreted as generalized Karush–Kuhn–Tucker multipliers whose value at a global minimizer are precisely the original Karush–Kuhn–Tucker scalar multipliers. This representation of nonnegative polynomials thus provides a natural optimality condition for global optimality.

When the optimal value is obtained at a particular LMI relaxation, the constrained *global* optimization problem thus has a natural "primal" LMI formulation, whose optimal solution provides a global minimizer, whereas an optimal solution of the dual LMI problem provides the Karush–Kuhn–Tucker polynomial multipliers in a representation of $p(x) - p_K^*$. Hence, the primal and dual LMI formulations perfectly match both sides of the same theory (moments and positive polynomials).

This approach is also valid for handling combinatorial problems, e.g., 0-1 programming problems, since the integrality constraint $x_i \in \{0, 1\}$ can be written $x_i^2 - x_i \geq 0$ and $x_i - x_i^2 \geq 0$. An elementary illustrative example is provided. We finally consider the general convex quadratic, quadratically constrained problem and provide a natural exact LMI formulation for both primal and dual problems (the Shor relaxation and its dual). The standard linear programming problem also appears as a particular case.

In [13], for the univariate case, the idea was to characterize the nonnegative polynomial $p(x) - p^*$ as a sum of squares. However, we will adopt a dual point of view. Namely, we replace $\mathbb{P}$ and $\mathbb{P}_K$ with the equivalent problems

$$\mathcal{P} \;\mapsto\; p^* := \min_{\mu \in \mathcal{P}(\mathbb{R}^n)} \int p(x)\, \mu(dx) \tag{1.3}$$

and

$$\mathcal{P}_K \;\mapsto\; p^* := \min_{\mu \in \mathcal{P}(K)} \int p(x)\, \mu(dx), \tag{1.4}$$

respectively, where $\mathcal{P}(\mathbb{R}^n)$ (respectively, $\mathcal{P}(K)$) is the space of finite Borel signed measures on $\mathbb{R}^n$ (respectively, on K). That $\mathcal{P}$ is equivalent to $\mathbb{P}$ is trivial. Indeed, as $p(x) \geq p^*$, then $\int p d\mu \geq p^*$ and thus $\inf \mathcal{P} \geq p^*$. Conversely, if x^* is a global minimizer of $\mathbb{P}$, then the probability measure $\mu^* := \delta_{x^*}$ (the Dirac at x^*) is admissible for $\mathcal{P}$. We then observe that if p is a polynomial of degree, say m, the criterion to minimize is a linear criterion $a'y$ on the finite collection of moments $\{y_\alpha\}$, up to order m, of the probability measure μ. We can then in turn replace $\mathcal{P}$ (respectively, $\mathcal{P}_K$) with an optimization problem on the y_α variables with the constraint that the y_α's must be moments of some probability measure μ. The theory of *moments* provides adequate conditions on the y_α variables. It has been known for a long time that the theory of moments is strongly related to—and in fact, in duality with—the theory of nonnegative polynomials and Hilbert's 17th problem on the representation of nonnegative polynomials. For the historical development and recent results on the theory of moments, the interested reader is referred to Berg [1], Curto and Fialkow [2], [3],

Jacobi [8], Putinar [15], Putinar and Vasilescu [14], Simon [19], Schmüdgen [16], and references therein.

The paper is organized as follows. We introduce the notation and some preliminary results in section 2. The unconstrained case is treated in section 3 and the constrained case (1.2) in section 4. Some elementary as well as nontrivial examples are presented for illustration. In the last section we show that when $p(x) - p^*_K$ is a weighted sum of squares, then the squared polynomials can be interpreted as generalized Karush–Kuhn–Tucker multipliers. The convex quadratic case is also investigated.

2. Notation and preliminary results. Let

$$1, x_1, x_2, \ldots, x_n, x_1^2, x_1x_2, \ldots, x_1x_n, x_2x_3, \ldots, x_n^2, \ldots, x_1^m, \ldots, x_n^m \tag{2.1}$$

be a basis for the m-degree real-valued polynomials $p(x) : \mathbb{R}^n \to \mathbb{R}$, and let $s(2m)$ be its dimension. We adopt the following standard notation. If $p(x) : \mathbb{R}^n \to \mathbb{R}$ is an m-degree polynomial, write

$$p(x) = \sum_\alpha p_\alpha x^\alpha \quad \text{with } x^\alpha := x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} \quad \text{and } \sum_i \alpha_i \le m, \tag{2.2}$$

where $p = \{p_\alpha\} \in \mathbb{R}^{s(m)}$ is the coefficient vector of $p(x)$ in the basis (2.1). When needed, a polynomial of degree m can be considered as a polynomial of higher degree, say r, with coefficient vector $p \in \mathbb{R}^{s(r)}$, where the coefficients of monomials of degree higher than m are set to zero.

Given an $s(2m)$-vector $y := \{y_\alpha\}$ with first element $y_{0,\ldots,0} = 1$, let $M_m(y)$ be the *moment* matrix of dimension $s(m)$, with rows and columns labeled by (2.1). For instance, for illustration and clarity of exposition, consider the two-dimensional case. The moment matrix $M_m(y)$ is the block matrix $\{M_{i,j}(y)\}_{0 \le i,j \le 2m}$ defined by

$$M_{i,j}(y) = \begin{bmatrix} y_{i+j,0} & y_{i+j-1,1} & \cdots & y_{i,j} \\ y_{i+j-1,1} & y_{i+j-2,2} & \cdots & y_{i-1,j+1} \\ \cdots & \cdots & \cdots & \cdots \\ y_{j,i} & y_{i+j-1,1} & \cdots & y_{0,i+j} \end{bmatrix}, \tag{2.3}$$

where $y_{i,j}$ represents the $(i+j)$-order moment $\int x^i y^j \, \mu(d(x,y))$ for some probability measure μ. To fix ideas, when $n = 2$ and $m = 2$, one obtains

$$M_2(y) = \left[\begin{array}{c|cc|cc} 1 & y_{1,0} & y_{0,1} & y_{2,0} & y_{1,1} & y_{0,2} \\ \hline y_{1,0} & y_{2,0} & y_{1,1} & y_{3,0} & y_{2,1} & y_{1,2} \\ y_{0,1} & y_{1,1} & y_{0,2} & y_{2,1} & y_{1,2} & y_{0,3} \\ \hline y_{2,0} & y_{3,0} & y_{2,1} & y_{4,0} & y_{3,1} & y_{2,2} \\ y_{1,1} & y_{2,1} & y_{1,2} & y_{3,1} & y_{2,2} & y_{1,3} \\ y_{0,2} & y_{1,2} & y_{0,3} & y_{2,2} & y_{1,3} & y_{0,4} \end{array}\right].$$

For the three-dimensional case, $M_m(y)$ is defined via blocks $\{M_{i,j,k}(y)\}$, $0 \le i,j,l \le 2m$ in a similar fashion, and so on.

Let $y = \{y_\alpha\}$ (with $y_{0,\ldots,0} = 1$) be the vector of moments up to order $2m$ of some probability measure μ_y. Let $\mathcal{A}_m$ be the vector space of real-valued polynomials $q(x) : \mathbb{R}^n \to \mathbb{R}$ of degree at most m. Identifying $q(x)$ with its vector $q \in \mathbb{R}^{s(m)}$ of

coefficients in the basis (2.1), one may then define a bilinear form $\langle ., \rangle_y : \mathcal{A}_m \times \mathcal{A}_m \to \mathbb{R}$ by

$$\langle q(x), p(x)\rangle_y \;=\; \langle q, M_m(y)p\rangle \;=\; \sum_\alpha (qp)_\alpha y_\alpha \;=\; \int q(x)p(x)\,\mu_y(dx). \tag{2.4}$$

This bilinear form also defines a positive semidefinite form on $\mathcal{A}_m$ since

$$\langle q(x), q(x)\rangle_y \;=\; \sum_\alpha (q^2)_\alpha y_\alpha \;=\; \int q(x)^2\,\mu_y(dx) \;\geq\; 0 \tag{2.5}$$

for all polynomials $q(x) \in \mathcal{A}_m$. The theory of moments identifies those sequences y with $M_m(y) \succeq 0$ that correspond to moments of some probability measure μ_y on $\mathbb{R}^n$.

We first briefly outline the idea developed in the next section: With K an arbitrary (Borel) subset of $\mathbb{R}^n$, one first reduces $\mathbb{P}_K$ to the equivalent convex optimization problem $\mathcal{P}_K$,

$$\mathcal{P}_K \mapsto \min_{\mu \in \mathcal{P}(K)} \int p(x)\,d\mu, \tag{2.6}$$

on the space of Borel probability measures μ with support contained in K. Indeed, we have the following.

PROPOSITION 2.1. *The problems $\mathcal{P}_K$ and $\mathbb{P}_K$ are equivalent, that is,*

(a) $\inf \mathbb{P}_K \;=\; \inf \mathcal{P}_K$.

(b) *if x^* is a global minimizer of $\mathbb{P}_K$, then $\mu^* := \delta_{x^*}$ is a global minimizer of $\mathcal{P}_K$.*

(c) *assuming $\mathbb{P}_K$ has a global minimizer, then, for every optimal solution μ^* of $\mathcal{P}_K$, $p(x) = \min \mathbb{P}_K$, μ^*-almost everywhere (μ^*-a.e.).*

(d) *if x^* is the unique global minimizer of $\mathbb{P}_K$, then $\mu^* := \delta_{x^*}$ is the unique global minimizer of $\mathcal{P}_K$.*

Proof. (a) As for every $x \in K$, $p(x) = \int p\,d\delta_x$, it follows that $\inf \mathcal{P}_K \leq \inf \mathbb{P}_K$ (including the case $-\infty$). Conversely, assume that $p^* := \inf \mathbb{P}_K > -\infty$. As $p(x) \geq p^*$ for all $x \in K$, it follows that $\int p d\mu \geq p^*$ for every probability measure μ with support contained in K.

(b) This proof is trivial.

(c) From (b), $\mathbb{P}_K$ has at least one optimal solution. For an arbitrary optimal solution μ^*, we have $\int p d\mu^* = p^*$ with $p^* = \min \mathbb{P}_K$. Assume that there is a Borel set $B \subset K$ such that $\mu^*(B) > 0$ and $p(x) \neq p^*$ on B, that is, $p(x) > p^*$ on B. Then,

$$\int p\,d\mu^* \;=\; \int_B p\,d\mu^* + \int_{K-B} p\,d\mu^* \;>\; p^*,$$

in contradiction with $\int p d\mu^* = p^*$.

(d) This proof follows from (c). □

Observe that since $p(x)$ is a polynomial of degree, say m, the criterion $\int p d\mu$ involves only the moments of μ, up to order m and, in addition, is *linear* in the moment variables. Therefore, one next replaces μ with the finite sequence $y = \{y_\alpha\}$ of all its moments, up to order m, that is,

$$y_\alpha \;:=\; \int x^\alpha\,d\mu, \qquad \sum_{i=1}^n \alpha_i = k,\; k = 0, 1, \ldots, m,$$

and one works with the finite sequence y of the moments of μ, up to order m, instead of μ itself. Of course, not every sequence y has a *representing measure* μ; that is, given an arbitrary finite sequence y, there might not be any probability measure μ, all of whose moments up to order m coincide with the y_α scalars. In the one-dimensional case, characterizing those sequences y that have a representing measure on X (respectively, on $[0,\infty)$ and $[a,b]$) is called the (truncated) *Hamburger* (respectively, *Stieltjes* and *Hausdorff*) *moment problem* (see Curto and Fialkow [3] or Simon [19] and references therein). The various necessary and sufficient conditions for the existence of a representing measure μ_y all invoke the positive semidefiniteness of the related (Hankel) moment matrix

$$H_m(y) := \begin{bmatrix} y_0 & y_1 & y_2 & \cdot & y_m \\ y_1 & y_2 & \cdot & \cdot & y_{m+1} \\ \cdot\cdot & \cdot & \cdot & \cdot & \cdot \\ y_m & y_{m+1} & \cdot & y_{2m-1} & y_{2m} \end{bmatrix} \tag{2.7}$$

(see, for instance, the various conditions related to the truncated Hamburger, Stieltjes, and Hausdorff moment problems in Curto and Fialkow [3]). For trigonometric polynomials, Toeplitz matrices are the analogues of the Hankel matrices.

As mentioned earlier, this theory of moments is in duality with the theory of nonnegative polynomials and Hilbert's 17th problem on the representation of nonnegative polynomials as sum of squares (always possible in the one-dimensional case). However, the multivariate case radically differs from the univariate case, for not every nonnegative polynomial can be written as a sum of squares. Also, in contrast to the univariate case, with $M_m(y)$ the moment matrix previously introduced (in lieu of the Hankel matrix (2.7)), there are vectors y for which $M_m(y) \succ 0$ but with *no* representing measure μ_y.

3. Unconstrained global optimization. Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a real-valued polynomial of degree $2m$ with coefficient vector $p \in \mathbb{R}^{s(2m)}$. Since we wish to minimize $p(x)$, we may and will assume that the constant term vanishes, that is, $p_0 = 0$. Let us introduce the following convex LMI optimization problem (or positive semidefinite (psd) program):

$$\mathbb{Q} \mapsto \left\{ \begin{array}{ll} \inf\limits_y \sum\limits_\alpha p_\alpha y_\alpha, & \\ \qquad M_m(y) & \succeq \quad 0, \end{array} \right. \tag{3.1}$$

or equivalently,

$$\mathbb{Q} \mapsto \left\{ \begin{array}{ll} \inf\limits_y \sum\limits_\alpha p_\alpha y_\alpha, & \\ \qquad \sum\limits_{\alpha \neq 0} y_\alpha B_\alpha & \succeq \quad -B_0, \end{array} \right. \tag{3.2}$$

where the matrices B_0 and B_α are easily understood from the definition of $M_m(y)$. The dual problem $\mathbb{Q}^*$ of $\mathbb{Q}$ is the convex LMI problem defined by

$$\mathbb{Q}^* \mapsto \left\{ \begin{array}{lll} \sup\limits_X \langle X, -B_0 \rangle \, (= -X(1,1)), & & \\ \qquad \langle X, B_\alpha \rangle & = \quad p_\alpha, & \alpha \neq 0, \\ \qquad X & \succeq \quad 0, & \end{array} \right. \tag{3.3}$$

where X is a real-valued symmetric matrix and $\langle A, B\rangle$ stands for the usual Frobenius inner product trace(AB) for real-valued symmetric matrices. The reader is referred to Vandenberghe and Boyd [20] for a survey on semidefinite programming.

We first have the following result.

PROPOSITION 3.1. *Assume that $\mathbb{Q}^*$ has a feasible solution. Then $\mathbb{Q}^*$ is solvable and there is no duality gap, that is,*

$$\inf \mathbb{Q} = \max \mathbb{Q}^*. \tag{3.4}$$

Proof. The result follows from the duality theory of convex programming if we can prove that there is a feasible solution y of $\mathbb{Q}$ with $M_m(y) \succ 0$. Let μ be a probability measure on $\mathbb{R}^n$ with a *strictly positive* density f with respect to the Lebesgue measure and with all its moments finite; that is, μ is such that

$$y_\alpha := \int x^\alpha \, d\mu < \infty$$

for every combination $\alpha_1 + \alpha_2 + \alpha_n = r$, $r = 1, 2, \ldots$. Then the matrix $M_m(y)$, with y as above, is such that $M_m(y) \succ 0$. Indeed, for every polynomial $q(x) : \mathbb{R}^m \to \mathbb{R}$, we have

$$\begin{aligned} \langle q(x), q(x)\rangle_y = \langle q, M_m(y) q\rangle &= \int q^2(x)\, \mu(dx) \ \text{(by (2.5))} \\ &= \int q(x)^2 f(x)\, dx \\ &> 0 \ \text{whenever } q \neq 0 \ (\text{as } f > 0). \end{aligned}$$

Therefore, y is feasible for $\mathbb{Q}$ and $M_m(y) \succ 0$, the desired result. □

Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a real-valued polynomial with $p_0 := p(0) = 0$. The first result of this paper is as follows.

THEOREM 3.2. *Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a $2m$-degree polynomial as in (2.2) with global minimum $p^* = \min \mathbb{P}$.*

(i) *If the nonnegative polynomial $p(x) - p^*$ is a sum of squares of polynomials, then $\mathbb{P}$ is equivalent to the convex LMI problem $\mathbb{Q}$ defined in (3.1). More precisely, $\min \mathbb{Q} = \min \mathbb{P}$ and, if x^* is a global minimizer of $\mathbb{P}$, then the vector*

$$y^* := (x_1^*, \ldots, x_n^*, (x_1^*)^2, x_1^* x_2^*, \ldots, (x_1^*)^{2m}, \ldots, (x_n^*)^{2m}) \tag{3.5}$$

is a minimizer of $\mathbb{Q}$.

(ii) *Conversely, if $\mathbb{Q}^*$ has a feasible solution, then $\min \mathbb{P} = \min \mathbb{Q}$ only if $p(x) - p^*$ is a sum of squares.*

Proof. (i) Let $p(x) - p^*$ be a sum of squares of polynomials, that is,

$$p(x) - p^* = \sum_{i=1}^{r} q_i(x)^2, \qquad x \in \mathbb{R}^n, \tag{3.6}$$

for some polynomials $q_i(x) : \mathbb{R}^n \to \mathbb{R}$, with coefficient vector $q_i \in \mathbb{R}^{s(m)}$, $i = 1, 2, \ldots, r$. Equivalently,

$$p(x) - p^* = \langle X, M_m(y)\rangle, \qquad x \in \mathbb{R}^n, \tag{3.7}$$

with $X = \sum_1^r q_i q_i'$ and $y = (x_1, \ldots, (x_1)^{2m}, \ldots, (x_n)^{2m})$. But from (3.7) and

$$p(x) - p^* \;=\; -p^* + \sum_\alpha p_\alpha x^\alpha,$$

it follows that

$$X(1,1) \;=\; -p^* \quad \text{and } \langle X, B_\alpha\rangle \;=\; p_\alpha \quad \text{for all } \alpha \neq 0$$

so that (as $X \succeq 0$) X is feasible for $\mathbb{Q}^*$ with value $-X(1,1) = p^*$. Next, observe that y^* in (3.5) is feasible for $\mathbb{Q}$ with value p^* so that $\min \mathbb{Q} = \max \mathbb{Q}^*$ and y^* and X are optimal solutions of $\mathbb{Q}$ and $\mathbb{Q}^*$, respectively.

(ii) Assume that $\mathbb{Q}^*$ has a feasible solution and $\min \mathbb{P} = \min \mathbb{Q}$. Then, from Proposition 3.1, $\mathbb{Q}^*$ is solvable and there is no duality gap, that is, $\max \mathbb{Q}^* = \inf \mathbb{Q} = \min \mathbb{Q}$. Let X^* be an optimal solution of $\mathbb{Q}^*$, guaranteed to exist. Write $X^* = \sum_{i=1}^r \lambda_i q_i q_i'$ with the q_i's being the eigenvectors of X^* corresponding to the positive eigenvalues λ_i, $i = 1, \ldots, r$.

As $\lambda^* := \max \mathbb{Q}^* = \min \mathbb{Q}$, and $\min \mathbb{Q} = \min \mathbb{P}$, let y^* as in (3.5) be an optimal solution of $\mathbb{Q}$. From the optimality of both X^* and y^*, we must have

$$\langle X^*, M_m(y^*)\rangle \;=\; 0.$$

Equivalently,

$$0 \;=\; \sum_{i=1}^r \lambda_i \langle q_i, M_m(y^*) q_i\rangle \;=\; \sum_{i=1}^r \lambda_i q_i(x^*)^2.$$

For an arbitrary $x \in \mathbb{R}^n$, let

$$y \;:=\; (x_1, \ldots, x_n, x_1^2, x_1x_2, \ldots, x_1^{2m}, \ldots, x_n^{2m})$$

so that, as we did for x^*,

$$\langle X^*, M_m(y)\rangle \;=\; \sum_{i=1}^r \lambda_i q_i(x)^2.$$

On the other hand,

$$\begin{aligned}\langle X^*, M_m(y)\rangle &= \lambda^* + \sum_{\alpha\neq 0} y_\alpha \langle X^*, B_\alpha\rangle \\ &= \lambda^* + \sum_{\alpha\neq 0} p_\alpha y_\alpha \;=\; \lambda^* + p(x).\end{aligned}$$

Therefore, as X^* is optimal, $-X^*(1,1) = -\lambda^* = p^*$, and we obtain

$$\sum_{i=1}^r \lambda_i q_i(x)^2 \;=\; p(x) - p^*,$$

the desired result. □

From the proof of Theorem 3.2, it is obvious that if $\min \mathbb{Q} = \min \mathbb{P}$, then x^* is a root of each polynomial $q_i(x)$, where $X^* = \sum_{i=1}^r q_i q_i'$ at an optimal solution X^* of

$\mathbb{Q}^*$. When $p(x) - p^*$ is a sum of squares, solving the dual LMI problem $\mathbb{Q}^*$ provides the q_i polynomials of such a decomposition. As a corollary, we obtain the following.

COROLLARY 3.3. *Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a real-valued polynomial of degree $2m$. Assume that $\mathbb{Q}^*$ has a feasible solution. Then,*

$$p(x) - p^* = \sum_{i=1}^{r} q_i(x)^2 - [\min \mathbb{P} - \inf \mathbb{Q}] \tag{3.8}$$

for some real-valued polynomials $q_i(x) : \mathbb{R}^n \to \mathbb{R}$ of degree at most m, $i = 1, 2, \ldots, r$.

The proof is the same as the proof of Theorem 3.2(ii), except now we may not have $\min \mathbb{Q} = \min \mathbb{P}$, but instead $\inf \mathbb{Q} \leq \min \mathbb{P}$. Hence, $\inf \mathbb{Q}$ always provides a lower bound on p^*.

Corollary 3.3 states that one may always write $p(x) - p^*$ as a sum of squares of polynomials up to some constant whenever $\mathbb{Q}^*$ has a feasible solution.

One may ask whether a nonnegative polynomial can be "approached" by polynomials that are the sum of squares. The answer is yes (see Remark 3.6 below).

Example 1. Consider the polynomial $p(x) : \mathbb{R}^2 \to \mathbb{R}$

$$(x_1, x_2) \mapsto (x_1^2 + 1)^2 + (x_2^2 + 1)^2 + (x_1 + x_2 + 1)^2.$$

It is not obvious a priori that with x^* a global minimizer, $p(x) - p^*$ is a sum of squares. Solving $\mathbb{Q}$ yields a minimum value of -0.4926, and from the solution y, one may check that

$$y = (x_1^*, x_2^*, (x_1^*)^2, x_1^* x_2^*, (x_2^*)^2, \ldots, (x_1^*)^4, \ldots, (x_2^*)^4),$$

with $x_1^* = x_2^* = -0.2428$, is a good approximation of a global minimizer of $\mathbb{P}$ since the gradient vector

$$\frac{\partial p(x_{1,2}^*)}{\partial x_1} = \frac{\partial p(x_1^*, x_2^*)}{\partial x_2} = 4 * 10^{-9}.$$

Solving $\mathbb{Q}^*$ yields

$$X^* \approx \begin{bmatrix} 0.4926 & 1.0000 & 1.0000 & -0.0196 & -0.0316 & -0.0668 \\ 1.0000 & 3.0392 & 1.0316 & 0 & -0.0276 & -0.1666 \\ 1.0000 & 1.0316 & 3.1335 & 0.0276 & 0.1666 & 0 \\ -0.0196 & 0 & 0.0276 & 1.0000 & 0 & -0.5539 \\ -0.0316 & -0.0276 & 0.1666 & 0 & 1.1078 & 0 \\ -0.0668 & -0.1666 & 0 & -0.5539 & 0 & 1.0000 \end{bmatrix}$$

with eigenvalues

$$[1.0899,\ 1.5414,\ 2.0885,\ 0.4410,\ 0.0000,\ 4.6123]$$

and corresponding eigenvectors

$$\begin{bmatrix} 0.0579 & 0.0144 & 0.0163 & 0.0675 & 0.9414 & -0.3246 \\ -0.0972 & 0.1118 & 0.6999 & -0.0759 & -0.2286 & -0.6559 \\ 0.1010 & -0.0657 & -0.6861 & 0.0114 & -0.2286 & -0.6800 \\ 0.0224 & -0.7105 & 0.0503 & -0.6993 & 0.0555 & -0.0092 \\ -0.9882 & -0.0334 & -0.1368 & -0.0028 & 0.0555 & -0.0242 \\ -0.0006 & 0.6907 & -0.1337 & -0.7075 & 0.0555 & 0.0377 \end{bmatrix}.$$

Example 2. Consider the polynomial $p(x) : \mathbb{R}^2 \to \mathbb{R}$

$$(x_1, x_2) \mapsto (x_1^2 + 1)^2 + (x_2^2 + 1)^2 - 2(x_1 + x_2 + 1)^2.$$

Solving $\mathbb{Q}$ yields an optimal value $p^* \approx -11.4581$ and an optimal solution

$$(x_1^*, x_2^*) = (1.3247, 1.3247)$$

with corresponding gradient

$$\frac{\partial p(x_1^*, x_2^*)}{\partial x_1} = \frac{\partial p(x_1^*, x_2^*)}{\partial x_2} = -8.7 * 10^{-6}.$$

Solving $\mathbb{Q}^*$ yields an optimal solution

$$X^* \approx \begin{bmatrix} 11.4581 & -2.0000 & -2.0000 & -1.0582 & -1.1539 & -1.2977 \\ -2.0000 & 2.1164 & -0.8461 & 0 & -0.0868 & 0.2676 \\ -2.0000 & -0.8461 & 2.5953 & 0.0868 & -0.2676 & 0 \\ -1.0582 & 0 & 0.0868 & 1.0000 & 0 & -0.4625 \\ -1.1539 & -0.0868 & -0.2676 & 0 & 0.9250 & 0 \\ -1.2977 & 0.2676 & 0 & -0.4625 & 0 & 1.0000 \end{bmatrix}.$$

The eigenvalues of X^* are

$$[1.2719,\ 1.4719,\ 0.5593,\ 0.0000,\ 3.2582,\ 12.5336]$$

with corresponding eigenvectors (in columns below)

$$\begin{bmatrix} 0.0854 & -0.0552 & -0.0615 & 0.2697 & 0.0177 & -0.9554 \\ 0.5477 & -0.1658 & -0.3615 & 0.3573 & -0.6204 & 0.1712 \\ 0.3274 & -0.2171 & -0.2965 & 0.3573 & 0.7740 & 0.1760 \\ 0.2384 & 0.6906 & 0.4831 & 0.4733 & 0.0403 & 0.0847 \\ -0.6736 & 0.2490 & -0.4967 & 0.4733 & -0.0744 & 0.0896 \\ -0.2740 & -0.6191 & 0.5454 & 0.4733 & -0.0919 & 0.1081 \end{bmatrix}.$$

Hence,

$$\begin{aligned} & p(x_1, x_2) - p^* \\ \approx & 1.2719(0.0854 + 0.5477x_1 + 0.3274x_2 + 0.2384x_1^2 - 0.6736x_1x_2 - 0.2740x_2^2)^2 \\ & + 1.4719(-0.0552 - 0.1658x_1 - 0.2171x_2 + 0.6906x_1^2 + 0.2490x_1x_2 - 0.6191x_2^2)^2 \\ & + 0.5593(-0.0615 - 0.3615x_1 - 0.2965x_2 + 0.4831x_1^2 - 0.4967x_1x_2 + 0.5454x_2^2)^2 \\ & + 3.2582(0.0177 - 0.6204x_1 + 0.7740x_2 + 0.0403x_1^2 - 0.0744x_1x_2 - 0.0919x_2^2)^2 \\ & + 12.5336(-0.9554 + 0.1712x_1 + 0.1760x_2 + 0.0847x_1^2 + 0.0896x_1x_2 + 0.1081x_2^2)^2. \end{aligned}$$

General case. We now provide a result valid in the general case, that is, when $p(x) - p^*$ is not necessarily a sum of squares.

We first need to introduce some notation: Let $q(x) : \mathbb{R}^n \to \mathbb{R}$ be a real-valued polynomial of degree w with coefficient vector $q \in \mathbb{R}^{s(w)}$.

If the entry (i, j) of the matrix $M_m(y)$ is y_β, let $\beta(ij)$ denote the subscript β of y_β. Let $M_m(qy)$ be the matrix defined by

$$M_m(qy)(i, j) = \sum_\alpha q_\alpha y_{\{\beta(i,j)+\alpha\}}. \tag{3.9}$$

For instance, with

$$M_1(y) = \begin{bmatrix} 1 & y_{10} & y_{01} \\ y_{10} & y_{20} & y_{11} \\ y_{01} & y_{11} & y_{02} \end{bmatrix} \text{ and } x \mapsto q(x) = a - x_1^2 - x_2^2,$$

we obtain

$$M_1(qy) = \begin{bmatrix} a - y_{20} - y_{02}, & ay_{10} - y_{30} - y_{12}, & ay_{01} - y_{21} - y_{03} \\ ay_{10} - y_{30} - y_{12}, & ay_{20} - y_{40} - y_{22}, & ay_{11} - y_{31} - y_{13} \\ ay_{01} - y_{21} - y_{03}, & ay_{11} - y_{31} - y_{13}, & ay_{02} - y_{22} - y_{04} \end{bmatrix}.$$

Let $\{y_\alpha\}$ (with $y_0 = 1$) be an $s(2m)$-vector of moments up to order $2m$ of some probability measure μ_y on $\mathbb{R}^n$. Then, for every polynomial $v(x) : \mathbb{R}^n \to \mathbb{R}$, of degree at most m, with coefficient vector $v \in \mathbb{R}^{s(m)}$,

$$\langle v, M_m(qy)v \rangle = \int q(x)v(x)^2 \, \mu_y(dx). \tag{3.10}$$

Therefore, with $K_q := \{x \in \mathbb{R}^n \,|\, q(x) \geq 0\}$, if μ_y has its support contained in K_q, then it follows from (3.10) that $M_m(qy) \succeq 0$.

Suppose that we know in advance that a global minimizer x^* of $p(x)$ has norm less than a for some $a > 0$, that is, $p(x^*) = p^* = \min \mathbb{P}$ and $\|x^*\| \leq a$. Then, with $x \mapsto \theta(x) = a^2 - \|x\|^2$, we have $p(x) - p^* \geq 0$ on $K_a := \{\theta(x) \geq 0\}$.

We will use the fact that every polynomial $p(x)$, strictly positive on K_a, can be written

$$p(x) = \sum_{i=1}^{r_1} q_i(x)^2 + \theta(x) \sum_{j=1}^{r_2} t_j(x)^2,$$

for some polynomials $q_i(x), t_j(x)$, $i = 1, \ldots, r_1$, $j = 1, \ldots, r_2$ (see, e.g., Berg [1, p. 119]). For every $N \geq m$, let $\mathbb{Q}_a^N$ be the convex LMI problem

$$\mathbb{Q}_a^N \begin{cases} \inf\limits_y \sum\limits_\alpha p_\alpha y_\alpha, \\ \qquad M_N(y) \succeq 0, \\ \qquad M_{N-1}(\theta y) \succeq 0. \end{cases} \tag{3.11}$$

Writing $M_{N-1}(\theta y) = \sum_\alpha y_\alpha C_\alpha$, for appropriate matrices $\{C_\alpha\}$, the dual of $\mathbb{Q}_a^N$ is the convex LMI problem

$$(\mathbb{Q}_a^N)^* \begin{cases} \sup\limits_{X, Z \succeq 0} -X(1,1) - a^2 Z(1,1), \\ \qquad \langle X, B_\alpha \rangle + \langle Z, C_\alpha \rangle = p_\alpha, \ \alpha \neq 0. \end{cases} \tag{3.12}$$

Now we have the following theorem.

THEOREM 3.4. *Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a $2m$-degree polynomial as in (2.2) with global minimum $p^* = \min \mathbb{P}$ and such that $\|x^*\| \leq a$ for some $a > 0$ at some global minimizer x^*. Then*

(a) *as $N \to \infty$, one has*

$$\inf \mathbb{Q}_a^N \uparrow p^*. \tag{3.13}$$

Moreover, for N sufficiently large, there is no duality gap between $\mathbb{Q}_a^N$ and its dual $(\mathbb{Q}_a^N)^*$, and $(\mathbb{Q}_a^N)^*$ is solvable.

(b) $\min \mathbb{Q}_a^N = p^*$ *if and only if*

$$p(x) - p^* = \sum_{i=1}^{r_1} q_i(x)^2 + \theta(x) \sum_{j=1}^{r_2} t_j(x)^2 \tag{3.14}$$

for some polynomials $q_i(x)$, $i = 1, \ldots, r_1$, of degree at most N, and some polynomials $t_j(x)$, $j = 1, \ldots, r_2$, of degree at most $N-1$. In this case, the vector

$$y^* := (x_1^*, \ldots, x_n^*, (x_1^*)^2, x_1^* x_2^*, \ldots, (x_1^*)^{2N}, \ldots, (x_n^*)^{2N}) \tag{3.15}$$

is a minimizer of $\mathbb{Q}_a^N$. In addition, $\max(\mathbb{Q}_a^N)^* = \min \mathbb{Q}_a^N$ and for every optimal solution (X^*, Z^*) of $(\mathbb{Q}_a^N)^*$,

$$p(x) - p^* = \sum_{i=1}^{r_1} \lambda_i q_i(x)^2 + \theta(x) \sum_{j=1}^{r_2} \gamma_j t_j(x)^2, \tag{3.16}$$

where the vectors of coefficients of the polynomials $q_i(x)$,$t_j(x)$ are the eigenvectors of X^* and Z^* with respective eigenvalues λ_i, γ_j.

Proof. (a) From $x^* \in K_a$, and with

$$y^* := (x_1^*, \ldots, (x_1^*)^{2N}, \ldots, (x_n^*)^{2N}),$$

it follows that $M_N(y^*), M_{N-1}(\theta y^*) \succeq 0$ so that y^* is admissible for $\mathbb{Q}_a^N$ and thus $\inf \mathbb{Q}_a^N \leq p^*$.

Now, fix $\epsilon > 0$ arbitrary. Then, $p(x) - (p^* - \epsilon) > 0$ and, therefore, there is some N_0 such that

$$p(x) - p^* + \epsilon = \sum_{i=1}^{r_1} q_i(x)^2 + \theta(x) \sum_{j=1}^{r_2} t_j(x)^2$$

for some polynomials $q_i(x)$, $i = 1, \ldots, r_1$, of degree at most N_0, and some polynomials $t_j(x)$, $j = 1, \ldots, r_2$, of degree at most $N_0 - 1$ (see Berg [1, p. 119]).

Let $q_i \in \mathbb{R}^{s(N_0)}, t_j \in \mathbb{R}^{s(N_0-1)}$ be the vector of coefficients of the polynomials $q_i(x), t_j(x)$, respectively, and let

$$X := \sum_{i=1}^{r_1} q_i q_i', \quad Z := \sum_{j=1}^{r_2} t_j t_j'$$

so that $X, Z \succeq 0$. It is immediate to check that (X, Z) is admissible for $(\mathbb{Q}_a^{N_0})^*$ with value $-X(1,1) - a^2 Z(1,1) = (p^* - \epsilon)$. From weak duality it follows that $\inf \mathbb{Q}_a^{N_0} \geq -(X(1,1) + a^2 Z(1,1)) = p^* - \epsilon$, and the desired result follows from

$$p^* - \epsilon \leq \inf \mathbb{Q}_a^{N_0} \leq p^*.$$

We next prove that there is no duality gap between $\mathbb{Q}_a^N$ and its dual $(\mathbb{Q}_a^N)^*$ as soon as $N \geq N_0$. Indeed, let μ be a probability measure with uniform distribution in K_a. Let $y_\mu = \{y_\alpha\}$ with

$$y_\alpha := \int x^\alpha \mu(dx)$$

for all combinations $(\alpha_1, \ldots, \alpha_n) = r$, $r = 1, \ldots, N$. All the y_α's are well defined since μ has its support contained in the compact set K_a. From (2.4),

$$\langle q, M_N(y_\mu)q\rangle \;=\; \int q(x)^2\mu(dx) \;>\; 0 \text{ whenever } 0 \neq q \in \mathbb{R}^{s(N)},$$

and from (3.10),

$$\langle q, M_{N-1}(\theta y_\mu)q\rangle \;=\; \int \theta(x)q(x)^2\mu(dx) \;>\; 0 \text{ whenever } 0 \neq q \in \mathbb{R}^{s(N-1)}.$$

It follows that $M_N(y_\mu), M_{N-1}(\theta y_\mu) \succ 0$; that is, y_μ is (strictly) admissible for $\mathbb{Q}_a^N$ and, as $(\mathbb{Q}_a^N)^*$ has an admissible solution, from a standard result in convex optimization, there is no duality gap between $(\mathbb{Q}_a^N)^*$ and $\mathbb{Q}_a^N$. In addition, $(\mathbb{Q}_a^N)^*$ is solvable, that is, $\sup(\mathbb{Q}_a^N)^* = \max(\mathbb{Q}_a^N)^*$.

That $\inf \mathbb{Q}_a^N \uparrow p^*$ follows from the fact that, obviously, $M_N(y) \succeq 0$ implies $M_{N'}(y) \succeq 0$ for every $N \geq N'$ (since $M_{N'}(y)$ is a submatrix of $M_N(y)$) and similarly for $M_{N-1}(\theta y)$. Therefore, for every solution y of $\mathbb{Q}_a^N$, the adequate truncated vector y' is admissible for $\mathbb{Q}_a^{N'}$, whenever $N' \leq N$, with the same value. Hence, $\inf \mathbb{Q}_a^N \geq \inf \mathbb{Q}_a^{N'}$ whenever $N \geq N'$.

(b) *Only if part.* That y^* in (3.15) is a minimizer of $\mathbb{Q}_a^N$ is obvious. From (a) we know that there is no duality gap between $\mathbb{Q}_a^N$ and $(\mathbb{Q}_a^N)^*$ for N sufficiently large, and $(\mathbb{Q}_a^N)^*$ is solvable. Therefore, for N sufficiently large, let (X^*, Z^*) be an optimal solution of $(\mathbb{Q}_a^N)^*$, guaranteed to exist.

As $X^* \succeq 0, Z^* \succeq 0$, write

$$X^* \;=\; \sum_{i=1}^{r_1} \lambda_i q_i q_i'; \; Z^* \;=\; \sum_{j=1}^{r_2} \gamma_j t_j t_j',$$

where the q_i's (respectively, the t_j's) are the eigenvectors of X^* (respectively, Z^*), with eigenvalues λ_i (respectively, γ_j). With

$$y = (x_1, \ldots, x_n, \ldots, (x_1)^{2N}, \ldots, (x_n)^{2N}),$$

we have

$$\begin{aligned}\langle X^*, M_N(y)\rangle + \langle Z^*, M_{N-1}(\theta y)\rangle &= X^*(1,1) + a^2 Z^*(1,1) \\ &\quad + \sum_{\alpha \neq 0} y_\alpha [\langle X^*, B_\alpha\rangle + \langle Z^*, C_\alpha\rangle] \\ &= X^*(1,1) + a^2 Z^*(1,1) + p(x) \\ &= p(x) - p^*,\end{aligned}$$

where the last equality follows from

$$\min \mathbb{Q}_a^N \;=\; p^* \;=\; \max(\mathbb{Q}_a^N)^* \;=\; -X^*(1,1) - a^2 Z^*(1,1).$$

On the other hand,

$$\langle X^*, M_N(y)\rangle \;=\; \sum_{i=1}^{r_1} \lambda_i \langle q_i, M_N(y) q_i\rangle \;=\; \sum_{i=1}^{r_1} \lambda_i q_i(x)^2$$

and

$$\langle Z^*, M_{N-1}(\theta y)\rangle \;=\; \sum_{j=1}^{r_2} \gamma_j \langle t_j, M_{N-1}(\theta y) t_j\rangle \;=\; \theta(x) \sum_{j=1}^{r_2} \gamma_j t_j(x)^2.$$

Therefore,

$$p(x) - p^* \;=\; \sum_{i=1}^{r_1} \lambda_i q_i(x)^2 + \theta(x) \sum_{j=1}^{r_2} \gamma_j t_j(x)^2,$$

the desired result.

If part. If (3.14) holds, then one proves as in (a) (but with $\epsilon = 0$) that $\sup(\mathbb{Q}_a^N)^* \geq p^*$ so that, in fact, $\max(\mathbb{Q}_a^N)^* = p^* = \min \mathbb{Q}_a^N$ for N sufficiently large. □

Thus, one may approach the global optimal value p^* as closely as desired by solving a finite number of convex LMI problems $\mathbb{Q}_a^N$, and if $p(x) - p^*$ (which is only nonnegative and not strictly positive) can be written as a weighted sum of squares, one obtains the exact optimal value by solving a finite number of problems $\mathbb{Q}_a^N$. However, from a computational point of view, the remark is irrelevant, especially if one solves $\mathbb{Q}_a^N$ with an interior point method.

REMARK 3.5. *Theorem* 3.4 *also applies for the global minimization of* $p(x)$ *on* K_a *if* K_a *does not contain any global minimizer of* $p(x)$ *on* $\mathbb{R}^n$. *It suffices to replace* p^* *with* $\delta^* := \min_{x \in K_a} p(x)$.

Example 3. Consider the polynomial $p(x) : \mathbb{R}^2 \to \mathbb{R}$,

$$x \;\mapsto\; p(x) \;:=\; x_1^2 x_2^2 (x_1^2 + x_2^2 - 1).$$

$1 + p(x)$ is positive but is not a sum of squares (see Berg [1]). A global minimizer of $p(x)$ is $x_1^2 = x_2^2 = 1/3$ with optimal value $p^* = -1/27$.

Solving the LMI problem $\mathbb{Q}$ in Theorem 3.2 yields an approximated optimal value of $-33.157352 < p^*$. With K_1 (the unit ball), solving $\mathbb{Q}_1^3$, one obtains *exactly* the global minimum p^* and a global minimizer x^*. In fact, as $p(x)$ contains only even powers of x_1 and x_2, y^* is the convex combination of $0.5y_1^* + 0.5y_2^*$ with y_1^*, y_2^* being the sequences of moments corresponding to the Dirac measures at $x_1^* = -\sqrt{1/3}$ and at $x_2^* = \sqrt{1/3}$, respectively.

This shows that in some cases one will obtain the exact global optimal value with few trials. In the present example, $p(x)$ is of degree 6 and we do not need to increase the degree to get the weighted sum of squares (3.14) when it exists; that is, $q_i(x)^2, t_j(x)^2$ in (3.14) are of degree at most 6, as $p(x)$.

REMARK 3.6. *One may ask whether a nonnegative polynomial can be "approached" by polynomials that are sums of squares. An answer is given in Berg* [1]. *Indeed, let* $\mathcal{A}$ *be the space of real-valued polynomials* $p(x) : \mathbb{R}^n \to \mathbb{R}$ *equipped with the norm* $\|p(x)\|_{\mathcal{A}} = \|p\|$ *with* p *the (finite-dimensional) vector of the coefficients of* $p(x)$ *(for instance, in the (extended) basis* (2.1)). *Then, the cone* Σ *of polynomials that are sums of squares is dense (for the norm* $\|.\|_{\mathcal{A}}$*) in the set of polynomials that are nonnegative on* $[-1, 1]^n$.

For instance, as we know that $p(x) - p^*$ is positive in $[-1, 1]^2$, for the polynomial $p(x)$ in the above example we may try to solve the LMI problem $\mathbb{Q}$ with $M_4(y) \succeq 0$ instead of $M_3(y) \succeq 0$ and perturbate $p(x)$ by adding the terms $0.01(x_1^8 + x_2^8)$ whose effect in $[-1, 1]^2$ is negligible. Solving $\mathbb{Q}$ for $\tilde{p}(x) = p(x) + 0.01(x_1^8 + x_2^8)$ yields the optimal value $\tilde{p}^* = -0.036792$ to compare with -0.037037 and a global minimizer $(\tilde{x}_1^*)^2 = (\tilde{x}_2^*)^2 = 0.3319$ to compare with $(x_1^*)^2 = (x_2^*)^2 = 1/3$. In this case, $\tilde{p}(x) - \tilde{p}^*$ is a sum of squares. However, the smaller perturbation $0.001(x_1^8 + x_2^8)$ does not work.

4. Constrained case. We now consider the constrained case, that is,

$$\mathbb{P}_K \;\mapsto\; p_K^* \;:=\; \min_{x\in K} p(x), \tag{4.1}$$

where

- $p(x) : \mathbb{R}^n \to \mathbb{R}$ is a real-valued polynomial of degree at most m.
- K is a *compact* set defined by polynomials inequalities $g_i(x) \geq 0$ with $g_i(x) : \mathbb{R}^n \to \mathbb{R}$ being a real-valued polynomial of degree at most w_i, $i = 1, 2, \ldots, r$.

Concerning the semi-algebraic compact set K, we make the following assumption.

ASSUMPTION 4.1. *The set K is compact and there exists a real-valued polynomial $u(x) : \mathbb{R}^n \to \mathbb{R}$ such that $\{u(x) \geq 0\}$ is compact, and*

$$u(x) \;=\; u_0(x) + \sum_{k=1}^{r} g_i(x)u_i(x) \;\; \textit{for all } x \in \mathbb{R}^n, \tag{4.2}$$

where the polynomials $u_i(x)$ are all sums of squares, $i = 0, \ldots, r$.

Assumption 4.1 is satisfied in many cases, for instance, if there is one polynomial $g_i(x)$ such that $\{g_i(x) \geq 0\}$ is compact (take $u_k(x) \equiv 0$ except $u_i(x) \equiv 1$ in (4.2)). It is also satisfied if all the p_i's are linear (see Jacobi and Prestel [9]) and for 0-1 programs, that is, when K includes the inequalities $x_i^2 \geq x_i$ and $x_i \geq x_i^2$ for all i. Therefore, one way to ensure that Assumption 4.1 holds is to add to the definition of K the extra constraint $g_{r+1}(x) = a^2 - \|x\|^2 \geq 0$ for some a sufficiently large.

It is important to emphasize that we do *not* assume that K is convex (it may even be disconnected). We will use the fact that whenever Assumption 4.1 holds, every polynomial $p(x)$, strictly positive on K, can be written

$$p(x) \;=\; q(x) + \sum_{k=1}^{r} g_k(x)t_k(x) \;\; \text{for all } x \in \mathbb{R}^n \tag{4.3}$$

for some polynomials $q(x)$, $t_k(x)$, $k = 1, \ldots, r$, that are all sums of squares (see, e.g., Lemma 4.1 in Putinar [15] and also Jacobi [8]). In fact, Assumption 4.1 is an *if and only if* condition for (4.3) to hold. Of course, one does not know in advance the degrees of these polynomials.

As we did for $\theta(x)$ in the previous section, for every $i = 1, \ldots, r$, let $M_m(g_i y)$ be the matrices defined as in (3.9), with $g_i(x)$ in lieu of $\theta(x)$. Therefore, if y is an $s(2m)$ moment vector for some probability measure μ on $\mathbb{R}^n$, then for every $i = 1, 2, \ldots, r$, and every polynomial $q(x)$ of degree at most m,

$$\langle q(x), q(x)\rangle_{g_i y} \;:=\; \langle q, M_m(g_i y)q\rangle \;=\; \int g_i(x)q(x)^2\,\mu(dx) \tag{4.4}$$

so that, if μ has its support contained in K, then $M_m(g_i y) \succeq 0$ for all $i = 1, 2, \ldots, r$.

Let $\tilde{w}_i := \lceil w_i/2 \rceil$ be the smallest integer larger than $w_i/2$, and with $N \geq \lceil m/2 \rceil$ and $N \geq \max_i \tilde{w}_i$, consider the convex LMI problem

$$\mathbb{Q}_K^N \left\{ \begin{array}{lrcl} \inf\limits_y & \sum\limits_\alpha p_\alpha y_\alpha, & & \\ & M_N(y) & \succeq & 0, \\ & M_{N-\tilde{w}_i}(g_i y) & \succeq & 0, \;\; i = 1, \ldots, r. \end{array} \right. \tag{4.5}$$

Writing $M_{N-\tilde{w}_i}(g_i y) = \sum_\alpha C_{i\alpha} y_\alpha$, for appropriate symmetric matrices $\{C_{i\alpha}\}$, the dual of $\mathbb{Q}_K^N$ is the convex LMI problem

$$(\mathbb{Q}_K^N)^* \left\{ \begin{array}{rcl} \sup\limits_{X,Z_i} -X(1,1) - \sum\limits_{i=1}^r g_i(0) Z_i(1,1), & & \\ \langle X, B_\alpha\rangle + \sum\limits_{i=1}^r \langle Z_i, C_{i\alpha}\rangle & = & p_\alpha,\ \alpha \neq 0, \\ X, Z_i & \succeq & 0,\ \ i = 1,\ldots,r. \end{array} \right. \tag{4.6}$$

Now we have the following theorem.

THEOREM 4.2. *Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be an m-degree polynomial and K be the compact set $\{g_i(x) \geq 0, i = 1,\ldots,r\}$. Let Assumption 4.1 hold, and let $p_K^* := \min_{x\in K} p(x)$. Then*

(a) *as $N \to \infty$, one has*

$$\inf \mathbb{Q}_K^N \uparrow p_K^*. \tag{4.7}$$

Moreover, for N sufficiently large, there is no duality gap between $\mathbb{Q}_K^N$ and its dual $(\mathbb{Q}_K^N)^*$ if K has a nonempty interior.

(b) *if $p(x) - p_K^*$ has the representation (4.3), that is,*

$$p(x) - p_K^* = q(x) + \sum_{i=1}^r g_i(x) t_i(x) \tag{4.8}$$

for some polynomial $q(x)$ of degree at most $2N$, and some polynomials $t_i(x)$ of degree at most $2N - w_i$, $i = 1,\ldots,r$, all sums of squares, then $\min \mathbb{Q}_K^N = p_K^* = \max\ (\mathbb{Q}_K^N)^*$ and the vector

$$y^* := (x_1^*, \ldots, x_n^*, (x_1^*)^2, x_1^* x_2^*, \ldots, (x_1^*)^{2N}, \ldots, (x_n^*)^{2N}) \tag{4.9}$$

is a global minimizer of $\mathbb{Q}_K^N$. In addition, for every optimal solution $(X^*, Z_1^*, \ldots, Z_r^*)$ of $(\mathbb{Q}_K^N)^*$,

$$p(x) - p_K^* = \sum_{i=1}^{r_0} \lambda_i q_i(x)^2 + \sum_{i=1}^r g_i(x) \sum_{j=1}^{r_i} \gamma_{ij} t_{ij}(x)^2, \tag{4.10}$$

where the vectors of coefficients of the polynomials $q_i(x)$,$t_{ij}(x)$ are the eigenvectors of X^* and Z_i^* with respective eigenvalues λ_i, γ_{ij}.

Proof. The proof is similar to that of Theorem 3.4. For (a) it is immediate that $\inf \mathbb{Q}_K^N \leq p_K^*$ since the sequence of moments y^* constructed from a global minimizer x^* is admissible with value p_K^*. Also, as in Theorem 3.4, the sequence $\{\inf \mathbb{Q}_K^N\}$ is easily seen to be monotone nondecreasing in N. Moreover,

(i) given $\epsilon > 0$ arbitrary, the polynomial $p(x) - p_K^* + \epsilon$ is strictly positive on K and thus can be written as in (4.3) for some polynomial $q(x)$ of degree at most $2N$ and some polynomials $t_i(x)$, $i = 1,\ldots,r$, of degree at most $2N - w_i$, that are all sums of squares. As in Theorem 3.4, writing $q(x) = \sum_i q_i(x)^2$ and $t_i(x) = \sum_j t_{ij}(x)^2$, from the vector of coefficients q_i of $q_i(x)$ (and t_{ij} of $t_{ij}(x)$), one may construct matrices $X := \sum_i q_i q_i' \succeq 0$ and $Z_i := \sum_j t_{ij} t_{ij}' \succeq 0$, $i = 1,\ldots,r$, that are admissible for $(\mathbb{Q}_K^N)^*$, with value $-X(1,1) - \sum_i g_i(0) Z_i(1,1) = p_K^* - \epsilon$. Indeed, with

$$y = (x_1, \ldots, (x_1)^{2N}, \ldots, (x_n)^{2N}),$$

we obtain

$$\langle X, M_N(y)\rangle + \sum_{i=1}^{r} \langle Z_i, M_{N-\tilde{w}_i}(g_i y)\rangle \;=\; p(x) - p_K^* + \epsilon$$

so that, as x was arbitrary,

$$\langle X, B_\alpha\rangle + \sum_{i=1}^{r} \langle Z_i, C_{i\alpha}\rangle \;=\; p_\alpha \;\text{ for all } \alpha \neq 0,$$

and $X(1,1) + \sum_i g_i(0) Z_i(1,1) = -(p_K^* - \epsilon)$. Hence, $p_K^* - \epsilon \leq \sup(\mathbb{Q}_K^N)^* \leq \inf \mathbb{Q}_K^N \leq p_K^*$. As ϵ was arbitrary, (4.7) follows.

(ii) That there is no duality gap between $\mathbb{Q}_K^N$ and its dual $(\mathbb{Q}_K^N)^*$ follows from the fact that $\mathbb{Q}_K^N$ admits a strictly admissible solution. It suffices to consider a probability measure μ with uniform distribution on K. The vector y_μ of its moments up to order $2N$ is such that $M_N(y) \succ 0$ and $M_{N-\tilde{w}_i}(g_i y) \succ 0$. Therefore, as $(\mathbb{Q}_K^N)^*$ has a feasible solution, by a standard result in convexity, $\sup(\mathbb{Q}_K^N)^* = \max(\mathbb{Q}_K^N)^* = \inf \mathbb{Q}_K^N$.

The proof of (b) is also similar. If $p(x) - p_K^*$ has the representation (4.3), then from the polynomials $q(x)$ and $\{t_i(x)\}$, of degree at most $2N$ and $2N - w_i$, respectively, one may construct matrices $X, Z_i \succeq 0$, $i = 1, \ldots, r$, as in (a), such that $(X, Z_1, \ldots, Z_r)$ is an admissible solution for $(\mathbb{Q}_K^N)^*$, with value $-X(1,1) - \sum_i g_i(0) Z_i(1,1) = p_K^*$. From $p_K^* \leq \sup(\mathbb{Q}_K^N)^* \leq \inf \mathbb{Q}_K^N \leq p_K^*$, it follows immediately that $\max(\mathbb{Q}_K^N)^* = p_K^* = \min \mathbb{Q}_K^N$ and $(X, Z_1, \ldots, Z_k)$ is an optimal solution of $(\mathbb{Q}_K^N)^*$. The last statement is obtained in a similar fashion. □

One may also prove that if K has a nonempty interior, then (4.8) is also necessary for $\min Q_K^N = p_K^*$ to hold.

When K is compact and Assumption 4.1 does not hold, there is still a representation of polynomials, strictly positive on K (see Corollary 3 in Schmüdgen [16]). But, instead of being "linear" as in (4.3), there are product terms of the form $g_{i_1}(x) g_{i_2}(x) \ldots g_{i_l}(x)$ times a sum of squares of polynomials, with $i_1, \ldots, i_l \in \{1, \ldots, r\}$. It then suffices to include the corresponding constraints $M_m(g_{i_1} \ldots g_{i_l} y) \succeq 0$ in the LMI problem Q_K^N. However, the number of LMI constraints in $\mathbb{Q}_K^N$ grows exponentially with the number of constraints.

Example 4. Let $p(x) : \mathbb{R}^2 \to \mathbb{R}$ be the polynomial $x \mapsto p(x) := -a_1 x_1^2 - a_2 x_2^2$ and K be the compact set

$$K \;:=\; \{x \in \mathbb{R}^2 \,|\, x_1 + x_2 \leq b_1; a x_1 + y \leq b_2;\; x_1, x_2 \geq 0\}.$$

Whenever $a_i > 0$, $p(x)$ is concave so that we have a concave minimization problem and thus, some vertex of K is a global minimizer.

We have solved $\mathbb{Q}_K^2$ for several values of $a_i > 0, b_i$, $i = 1, 2$, and $a < 0$, each time providing a global minimizer exactly, so that

$$p(x) - p_K^* \;=\; q(x) + (b_1 - x_1 - x_2) t_1(x) + (b_2 - a x_1 - x_2) t_2(x) + x_1 t_3(x) + x_2 t_4(x)$$

for some 4-degree polynomial $q(x)$ and 2-degree polynomials $t_i(x)$, all sums of squares, $i = 1, \ldots, 4$.

Example 5. Let $p(x) : \mathbb{R}^2 \to \mathbb{R}$ be the concave polynomial $x \mapsto p(x) := -(x_1 - 1)^2 - (x_1 - x_2)^2 - (x_2 - 3)^2$ and

$$K := \{(x_1, x_2) \in \mathbb{R}^2 \,|\, 1 - (x_1 - 1)^2 \geq 0;\; 1 - (x_1 - x_2)^2 \geq 0;\; 1 - (x_2 - 3)^2 \geq 0\}.$$

The point $(1,2)$ is a global minimizer with optimal value -2. Solving $\mathbb{Q}_K^1$, that is, with $N=1$ and $\tilde{p}(x)=p(x)+10$ (since we eliminate the constant term -10), yields an optimal value of 7 instead of the desired value 8. On the other hand, solving $\mathbb{Q}_K^2$ yields an optimal value 8.00017 and an approximate global minimizer $(1.0043, 2.0006)$ (the error 0.00017 is likely due to the use of an interior point method in the LMI toolbox of MATLAB). Hence, with polynomials of degree 4 instead of 2, one obtains a good approximation of the correct value. Observe that there exist $\lambda_i = 1 \geq 0$ such that

$$p(x) + 3 \;=\; 0 + \sum_{i=1}^{r} \lambda_i g_i(x),$$

but $p(x) - p_K^*$ $(= p(x)+2)$ cannot be written that way.

Therefore, for the general nonconvex and quadratically constrained quadratic problem, $\mathbb{Q}_K^1$ may sometimes provide directly the exact global minimum, but in general a lower bound only (if $(\mathbb{Q}_K^1)^*$ has a feasible solution).

Solving some test problems. We have also solved the following test problems proposed in Floudas and Pardalos [6].

Problem 2.2 in [6].

$$\begin{cases} \min\limits_{x,y} \; p(x,y) \;:=\; c^T x - 0.5 x^T Q x + d^T y; \\ 6x_1 + 3x_2 + 3x_3 + 2x_4 + x_5 \;\leq\; 6.5; \\ 10x_1 + 10x_3 + y \;\leq\; 20; \\ 0 \;\leq\; y;\; 0 \;\leq\; x_i \;\leq\; 1,\; i = 1, \ldots, 5 \end{cases}$$

with $Q := I$ and $c = [-10.5, -7.5, -3.5, -2.5, -1.5]$. The optimal value -213 is obtained at the $\mathbb{Q}_K^2$ relaxation.

Problem 2.6 in [6].

$$\begin{cases} \min\limits_{x} \; p(x) \;:=\; c^T x - 0.5 x^T Q x; \\ Ax \;\leq\; b; \\ 0 \;\leq\; x_i \leq 1,\; i = 1, \ldots, 10 \end{cases}$$

with A being the matrix

$$\begin{bmatrix} -2 & -6 & -1 & 0 & -3 & -3 & -2 & -6 & -2 & -2 \\ 6 & -5 & 8 & -3 & 0 & 1 & 3 & 8 & 9 & -3 \\ -5 & 6 & 5 & 3 & 8 & -8 & 9 & 2 & 0 & -9 \\ 9 & 5 & 0 & -9 & 1 & -8 & 3 & -9 & -9 & -3 \\ -8 & 7 & -4 & -5 & -9 & 1 & -7 & -1 & 3 & -2 \end{bmatrix},$$

$c = [48, 42, 48, 45, 44, 41, 47, 42, 45, 46]$, $b = [-4, 22, -6, -23, -12]$, and $Q = 100I$. The optimal value -39 is obtained at the $\mathbb{Q}_K^2$ relaxation.

Problem 2.9 in [6].

$$\begin{cases} \max\limits_{x} \; p(x) \;:=\; \sum\limits_{i=1}^{9} x_i x_{i+1} + \sum\limits_{i=1}^{8} x_i x_{i+2} + x_1 x_7 + x_1 x_9 + x_1 x_{10} + x_2 x_{10} + x_4 x_7; \\ \sum_{i=1}^{10} x_i \;=\; 1;\; x_i \;\geq\; 0,\; i = 1, \ldots, 10. \end{cases}$$

The optimal value 0.375 is obtained at the $\mathbb{Q}_K^2$ relaxation.

Problem 3.3 in [6].

$$\begin{cases} \min\limits_x p(x) := -25(x_1-2)^2-(x_2-2)^2 \\ \qquad\qquad -(x_3-1)^2-(x_4-4)^2-(x_5-1)^2-(x_6-4)^2; \\ (x_3-3)^2+x_4 \geq 4;\ (x_5-3)^2+x_6 \geq 4; \\ x_1-3x_2 \leq 2;\ -x_1+x_2 \leq 2; \\ x_1+x_2 \leq 6;\ x_1+x_2 \geq 2; \\ 1 \leq x_3 \leq 5;\ 0 \leq x_4 \leq 6; \\ 1 \leq x_5 \leq 5;\ 0 \leq x_6 \leq 10; \\ x_1, x_2, \geq 0. \end{cases}$$

The optimal value -310 is obtained at the $\mathbb{Q}_K^2$ relaxation.

Problem 3.4 in [6].

$$\begin{cases} \min\limits_x p(x) := -2x_1+x_2-x_3; \\ x_1+x_2+x_3 \leq 4; \\ x_1 \leq 2;\ x_3 \leq 3;\ 3x_2+x_3 \leq 6; \\ x_i \geq 0,\ i=1,2,3; \\ x^TB^TBx-2r^TBx+\|r\|^2-0.25\|b-v\|^2 \geq 0 \end{cases}$$

with $r=[1.5,-0.5,-5]$ and

$$B=\begin{bmatrix} 0 & 0 & 1 \\ 0 & -1 & 0 \\ -2 & 1 & -1 \end{bmatrix};\ b=\begin{bmatrix} 3 \\ 0 \\ -4 \end{bmatrix};v=\begin{bmatrix} 0 \\ -1 \\ -6 \end{bmatrix}.$$

The optimal value -4 is obtained at the $\mathbb{Q}_K^4$ relaxation, whereas $\inf \mathbb{Q}_K^3 = -4.0685$.

Problem 4.6 in [6].

$$\begin{cases} \min\limits_x p(x) := -x_1-x_2; \\ x_2 \leq 2x_1^4-8x_1^3+8x_1^2+2; \\ x_2 \leq 4x_1^4-32x_1^3+88x_1^2-96x_1+36; \\ 0 \leq x_1 \leq 3;\ 0 \leq x_2 \leq 4. \end{cases}$$

The feasible set K is almost disconnected. The $\mathbb{Q}_K^4$ relaxation provides the optimal value -5.5079, the best value known so far, and therefore proves its global optimality.

Problem 4.7 in [6].

$$\begin{cases} \min\limits_x p(x) := -12x_1-7x_2+x_2^2; \\ -2x_1^4+2-x_2 = 0; \\ 0 \leq x_1 \leq 2;\ 0 \leq x_2 \leq 3. \end{cases}$$

The $\mathbb{Q}_K^5$ relaxation provides the optimal value -16.73889, the best known solution so far, and therefore proves its global optimality.

0-1 programming. It is also worth mentioning that constrained and unconstrained 0-1 programming problems can also be treated by solving convex LMI

problems $\mathbb{Q}_K^N$ since the integral constraints $x_i \in \{0,1\}$ can be written $x_i^2 \geq x_i; x_i^2 \leq x_i$ for all $i = 1,\ldots,n$. Therefore, the set

$$K_1 := \{x_i - x_i^2 \geq 0;\ x_i^2 - x_i \geq 0;\ i = 1,2,\ldots,n\} \tag{4.11}$$

(and its intersection with other additional polynomial constraints) is compact and Assumption 4.1 holds. However, there is no strictly admissible solution (or interior point). For illustration we have solved the elementary problem

$$\min\{-ax_1 - bx_2 \,|\, r - x_1 - cx_2 \geq 0; x_1, x_2 \geq 0;\ x_1, x_2 \in \{0,1\}\},$$

replacing the integrality constraints with $(x_1, x_2) \in K_1$, and K_1 as in (4.11).

Solving $\mathbb{Q}_K^2$ with $0 < a < b$, and several random values of c, yields the global optimal value in all cases and a global minimizer at one of the integral points $(0,1)$, $(1,0)$, and $(1,1)$ of K.

Also, the first experimental results on a sample of randomly generated MAX-CUT problems in $\mathbb{R}^n$ (that is, maximizing a quadratic form with no squared terms under the integrality constraints $x_i^2 = 1$ for all i) are encouraging. Indeed, the optimal value was obtained at the $\mathbb{Q}_K^2$ relaxation in all cases (see Lasserre [12]) for $n = 5$ and even $n = 10$.

5. Karush–Kuhn–Tucker global optimality conditions. In this section we still consider the problem $\mathbb{P}_K$ with a compact set K defined by polynomials inequalities $g_i(x) \geq 0$, $i = 1,\ldots,r$.

PROPOSITION 5.1. *Let $p_K^* := \min \mathbb{P}_K$ and assume that $x^* \in K$ is a global minimizer. If $p(x) - p_K^*$ can be written*

$$p(x) - p_K^* = \sum_{i=1}^{r_0} q_i(x)^2 + \sum_{k=1}^{r} g_k(x) \sum_{j=1}^{r_k} t_{kj}(x)^2, \qquad x \in \mathbb{R}^n, \tag{5.1}$$

for some polynomials $q_i(x)$, $t_{kj}(x)$, $i = 1,\ldots,r_0$, $k = 1,\ldots,r$, $j = 1,\ldots r_k$, then

$$0 = g_k(x^*) \left[\sum_{j=1}^{r_k} t_{kj}(x^*)^2\right], \qquad k = 1,\ldots,r. \tag{5.2}$$

$$\nabla p(x^*) = \sum_{k=1}^{r} \nabla g_k(x^*) \left[\sum_{j=1}^{r_k} t_{kj}(x^*)^2\right]. \tag{5.3}$$

Moreover, if there exist associated Lagrange Karush–Kuhn–Tucker multipliers $\lambda^* \in (\mathbb{R}^r)^+$ and if the gradients $\nabla g_k(x^*)$ are linearly independent, then

$$\sum_{j=1}^{r_k} t_{kj}(x^*)^2 = \lambda_k^*, \qquad k = 1,\ldots,r. \tag{5.4}$$

Proof. As x^* is a global minimizer of $\mathbb{P}_K$, it follows from $p(x^*) - p_K^* = 0$ and (5.1) that

$$0 = \sum_{i=1}^{r_0} q_i(x^*)^2 + \sum_{k=1}^{r} g_k(x^*) \sum_{j=1}^{r_k} t_{kj}(x^*)^2$$

so that

$$0 = q_i(x^*), \quad i = 1, \ldots, r_0, \quad \text{and } 0 = g_k(x^*) \sum_{j=1}^{r_k} t_{kj}(x^*)^2, \quad k = 1, \ldots, r.$$

Moreover, from (5.1) and in view of the above,

$$\begin{aligned} \nabla p(x^*) &= \sum_{k=1}^{r} \nabla g_k(x^*) \sum_{j=1}^{r_k} t_{kj}(x^*)^2 \\ &= \sum_{k=1}^{r} \lambda_k^* \nabla g_k(x^*) \end{aligned}$$

so that (5.4) follows from the linear independence of the $\nabla g_k(x^*)$. □

Hence, the representation (5.1) can be viewed as a global optimality condition of the Karush–Kuhn–Tucker type, where the multipliers are now nonnegative polynomials instead of nonnegative constants. In general, and in contrast to the usual (local) Karush–Kuhn–Tucker optimality conditions, the polynomial multiplier associated to a constraint $g_k(x) \geq 0$, nonactive at x^*, is *not* identically null, but vanishes at x^*.

If $p(x) - p_K^*$ cannot be written as (5.1), we still have that $p(x) - p_K^* + \epsilon$ can be written as (5.1) for every $\epsilon > 0$. Of course, the degrees of $q_i(x)$ and $t_{kj}(x)$ in (5.1) depend on ϵ, but we have

$$\lim_{\epsilon \to 0} \sum_{i=1}^{r_0(\epsilon)} q_i(x^*)^2 = 0 \quad \text{and} \quad \lim_{\epsilon \to 0} \sum_{j=1}^{r_k(\epsilon)} t_{kj}(x^*)^2 = 0$$

for every k such that $g_k(x^*) > 0$.

Convex quadratic programming. In the case where $p(x)$ is a convex quadratic polynomial and $g_k(x)$ are concave quadratic (or linear) polynomials, then, at a Karush–Kuhn–Tucker point (x^*, λ^*), and with the Lagrangian $L(x, \lambda^*) := p(x) - \sum_{k=1}^{r} \lambda_k^* g_k(x)$, we have

$$\begin{aligned} p(x) - p_K^* &= L(x, \lambda^*) - L(x^*, \lambda^*) + \sum_{k=1}^{r} \lambda_k^* g_k(x) \\ &= \frac{1}{2} \langle x - x^*, \nabla_{xx}^2 L(x^*, \lambda^*)(x - x^*) \rangle + \sum_{k=1}^{r} \lambda_k^* g_k(x) \\ &= \sum_{i=1}^{n} \alpha_i (\langle q_i, x - x^* \rangle)^2 + \sum_{k=1}^{r} \lambda_k^* g_k(x), \end{aligned}$$

where the q_i's are the eigenvectors of the psd form $\nabla_{xx}^2 L/2$ with respective eigenvalues α_i, $i = 1, \ldots, n$.

In this case, $p(x) - p_K^*$ can be written as (5.1) with $r_k = 1$ and $t_k(x) \equiv \sqrt{\lambda_k^*}$, and

$$q_i(x) = \sqrt{\alpha_i} \langle q_i, x - x^* \rangle, \qquad i = 1, \ldots, n.$$

That is, the polynomial $\sum_j t_{kj}(x)^2$ is just the constant λ_k^*. Therefore, we have the following theorem.

THEOREM 5.2. *Let $p(x) : \mathbb{R}^n \to \mathbb{R}$ be a convex quadratic polynomial and $K := \{g_i(x) \geq 0\}$ be a compact convex set defined by concave quadratic polynomials $g_i(x)$,*

$i = 1, \ldots, r$. Let x^* be a local (hence global) minimum of $\mathbb{P}_K$ with associated Karush–Kuhn–Tucker multipliers $\lambda^* \in (\mathbb{R}^r)^+$. Then,

$$y^* = (x_1^*, \ldots, x_n^*, (x_1^*)^2, \ldots, (x_n^*)^2)$$

is an optimal solution of the convex LMI problem

$$\mathbb{Q}_K^1 \left\{ \begin{array}{rcll} \min\limits_y \sum\limits_\alpha p_\alpha y_\alpha, & & & \\ \sum\limits_\alpha (g_i)_\alpha y_\alpha & \geq & -g_i(0), & i = 1, \ldots, r, \\ M_1(y) & \succeq & 0 & \end{array} \right.$$

and λ^* is an optimal solution of the dual LMI problem

$$(\mathbb{Q}_K^1)^* \left\{ \begin{array}{rcl} \max\limits_{X \succeq 0, \lambda \geq 0} -X(1,1) - \sum\limits_{i=1}^r \lambda_i g_i(0), & & \\ \langle X, B_\alpha \rangle + \sum\limits_{i=1}^r \lambda_i (g_i)_\alpha & = & p_\alpha, \ \ \alpha \neq 0. \end{array} \right.$$

Hence $(\mathbb{Q}_K^1)^*$, which is the well-known Shor's relaxation for nonconvex quadratic programs, is also the natural dual problem of the general convex quadratically constrained quadratic program. In fact, Theorem 5.2 is also true in the more general case where $\nabla^2_{xx} L(x^*, \lambda^*) \succeq 0$, which may also happen at a global minimizer of some nonconvex quadratic programs. For instance, the particular nonconvex quadratic problems investigated in [4] reduce to solving the single LMI problem Q_K^1.

The difference between the convex and nonconvex cases is that $\mathbb{Q}_K^1$ provides an exact solution in the convex case, whereas one has to solve an (often finite) sequence of problems $\{Q_K^N\}$ in the nonconvex case.

In the case where $p(x), g_i(x)$ are all linear, then the standard linear programming problem $\min_x \{c'x | Ax \geq b\}$ is just $\mathbb{Q}_K^0$, with $K := \{Ax \geq b\}$.

6. Conclusion. We have shown that the constrained and unconstrained global optimization problem with polynomials has a natural sequence of convex LMI relaxations $\{Q_K^N\}$ whose optimal values converge to the optimal value p_K^*. In some cases, the exact optimal value and a global minimizer are obtained at a particular relaxation. When this happens, every optimal solution of the dual LMI problem provides the Karush–Kuhn–Tucker polynomials in the representation of the polynomial $p(x) - p_K^*$, nonnegative on K, the analogues of the scalar multipliers in the standard Karush–Kuhn–Tucker (local) optimality condition. Identifying classes of problems, for which the dimension of the LMI problem $\mathbb{Q}_K^N$ to solve is known in advance, is a topic of further research.

Acknowledgments. The author wishes to thank Prof. Jim Renegar and an anonymous referee for their fruitful suggestions that helped improve the paper.

REFERENCES

[1] C. Berg, *The multidimensional moment problem and semi-groups*, in Moments in Mathematics, H.J. Landau, ed., AMS, Providence, RI, 1980, pp. 110–124.

[2] R.E. Curto and L.A. Fialkow, *Recursiveness, positivity, and truncated moment problems*, Houston J. Math., 17 (1991), pp. 603–635.

[3] R.E. Curto, *Flat Extensions of Positive Moment Matrices: Recursively Generated Relations*, Mem. Amer. Math. Soc., 136 (1998), no. 648.
[4] P.-D. Tao and L.-T.-H. An, *Lagrangian stability and global optimality in nonconvex quadratic minimization over Euclidian balls and spheres*, J. Convex Anal., 2 (1995), pp. 263–276.
[5] C. Ferrier, *Hilbert's* 17*th problem and best dual bounds in quadratic minimization*, Cybernet. Systems Anal., 34 (1998), pp. 696–709.
[6] C.A. Floudas and P.M. Pardalos, *A Collection of Test Problems for Constrained Global Optimization Algorithms*, Lecture Notes in Comput. Sci. 455, Springer-Verlag, Berlin, 1990.
[7] J.B. Hiriart-Urruty, *Conditions for global optimality*, in Handbook of Global Optimization, Kluwer, Dordrecht, 1995, pp. 1–24.
[8] T. Jacobi, *A representation theorem for certain partially ordered commutative rings*, Math. Z., to appear.
[9] T. Jacobi and A. Prestel, *On Special Representations of Strictly Positive Polynomials*, Tech. report, Konstanz University, Konstanz, Germany, January 2000.
[10] M. Krein and A. Nudel'man, *The Markov Moment Problem and Extremal Problems*, Ideas and Problems of P.L. Čebyšev and A.A. Markov and Their Further Development, Transl. Math. Monographs 50, AMS, Providence, RI, 1977.
[11] I.J. Landau, *Moments in Mathematics*, AMS, Providence, RI, 1987.
[12] J.B. Lasserre, *Optimality Conditions and LMI Relaxations for* 0-1 *Programs*, Tech. report 2000099, LAAS, Toulouse, France, March 2000, submitted.
[13] Y. Nesterov, *Squared functional systems and optimization problems*, in High Performance Optimization, H. Frenk, K. Roos, T. Terlaky, and S. Zhang, eds., Kluwer, Dordrecht, 2000.
[14] M. Putinar and F.-H. Vasilescu, *Solving moment problems by dimensional extension*, Ann. Math., 149 (1999), pp. 1087–1107.
[15] M. Putinar, *Positive polynomials on compact semi-algebraic sets*, Ind. Univ. Math. J., 42 (1993), pp. 969–984.
[16] K. Schmüdgen, *The K-moment problem for compact semi-algebraic sets*, Math. Ann., 289 (1991), pp. 203–206.
[17] N.Z. Shor, *Quadratic optimization problems*, Soviet J. Comput. Systems Sci., 25 (1987), pp. 1–11.
[18] N.Z. Shor, *Nondifferentiable Optimization and Polynomial Problems*, Kluwer, Dordrecht, 1998.
[19] B. Simon, *The classical moment problem as a self-adjoint finite difference operator*, Adv. Math., 137 (1998), pp. 82–203.
[20] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 818–836

CONDITION-MEASURE BOUNDS ON THE BEHAVIOR OF THE CENTRAL TRAJECTORY OF A SEMIDEFINITE PROGRAM*

MANUEL A. NUNEZ† AND ROBERT M. FREUND‡

Abstract. We present bounds on various quantities of interest regarding the central trajectory of a semidefinite program, where the bounds are functions of Renegar's condition number $\mathcal{C}(d)$ and other naturally occurring quantities such as the dimensions n and m. The condition number $\mathcal{C}(d)$ is defined in terms of the data instance $d = (A, b, C)$ for a semidefinite program; it is the inverse of a relative measure of the distance of the data instance to the set of ill-posed data instances, that is, data instances for which arbitrary perturbations would make the corresponding semidefinite program either feasible or infeasible. We provide upper and lower bounds on the solutions along the central trajectory, and upper bounds on changes in solutions and objective function values along the central trajectory when the data instance is perturbed and/or when the path parameter defining the central trajectory is changed. Based on these bounds, we prove that the solutions along the central trajectory grow at most linearly and at a rate proportional to the inverse of the distance to ill-posedness, and grow at least linearly and at a rate proportional to the inverse of $\mathcal{C}(d)^2$, as the trajectory approaches an optimal solution to the semidefinite program. Furthermore, the change in solutions and in objective function values along the central trajectory is at most linear in the size of the changes in the data. All such bounds involve polynomial functions of $\mathcal{C}(d)$, the size of the data, the distance to ill-posedness of the data, and the dimensions n and m of the semidefinite program.

Key words. semidefinite programming, perturbation of convex programs, central trajectory, interior-point methods, ill-posed problems, condition numbers

AMS subject classifications. 90C22, 90C31, 90C51

PII. S105262349936063X

1. Introduction.

We study various properties of the central trajectory of a semidefinite program $P(d) : \min\{C \bullet X : AX = b, X \succeq 0\}$. Here X and C are symmetric matrices; $C \bullet X$ denotes the trace inner product; A is a linear operator that maps symmetric matrices into $\Re^m$; $b \in \Re^m$; $X \succeq 0$ denotes that X is a symmetric positive semidefinite matrix; and the data for $P(d)$ is the array $d = (A, b, C)$. The central trajectory of $P(d)$ is the solution to the logarithmic barrier problem $P_\mu(d) : \min\{C \bullet X - \mu \ln \det(X) : AX = b, X \succeq 0\}$ as the trajectory parameter μ ranges over the interval $(0, \infty)$. Semidefinite programming (SDP) has been the focus of an enormous amount of research in the past decade and has proven to be a unifying model for many convex programming problems amenable to efficient solution by interior-point methods; see [1, 14, 27], and [2], among others. Our primary concern lies in bounding a variety of measures of the behavior of the central trajectory of $P(d)$ in terms of the condition number $\mathcal{C}(d)$ for $P(d)$ originally developed by Renegar.

By the condition number $\mathcal{C}(d)$ of the data $d = (A, b, C)$, we mean a scale-invariant positive measure depending on a given feasible data instance $d = (A, b, C)$ and with the following property: the condition number approaches infinity as the data approaches the set of data instances for which the problem $P(d)$, or its dual, becomes infeasible. In particular, we say that a data instance is *ill-posed* whenever its

*Received by the editors August 26, 1999; accepted for publication (in revised form) August 25, 2000; published electronically February 2, 2001.
http://www.siam.org/journals/siopt/11-3/36063.html
†Argyros School of Business and Economics, Chapman University, One University Drive, Orange, CA 92866 (mnunez@chapman.edu).
‡Sloan School of Management, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139-4307 (rfreund@mit.edu).

corresponding condition number is unbounded, that is, whenever the data instance is on the boundary of the set of primal-dual feasible data instances. This notion of conditioning (formally presented in subsection 2.3) was originally developed by Renegar in [17] within a more general convex programming context and has proven to be a key concept in the understanding of the continuous complexity of convex optimization methods (see, for instance, [4, 5, 6, 7, 8, 9, 16, 17, 18, 19, 20, 28, 29] among others). In this paper, we show the relevance of using this measure of conditioning in the analysis of the central trajectory of a semidefinite program of the form $P(d)$.

More specifically, in section 3 we present a variety of results that bound certain behavioral measures of the central trajectory of $P(d)$ in terms of the condition number $\mathcal{C}(d)$. In Theorem 3.1, we present upper bounds on the norms of solutions along the central trajectory. These bounds show that the solutions along the central trajectory grow at most linearly in the trajectory parameter μ and at a rate proportional to the inverse of the distance to ill-posedness of d. In Theorem 3.2, we present lower bounds on the values of the eigenvalues of solutions along the central trajectory. These bounds show that the eigenvalues of solutions along the central trajectory grow at least linearly in the trajectory parameter μ and at a rate proportional to $\mathcal{C}(d)^{-2}$.

In Theorem 3.3, we present bounds on changes in solutions along the central trajectory under simultaneous changes (perturbations) in the data d as well as changes in the trajectory parameter μ. These bounds are linear in the size of the data perturbation, quadratic in the inverse of the trajectory parameter, and are polynomial functions of the condition number and the dimensions m and n. Finally, in Theorem 3.4 we present similar bounds on the change in the optimal objective function values of the barrier problem along the central trajectory, under data and trajectory parameter perturbations. These bounds also are linear in the size of the data perturbation and in the size of the change in the trajectory parameter.

The use of continuous complexity theory in convex optimization, especially the theory developed by Renegar in [17, 18, 19, 20], has added significant insight into what makes certain convex optimization problems better or worse behaved (in terms of the deformation of problem characteristics under data perturbations) and consequently what makes certain convex optimization problems easier or harder to solve. We believe that the results presented in this paper contribute to this understanding by providing behavioral bounds on relevant aspects of the central trajectory of a semidefinite program.

The main results presented in this paper can be viewed as extensions of related results for the linear programming (LP) case presented in [15]. While some of the extensions contained herein are rather straightforward generalizations of analogous results for the LP case, other extensions have proven to be mathematically challenging to us and have necessitated (in their proofs) the development of further properties of matrices arising in the analysis of SDP; see Propositions 5.1 and 5.3, for example. One reason why we have found the extension from LP to SDP to be mathematically challenging has to do with the linear algebra of certain linear operators that arise in the study of the central trajectory. In the case of LP, we have $X\bar{X} = \bar{X}X$ whenever X and $\bar{X}$ are diagonal matrices. Matrix products like this appear when dealing with solutions x and $\bar{x}$ on the central trajectory of a data instance and its perturbation, respectively, thus streamlining the proofs of results in the LP case. When dealing with analogous solutions in the case of SDP, we no longer have the same commutative property of the matrix product, and so it is necessary to develop more complicated linear operators in the analysis of the central trajectory. Another difficulty in the extension from the

LP case to the SDP case is the lack of closedness of certain projections of the cone of positive semidefinite symmetric matrices. This lack of closedness prevents the use of "nice" LP properties such as strict complementarity of solutions.

Literature review. The study of perturbation theory and continuous complexity for convex programs in terms of the distance to ill-posedness and condition number of a given data instance was introduced in [17] by Renegar, who studied perturbations in a very general setting of the problem $(RLP) : \sup\{c^*x : Ax \leq b, x \geq 0, x \in \mathcal{X}\}$, where $\mathcal{X}$ and $\mathcal{Y}$ denote real normed vector spaces, $A : \mathcal{X} \to \mathcal{Y}$ is a continuous linear operator, $c^* : \mathcal{X} \to \Re$ is a continuous linear functional, and the inequalities $Ax \leq b$ and $x \geq 0$ are induced by any closed convex cones (linear or nonlinear) containing the origin in $\mathcal{X}$ and $\mathcal{Y}$, respectively. Previous to the paper of Renegar, many papers were written on perturbations of linear programs and systems of linear inequalities, but not in terms of the distance to ill-posedness (see, for instance, [12, 22, 23, 24, 25]).

Even though there is now a vast literature on SDP, there are only a few papers that study SDP in terms of some notion of a condition measure. Renegar [17] presents a bound on solutions to RLP, a bound on the change in optimal solutions when only the right-hand side vector b is perturbed, and a bound on changes in optimal objective function values when the whole data instance is perturbed. All of these bounds depend on the distance to ill-posedness of the given data instance. Because of their generality, these results also apply to the SDP case studied in this paper. Later, in [19] and [20] Renegar presented upper and lower bounds on the inverse of the Hessian matrix resulting from the application of Newton's method to the optimality conditions of RLP along the central trajectory. Again, these bounds depend on the distance to ill-posedness of the data instance, and they apply to the SDP case. These bounds are important because they can be used to study the continuous complexity of interior-point methods for solving semidefinite programs (see [19]) as well as the use of the conjugate gradient method in the solution of semidefinite programs (see [20]).

Nayakkankuppam and Overton in [13] study the conditioning of SDP in terms of a condition measure that depends on the inverse of a certain Jacobian matrix. This Jacobian matrix arises when applying Newton's method to find a root of a semidefinite system of equations equivalent to the system of equations that arise from the Karush–Kuhn–Tucker optimality conditions for $P(d)$. In particular, under the assumption that both $P(d)$ and its dual have unique optimal solutions, they present a bound on the change in the optimal solution to $P(d)$ and $P(d + \Delta d)$, where Δd is a data perturbation, in terms of their condition number. This bound is linear in the norm of Δd. Their analysis pertains to the study of the optimal solution of $P(d)$, but is not readily applicable to the central trajectory of a semidefinite program.

Sturm and Zhang [26] study the sensitivity of the central trajectory of a semidefinite program in terms of changes in the right-hand side of the constraints $AX = b$ in $P(d)$. Given a data instance $d = (A, b, C)$ of a semidefinite program, they consider data perturbations of the form $d + \Delta d = (A, b + \Delta b, C)$. Using this kind of perturbation, and under a primal and dual Slater condition as well as a strict complementarity condition, they show several properties of the derivatives of central trajectory solutions with respect to the right-hand side vector. The results presented herein differ from these results in that we use data perturbations of the form $d+\Delta d = (A+\Delta A, b+\Delta b, C+\Delta C)$, and we express our results in terms of the distance to ill-posedness of the data. As a result, our results are not as strong in terms of the size of bounds, but our results are more general, as they do not rely on any particular assumptions.

2. Notation, definitions, and preliminaries.

2.1. Space of symmetric matrices. Given two matrices U and V in $\Re^{n\times n}$, we define the inner product of U and V as $U \bullet V := \mathit{trace}(U^T V)$, where $\mathit{trace}(W) := \sum_{j=1}^{n} W_{jj}$ for all $W \in \Re^{n\times n}$. Given a matrix $U \in \Re^{n\times n}$, we denote by $\sigma(U) = (\sigma_1, \ldots, \sigma_n)^T$ the vector in $\Re^n$ whose components are the ordered singular values of U; that is, each σ_j is a singular value of U, and $0 \le \sigma_1 \le \cdots \le \sigma_n$. Furthermore, we denote by $\sigma_j(U)$ the jth singular value of U chosen according to the increasing order in $\sigma(U)$. In particular, $\sigma_1(U)$ and $\sigma_n(U)$ are the smallest and the largest singular values of U, respectively. We use the following norms in the space $\Re^{n\times n}$:

$$\|U\|_1 := \sum_{j=1}^{n} \sigma_j(U), \tag{1}$$

$$\|U\|_2 := \left(\sum_{j=1}^{n} \sigma_j(U)^2\right)^{1/2} = (U \bullet U)^{1/2}, \tag{2}$$

$$\|U\|_\infty := \max_{1\le j\le n} \sigma_j(U) = \sigma_n(U) \tag{3}$$

for all matrices $U \in \Re^{n\times n}$. The norm (1) is known as the *Ky Fan n*-norm or *trace* norm (see [3]); the norm (2) is known as the *Hilbert–Schmidt* norm or *Frobenius* norm and is induced by the inner product $\bullet$ defined above; (3) is the operator norm induced by the Euclidean norm on $\Re^n$. Notice that all these norms are *unitarily invariant* in that $\|U\| = \|PUQ\|$ for all unitary matrices P and Q in $\Re^{n\times n}$. We also have the following proposition that summarizes a few properties of these norms.

PROPOSITION 2.1. *For all $U, V \in \Re^{n\times n}$ we have*

(i) *Hölder's inequalities* (*see* [3])

$$|U \bullet V| \le \|U\|_\infty \|V\|_1, \tag{4}$$

$$|U \bullet V| \le \|U\|_2 \|V\|_2. \tag{5}$$

(ii) $\|UV\|_2 \le \|U\|_2 \|V\|_2$.

(iii) $\|U\|_\infty \le \|U\|_2 \le \sqrt{n}\|U\|_\infty$.

(iv) $\frac{1}{\sqrt{n}}\|U\|_1 \le \|U\|_2 \le \|U\|_1$.

From now on, whenever we use a Euclidean norm over any space, we will omit subscripts. Hence, $\|U\| := \|U\|_2$ for all U in $\Re^{n\times n}$.

Let $\mathcal{S}_n$ denote the subspace of $\Re^{n\times n}$ consisting of symmetric matrices. Given a matrix $U \in \mathcal{S}_n$, let $U \succeq 0$ denote that U is a positive semidefinite matrix, and let $U \succ 0$ denote that U is a positive definite matrix. We denote by $\mathcal{S}_n^+$ the set of positive semidefinite matrices in $\mathcal{S}_n$, that is, $\mathcal{S}_n^+ = \{U \in \mathcal{S}_n : U \succeq 0\}$. Observe that $\mathcal{S}_n^+$ is a closed convex pointed cone in $\mathcal{S}_n$ with nonempty interior given by $\{U \in \mathcal{S}_n : U \succ 0\}$. Furthermore, notice that the polar $(\mathcal{S}_n^+)^*$ of the cone $\mathcal{S}_n^+$ is the cone $\mathcal{S}_n^+$ itself. When $U \in \mathcal{S}_n$, we denote by $\lambda(U) := (\lambda_1, \ldots, \lambda_n)^T$ the vector in $\Re^n$ whose components are the real eigenvalues of U ordered as $0 \le |\lambda_1| \le \cdots \le |\lambda_n|$. Moreover, we denote by $\lambda_j(U)$ the jth eigenvalue of U chosen according to the order in $\lambda(U)$. In particular, notice that $\sigma_j(U) = |\lambda_j(U)|$ whenever $U \in \mathcal{S}_n$.

Given matrices $A_1, \ldots, A_m \in \mathcal{S}_n$, we define the linear operator $A = (A_1, \ldots, A_m)$ from $\mathcal{S}_n$ to $\Re^m$ as follows:

$$AX := (A_1 \bullet X, \ldots, A_m \bullet X)^T \tag{6}$$

for all $X \in \mathcal{S}_n$. We denote by $\mathcal{L}_{m,n}$ the space of linear operators from $\mathcal{S}_n$ to $\Re^m$ of the form (6). Given a linear operator $A = (A_1, \ldots, A_m) \in \mathcal{L}_{m,n}$, we define the *rank* of A as the dimension of the subspace generated by the matrices $A_1, \ldots, A_m$, that is, $\text{rank}(A) := \dim(\langle A_1, \ldots, A_m \rangle)$. We say that A has full-rank whenever $\text{rank}(A) = \min\{m, n(n-1)/2\}$. Throughout the remainder of this paper we will assume that $m \leq n(n-1)/2$, so that A has full-rank if and only if $\text{rank}(A) = m$. The corresponding adjoint transformation $A^T : \Re^m \mapsto \mathcal{S}_n$, associated with A, is given by

$$A^T[y] = \sum_{i=1}^{m} y_i A_i$$

for all $y \in \Re^m$. Furthermore, we endow the space $\mathcal{L}_{m,n}$ with the operator norm $\|A\| := \max\{\|AX\| : X \in \mathcal{S}_n, \|X\| \leq 1\}$ for all operators $A \in \mathcal{L}_{m,n}$. Finally, if we define the norm of the adjoint operator as $\|A^T\| := \max\{\|A^T[y]\| : y \in \Re^m, \|y\| \leq 1\}$, then it follows that $\|A^T\| = \|A\|$.

2.2. Data instance space. Consider the vector space $\mathcal{D}$ defined as $\mathcal{D} := \{d = (A, b, C) : A \in \mathcal{L}_{m,n}, b \in \Re^m, C \in \mathcal{S}_n\}$. We regard $\mathcal{D}$ as the space of data instances associated with the following pair of dual semidefinite programs:

$$\begin{aligned} P(d) &: \min \{C \bullet X : AX = b, X \succeq 0\}, \\ D(d) &: \max \left\{b^T y : A^T[y] + S = C, S \succeq 0\right\}, \end{aligned}$$

where $d = (A, b, C) \in \mathcal{D}$. To study the central trajectory of a data instance in $\mathcal{D}$, we use the functional $p(\cdot)$ defined as $p(U) = -\ln \det U$ for all $U \succ 0$. Notice that, as proven in [14], $p(\cdot)$ is a strictly convex n-normal barrier for the cone $\mathcal{S}_n^+$. Given a data instance $d = (A, b, C) \in \mathcal{D}$ and a fixed scalar $\mu > 0$, we study the following parametric family of dual logarithmic barrier problems associated with $P(d)$ and $D(d)$:

$$\begin{aligned} P_\mu(d) &: \min \{C \bullet X + \mu p(X) : AX = b, X \succ 0\}, \\ D_\mu(d) &: \max \left\{b^T y - \mu p(S) : A^T[y] + S = C, S \succ 0\right\}. \end{aligned}$$

Let $X(\mu)$ and $(y(\mu), S(\mu))$ denote the optimal solutions of $P_\mu(d)$ and $D_\mu(d)$, respectively (when they exist). Then the primal central trajectory is the set $\{X(\mu) : \mu > 0\}$ and is a smooth mapping from $(0, \infty)$ to S_n^+ [10, 27]. Similarly, the dual central trajectory is the set $\{(y(\mu), S(\mu)) : \mu > 0\}$ and is a smooth mapping from $(0, \infty)$ to $\Re^m \times S_n^+$.

We provide the data instance space $\mathcal{D}$ with the norm

$$\|d\| := \max\{\|A\|, \|b\|, \|C\|\} \tag{7}$$

for all data instances $d = (A, b, C) \in \mathcal{D}$. Using this norm, we denote by $B(d, \delta)$ the open ball $\{d + \Delta d \in \mathcal{D} : \|\Delta d\| < \delta\}$ in $\mathcal{D}$ centered at d and with radius $\delta > 0$ for all $d \in \mathcal{D}$.

2.3. Distance to ill-posedness. We are interested in studying data instances for which both programs $P(\cdot)$ and $D(\cdot)$ are feasible. Consequently, consider the following subset of the data set $\mathcal{D}$:

$$\mathcal{F} := \left\{(A, b, C) \in \mathcal{D} : b \in A(\mathcal{S}_n^+) \text{ and } C \in A^T[\Re^m] + \mathcal{S}_n^+\right\},$$

that is, the elements in $\mathcal{F}$ correspond to those data instances d in $\mathcal{D}$ for which $P(d)$ and $D(d)$ are feasible. The complement of $\mathcal{F}$, denoted by $\mathcal{F}^C$, is the set of data

instances $d = (A, b, C)$ for which $P(d)$ or $D(d)$ is infeasible. The boundary of $\mathcal{F}$, denoted by $\partial\mathcal{F}$, is called the set of *ill-posed* data instances. This is because arbitrarily small changes in a data instance $d \in \partial\mathcal{F}$ 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

$$\rho(d) := \inf\{\|\Delta d\| : d + \Delta d \in \partial\mathcal{F}\}$$

(see [17, 21, 18]), and so $\rho(d)$ is the distance of the data instance d to the set of ill-posed instances $\partial\mathcal{F}$. The *condition number* $\mathcal{C}(d)$ of the data instance d is defined as

$$\mathcal{C}(d) := \begin{cases} \frac{\|d\|}{\rho(d)} & \text{if } \rho(d) > 0, \\ \infty & \text{if } \rho(d) = 0. \end{cases}$$

The condition number $\mathcal{C}(d)$ can be viewed as a scale-invariant reciprocal of $\rho(d)$, as it is elementary to demonstrate that $\mathcal{C}(d) = \mathcal{C}(\alpha d)$ for any positive scalar α. Moreover, for $d = (A, b, C) \notin \partial\mathcal{F}$, let $\Delta d = (-A, -b, -C)$. Observe that $d + \Delta d = (0, 0, 0) \in \partial\mathcal{F}$ and, since $\partial\mathcal{F}$ is a closed set, we have $\|d\| = \|\Delta d\| \geq \rho(d) > 0$ so that $\mathcal{C}(d) \geq 1$. The value of $\mathcal{C}(d)$ is a measure of the relative conditioning of the data instance d.

As proven in [24], the interior of $\mathcal{F}$, denoted $\mathrm{Int}(\mathcal{F})$, is characterized as follows:

(8)
$$\mathrm{Int}(\mathcal{F}) = \left\{(A, b, C) \in \mathcal{D} : b \in A(\mathrm{Int}(\mathcal{S}_n^+)), C \in A^T[\Re^m] + \mathrm{Int}(\mathcal{S}_n^+), A \text{ has full-rank}\right\}.$$

In particular, notice that data instances in $\mathrm{Int}(\mathcal{F})$ correspond to data instances for which both $P_\mu(\cdot)$ and $D_\mu(\cdot)$ are feasible (for any $\mu > 0$). Also, observe that $d = (A, b, C) \in \mathcal{F}$ and $\rho(d) > 0$ if and only if $d \in \mathrm{Int}(\mathcal{F})$, and so, if and only if the characterization given in (8) holds for d. We will use this characterization of the interior of $\mathcal{F}$ throughout the remainder of this paper.

We will also make use of the following elementary sufficient certificates of infeasibilty.

PROPOSITION 2.2. *Let $d = (A, b, C) \in \mathcal{D}$.*

1. *If there exists $y \in \Re^m$ satisfying $A^T[y] \prec 0$ and $b^T y \geq 0$, then $P_\mu(d)$ is infeasible.*
2. *If there exists $X \in \mathcal{S}_n$ satisfying $AX = 0, X \succ 0$, and $C \bullet X \leq 0$, then $D_\mu(d)$ is infeasible.*

3. Statement of main results. For a given data instance $d \in \mathrm{Int}(\mathcal{F})$ and a scalar $\mu > 0$, we denote by $X(d, \mu)$ the unique optimal solution to $P_\mu(d)$ and by $(y(d, \mu), S(d, \mu))$ the unique optimal solution to $D_\mu(d)$. Furthermore, we use the following function of d and μ as a condition measure for the programs $P_\mu(d)$ and $D_\mu(d)$:

$$\mathcal{K}(d, \mu) := \mathcal{C}(d)^2 + \frac{\mu n}{\rho(d)}. \tag{9}$$

As with the condition number $\mathcal{C}(d)$, it is not difficult to show that $\mathcal{K}(d, \mu) \geq 1$ and $\mathcal{K}(d, \mu)$ is scale-invariant in the sense that $\mathcal{K}(\lambda d, \lambda\mu) = \mathcal{K}(d, \mu)$ for all $\lambda > 0$. The reason why we call $\mathcal{K}(d, \mu)$ a condition measure will become apparent from the theorems stated in this section.

Our first theorem concerns upper bounds on the optimal solutions to $P_\mu(d)$ and $D_\mu(d)$, respectively. The bounds are given in terms of the condition measure $\mathcal{K}(d,\mu)$ and the size of the data $\|d\|$. In particular, the theorem shows that the norm of the optimal primal solution along the central trajectory grows at most linearly in the barrier parameter μ and at a rate no larger than $n/\rho(d)$. The proof of this theorem is presented in section 4.

THEOREM 3.1. *Let $d \in \text{Int}(\mathcal{F})$ and μ be a positive scalar. Then*

$$\|X(d,\mu)\| \le \mathcal{K}(d,\mu), \tag{10}$$

$$\|y(d,\mu)\| \le \mathcal{K}(d,\mu), \tag{11}$$

$$\|S(d,\mu)\| \le 2\|d\|\mathcal{K}(d,\mu), \tag{12}$$

where $\mathcal{K}(d,\mu)$ is the condition measure defined in (9).

As the proof of Theorem 3.1 will show, there is a tighter bound on $\|X(d,\mu)\|$, namely, $\|X(d,\mu)\| \le \mathcal{M}(d,\mu)$, where

$$\mathcal{M}(d,\mu) := \begin{cases} \mathcal{C}(d) & \text{if } C \bullet X(d,\mu) \le 0, \\ \max\left\{\mathcal{C}(d), \frac{\mu n}{\rho(d)}\right\} & \text{if } 0 < C \bullet X(d,\mu) \le \mu n, \\ \mathcal{C}(d)^2 + \frac{\mu n}{\rho(d)} & \text{if } \mu n < C \bullet X(d,\mu), \end{cases} \tag{13}$$

whenever $d \in \text{Int}(\mathcal{F})$ and $\mu > 0$. Notice that because of the uniqueness of the optimal solution to $P_\mu(d)$ for $\mu > 0$, the condition measure $\mathcal{M}(d,\mu)$ is well defined. Also, observe that $\mathcal{M}(d,\mu)$ can always be bounded from above by $\mathcal{K}(d,\mu)$.

It is not difficult to create data instances for which the condition measure $\mathcal{M}(d,\mu)$ is a tight bound on $\|X(d,\mu)\|$. Even though the condition measure $\mathcal{M}(d,\mu)$ provides a tighter bound on $\|X(d,\mu)\|$ than $\mathcal{K}(d,\mu)$, we will use the condition measure $\mathcal{K}(d,\mu)$ for the remainder of this paper. This is because $\mathcal{K}(d,\mu)$ conveys the same general asymptotic behavior as $\mathcal{M}(d,\mu)$ and also because using $\mathcal{K}(d,\mu)$ simplifies most of the expressions in the theorems to follow. Similar remarks apply to the bounds on $\|y(d,\mu)\|$ and $\|S(d,\mu)\|$.

In particular, when $C = 0$, that is, when we are solving a semidefinite analytic center program, we obtain the following corollary.

COROLLARY 3.1. *Let $d = (A,b,C) \in \text{Int}(\mathcal{F})$ be such that $C = 0$ and μ be a positive scalar. Then*

$$\|X(d,\mu)\| \le \mathcal{C}(d).$$

The following theorem presents lower bounds on the eigenvalues of solutions along the primal and dual central trajectories. In particular, the lower bound on the eigenvalues of solutions along the primal central trajectory implies that the convergence of $X(d,\mu)$ to an optimal solution to $P(d)$, as μ goes to zero, is at least asymptotically linear in μ and at a rate of $1/(2\|d\|\mathcal{C}(d)^2)$.

THEOREM 3.2. *Let $d \in \text{Int}(\mathcal{F})$ and μ be a positive scalar. Then for all $j = 1,\dots,n$,*

$$\lambda_j(X(d,\mu)) \ge \frac{\mu}{2\|d\|\mathcal{K}(d,\mu)},$$

$$\lambda_j(S(d,\mu)) \ge \frac{\mu}{\mathcal{K}(d,\mu)}.$$

The proof of Theorem 3.2 is presented in section 4.

The next theorem concerns bounds on changes in optimal solutions to $P_\mu(d)$ and $D_\mu(d)$ as the data instance d and the parameter μ are perturbed. In particular, we present these bounds in terms of an asymptotically polynomial expression of the condition number $\mathcal{C}(d)$, the condition measure $\mathcal{K}(d,\mu)$, the size of the data $\|d\|$, the scalar μ, and the dimensions m and n. It is also important to notice the linear dependence of the bound on the size of the data perturbation $\|\Delta d\|$ and the parameter perturbation $|\Delta\mu|$.

THEOREM 3.3. *Let $d = (A, b, C)$ be a data instance in $\mathrm{Int}(\mathcal{F})$, μ be a positive scalar, $\Delta d = (\Delta A, \Delta b, \Delta C) \in \mathcal{D}$ be a data perturbation such that $\|\Delta d\| \le \rho(d)/3$, and $\Delta\mu$ be a scalar such that $|\Delta\mu| \le \mu/3$. Then,*

$$\begin{aligned}\|X(d+\Delta d,\mu+\Delta\mu)-X(d,\mu)\| \le \|\Delta d\|\ &\frac{640n\sqrt{m}\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu+\|d\|)}{\mu^2}\\ &+|\Delta\mu|\ \frac{6n\|d\|\mathcal{K}(d,\mu)^2}{\mu^2},\end{aligned}\tag{14}$$

$$\begin{aligned}\|y(d+\Delta d,\mu+\Delta\mu)-y(d,\mu)\| \le \|\Delta d\|\ &\frac{640\sqrt{m}\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu+\|d\|)}{\mu^2}\\ &+|\Delta\mu|\ \frac{32\sqrt{m}\|d\|\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2},\end{aligned}\tag{15}$$

$$\begin{aligned}\|S(d+\Delta d,\mu+\Delta\mu)-S(d,\mu)\| \le \|\Delta d\|\ &\frac{640\sqrt{m}\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu+\|d\|)^2}{\mu^2}\\ &+|\Delta\mu|\ \frac{32\sqrt{m}\|d\|^2\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2}.\end{aligned}\tag{16}$$

The proof of Theorem 3.3 is presented in section 5.

Finally, we present a theorem concerning changes in optimal objective function values of the program $P_\mu(d)$ as the data instance d and the parameter μ are perturbed. We denote by $z(d,\mu)$ the optimal objective function value of the program $P_\mu(d)$, namely, $z(d,\mu) := C \bullet X(d,\mu) + \mu p(X(d,\mu))$, where $X(d,\mu)$ is the optimal solution of $P_\mu(d)$.

THEOREM 3.4. *Let $d = (A, b, C)$ be a data instance in $\mathrm{Int}(\mathcal{F})$, μ be a positive scalar, $\Delta d = (\Delta A, \Delta b, \Delta C) \in \mathcal{D}$ be a data perturbation such that $\|\Delta d\| \le \rho(d)/3$, and $\Delta\mu$ be a scalar such that $|\Delta\mu| \le \mu/3$. Then*

$$\begin{aligned}|z(d+\Delta d,\mu+\Delta\mu)-z(d,\mu)| \le \|\Delta d\|\ &9\mathcal{K}(d,\mu)^2\\ &+|\Delta\mu|\ n\left(\ln 16+|\ln\mu|+|\ln\|d\||+\ln\mathcal{K}(d,\mu)\right).\end{aligned}$$

Notice that it follows from Theorem 3.4 that changes in objective function values of $P_\mu(d)$ are at most linear in the size of the perturbation of the data instance d and the parameter μ. As with Theorem 3.3, the bound is polynomial in terms of the condition measure $\mathcal{K}(d,\mu)$ and the size of the data instance d. Also observe that if $\Delta d = 0$, and we let $\Delta\mu$ go to zero, from the analytic properties of the central trajectory [10, 27] we obtain the following bound on the derivative of $z(\cdot)$ with respect to μ:

$$\left|\frac{\partial z(d,\mu)}{\partial\mu}\right| \le n\left(\ln 16+|\ln\mu|+|\ln\|d\||+\ln\mathcal{K}(d,\mu)\right).$$

We remark that it is not known to us if the bounds in Theorem 3.1, 3.2, 3.3, or 3.4 are tight (even up to a constant) for some data instances, but we suspect that they are not. However, our concern herein is not the exploration of the best possible bounds but rather the demonstration of bounds that are some polynomial function of appropriate natural behavior measures of a semidefinite program.

The remaining two sections of this paper are devoted to proving the four theorems stated in this section.

4. Proof of bounds on optimal solutions. This section presents the proofs of the results on lower and upper bounds on sizes of optimal solutions along the central trajectory for the pair of dual logarithmic barrier problems $P_\mu(d)$ and $D_\mu(d)$. We start by proving Theorem 3.1. Our proof is an immediate generalization to the semidefinite case of the proof of Theorem 3.1 in [15] for the case of a linear program.

Proof of Theorem 3.1. Let $\hat{X} := X(d,\mu)$ be the optimal solution to $P_\mu(d)$ and $(\hat{y},\hat{S}) := (y(d,\mu), S(d,\mu))$ be the optimal solution to the corresponding dual problem $D_\mu(d)$. Notice that the optimality conditions of $P_\mu(d)$ and $D_\mu(d)$ imply that $C \bullet \hat{X} = b^T\hat{y} + \mu n$.

Observe that since $\hat{S} = C - A^T[\hat{y}]$, then $\|\hat{S}\| \le \|C\| + \|A^T\|\|\hat{y}\|$. Since $\|A^T\| = \|A\|$, we have that $\|\hat{S}\| \le \|d\|(1+\|\hat{y}\|)$, and using the fact that $\mathcal{K}(d,\mu) \ge 1$ the bound (12) on $\|\hat{S}\|$ is a consequence of the bound (11) on $\|\hat{y}\|$. It therefore is sufficient to prove the bounds on $\|\hat{X}\|$ and on $\|\hat{y}\|$. Furthermore, the bound on $\|\hat{y}\|$ is trivial if $\hat{y} = 0$; therefore from now on we assume that $\hat{y} \ne 0$. Also, let $\bar{X} = \hat{X}/\|\hat{X}\|$ and $\bar{y} = \hat{y}/\|\hat{y}\|$. Clearly, $\bar{X} \bullet \hat{X} = \|\hat{X}\|$, $\|\bar{X}\| = 1$, $\bar{y}^T\hat{y} = \|\hat{y}\|$, and $\|\bar{y}\| = 1$.

The rest of the proof proceeds by examining three cases:

(i) $C \bullet \hat{X} \le 0$,
(ii) $0 < C \bullet \hat{X} \le \mu n$, and
(iii) $\mu n < C \bullet \hat{X}$.

In case (i), let $\Delta A_i := -b_i\bar{X}/\|\hat{X}\|$ for $i = 1,\dots,m$. Then, by letting the operator $\Delta A := (\Delta A_1,\dots,\Delta A_m)$ and $\Delta d := (\Delta A, 0, 0) \in \mathcal{D}$, we have $(A+\Delta A)\hat{X} = 0$, $\hat{X} \succ 0$, and $C \bullet \hat{X} \le 0$. It then follows from Proposition 2.2 that $D_\mu(d+\Delta d)$ is infeasible, and so $\rho(d) \le \|\Delta d\| = \|\Delta A\| = \|b\|/\|\hat{X}\| \le \|d\|/\|\hat{X}\|$. Therefore, $\|\hat{X}\| \le \|d\|/\rho(d) = \mathcal{C}(d) \le \mathcal{K}(d,\mu)$. This proves (10) in this case.

Consider the following notation: $\theta := b^T\hat{y}$, $\Delta b := -\theta\bar{y}/\|\hat{y}\|$, $\Delta A_i := -\bar{y}_i C/\|\hat{y}\|$ for $i = 1,\dots,m$, $\Delta A := (\Delta A_1,\dots,\Delta A_m)$, and $\Delta d := (\Delta A, \Delta b, 0) \in \mathcal{D}$. Observe that $(b+\Delta b)^T\hat{y} = 0$ and $(A+\Delta A)^T[\hat{y}] \prec 0$, so from Proposition 2.2 we conclude that $P_\mu(d+\Delta d)$ is infeasible. Therefore, $\rho(d) \le \|\Delta d\| = \max\{\|C\|, |\theta|\}/\|\hat{y}\|$. Hence, $\|\hat{y}\| \le \max\{\mathcal{C}(d), |\theta|/\rho(d)\}$. Furthermore, $|\theta| = |b^T\hat{y}| = |C\bullet\hat{X} - \mu n| \le \|\hat{X}\|\|C\| + \mu n \le \mathcal{C}(d)\|d\| + \mu n$. Therefore, using the fact that $\mathcal{C}(d) \ge 1$ for any d, we have (11).

In case (ii), let $\Delta d := (\Delta A, 0, \Delta C) \in \mathcal{D}$, where $\Delta A_i := -b_i\bar{X}/\|\hat{X}\|$ for $i = 1,\dots,m$ and $\Delta C := -\mu n\bar{X}/\|\hat{X}\|$. Observe that $(A+\Delta A)\hat{X} = 0$ and $(C+\Delta C)\bullet\hat{X} \le 0$. Hence, from Proposition 2.2 $D_\mu(d+\Delta d)$ is infeasible, and so we conclude that $\rho(d) \le \|\Delta d\| = \max\{\|\Delta A\|, \|\Delta C\|\} = \max\{\|b\|, \mu n\}/\|\hat{X}\| \le \max\{\|d\|, \mu n\}/\|\hat{X}\|$. Therefore, $\|\hat{X}\| \le \max\{\mathcal{C}(d), \mu n/\rho(d)\} \le \mathcal{K}(d,\mu)$. This proves (10) for this case.

Now let $\Delta d := (\Delta A, \Delta b, 0)$, where $\Delta A_i := -\bar{y}_i C/\|\hat{y}\|$ for $i = 1,\dots,m$ and $\Delta b := \mu n\bar{y}/\|\hat{y}\|$. Observe that $(b+\Delta b)^T\hat{y} = b^T\hat{y} + \mu n = C\bullet\hat{X} > 0$ and $(A+\Delta A)^T[\hat{y}] \prec 0$. As before, we have from Proposition 2.2 that $P_\mu(d+\Delta d)$ is infeasible, and so we conclude that $\rho(d) \le \|\Delta d\| = \max\{\|\Delta A\|, \|\Delta b\|\} = \max\{\|C\|, \mu n\}/\|\hat{y}\| \le \max\{\|d\|, \mu n\}/\|\hat{y}\|$. Therefore, we obtain $\|\hat{y}\| \le \max\{\mathcal{C}(d), \mu n/\rho(d)\} \le \mathcal{K}(d,\mu)$.

In case (iii), we first consider the bound on $\|\hat{y}\|$. Let $\Delta d := (\Delta A, 0, 0) \in \mathcal{D}$, where $\Delta A_i := -\bar{y}_i C/\|\hat{y}\|$ for $i = 1,\dots,m$. Since $(A+\Delta A)^T[\hat{y}] \prec 0$ and $b^T\hat{y} = C\bullet\hat{X} - \mu n > 0$,

it follows from Proposition 2.2 that $P_\mu(d+\Delta d)$ is infeasible, and so $\rho(d) \le \|\Delta d\| = \|C\|/\|\hat{y}\|$. Therefore, $\|\hat{y}\| \le \mathcal{C}(d) \le \mathcal{K}(d,\mu)$.

Finally, let $\Delta A_i := -b_i\bar{X}/\|\hat{X}\|$ for $i = 1,\dots,m$, and $\Delta C := -\theta\bar{X}/\|\hat{X}\|$, where $\theta := C \bullet \hat{X}$. Observe that $(A+\Delta A)\hat{X} = 0$ and $(C+\Delta C)\bullet\hat{X} = 0$. Thus, from Proposition 2.2 we conclude that $D_\mu(d+\Delta d)$ is infeasible, and so $\rho(d) \le \|\Delta d\| = \max\{\|\Delta A\|, \|\Delta C\|\} = \max\{\|b\|,\theta\}/\|\hat{X}\|$ so that $\|\hat{X}\| \le \max\{\mathcal{C}(d), \theta/\rho(d)\}$. Furthermore, $\theta = C\bullet\hat{X} = b^T\hat{y} + \mu n \le \|b\|\|\hat{y}\| + \mu n \le \|d\|\mathcal{C}(d) + \mu n$. Therefore, $\|\hat{X}\| \le \mathcal{K}(d,\mu)$. ∎

The following corollary presents upper bounds on optimal solutions to $P_{\mu+\Delta\mu}(d+\Delta d)$ and $D_{\mu+\Delta\mu}(d+\Delta d)$, where Δd is a data instance in $\mathcal{D}$ representing a small perturbation of the data instance d, and $\Delta\mu$ is a scalar.

COROLLARY 4.1. *Let $d \in \mathrm{Int}(\mathcal{F})$ and $\mu > 0$. If $\|\Delta d\| \le \rho(d)/3$ and $|\Delta\mu| \le \mu/3$, then*

$$\begin{aligned}\|X(d+\Delta d,\mu+\Delta\mu)\| &\le 4\mathcal{K}(d,\mu),\\ \|y(d+\Delta d,\mu+\Delta\mu)\| &\le 4\mathcal{K}(d,\mu),\\ \|S(d+\Delta d,\mu+\Delta\mu)\| &\le 6\|d\|\mathcal{K}(d,\mu).\end{aligned}$$

Proof. The proof follows by observing that

$$\begin{aligned}\|d+\Delta d\| &\le \|d\| + \frac{\rho(d)}{3},\\ \mu+\Delta\mu &\le \frac{4\mu}{3},\\ \rho(d+\Delta d) &\ge \frac{2\rho(d)}{3}.\end{aligned}$$

From these inequalities, we have $\mathcal{C}(d+\Delta d) \le \frac{3}{2}(\|d\| + \rho(d)/3)/\rho(d) = \frac{3}{2}(\mathcal{C}(d) + 1/3) \le 2\mathcal{C}(d)$ and $\|d+\Delta d\| \le \frac{4}{3}\|d\| \le 1.5\|d\|$, since $\mathcal{C}(d) \ge 1$. Furthermore, $(\mu+\Delta\mu)n/\rho(d+\Delta d) \le 2\mu n/\rho(d)$. Therefore, $\mathcal{K}(d+\Delta d,\mu+\Delta\mu) \le 4\mathcal{K}(d,\mu)$, and the result follows. ∎

The following proof of Theorem 3.2 is a generalization of part of the proof of Theorem 3.2 in [15] for the case of a linear program.

Proof of Theorem 3.2. Because of the Karush–Kuhn–Tucker optimality conditions of the dual pair of programs $P_\mu(d)$ and $D_\mu(d)$, we have $X(d,\mu)S(d,\mu) = \mu I$. This being the case, $X(d,\mu)$ and $S(d,\mu)$ can be simultaneously diagonalized, and so there exists an orthogonal matrix U such that $X(d,\mu) = UDU^T$, where $D = \mathrm{diag}(\lambda(X(d,\mu)))$ and $S(d,\mu) = \mu UD^{-1}U^T$. Then

$$\frac{1}{\lambda_j(X(d,\mu))} \le \frac{1}{D_{11}} = \frac{\|S(d,\mu)\|_\infty}{\mu},$$

and so $\lambda_j(X(d,\mu)) \ge \frac{\mu}{\|S(d,\mu)\|_\infty}$ for $j = 1,\dots,n$, and the result for $\lambda_j(X(d,\mu))$ follows from Theorem 3.1. Similarly,

$$\frac{1}{\lambda_j(S(d,\mu))} \le \frac{D_{nn}}{\mu} = \frac{\|X(d,\mu)\|_\infty}{\mu},$$

and so $\lambda_j(S(d,\mu)) \ge \frac{\mu}{\|X(d,\mu)\|_\infty}$ for $j = 1,\dots,n$, and the result for $\lambda_j(S(d,\mu))$ again follows from Theorem 3.1. ∎

COROLLARY 4.2. *Let $d \in \text{Int}(\mathcal{F})$ and $\mu > 0$. If $\|\Delta d\| \leq \rho(d)/3$ and $|\Delta\mu| \leq \mu/3$, then for all $j = 1, \ldots, n$,*

$$\lambda_j \left(X(d + \Delta d, \mu + \Delta\mu)\right) \geq \frac{\mu}{16\|d\|\mathcal{K}(d,\mu)},$$
$$\lambda_j \left(S(d + \Delta d, \mu + \Delta\mu)\right) \geq \frac{\mu}{6\mathcal{K}(d,\mu)}.$$

Proof. The proof follows immediately from Theorem 3.2 by observing that $\|d + \Delta d\| \leq \frac{4}{3}\|d\|$, $\mu + \Delta\mu \geq \frac{2}{3}\mu$, and $\mathcal{K}(d + \Delta d, \mu + \Delta\mu) \leq 4\mathcal{K}(d,\mu)$. ∎

5. Proof of bounds on changes in optimal solutions. In this section we prove Theorems 3.3 and 3.4. Before presenting the proofs, we first present properties of certain linear operators that arise in our analysis, in Propositions 5.1–5.5, and Corollary 5.1.

PROPOSITION 5.1. *Given a data instance $d = (A, b, C) \in \mathcal{D}$ and matrices X and $\bar{X}$ such that $X, \bar{X} \succ 0$, let P be the linear operator from $\Re^m$ to $\Re^m$ defined as*

$$Pw := A\left(X\left(A^T[w]\right)\bar{X}\right)$$

for all $w \in \Re^m$. If A has rank m, then the following statements hold true:

1. *P corresponds to a symmetric positive definite matrix in $\Re^{m\times m}$,*
2. *$Pw = A\left(\bar{X}\left(A^T[w]\right)X\right)$ for all $w \in \Re^m$.*

Proof. By using the canonical basis for $\Re^m$ and slightly amending the notation, we have that the (i,j)-coordinate of the matrix corresponding to P is given by

$$P_{ij} = A_i \bullet (XA_j\bar{X}). \tag{17}$$

Hence, if w is such that $Pw = 0$, then for all $i = 1, \ldots, m$,

$$\sum_{j=1}^{m} \left(A_i \bullet (XA_j\bar{X})\right) w_j = 0,$$
$$\sum_{j=1}^{m} \left((X^{1/2}A_i) \bullet (X^{1/2}A_j\bar{X})\right) w_j = 0,$$
$$\sum_{j=1}^{m} \left((X^{1/2}A_i\bar{X}^{1/2}) \bullet (X^{1/2}A_j\bar{X}^{1/2})\right) w_j = 0.$$

It therefore follows that

$$w^T Pw = \sum_{i=1}^{m}\sum_{j=1}^{m} w_i \left((X^{1/2}A_i\bar{X}^{1/2}) \bullet (X^{1/2}A_j\bar{X}^{1/2})\right) w_j = 0. \tag{18}$$

This in turn can be written as

$$\|X^{1/2}(A^T[w])\bar{X}^{1/2}\|_2^2 = 0,$$

from which we obtain $A^T[w] = 0$. Using the fact that A has rank m, we have $w = 0$. Therefore the matrix corresponding to P is nonsingular.

On the other hand, notice that from (17) we have $P_{ij} = A_i \bullet (XA_j\bar{X}) = (XA_i\bar{X}) \bullet A_j = A_j \bullet (XA_i\bar{X}) = P_{ji}$ for all $1 \leq i, j \leq m$. Hence, P is a symmetric operator.

Furthermore, if we let $\hat{A} := (X^{1/2}A_1\bar{X}^{1/2}, \ldots, X^{1/2}A_m\bar{X}^{1/2})$, we obtain from (18) $w^T P w = \|\hat{A}^T[w]\|_2^2 \geq 0$ for all $w \in \Re^m$. Hence, P is a positive semidefinite operator. Using that P is nonsingular, we conclude the first statement.

Finally, the second statement follows from well-known properties of the trace:

$$\begin{aligned} \mathit{trace}\left(A_i X A_j \bar{X}\right) &= \mathit{trace}\left(X A_j \bar{X} A_i\right) \\ &= \mathit{trace}\left((X A_j \bar{X} A_i)^T\right) \\ &= \mathit{trace}\left(A_i \bar{X} A_j X\right) \end{aligned}$$

for all $1 \leq i, j \leq m$. Therefore, $Pw = A\left(\bar{X}\left(A^T[w]\right)X\right)$ for all $w \in \Re^m$, and the result follows. ∎

PROPOSITION 5.2. *Let $d = (A, b, C) \in \mathrm{Int}(\mathcal{F})$ and P be the linear operator from $\Re^m$ to $\Re^m$ defined as*

$$Pw := A\left(A^T[w]\right)$$

for all $w \in \Re^m$. Then P is a symmetric positive definite matrix and

$$\rho(d) \leq \sqrt{\lambda_1(P)}.$$

Proof. Observe that since $d \in \mathrm{Int}(\mathcal{F})$, A has rank m, and so from Proposition 5.1 (setting $\bar{X} := X := I$), P is a symmetric and positive semidefinite matrix.

Let $\lambda := \lambda_1(P)$. There exists a vector $v \in \Re^m$ with $\|v\| = 1$ and $Pv = \lambda v$. Hence, $v^T P v = \lambda$. Let $\Delta A \in \mathcal{L}_{m,n}$ be defined as

$$\Delta A := \left(-v_1(A^T[v]), \ldots, -v_m(A^T[v])\right)^T,$$

and $\Delta b = \epsilon v$ for any $\epsilon > 0$ and small. Then, $(A + \Delta A)^T[v] = 0$ and $(b + \Delta b)^T v = b^T v + \epsilon \neq 0$ for all $\epsilon > 0$ and small. Hence, $(A + \Delta A)X = b + \Delta b$ is an inconsistent system of equations for all $\epsilon > 0$ and small. Therefore, $\rho(d) \leq \max\{\|\Delta A\|, \|\Delta b\|\} = \|\Delta A\| = \|A^T[v]\| = \sqrt{\lambda}$, thus proving this proposition. ∎

PROPOSITION 5.3. *Given a data instance $d = (A, b, C) \in \mathcal{D}$ such that A has rank m, and matrices X and $\bar{X}$ such that $X, \bar{X} \succ 0$, let Q be the linear operator from $\Re^{n\times n}$ to $\Re^{n\times n}$ defined as*

$$QV := V - \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}VX^{1/2}\right)\right]\right)X^{1/2}$$

for all $V \in \Re^{n\times n}$, where P is the matrix from Proposition 5.1. Then Q corresponds to a symmetric projection operator.

Proof. Let $RV := \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}VX^{1/2}\right)\right]\right)X^{1/2}$ for all $V \in \Re^{n\times n}$. Since $QV = V - RV = (I - R)V$, then Q is a symmetric projection if and only if R is a symmetric projection. It is straightforward to show that

$$RV = \sum_{i=1}^{m}\sum_{j=1}^{m} P_{ij}^{-1}\left(A_j \bullet \left(\bar{X}^{1/2}VX^{1/2}\right)\right)\left(\bar{X}^{1/2}A_iX^{1/2}\right) \tag{19}$$

for all $V \in \Re^{n\times n}$. For a fixed matrix V in $\Re^{n\times n}$, it follows from this identity that

$$W \bullet (RV) = \left(\sum_{i=1}^{m}\sum_{j=1}^{m} P_{ij}^{-1}\left(A_i \bullet \left(\bar{X}^{1/2}WX^{1/2}\right)\right)\left(\bar{X}^{1/2}A_jX^{1/2}\right)\right) \bullet V$$

for all W in $\Re^{n\times n}$. Hence, we have

$$R^T[W] = \sum_{i=1}^{m}\sum_{j=1}^{m} P_{ij}^{-1}\left(A_i \bullet \left(\bar{X}^{1/2} W X^{1/2}\right)\right)\left(\bar{X}^{1/2} A_j X^{1/2}\right)$$

for all W in $\Re^{n\times n}$. By noticing that P is a symmetric matrix and using (19), we obtain $R = R^T$; that is, R is a symmetric operator.

On the other hand, for a given V in $\Re^{n\times n}$, let $w := P^{-1}A\left(\bar{X}^{1/2} V X^{1/2}\right)$. Thus, using Proposition 5.1, statement 2, we obtain

$$\begin{aligned}
RRV &= \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}(RV)X^{1/2}\right)\right]\right)X^{1/2}\\
&= \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}\left(\bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}VX^{1/2}\right)\right]\right)X^{1/2}\right)X^{1/2}\right)\right]\right)X^{1/2}\\
&= \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}\left(A^T[w]\right)X\right)\right]\right)X^{1/2}\\
&= \bar{X}^{1/2}\left(A^T\left[P^{-1}Pw\right]\right)X^{1/2}\\
&= \bar{X}^{1/2}\left(A^T[w]\right)X^{1/2}\\
&= RV,
\end{aligned}$$

where the fourth equality above follows from statement 2 of Proposition 5.1. Therefore, from [11, Theorem 1, page 73], R is a projection and the result follows. □

PROPOSITION 5.4. *Given a data instance $d = (A, b, C) \in \mathcal{D}$ and matrices X and $\bar{X}$ such that $X, \bar{X} \succ 0$, let P be the linear operator from $\Re^m$ to $\Re^m$ defined as*

$$Pw := A\left(X\left(A^T[w]\right)\bar{X}\right)$$

for all $w \in \Re^m$. Then if A has rank m,

$$\|P^{-1}\|_\infty \le \|X^{-1}\|_\infty \|\bar{X}^{-1}\|_\infty \|(AA^T)^{-1}\|_\infty.$$

Proof. From Proposition 5.1, it follows that P is nonsingular. Let w be any vector in $\Re^m$ normalized so that $\|w\| = 1$, and consider a spectral decomposition of X as

$$X = \sum_{k=1}^{n} \lambda_k(X) u_k u_k^T,$$

where $\{u_1, \ldots, u_n\}$ is an orthonormal basis for $\Re^n$. By using that $trace(u_k u_k^T) \ge 0$ for all $1 \le k \le n$, and $\sum_{k=1}^{n} u_k u_k^T = I$, we have

$$\begin{aligned}
w^T P w &= \sum_{i=1}^{m}\sum_{j=1}^{m} trace\left(A_i X A_j \bar{X}\right) w_i w_j\\
&= trace\left(\sum_{i=1}^{m}\sum_{j=1}^{m} A_i X A_j \bar{X} w_i w_j\right)\\
&= trace\left(\bar{X}^{1/2}\left(\sum_{i=1}^{m} A_i w_i\right) X \left(\sum_{j=1}^{m} A_j w_j\right)\bar{X}^{1/2}\right)\\
&= \sum_{k=1}^{n} \lambda_k(X)\, trace\left(\bar{X}^{1/2}\left(\sum_{i=1}^{m} A_i w_i\right) u_k u_k^T \left(\sum_{j=1}^{m} A_j w_j\right)\bar{X}^{1/2}\right)
\end{aligned}$$

$$\geq \|X^{-1}\|_\infty^{-1} trace\left(\bar{X}^{1/2}\left(\sum_{i=1}^m A_i w_i\right)\left(\sum_{j=1}^m A_j w_j\right)\bar{X}^{1/2}\right)$$
$$= \|X^{-1}\|_\infty^{-1} trace\left(\left(\sum_{i=1}^m A_i w_i\right)\bar{X}\left(\sum_{j=1}^m A_j w_j\right)\right).$$

Now, consider a spectral decomposition of $\bar{X}$ as

$$\bar{X} = \sum_{k=1}^n \lambda_k(\bar{X}) v_k v_k^T,$$

where, as before, $\{v_1, \ldots, v_n\}$ is an orthonormal basis for $\Re^n$, and so $\sum_{k=1}^n v_k v_k^T = I$. Notice that from Proposition 5.1, it follows that the operator AA^T is nonsingular. Then, we have

$$w^T P w \geq \|X^{-1}\|_\infty^{-1} \sum_{k=1}^n \lambda_k(\bar{X}) trace\left(\left(\sum_{i=1}^m A_i w_i\right) v_k v_k^T \left(\sum_{j=1}^m A_j w_j\right)\right)$$
$$\geq \|X^{-1}\|_\infty^{-1} \|\bar{X}^{-1}\|_\infty^{-1} trace\left(\left(\sum_{i=1}^m A_i w_i\right)\left(\sum_{j=1}^m A_j w_j\right)\right)$$
$$\geq \|X^{-1}\|_\infty^{-1} \|\bar{X}^{-1}\|_\infty^{-1} \|(AA^T)^{-1}\|_\infty^{-1}.$$

Notice that in the last inequality we used

$$trace\left(\left(\sum_{i=1}^m A_i w_i\right)\left(\sum_{j=1}^m A_j w_j\right)\right) = w^T \hat{P} w \geq \min_k \lambda_k(\hat{P}) = \|(AA^T)^{-1}\|_\infty^{-1},$$

where $\hat{P} = AA^T$.

Now let $\hat{w}$ be the normalized eigenvector corresponding to the smallest eigenvalue of P, i.e., $\|\hat{w}\| = 1$ and $P\hat{w} = \lambda_1(P)\hat{w}$. Then from above we have

$$\|P^{-1}\|_\infty^{-1} = \lambda_1(P) = \hat{w}^T P \hat{w} \geq \|X^{-1}\|_\infty^{-1} \|\bar{X}^{-1}\|_\infty^{-1} \|(AA^T)^{-1}\|_\infty^{-1}$$

and the result follows. □

COROLLARY 5.1. *Let $d = (A, b, C)$ be a data instance in $\mathrm{Int}(\mathcal{F})$, μ be a positive scalar, $\Delta d = (\Delta A, \Delta b, \Delta C) \in \mathcal{D}$ be a data perturbation such that $\|\Delta d\| \leq \rho(d)/3$, and $\Delta\mu$ be a scalar such that $|\Delta\mu| \leq \mu/3$. Then*

$$\|P^{-1}\| \leq 32\sqrt{m}\left(\frac{\mathcal{C}(d)\mathcal{K}(d,\mu)}{\mu}\right)^2,$$

where P is the linear operator from $\Re^m$ to $\Re^m$ defined as

$$Pw := A\left(X(d,\mu)\left(A^T[w]\right)X(d+\Delta d, \mu+\Delta\mu)\right)$$

for all $w \in \Re^m$, and $\mathcal{K}(d,\mu)$ is the scalar defined in (9).

Proof. Let $X = X(d,\mu)$ and $\bar{X} = X(d+\Delta d, \mu+\Delta\mu)$. From Proposition 5.4 we know that

$$\|P^{-1}\|_\infty \leq \|X^{-1}\|_\infty \|\bar{X}^{-1}\|_\infty \|(AA^T)^{-1}\|_\infty.$$

From Theorem 3.2 and Corollary 4.2, respectively, we have

$$\|X^{-1}\|_\infty \leq \frac{2\|d\|\mathcal{K}(d,\mu)}{\mu},$$
$$\|\bar{X}^{-1}\|_\infty \leq \frac{16\|d\|\mathcal{K}(d,\mu)}{\mu}.$$

Furthermore, from Proposition 5.2 we have

$$\|(AA^T)^{-1}\|_\infty \leq \frac{1}{\rho(d)^2}.$$

By combining these results and using Proposition 2.1, we obtain the corollary. ∎

PROPOSITION 5.5. *Let $d = (A, b, C)$ be a data instance in $\mathrm{Int}(\mathcal{F})$, μ be a positive scalar, $\Delta d = (\Delta A, \Delta b, \Delta C) \in \mathcal{D}$ be a data perturbation such that $\|\Delta d\| \leq \rho(d)/3$, and $\Delta\mu$ be a scalar such that $|\Delta\mu| \leq \mu/3$. Then,*

$$\|\Delta b - \Delta A X(d + \Delta d, \mu + \Delta\mu)\| \leq 5\|\Delta d\|\mathcal{K}(d,\mu),$$
$$\|\Delta C - \Delta A^T[y(d + \Delta d, \mu + \Delta\mu)]\| \leq 5\|\Delta d\|\mathcal{K}(d,\mu).$$

Proof. Let $\bar{X} := X(d + \Delta d, \mu + \Delta\mu)$ and $\bar{y} := y(d + \Delta d, \mu + \Delta\mu)$. From Corollary 4.1, we have

$$\begin{aligned}\|\Delta b - \Delta A\bar{X}\| &\leq \|\Delta d\|\left(1 + \|\bar{X}\|\right)\\ &\leq \|\Delta d\|\left(1 + 4\mathcal{K}(d,\mu)\right)\\ &\leq 5\|\Delta d\|\mathcal{K}(d,\mu),\end{aligned}$$

$$\begin{aligned}\|\Delta C - \Delta A^T[\bar{y}]\| &\leq \|\Delta d\|\left(1 + \|\bar{y}\|\right)\\ &\leq \|\Delta d\|\left(1 + 4\mathcal{K}(d,\mu)\right)\\ &\leq 5\|\Delta d\|\mathcal{K}(d,\mu),\end{aligned}$$

and so the proposition follows. ∎

Now we are ready to present the proof of Theorem 3.3.

Proof of Theorem 3.3. To simplify the notation, let $(X, y, S) := (X(d,\mu), y(d,\mu), S(d,\mu))$, $(\bar{X}, \bar{y}, \bar{S}) := (X(d+\Delta d, \mu+\Delta\mu), y(d+\Delta d, \mu+\Delta\mu), S(d+\Delta d, \mu+\Delta\mu))$, and $\bar{\mu} := \mu + \Delta\mu$. From the Karush–Kuhn–Tucker optimality conditions associated with the programs $P_\mu(d)$ and $P_{\mu+\Delta\mu}(d + \Delta d)$, respectively, we obtain

$$\begin{array}{rclcrcl} XS &=& \mu I, & & \bar{X}\bar{S} &=& \bar{\mu} I,\\ A^T[y] + S &=& C, & (A+\Delta A)^T[\bar{y}] + \bar{S} &=& C + \Delta C,\\ AX &=& b, & (A+\Delta A)\bar{X} &=& b + \Delta b,\\ X &\succ& 0, & \bar{X} &\succ& 0. \end{array}$$

Let $\Delta E := \Delta b - \Delta A\bar{X}$ and $\Delta F := \Delta C - \Delta A^T[\bar{y}]$. Therefore,

$$\begin{aligned}\bar{X} - X &= \frac{1}{\mu\bar{\mu}}\bar{X}(\bar{\mu}S - \mu\bar{S})X\\ &= \frac{1}{\mu\bar{\mu}}\bar{X}\left(\bar{\mu}(C - A^T[y]) - \mu(C + \Delta C - (A+\Delta A)^T[\bar{y}])\right)X\\ &= \frac{1}{\mu\bar{\mu}}\bar{X}\left(\Delta\mu C - \mu(\Delta C - \Delta A^T[\bar{y}]) - A^T[\bar{\mu}y - \mu\bar{y}]\right)X\\ &= \frac{\Delta\mu}{\mu\bar{\mu}}\bar{X}CX - \frac{1}{\bar{\mu}}\bar{X}\Delta FX - \frac{1}{\mu\bar{\mu}}\bar{X}\left(A^T[\bar{\mu}y - \mu\bar{y}]\right)X. \qquad (20)\end{aligned}$$

On the other hand, $A(\bar{X} - X) = \Delta b - \Delta A\bar{X} = \Delta E$. Since $d \in \mathrm{Int}(\mathcal{F})$, then A has full-rank (see (8)). It follows from Proposition 5.1 that the linear operator P from $\Re^m$ to $\Re^m$ defined as $Pw := A(\bar{X}(A^T[w])X)$, for all $w \in \Re^m$, corresponds to a positive definite matrix in $\Re^{m\times m}$. By combining this result with (20), we obtain

$$\Delta E = \frac{\Delta\mu}{\mu\bar{\mu}} A\left(\bar{X}CX\right) - \frac{1}{\bar{\mu}} A\left(\bar{X}\Delta FX\right) - \frac{1}{\mu\bar{\mu}} P(\bar{\mu}y - \mu\bar{y}),$$

and so

$$P^{-1}\Delta E = \frac{\Delta\mu}{\mu\bar{\mu}} P^{-1}A\left(\bar{X}CX\right) - \frac{1}{\bar{\mu}} P^{-1}A\left(\bar{X}\Delta FX\right) - \frac{1}{\mu\bar{\mu}}(\bar{\mu}y - \mu\bar{y}).$$

Therefore, we have the following identity:

$$\frac{1}{\mu\bar{\mu}}(\bar{\mu}y - \mu\bar{y}) = \frac{\Delta\mu}{\mu\bar{\mu}} P^{-1}A\left(\bar{X}CX\right) - \frac{1}{\bar{\mu}} P^{-1}A\left(\bar{X}\Delta FX\right) - P^{-1}\Delta E. \tag{21}$$

Combining (21) and (20), we obtain

$$\begin{aligned}
\bar{X} - X &= \frac{\Delta\mu}{\mu\bar{\mu}}\bar{X}CX - \frac{1}{\bar{\mu}}\bar{X}\Delta FX \\
&\quad - \bar{X}\left(A^T\left[\frac{\Delta\mu}{\mu\bar{\mu}} P^{-1}A\left(\bar{X}CX\right) - \frac{1}{\bar{\mu}} P^{-1}A\left(\bar{X}\Delta FX\right) - P^{-1}\Delta E\right]\right)X \\
&= \frac{\Delta\mu}{\mu\bar{\mu}}\left(\bar{X}CX - \bar{X}\left(A^T\left[P^{-1}A\left(\bar{X}CX\right)\right]\right)X\right) \\
&\quad - \frac{1}{\bar{\mu}}\left(\bar{X}\Delta FX - \bar{X}\left(A^T\left[P^{-1}A\left(\bar{X}\Delta FX\right)\right]\right)X\right) \\
&\quad + \bar{X}\left(A^T\left[P^{-1}\Delta E\right]\right)X \\
&= \frac{\Delta\mu}{\mu\bar{\mu}}\bar{X}^{1/2}Q\left(\bar{X}^{1/2}CX^{1/2}\right)X^{1/2} - \frac{1}{\bar{\mu}}\bar{X}^{1/2}Q\left(\bar{X}^{1/2}\Delta FX^{1/2}\right)X^{1/2} \\
&\quad + \bar{X}\left(A^T\left[P^{-1}\Delta E\right]\right)X, \qquad (22)
\end{aligned}$$

where by Q we denote the following linear operator from $\Re^{n\times n}$ to $\Re^{n\times n}$:

$$Q(V) := V - \bar{X}^{1/2}\left(A^T\left[P^{-1}A\left(\bar{X}^{1/2}VX^{1/2}\right)\right]\right)X^{1/2}$$

for all $V \in \Re^{n\times n}$. By using Proposition 5.3, it follows that Q is a symmetric projection operator, and so $\|QV\| \le \|V\|$ for all $V \in \Re^{n\times n}$. Since $\|V^{1/2}\|^2 \le \sqrt{n}\|V\|$ for all V in $\mathcal{S}_n^+$, from (22), Theorem 3.1, Corollary 4.1, Corollary 5.1, and Proposition 5.5, it follows that

$$\begin{aligned}
\|\bar{X} - X\| &\le \frac{|\Delta\mu|}{\mu\bar{\mu}}\|\bar{X}^{1/2}\|^2\|C\|\|X^{1/2}\|^2 + \frac{1}{\bar{\mu}}\|\bar{X}^{1/2}\|^2\|\Delta F\|\|X^{1/2}\|^2 \\
&\qquad + \|\bar{X}\left(A^T[P^{-1}\Delta E]\right)X\| \\
&\le \frac{n|\Delta\mu|}{\mu\bar{\mu}}\|\bar{X}\|\|C\|\|X\| + \frac{n}{\bar{\mu}}\|\bar{X}\|\|\Delta F\|\|X\| + \|\bar{X}\|\|A^T\|\|P^{-1}\|\|\Delta E\|\|X\| \\
&\le \frac{4n|\Delta\mu|}{\mu\bar{\mu}}\|d\|\mathcal{K}(d,\mu)^2 + \frac{20n\|\Delta d\|}{\bar{\mu}}\mathcal{K}(d,\mu)^3 + \frac{640}{\mu^2}\sqrt{m}\|\Delta d\|\|d\|\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5.
\end{aligned}$$

Therefore, by noticing that $\bar{\mu} \geq \frac{2}{3}\mu$, we obtain

$$\|\bar{X} - X\| \leq \frac{6n}{\mu^2}|\Delta\mu|\|d\|\mathcal{K}(d,\mu)^2 + \frac{640n\sqrt{m}}{\mu^2}\|\Delta d\|\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5\left(\mu + \|d\|\right),$$

and so (14) follows.

Next, we prove the bound on $\|\bar{y} - y\|$. From the identities $(A + \Delta A)^T[\bar{y}] + \bar{S} = C + \Delta C$ and $A^T[y] + S = C$, it follows that

$$\begin{aligned}
\bar{S} - S &= \Delta F - A^T[\bar{y} - y], \\
\bar{\mu}\bar{X}^{-1} - \mu X^{-1} &= \Delta F - A^T[\bar{y} - y], \\
\bar{X}^{-1}(\bar{\mu}X - \mu\bar{X})X^{-1} &= \Delta F - A^T[\bar{y} - y].
\end{aligned}$$

Hence,

$$\begin{aligned}
\bar{\mu}X - \mu\bar{X} &= \bar{X}\left(\Delta F - A^T[\bar{y} - y]\right)X \\
&= \bar{X}\Delta F X - \bar{X}\left(A^T[\bar{y} - y]\right)X.
\end{aligned}$$

By premultiplying this identity by A, we obtain

$$\Delta\mu b - \mu\Delta E = A\left(\bar{X}\Delta F X\right) - P(\bar{y} - y),$$

and so,

$$\begin{aligned}
P(\bar{y} - y) &= -\Delta\mu b + \mu\Delta E + A\left(\bar{X}\Delta F X\right), \\
\bar{y} - y &= -\Delta\mu P^{-1}b + \mu P^{-1}\Delta E + P^{-1}A\left(\bar{X}\Delta F X\right).
\end{aligned}$$

Therefore, using this identity, Theorem 3.1, Corollary 4.1, Corollary 5.1, and Proposition 5.5, we obtain

$$\begin{aligned}
\|\bar{y} - y\| &\leq |\Delta\mu|\|P^{-1}\|\|b\| + \mu\|P^{-1}\|\|\Delta E\| + \|P^{-1}\|\|A\|\|\bar{X}\|\|\Delta F\|\|X\| \\
&\leq 32\sqrt{m}|\Delta\mu|\|d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2} + 160\sqrt{m}\|\Delta d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^3}{\mu} \\
&\quad + 640\sqrt{m}\|\Delta d\|\|d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5}{\mu^2} \\
&\leq 32\sqrt{m}|\Delta\mu|\|d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2} + 640\sqrt{m}\|\Delta d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu + \|d\|)}{\mu^2},
\end{aligned}$$

and so we obtain inequality (15).

Finally, to obtain the bound on $\|\bar{S} - S\|$, we proceed as follows. Notice that $\bar{S} - S = \Delta F - A^T[\bar{y} - y]$. Hence, from (15) and Proposition 5.5, we have

$$\begin{aligned}
\|\bar{S} - S\| &\leq \|\Delta F\| + \|A^T\|\|\bar{y} - y\| \\
&\leq 5\|\Delta d\|\mathcal{K}(d,\mu) + \|d\|\left(32\sqrt{m}|\Delta\mu|\|d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2}\right. \\
&\qquad\qquad \left. + 640\sqrt{m}\|\Delta d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu + \|d\|)}{\mu^2}\right) \\
&\leq 32\sqrt{m}|\Delta\mu|\|d\|^2\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^2}{\mu^2} + 640\sqrt{m}\|\Delta d\|\frac{\mathcal{C}(d)^2\mathcal{K}(d,\mu)^5(\mu + \|d\|)^2}{\mu^2},
\end{aligned}$$

which establishes (16), concluding the proof of this theorem. ∎

Finally, we present the proof of Theorem 3.4.

Proof of Theorem 3.4. To simplify the notation, let $\bar{z} := z(d+\Delta d, \mu+\Delta\mu)$ and $z = z(d,\mu)$. Consider the Lagrangian functions associated with $P_\mu(d)$ and $P_{\mu+\Delta\mu}(d+\Delta d)$, respectively:

$$L(X,y) := C \bullet X + \mu p(X) + y^T(b - AX),$$
$$\bar{L}(X,y) := (C+\Delta C) \bullet X + (\mu+\Delta\mu)p(X) + y^T(b+\Delta b - (A+\Delta A)X),$$

and define $M(X,y) := L(X,y) - \bar{L}(X,y)$. Let $\hat{X}$ and $(\hat{y},\hat{S})$ denote the optimal solutions to $P_\mu(d)$ and $D_\mu(d)$, respectively, and let $\bar{X}$ and $(\bar{y},\bar{S})$ denote the optimal solutions to $P_{\mu+\Delta\mu}(d+\Delta d)$ and $D_{\mu+\Delta\mu}(d+\Delta d)$, respectively. Hence, we have

$$\begin{aligned} z &= L(\hat{X},\hat{y}) \\ &= \max_y L(\hat{X},y) \\ &= \max_y \left\{ \bar{L}(\hat{X},y) + M(\hat{X},y) \right\} \\ &\geq \bar{L}(\hat{X},\bar{y}) + M(\hat{X},\bar{y}) \\ &\geq \min_{X\succ 0} \bar{L}(X,\bar{y}) + M(\hat{X},\bar{y}) \\ &= \bar{z} + M(\hat{X},\bar{y}). \end{aligned}$$

Thus, $z - \bar{z} \geq M(\hat{X},\bar{y})$. Similarly, we can prove that $z - \bar{z} \leq M(\bar{X},\hat{y})$. Therefore, we obtain that either $|\bar{z}-z| \leq |M(\hat{X},\bar{y})|$ or $|\bar{z}-z| \leq |M(\bar{X},\hat{y})|$. On the other hand, by using Theorem 3.1 and Corollary 4.1, we have

$$\begin{aligned} |M(\hat{X},\bar{y})| &= |\Delta C \bullet \hat{X} + \Delta\mu p(\hat{X}) + \bar{y}^T\Delta b - \bar{y}^T\Delta A\hat{X}| \\ &\leq \|\Delta C\|\|\hat{X}\| + |\Delta\mu||p(\hat{X})| + \|\bar{y}\|\|\Delta b\| + \|\bar{y}\|\|\Delta A\|\|\hat{X}\| \\ &\leq \|\Delta d\| \left(\|\hat{X}\| + \|\bar{y}\| + \|\bar{y}\|\|\hat{X}\| \right) + |\Delta\mu||p(\hat{X})| \\ &\leq 9\|\Delta d\|\mathcal{K}(d,\mu)^2 + |\Delta\mu||p(\hat{X})|. \end{aligned}$$

Similarly, it is not difficult to show that

$$|M(\bar{X},\hat{y})| \leq 9\|\Delta d\|\mathcal{K}(d,\mu)^2 + |\Delta\mu||p(\bar{X})|.$$

Therefore,

$$|\bar{z}-z| \leq 9\|\Delta d\|\mathcal{K}(d,\mu)^2 + |\Delta\mu| \max\left\{ |p(\hat{X})|, |p(\bar{X})| \right\}.$$

By using Theorems 3.1 and 3.2 and Corollaries 4.1 and 4.2, we obtain

$$\begin{aligned} -n\ln\left(\mathcal{K}(d,\mu)\right) \leq & p(\hat{X}) \leq -n\ln\left(\frac{\mu}{2\|d\|\mathcal{K}(d,\mu)}\right) \\ -n\ln\left(4\mathcal{K}(d,\mu)\right) \leq & p(\bar{X}) \leq -n\ln\left(\frac{\mu}{16\|d\|\mathcal{K}(d,\mu)}\right). \end{aligned}$$

Thus, we have the following bound:

$$\begin{aligned} \max\left\{ |p(\hat{X})|, |p(\bar{X})| \right\} &\leq n\max\left\{ \ln\left(4\mathcal{K}(d,\mu)\right), \left| \ln\left(\frac{\mu}{16\|d\|\mathcal{K}(d,\mu)}\right) \right| \right\} \\ &\leq n\left(\ln 16 + |\ln\mu| + |\ln\|d\|| + \ln\mathcal{K}(d,\mu)\right), \end{aligned}$$

and so the result follows. ∎

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. 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] R. Bhatia, *Matrix Analysis*, Springer-Verlag, New York, 1997.

[4] M. Epelman and R. M. Freund, *Condition number complexity of an elementary algorithm for computing a reliable solution of a conic linear system*, Math. Program., 88 (2000), pp. 451–485.

[5] S. Filipowski, *On the complexity of solving feasible systems of linear inequalities specified with approximate data*, Math. Program., 71 (1995), pp. 259–288.

[6] S. Filipowski, *On the complexity of solving sparse symmetric linear programs specified with approximate data*, Math. Oper. Res., 22 (1997), pp. 769–792.

[7] S. Filipowski, *On the complexity of solving feasible linear programs specified with approximate data*, SIAM J. Optim., 9 (1999), pp. 1010–1040.

[8] R. M. Freund and J. R. Vera, *Some characterizations and properties of the "distance to ill-posedness" and the condition measure of a conic linear system*, Math. Program., 86 (1999), pp. 225–260.

[9] R. M. Freund and J. R. Vera, *Condition-based complexity of convex optimization in conic linear form via the ellipsoid algorithm*, SIAM J. Optim., 10 (1999), pp. 155–176.

[10] D. Goldfarb and K. Scheinberg, *Interior point trajectories in semidefinite programming*, SIAM J. Optim., 8 (1998), pp. 871–886.

[11] P. R. Halmos, *Finite-Dimensional Vector Spaces*, Springer-Verlag, New York, 1993.

[12] O. L. Mangasarian, *A stable theorem of the alternative*: *An extension of the Gordan theorem*, Linear Algebra Appl., 41 (1981), pp. 209–223.

[13] M. V. Nayakkankuppam and M. L. Overton, *Conditioning of semidefinite programs*, Math. Program., 85 (1999), pp. 525–540.

[14] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[15] M. A. Nunez and R. M. Freund, *Condition measures and properties of the central trajectory of a linear program*, Math. Program., 83 (1998), pp. 1–28.

[16] J. Peña and J. Renegar, *Computing approximate solutions for convex conic systems of constraints*, Math. Program., 87 (2000), pp. 351–383.

[17] J. Renegar, *Some perturbation theory for linear programming*, Math. Program., 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. Program., 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] J. Renegar, *A Mathematical View of Interior-Point Methods in Convex Optimization*, SIAM, Philadelphia, forthcoming.

[22] S. M. Robinson, *Stability theory for systems of inequalities. Part* I: *Linear systems*, SIAM J. Numer. Anal., 12 (1975), pp. 754–769.

[23] S. M. Robinson, *Stability theory for systems of inequalities. Part* II: *Differential nonlinear systems*, SIAM J. Numer. Anal., 13 (1976), pp. 497–513.

[24] S. M. Robinson, *A characterization of stability in linear programming*, Oper. Res., 25 (1977), pp. 435–447.

[25] S. Scholtes, *On the Computability of the Condition Number for Certain Inconsistent Systems*, Tech. report, Department of Engineering and The Judge Institute of Management Studies, University of Cambridge, Cambridge, England, 1996.

[26] J. F. Sturm and S. Zhang, *On Sensitivity of Central Solutions in Semidefinite Programming*, Tech. report 9813/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1998.

[27] L. Vandenberghe and S. Boyd, *Semidefinite programming*, SIAM Rev., 38 (1996), pp. 49–95.

[28] J. R. Vera, *Ill-posedness and the Computation of Solutions to Linear Programs with Approximate Data*, Tech. report, Cornell University, Ithaca, NY, 1992.

[29] J. R. Vera, *Ill-posedness and the complexity of deciding existence of solutions to linear programs*, SIAM J. Optim., 6 (1996), pp. 549–569.

SIAM J. OPTIM.
Vol. 11, No. 3, pp. 837–857

A DIRECT SEARCH ALGORITHM FOR OPTIMIZATION WITH NOISY FUNCTION EVALUATIONS*

EDWARD J. ANDERSON† AND MICHAEL C. FERRIS‡

Abstract. We consider the unconstrained optimization of a function when each function evaluation is subject to a random noise. We assume that there is some control over the variance of the noise term, in the sense that additional computational effort will reduce the amount of noise. This situation may occur when function evaluations involve simulation or the approximate solution of a numerical problem. It also occurs in an experimental setting when averaging repeated observations at the same point can lead to a better estimate of the underlying function value. We describe a new direct search algorithm for this type of problem. We prove convergence of the new algorithm when the noise is controlled so that the standard deviation of the noise approaches zero faster than the step size. We also report some numerical results on the performance of the new algorithm.

Key words. optimization, direct search, noisy functions

AMS subject classifications. 49M30, 90C56

PII. S1052623496312848

1. Introduction.

During the past few years there has been an increasing interest in direct search methods for unconstrained optimization (Hooke and Jeeves (1961), Nelder and Mead (1965), Spendley, Hext, and Himsworth (1962)). These methods do not make gradient estimates and involve relatively few function evaluations at each iteration. The most commonly used method in this class is due to Nelder and Mead (1965): this is a simplicial method which works using the repeated operations of reflection, expansion, and contraction applied to a simplex of $n+1$ points in $\Re^n$. Although the Nelder–Mead method was invented more than 30 years ago, some version of this approach probably still is the most common way to carry out optimization when each function evaluation requires a separate experiment; this is despite the apparent superiority of other direct search methods, such as that due to Powell (see Brent (1973), Powell (1964), and del Valle et al. (1990)).

Methods based on a simplicial approach have the disadvantage that for poorly behaved problems they can fail to converge. (There are even examples of well-behaved problems for which Nelder–Mead converges in theory to a nonstationary point; see McKinnon (1998).) This has been recognized for some time and has led to a variety of suggestions for modifications which can help the convergence behavior in practice (e.g., Parker, Cave, and Barnes (1985) and Hedlund and Gustavsson (1992)). At the same time researchers have developed versions of the basic simplicial algorithm that have provable convergence for certain classes of objective function. Examples include the work of Yu (1979), Rykov (1980), Torczon (1991, 1997), Lagarias et al. (1998), and Tseng (2000).

Another important restriction on methods in this class is that they have computation times that are heavily dependent on the dimension of the problem; they are not

*Received by the editors November 27, 1996; accepted for publication (in revised form) August 22, 2000; published electronically February 2, 2001.
http://www.siam.org/journals/siopt/11-3/31284.html
†Australian Graduate School of Management, University of New South Wales, Sydney 2052, Australia (eddiea@agsm.unsw.edu.au).
‡Computer Sciences Department, University of Wisconsin, Madison, WI 53706 (ferris@cs.wisc.edu). The research of this author was partially funded by the National Science Foundation and the Air Force Office of Scientific Research.

usually suitable for problems with more than a small number of variables. Indeed, for Nelder–Mead, there are even difficulties with convergence when the dimension of the problems becomes reasonably large. However, this disadvantage may be partially offset by the fact that some direct search algorithms are capable of easy parallelization; see Dennis and Torczon (1991).

A major factor in the continuing popularity of simplicial methods among users of optimization software is their ability to deal effectively with situations in which function evaluations are inaccurate. In this case more complex methods which approximate the function with some polynomial based on recent function evaluations (see Conn and Toint (1996) and Powell (1994)) may be led seriously astray: even the estimation of gradient information by finite differencing must be carried out carefully; see Gill et al. (1983).

In this paper we deal explicitly with the optimization of functions where the accuracy of the function evaluation depends on the time devoted to it. An example occurs when the function evaluation involves the solution of a PDE, where the accuracy depends on the grid size used. A second example occurs when attempting to optimize settings to achieve the maximum yield from a chemical reaction. Here the objective function is evaluated by carrying out a chemical experiment which is subject to random errors. For a given set of parameter values the experiment can be carried out many times over and then the average yield over the whole set of experiments gives an improved estimate of the objective function. Finally, the same situation occurs in the design of a facility, for example, a new warehouse, using a simulation model. The performance of the facility for some particular set of parameter values can be estimated using the simulation, and the longer the simulation is run the more accurate will be the result. In each of these examples, finding the best choice of parameter values requires the judicious balancing of time spent on improving the estimation of the objective function at a single point against time spent in making function evaluations at different parameter settings.

There are a variety of approaches to the problems of optimization with noise in the function evaluations. One approach is called the *response surface methodology* (see Khuri and Cornell (1987)). This is straightforward: an estimation of the behavior of the objective function around the current point x is obtained by making some kind of factorial experiment using points in the neighborhood of x. A regression fit of a low order polynomial (usually linear) is then made to these points. Then a line search is carried out in the negative gradient direction before the whole process is repeated.

A related method that uses a quadratic function which best fits the function values and chooses a descent direction based on this quadratic approximation has been proposed by Glad and Goldstein (1977). They establish a form of convergence result for the case where the noise is bounded. A similar approach has been used by Elster and Neumaier (1995), whose grid algorithm can be shown to be superior to Nelder–Mead on a variety of test problems when noise is included.

Another approach from the area of *stochastic approximation* is the Keifer–Wolkowitz method; see, for example, Kushner and Clark (1978) and Polyak (1987). This method estimates the gradient by evaluating the function at points $x \pm \alpha_k e_i$, where e_i are the n unit vectors and α_k is a constant depending on the iteration, and then taking a step of length γ_k in the negative gradient direction (rather than carrying out a line search). In order to obtain convergence when there is noise it is necessary to let γ_k and, especially α_k, tend to zero very slowly. There are various conditions but the crucial ones are that the infinite sum $\sum \gamma_k$ diverges and that the infinite sum

$\sum(\gamma_k/\alpha_k)^2$ converges. There are other stochastic approximation techniques which use increased sampling of $f(x)$ rather than decreasing step lengths to ensure convergence; see, for example, Dupuis and Simha (1991). There are also some stochastic approximation techniques that involve a line search; see Wardi (1988, 1990).

Barton and Ivey (1996) consider variations of the Nelder–Mead algorithm designed to cope with noisy function evaluations. These authors test alternative Nelder–Mead variants on a suite of test problems and use a stochastic noise term sampled from a truncated normal distribution which is added to the underlying function.

In this paper we introduce a new simplicial direct search method which is designed for use with function evaluations subject to noise. We will prove convergence of this method subject to some assumptions on the behavior of the objective function. We believe that this is the first time that an analysis of the convergence behavior of a direct search algorithm with unbounded random noise has been carried out. We also present some preliminary computational results which demonstrate that the new method can be effective in practice.

There are a number of points of interest. First, the algorithm includes a stochastic element. This is found to be advantageous in practice and is easily incorporated into the stochastic framework of our analysis. The reader should note that the new method is not a pattern search method in the usual sense: there is no reason for the points at which the function is evaluated to be drawn from any kind of regular grid of points in $\Re^n$.

Second, the stochastic nature of the function noise actually acts as an advantage in establishing convergence of the new algorithm. Paradoxically, we do not have a proof of convergence for this algorithm in the case where there is no function noise. The reason for this is that, in the absence of noise, our algorithm might make an infinite sequence of iterations without contracting the size of the structure, and with function improvements tending to zero, without this implying that the gradient is close to zero. Our algorithm contains nothing to stop it from cycling back to points closer to points that have been visited before.

Finally, the balance between function accuracy and step size is of interest. We show that convergence can be obtained when the standard deviation of the error of the function estimate decreases to zero faster than the step length size. There are valid reasons for thinking that convergence is unlikely if the standard deviation of the error becomes large in comparison to the step length, so our result may be the best possible result.

2. Description of the algorithm. The algorithm we propose operates with a set of m points in $\Re^n$ at each iteration (with $m \geq n+1$). This set of points is called a *structure*.

In specifying the algorithm we need to ensure that each structure is of full dimension. For our purposes, a convenient way to measure the extent to which the structure is "flat" is to define, for any structure $S = \{x_1, x_2, \ldots, x_m\}$,

$$d(S) = \min_{j=1,2,\ldots,m,\, u\in\Re^n,\, |u|=1} \left\{ \max_{k=1,2,\ldots,m} |(x_j - x_k)^T u| \right\}.$$

We also define the size of a structure S as

$$D(S) = \max_{j,k=1,2,\ldots,m} |x_j - x_k|.$$

We assume that there is an underlying objective function f defined on $\Re^n$ which we wish to minimize and that each function evaluation we make is subject to some random noise. Thus the apparent function value at a point x is $\hat{f}(x) = f(x) + \xi$, where ξ is drawn from a distribution with zero mean and finite variance σ^2. We will assume that the size of σ is under our control. For example, we might take N function evaluations at a single point x and estimate the underlying function value $f(x)$ by averaging the results, in which case varying N allows us to control σ. The noise on successive function evaluations is independent. We will require increasing accuracy in our function evaluations as the structure size decreases.

We can generate new structures from a given structure S with the operations of reflection around a point x or expansion around a point x (in each case x is one of the points of S):

$$\textbf{reflect}(S, x) = \{2x - x_i | x_i \in S\},$$

$$\textbf{expand}(S, x) = \{2x_i - x | x_i \in S\}.$$

We also need to define a structure $\textbf{contract}(S, x)$ which is the result of a contraction operation. Just as expansion will double the size of a structure, contraction is defined in a way which essentially halves the size of a structure. This enables us to define the *level*, $l(S)$, of a structure, S, such that the size of a structure S is a multiple $2^{-l(S)}$ of the size of the initial structure. A contraction operation increases the level by 1 and an expansion operation decreases the level by 1. More precisely, we assume that there are constants z_1 and z_2 such that for a structure S at level $l(S)$,

$$\frac{z_1}{2^{l(S)}} < d(S) < D(S) < \frac{z_2}{2^{l(S)}}. \tag{2.1}$$

We allow considerable freedom in the contraction operation. We require the structure $\textbf{contract}(S, x)$ to contain the point x (which again is a point in S) and be such that inequalities (2.1) will hold at all stages. There are a number of ways in which this can be done. One straightforward option is to take

$$\textbf{contract}(S, x) = \{0.5(x + x_i) | x_i \in S\}. \tag{2.2}$$

Another possibility is to apply a random rotation around x to the set $\{0.5(x+x_i)|x_i \in S\}$. A third option is to apply a random perturbation to the elements in a core structure.

We suppose that the accuracy of the function evaluations made at any point depends on the level of the structure. At a higher level, when structures are smaller, we will need greater accuracy. We assume that the points in a structure S at level l are each evaluated in such a way that the noise has standard deviation σ_l, where

$$2k_1 2^{-l(1+k_2)} \geq \sigma_l < k_1 2^{-l(1+k_2)}$$

for (small) constants $k_1 > 0$, $1/(m-2) > k_2 > 0$. Essentially this halves the standard deviation of the noise for each halving of structure size. Ignoring the small constant k_2, the standard deviation of the errors decreases at essentially the same rate as the size of the structure $D(S)$.

```
Given S^0 of full dimension and a sequence η_i > 0, i = 1, 2, ...
set l = l_0, i = 0, b = F(S^0)
while not satisfying termination criteria
begin
        v^i = v(S^i)
        T = reflect(S^i, v^i)
        if F(T) < F(S^i) then
        begin
                if F(T) < b then b = F(T)
                U = expand(T, v^i)
                if F(U) < b − η_l and l > 0 then
                begin
                        S^{i+1} = U
                        b = F(U)
                        l = l − 1
                end
                else S^{i+1} = T
        end
        else
        begin
                S^{i+1} = contract(S^i, v^i)
                if F(S^{i+1}) < b then b = F(S^{i+1})
                l = l + 1
         end
         i = i + 1
end
```

FIG. 2.1. *Pseudocode description of the algorithm.*

For a structure S we define its value function as

$$F(S) = \min\{\hat{f}(x_j) | x_j \in S\}$$

and its best point (that we call the pivot point) as

$$v(S) = \arg\min\{\hat{f}(x_j) | x_j \in S\}.$$

The algorithm operates at each stage by pivoting about the point $v(S)$ for the current structure. The basic operation of reflection in the pivot point is carried out until no further improvement can be made, when there is contraction around the pivot point and the whole process repeats. At each stage that reflection produces an improvement, an expanded structure is tested and accepted if this produces sufficient improvement on the best value so far. We can give a more formal definition of the algorithm with the pseudocode of Figure 2.1.

We shall assume that the algorithm has no memory of the points it has already evaluated, so that previous function evaluations at any point are not re-used if that point is revisited. This is also true for the pivot point when contraction takes place, but previous function values for the pivot point are used again for the reflection and expansion operations.

3. Convergence of the algorithm. In this section we establish convergence of the algorithm under the following assumptions:

A1 The function f is uniformly Lipschitz, continuously differentiable, and has compact lower level sets.

A2 The noise distribution is normal.

A3 The sequence η_i is bounded away from zero.

The first assumption is stronger than the assumptions made by Tseng (2000) and Torczon (1991) in their proofs of convergence for direct search methods where there is no noise component. These authors require that the function f be continuously differentiable and that the lower level set $L_0 = \{x \in \Re^n | f(x) \leq F(S^{(0)}\}$ be compact. Because of the stochastic nature of our algorithm, the structures generated do not necessarily contain a point in L_0 and this is one reason for the stronger assumption. We believe that some form of convergence will occur with a condition weaker than assumption A1: as it stands, this restriction rules out well-behaved functions such as $f(x) = x^T x$. The final assumption effectively ensures that the probability of expansion decreases to zero as the algorithm proceeds.

Our analysis here is stochastic and all probability statements we make are to be understood with respect to realizations of the noise process. We shall write $\gamma(x)$ for the probability in the tail of the standardized normal distribution, so $\gamma(x) = \text{Prob}(\xi \geq \sigma x)$, where the noise ξ has an $N(0, \sigma^2)$ distribution.

We need to make use of the following inequality on $\gamma(x)$; see, for example, Feller (1968):

$$\frac{1}{\sqrt{2\pi}} \left(\frac{1}{x} - \frac{1}{x^3} \right) e^{-x^2/2} \leq \gamma(x) \leq \frac{1}{\sqrt{2\pi}} \frac{1}{x} e^{-x^2/2}. \tag{3.1}$$

We write $\gamma_l(x)$ for the probability in the tail of the distribution for the errors in evaluations at level l; thus $\gamma_l(x) = \gamma(x/\sigma_l)$.

Before beginning the detailed analysis of the behavior of the algorithm it may be helpful to make some general remarks. The difficulty of the proof we give below arises primarily because the apparent function values are likely to increase whenever there is a contraction. To see why this is so, observe that, during the operation of the algorithm, we continually pivot around the best point in the structure: especially when the function is relatively flat, this is likely to be a point with a negative evaluation error. The more negative the error, and the lower the apparent function value, the more likely it is that none of the points in the reflected structure have an apparent value as small, and a contraction takes place. The pivot point is then re-evaluated and is equally likely to have positive or negative error, giving a high probability of an increase in the apparent function value. Most of the work in the proof (Lemmas 3.2 and 3.3) is required to establish that there is only a small probability of a large increase in apparent function value arising from a series of contractions.

We begin by showing that the structure size decreases to zero, i.e., that the level increases indefinitely.

LEMMA 3.1. *Under assumptions* A1 *and* A2, $l(S^i) \to \infty$ *as* $i \to \infty$ *with probability* 1. *If, in addition, assumption* A3 *holds, then with probability* 1 *there is only a finite number of expansions.*

Proof. We will prove this by regarding the algorithm as defining a stochastic process moving on the set of levels. First observe that, from assumption A1, the function f has a lower bound, which we will denote f^*.

We first suppose that the stochastic process is recurrent, so that there are infinitely many returns to some level i.

Since the function is uniformly Lipschitz, there is some number k such that the real function difference between two points in a structure S at level i is less than $kD(S) < kz_2(0.5)^i$. From the above remarks we know that there is a probability of at least $1/2$ that the pivot point has a negative error. (In what follows we use the term "error" to refer to $\hat{f}(x) - f(x)$ at some point x, where the error is "signed" rather than being an absolute magnitude.) Thus the probability of a contraction is always at least $q_i/2$, where q_i is the probability that all the points in the reflected structure (other than the pivot point) have errors greater than $kz_2(0.5)^i$. This means that the probability of remaining at level i indefinitely (without steps to other levels) is zero.

Suppose that there are infinitely many steps taken away from the level i. The implication is that there are infinitely many expansion steps at either level i or $i-1$. Without loss of generality we suppose that these occur at level i. At each expansion step at level i, b decreases at least by a fixed amount. Thus we may choose a number M, so that after $M+j$ expansions at level i, $b < f^* - Kj\sigma_{i-1}$ for some $K > 0$.

Expansion can take place only if the apparent function value at one of the points in the expanded structure is less than b. Thus the probability that an expansion step after $M+j$ returns to level i is less than the probability that one or more of the errors in the points used in the expanded structure is more negative than $-Kj\sigma_{i-1}$. Since we re-use the pivot point in expansion, there are $m-1$ of these points we need to consider. The probability that one or more errors are less than $-Kj\sigma_{i-1}$ is thus less than $m-1$ times the probability that a single error is less than $-Kj\sigma_{i-1}$. Hence, if we write p_j for the probability of an expansion at level i after the $M+j$th return to that level, then

$$\begin{aligned} p_j &< (m-1)\gamma_{i-1}(Kj\sigma_{i-1}) \\ &< \frac{m-1}{Kj\sqrt{2\pi}}\, e^{-(Kj)^2/2} \\ &< L^{-j^2} \end{aligned}$$

for some constant $L > 1$, provided j is chosen large enough.

We write $\tilde{p}_j$ for the probability that, after $M+j$ returns to level i, the next step away from i is to $i-1$ (rather than $i+1$). Now $\tilde{p}_j$ is equal to p_j divided by the probability of an expansion or contraction. This is no greater than $p_j/(p_j + q_i/2) < 2p_j/q_i$. The expected value of the remaining number of jumps from i to $i-1$ after M returns to level i is given by $\sum_{j=1}^{\infty} \tilde{p}_j < \sum_{j=1}^{\infty} 2p_j/q_i < (2/q_i)\sum_{j=1}^{\infty} L^{-j^2}$ which converges. Since this expectation is finite we can deduce that with probability 1 there is only a finite number of such jumps.

Thus with probability 1, the algorithm produces a sequence of levels which is not recurrent. Since the algorithm statement precludes l from becoming negative, the level must tend to ∞.

Under assumption A3, b decreases by at least some fixed amount at each expansion step, regardless of the level. We suppose that a sufficient number of expansion steps has taken place so that $b < f^* - K$, $K > 0$. Now consider the expected number of levels at which there is an expansion before a contraction on the first visit to the level.

This is less than

$$\begin{aligned}
&\sum_{l=1}^{\infty} \text{Prob (expansion before contraction at level l)} \\
&< \sum_{l=1}^{\infty} \frac{2(m-1)\gamma_l(K)}{q_l} \\
&< 2(m-1)\sum_{l=1}^{\infty} \frac{\gamma_l(K)}{\gamma_l(kz_2(0.5)^l)^{m-1}} \\
&< 2(m-1)\sum_{l=1}^{\infty} \frac{\gamma(K/\sigma_l)}{\gamma(kz_2(0.5)^l/\sigma_l)^{m-1}} \\
&< C_1 \sum_{l=1}^{\infty} \frac{\exp(-C_2/\sigma_l^2)}{\exp(-C_3(0.5)^{2l}/\sigma_l^2)} \\
&< C_1 \sum_{l=1}^{\infty} \exp(-C_2 k_1 4^{(1+k_2)l} + C_3 2k_1 4^{k_2 l})
\end{aligned}$$

for some positive constants C_1, C_2, C_3.

The constant C_1 here must be chosen greater than

$$Q(l) = 2(m-1)(\sqrt{2\pi})^{m-2}\frac{\sigma_l}{K}\left(\frac{1}{x} - \frac{1}{x^3}\right)^{-(m-1)}$$

for $x = kz_2(0.5)^l/\sigma_l > k\sqrt{k_1}z_2 2^{lk_2}/\sigma$. To show that this is possible observe that, when l is large, x will be large enough for $\left(\frac{1}{x} - \frac{1}{x^3}\right) > \frac{1}{2x}$ and so

$$\begin{aligned}
Q(l) &< 2(m-1)(\sqrt{2\pi})^{m-2}\frac{\sigma_l}{K}(2x)^{m-1} \\
&< C_4 2^{l(mk_2 - 2k_2 - 1)}
\end{aligned}$$

for some constant C_4, due to the upper bound on $N(l)$. This expression is bounded because of the choice of k_2.

Since for large enough l the term involving C_2 dominates that involving C_3, it is easy to see that this series has a finite value. Consequently there is probability 0 of there being an infinite number of levels at which there is an expansion before a contraction on the first visit to the level. Since we have already shown that with probability 1 there is only a finite number of expansions at any given level, we are done. □

The next step is to show that the probability of an increase in $\hat{f}$ of a fixed size is bounded in an appropriate way. Throughout our analysis we will continue to make assumptions A1, A2, and A3.

Note that Lemma 3.1 shows that with probability 1 there is an iteration I after which there are no further expansion steps. Let $L(j)$ be an iteration at which a contraction at level j takes place. We write f_j for the function value f at $v(S^{L(j)})$ and $\hat{f}_j$ for the apparent function value at $v(S^{L(j)})$ *after* contraction takes place. At this stage $v(S^{L(j)})$ is one of the points in $S^{L(j)+1}$ but need not be the pivot point. We will keep track of both the function values f_j and the apparent function values $\hat{f}_j$.

LEMMA 3.2. *Suppose that at step i of the algorithm at level $l = l(S^i)$, θ is chosen large enough so that*

$$2t > u > 2\sigma_l\sqrt{2},$$

where $t = \theta - f(v(S^i))$ *and* $u = \theta - \hat{f}(v(S^i))$*; then, if* $i > I$,

$$\text{Prob}(\max(f_l, \hat{f}_l) > \theta) \leq \gamma_l(t) + \alpha\gamma_l(u/2), \tag{3.2}$$

where $\alpha = m/\gamma(2\sqrt{2})^{m-1} - 1$.

Proof. The probability we require depends on the function values which occur in all the structures that are evaluated between step i of the algorithm and the point at which contraction takes place for this level. We establish the bound we require by maximizing this probability over choices of function values.

Let $V_n(t, u)$ be the maximum value for the probability that the maximum of f_l and $\hat{f}_l$ is greater than θ conditional on a contraction occurring before n iterations. This is the maximum value of the probability in (3.2) with the restriction that a contraction at level l occurs within n steps. The maximum is taken over the choice of function values at the points in the structure $\textbf{reflect}(S^i, v^i)$ and in succeeding structures.

There is a dynamic programming recursion which links V_n and V_{n-1} of the form

$$\begin{aligned} V_n(t,u) = \max_{\theta_1,\theta_2,\ldots,\theta_{m-1}} &(\text{Prob}\,(\text{immediate contraction})\gamma_{l+1}(t) \\ &+ \text{Prob}\,(\text{reflection})E[V_{n-1}(t',u')])\,, \end{aligned}$$

where θ_j are the real function values in $\textbf{reflect}(S^i, v^i)$. Here t' and u' are the new values of t and u after reflection. The $\gamma_{l+1}(t)$ in the first term is the probability that $\hat{f}_l$ is greater than θ if contraction occurs immediately (we have assumed $t > 0$ so $f(v^i) < \theta$).

Observe that $V_0(t, u)$ is maximized by taking each θ_j very large, forcing contraction. Hence $V_0(t, u) = \gamma_{l+1}(t) < \gamma_l(t)$ which satisfies the bound in (3.2). We will prove the result by induction on V_n. Once the bound is established for all values of n we are done. So we assume that the bound holds for $V_{n-1}(t, u)$.

Let $w = \hat{f}(v^i) - 2\sigma_l\sqrt{2}$. We shall consider two cases.

Case 1. Each $\theta_k > w$, $k = 1, 2, \ldots, m-1$.

We consider three possible events:

a. Contraction occurs at this step;
b. reflection occurs at this step and $f(v^{i+1}) > \theta - u/2$;
c. reflection occurs at this step and $f(v^{i+1}) \leq \theta - u/2$.

Let p_A, p_B, and p_C be the probabilities of these events. Since there will be a contraction if all the errors are greater than $2\sigma_l\sqrt{2}$,

$$p_A > \gamma_l(2\sigma_l\sqrt{2})^{m-1} = \gamma(2\sqrt{2})^{m-1}.$$

If B occurs, then, since the apparent pivot value is less than $\hat{f}(v^i)$, the error is less than $-u/2$. The probability that one of the errors in the reflected structure is less than $-u/2$ is less than $(m-1)\gamma_l(u/2)$ which is thus an upper bound on p_B.

Finally, note that if C occurs, then $t' = \theta - f(v^{i+1}) \geq u/2$ and, as the apparent function value at the new pivot is smaller than at the old one, $u' = \theta - \hat{f}(v^{i+1}) \geq u$. So if C occurs, then $V_{n-1}(t', u') \leq V_{n-1}(u/2, u)$. Using the fact that $V_{n-1}(t, u) \leq 1$, we have

$$V_n(t,u) \leq p_A\gamma_{l+1}(t) + p_B + p_C V_{n-1}\left(\frac{u}{2}, u\right).$$

Since $p_A \le 1$ and $p_C \le 1 - p_A \le 1 - \gamma(2\sqrt{2})^{m-1}$, we obtain the bound

$$\begin{aligned} V_n(t,u) &\le \gamma_{l+1}(t) + (m-1)\gamma_l\left(\frac{u}{2}\right) + (1-\gamma(2\sqrt{2})^{m-1})V_{n-1}\left(\frac{u}{2},u\right) \\ &\le \gamma_l(t) + (m-1)\gamma_l\left(\frac{u}{2}\right) + (1-\gamma(2\sqrt{2})^{m-1})(\alpha+1)\gamma_l\left(\frac{u}{2}\right) \\ &= \gamma_l(t) + (m+\alpha-\gamma(2\sqrt{2})^{m-1}(\alpha+1))\gamma_l\left(\frac{u}{2}\right) \\ &= \gamma_l(t) + \alpha\gamma_l\left(\frac{u}{2}\right) \end{aligned}$$

as required.

Case 2. At least one θ_k is less than w.

We now need to consider four possibilities:

a. Contraction occurs at this step;
b. reflection occurs and $f(v^{i+1}) \ge \theta - u/2$;
c. reflection occurs, $f(v^{i+1}) < \theta - u/2$, and $\hat{f}(v^{i+1}) \ge w$;
d. reflection occurs, $f(v^{i+1}) < \theta - u/2$, and $\hat{f}(v^{i+1}) < w$.

As in Case 1, we write p_A, p_B, etc. for the probabilities of these events, and we have the same upper bound on p_B. Also, as before, if C occurs, then $V_{n-1}(t',u') \le V_{n-1}(u/2,u)$. If D occurs, u' is larger and we obtain the stronger bound $V_{n-1}(t',u') \le V_{n-1}(u/2, u+2\sigma_l\sqrt{2})$. Thus

$$V_n(t,u) \le p_A\gamma_{l+1}(t) + p_B + p_C V_{n-1}\left(\frac{u}{2},u\right) + p_D V_{n-1}\left(\frac{u}{2},u+2\sigma_l\sqrt{2}\right).$$

Since $p_D \le 1 - p_C$, we have

$$\begin{aligned} V_n(t,u) \le \gamma_{l+1}(t) &+ (m-1)\gamma_l\left(\frac{u}{2}\right) + p_C V_{n-1}\left(\frac{u}{2},u\right) \\ &+ (1-p_C)V_{n-1}\left(\frac{u}{2},u+2\sigma_l\sqrt{2}\right). \end{aligned}$$

The event C can occur only if all θ_k with a value less than w have a positive error, so $p_C \le 1/2$. The right-hand side of the inequality above is maximized when $p_C = 1/2$, so we obtain

$$\begin{aligned} V_n(t,u) &\le \gamma_l(t) + (m-1)\gamma_l\left(\frac{u}{2}\right) + \frac{\alpha+1}{2}\gamma_l\left(\frac{u}{2}\right) \\ &\quad + \frac{1}{2}\gamma_l\left(\frac{u}{2}\right) + \frac{\alpha}{2}\gamma_l\left(\frac{u}{2}+\sqrt{2}\sigma_l\right) \\ &= \gamma_l(t) + \left(m+\frac{\alpha}{2}\right)\gamma_l\left(\frac{u}{2}\right) + \frac{\alpha}{2}\gamma_l\left(\frac{u}{2}+\sqrt{2}\sigma_l\right). \end{aligned}$$

Now

$$\begin{aligned} \gamma_l\left(\frac{u}{2}+\sqrt{2}\sigma_l\right) &= \gamma\left(\frac{u}{2\sigma_l}+\sqrt{2}\right) \\ &< \frac{1}{\sqrt{2\pi}}\,\frac{2\sigma_l}{u+2\sigma_l\sqrt{2}}\,\exp\left(\frac{-u^2}{8\sigma^2}\right)\exp(-1) \\ &< \frac{2}{e\sqrt{2\pi}}\left(\frac{2\sigma_l}{u} - \left[\frac{2\sigma_l}{u}\right]^3\right)\exp\left(\frac{-u^2}{8\sigma^2}\right), \end{aligned}$$

since $(2\sigma_l/u)^3 < \sigma_l/u$. Thus, again using inequality (3.1),

$$\gamma_l\left(\frac{u}{2}+\sqrt{2}\sigma_l\right) < \frac{2}{e}\gamma\left(\frac{u}{2\sigma_l}\right) = \frac{2}{e}\gamma_l\left(\frac{u}{2}\right).$$

So

$$V_n(t,u) \le \gamma_l(t) + \left(m+\frac{\alpha}{2}+\frac{\alpha}{e}\right)\gamma_l\left(\frac{u}{2}\right).$$

It is now easy to check that α is large enough for $m+(\alpha/2)+(\alpha/e)<\alpha$ which is the inequality we require to establish the bound in this case. □

Lemma 3.2 relates to the probability of achieving a high function value (or apparent function value) immediately after the next contraction. The next result is concerned with the probability of achieving a high value at any point after the current iteration. We will prove this by stringing together applications of Lemma 3.2. Let $W(\delta,i)$ be the probability that there is some level j, $j\ge i$, with either $\hat{f}_j > \hat{f}_i+\delta$ or $f_j > \hat{f}_i+\delta$. Let $\bar{W}(\delta,i)$ be the probability $W(\delta,i)$ conditional on $i>I$.

LEMMA 3.3. *If i_0 is large enough, then $\bar{W}(\delta,i_0)\to 0$ as $\delta\to\infty$, and for any $\delta>0$, $W(\delta,i_0)\to 0$ as $i_0\to\infty$.*

Proof. First note the identity $\sum_{k=1}^{\infty} k/2^k = 2$. Thus $\bar{W}(\delta,i_0)$ is less than the probability, conditional on $i>I$, that for some j, $j\ge i_0$, either $\hat{f}_j$ or f_j is greater than

$$\Delta_j = \hat{f}_{i_0} + \delta/2 + (\delta/4)\sum_{k=1}^{j-i_0} k/2^k.$$

We can bound $\bar{W}(\delta,i_0)$ by the sum of the probabilities $q_j, j\ge i_0$, where q_j is the probability that either $\hat{f}_j$ or f_j is greater than Δ_j, but that $\hat{f}_h$ and f_h are both less than Δ_h for $i_0\le h<j$; i.e., q_j is the probability that one of the inequalities is broken for the first time at level j.

Now q_{i_0} is the probability that $f_{i_0} > \hat{f}_{i_0}+\delta/2$, so $q_{i_0}=\gamma_{i_0}(\delta/2)$.

In general, for $j > i_0$ we wish to apply Lemma 3.2 to bound q_j. We know that both $\hat{f}_{j-1}$ and f_{j-1} are less than Δ_{j-1}. If $v^{L(j-1)}$ is also the pivot point in $S^{L(j-1)+1}$, then we will apply Lemma 3.2 with $i=L(j-1)+1$. Otherwise we let $S^* = \textbf{reflect}(S^{L(j-1)+1}, v^{L(j-1)})$ be an artificial predecessor for $S^{L(j-1)+1}$. The bound in Lemma 3.2 is independent of the apparent function values at the points in structure S^i other than the pivot point. Hence we can apply Lemma 3.2 with $i=L(j-1)$ and using S^* instead of $S^{L(j-1)}$. In either case we obtain

$$\begin{aligned} q_j &< \gamma_j(\Delta_j - f_{j-1}) + \alpha\gamma_j\left(\frac{\Delta_j-\hat{f}_{j-1}}{2}\right) \\ &< \gamma_j\left(\frac{\delta(j-i_0)}{2^{j-i_0+2}}\right) + \alpha\gamma_j\left(\frac{\delta(j-i_0)}{2^{j-i_0+3}}\right) \\ &< (1+\alpha)\gamma_j\left(\frac{\delta(j-i_0)}{2^{j-i_0+3}}\right). \end{aligned}$$

In order to apply Lemma 3.2 we require that $\delta(j-i_0)/(2^{j-i_0+2})$ be greater than $2\sigma_j\sqrt{2}$. This inequality will hold provided i_0, and hence j, is chosen large enough.

Hence

$$\bar{W}(\delta, i_0) \leq \gamma_{i_0}(\delta/2) + \sum_{j=i_0+1}^{\infty} (1+\alpha)\gamma_j \left(\frac{\delta(j-i_0)}{2^{j-i_0+3}} \right).$$

Now $\sigma^2_{i_0+j} \leq \sigma^2_{i_0}/4^{j(1+k_2)}$, so we can use Chebyshev's inequality to show that

$$\gamma_{i_0+j}(x) \leq \frac{\sigma^2_{i_0}}{x^2 4^{j(1+k_2)}}.$$

Thus, for $j \geq 1$,

$$\gamma_{i_0+j}\left(\frac{\delta j}{2^{3+j}} \right) \leq \frac{\sigma^2_{i_0}}{\delta^2 j^2 4^{jk_2-3}}.$$

It also follows from Chebyshev's inequality that $\gamma_{i_0}(\delta/2) \leq 4\sigma^2_{i_0}/\delta^2$ and so

$$\bar{W}(\delta, i_0) \leq \frac{\sigma^2_{i_0}}{\delta^2} \left(4 + 64(1+\alpha) \sum_{j=1}^{\infty} \frac{1}{j^2 4^{jk_2}} \right).$$

Since the infinite sum converges, the first part of the result follows.

The second part of the result follows from observing that

$$W(\delta, i_0) \leq \text{Prob}(I > i_0) + \bar{W}(\delta, i_0),$$

where the first term tends to zero from Lemma 3.1 and the second term also tends to zero since $\sigma_{i_0} \to 0$ as $i_0 \to \infty$. □

With these three lemmas established we can now go on to prove our main result. We will show that, starting from a point with nonzero gradient, there is, for a high enough level, a high probability of a succession of reflection steps leading to a reduction in $\hat{f}_j$ of a certain size, where the amount of reduction is independent of the level. Since the probability of ever seeing the same increase in $\hat{f}_j$ decreases to zero from Lemma 3.3, the probability of an infinite number of returns to a neighborhood of the initial point is zero. The consequence is that any cluster point has zero gradient—though we cannot establish that the cluster point is a local minimum.

This theorem concerns the behavior of a single sequence $\{v^i\}$ generated from running the algorithm. This may have multiple cluster points, but we will show that with probability 1 they will all have the same function value. Notice, however, that a new sequence generated from running the algorithm again might converge to a point with a different function value.

THEOREM 3.4. *If $\{v^i\}$ is the sequence of pivot points occurring when the algorithm is run, and assumptions* A1, A2, *and* A3 *hold, then with probability* 1 *there is a cluster point v^* for the sequence v^i and with probability* 1, *$\nabla f(v^*) = 0$ for each such cluster point. Moreover, with probability* 1 *each cluster point has the same function value $\tilde{f} = f(v^*)$ and $\hat{f}(v^i)$ converges in probability to $\tilde{f}$.*

Proof. It will be convenient in the proof below to assume that $z_2 \leq 1$. Since z_2 is an upper bound on the size of the structure at level 0, we can make this assumption without any loss of generality.

We begin by discarding the first iterations of the algorithm until we reach a position in which the first statement of Lemma 3.3 applies. We choose an arbitrary

$H_0 > 0$ and write v^0 for the pivot point after H_0 further steps of the algorithm. Let $\epsilon_1 > 0$ be arbitrary. Using Lemma 3.3 we can choose M large enough so that, conditional on there being no more expansion steps, there is a probability of less than ϵ_1 that $f_j > \hat{f}(v^0) + M$. Let $S = \{x | f(x) \leq \hat{f}(v^0) + M\}$ which is compact by assumption A1. Hence there is a probability of at least $1 - \epsilon_1$ that all the points at which there is a contraction are in S. However, in this case there must be a cluster point in S. Since ϵ_1 was arbitrary, this establishes that, conditional on $I < H_0$, there is a cluster point with probability 1. Now, since this holds for all choices of H_0, we deduce that there is a cluster point with probability 1.

If v^* is a cluster point for v^i and we choose a subsequence $v^{k(i)}$ which converges to v^*, then $f(v^{k(i)}) \to f(v^*)$. Now observe that the choice of pivot point is influenced by the errors in evaluating all the points of a structure, but the error is equal to one of at most $2m - 1$ independent evaluation errors which can be involved at each step. Using Lemma 3.1 all evaluation errors approach zero, and hence $\hat{f}(v^{k(i)})$ converges in probability to $f(v^*)$.

Suppose there is a subsequence $v^{k(i)}$ with $\hat{f}(v^{k(i)})$ converging to $f' > f(v^*)$. Since the apparent function values of the pivots can decrease only when the step is not a contraction, this implies that there is an infinite subsequence of $\hat{f}_j$ which is greater than or equal to f'. But since the $\hat{f}_j$ values must also make infinitely many visits to a neighborhood of $f(v^*)$, Lemma 3.3 implies that this occurs with probability 0. The same argument can be used to show that there is probability 0 of a subsequence with apparent values converging to $f' < f(v^*)$.

We continue under the assumption of arbitrary $\epsilon_1 > 0$ and hence of the compact set S. Since f is continuously differentiable, ∇f is uniformly continuous on S. Let $\epsilon_2 > 0$ be arbitrary. Choose $\delta_1 > 0$ such that for any $y \in S$, $|\nabla f(x) - \nabla f(y)| < (z_1/z_2)(\epsilon_2/4)$ for all $|x - y| < \delta_1$. Let $\delta_2 = z_1\delta_1/(64 + z_1)$.

Since S is compact we can find a finite set $y_1, y_2, \ldots, y_k$ with

$$S \subset \bigcup_{j=1,2,\ldots,k} B_{\delta_2}(y_j).$$

Let $A = B_{\delta_2}(y^*)$ be chosen from among this cover of S. We suppose that $3\epsilon_2/2 > |\nabla f(x')| > \epsilon_2$ for some $x' \in A$. From the definition of δ_1, $2\epsilon_2 > |\nabla f(x)| > \epsilon_2/2$ for every $x \in B = B_{\delta_1}(y^*)$.

Consider the steps of the algorithm while within B. We suppose that the current pivot point is v^h and $l = l(S^h)$. If ∇f was constant within B and there were no evaluation errors, then we would obtain an improvement of $\max_j |\nabla f^T(v^h - x_j)|$, where x_j are the other points in the current structure. This happens because we can choose either $v^h + (v^h - x_j)$ or $v^h - (v^h - x_j)$ as the next pivot: since the function values at these two points bracket that at v^h, only the one with the lower value appears in **reflect**(S^h, v^h).

From the definition of $d(S)$ we see that this improvement is at least $|\nabla f|\, d(S^h) > d(S^h)\epsilon_2/2$. Now we need to consider the variation in ∇f over B. The actual function value at the new and improved point x_j may not be as low as would be predicted on the basis of a constant gradient of $\nabla f(v^h)$. Using the mean value theorem we have

$$f(x_j) = f(v^h) + \nabla f(v^h)^T(x_j - v^h) + (\nabla f(x) - \nabla f(v^h))^T(x_j - v^h)$$

for some x on the line segment (v^h, x_j). Thus the overall improvement predicted could be reduced by at most $|\nabla f(x) - \nabla f(v^h)||x_j - v^h|$, and since $|x_j - v^h| < D(S^h)$, the

reduction is less than $d(S^h)\epsilon_2/4$. Hence, unless at least one evaluation error among the $2m-1$ errors in the current and reflected structure is greater than $d(S^h)\epsilon_2/8$ in magnitude, we will observe an improvement of at least $d(S^h)\epsilon_2/8$ in both the actual and apparent values of f.

Suppose we start with the pivot point v^h in the smaller ball A. Consider a sequence of $\kappa = (\delta_1 - \delta_2)/D(S^h)$ steps of the algorithm. Note that throughout these steps the pivot point will be within B. If there is an improvement of $d(S^h)\epsilon_2/8$ at each step, then the total improvement after these κ steps is at least

$$(\delta_1 - \delta_2)\frac{d(S^h)\epsilon_2}{8D(S^h)} > \frac{64\delta_1}{64+z_1}\frac{\epsilon_2 z_1}{8z_2} > 8\epsilon_2\delta_2,$$

where we use the fact that $z_2 < 1$.

We will obtain this overall improvement unless at one of these κ steps the improvement is less than $d(S^h)\epsilon_2/8$. At each step there is an improvement of less than $d(S^h)\epsilon_2/8$ with probability that is less than $(2m-1)\gamma_l(d(S^h)\epsilon_2/8)$. Hence the probability of not seeing the overall improvement is less than

$$\kappa(2m-1)\gamma_l(d(S^h)\epsilon_2/8) = \frac{\delta_1-\delta_2}{D(S^h)}(2m-1)\gamma(d(S^h)\epsilon_2/(8\sigma_l)).$$

However, $d(S^h)/\sigma_l > z_1\sqrt{k_1}2^{lk_2}$, and so, using inequality (3.1), it is easy to see that there will be a term in $\exp(-2^{2lk_2})$ arising from the γ function which will dominate the multiplier $1/D(S^h)$. Thus the probability of not obtaining the overall improvement $8\epsilon_2\delta_2$ approaches zero as the level l approaches ∞. Hence, we can choose an iteration h large enough that the probability of an improvement of this size is greater than $1/2$.

The maximum difference in function values between points in A is $4\epsilon_2\delta_2$. We choose h large enough so that the probability of the error at v^h being greater than $\epsilon_2\delta_2$ is less than $1/2$. Thus with probability at least $1/4$ both the apparent and actual function values are at least $3\epsilon_2\delta_2$ less than the smallest function value in A by the end of the sequence of κ steps in B. We choose h large enough for $W(3\epsilon_2\delta_2, l) < 1/2$. So provided h is sufficiently large, the probability of never returning to A is at least $1/8$. Thus the probability of an infinite number of returns is zero.

This establishes that the probability of a cluster point occurring in A is zero. Hence, with probability 1 the cluster points of the pivot sequence occur in sets $B_{\delta_2}(y_j)$ in which every point has either $|\nabla f| > 3\epsilon_2/2$ or $|\nabla f| < \epsilon_2$. Therefore at each cluster point v^* either $|\nabla f(v^*)| > 3\epsilon_2/2$ or $|\nabla f(v^*)| < \epsilon_2$. Since ϵ_2 is arbitrary, the appropriate choice of ϵ_2 rules out any strictly positive value of $\nabla f(v^*)$. Thus $\nabla f(v^*) = 0$. □

4. Computational results. In this section we report the results of some limited computational testing of the algorithm. Our aim is to investigate the performance of the new algorithm on a small set of test functions, and we will look at its behavior both when function evaluations are noisy and when they are not. The algorithm requires the user to set some parameters, to choose an initial structure, and to determine whether or not to apply some sort of perturbation on contraction. The experiments we report are enough to indicate a reasonable choice of structure and settings for the parameters, as well as to demonstrate the value of allowing a perturbation when there is controlled noise.

The algorithm we use is as described in section 2. The contraction operation (in the absence of a perturbation) just uses the definition of (2.2).

We have chosen to use a test suite of 11 problems. Two of these are well known: *rosen* and *powell*. The Rosenbrock function *rosen* (Moré, Garbow, and Hillstrom 1981) is a two-dimensional problem with a single banana-shaped valley and is known to cause difficulties, particularly for direct search algorithms. The test function *powell* (Powell 1970) of dimension four is also well known. The problems *camel*, *ill*3*a*, *ill*3*b*, *sincusp*, and *smalla* are taken from the problem set given in Aluffi-Pentini, Parisi, and Zirilli (1988), being problem numbers 6, 10, 13, 35, and 37 from this set. The other four problems are straightforward quadratic functions. Problems *quad*6 and *quad*6*bad* are of dimension six with *quad*6*bad* being poorly conditioned (the reciprocal condition number is approximately 1e-6). Similarly, *quad*10 and *quad*10*bad* are quadratics of dimension 10, with the second function being poorly conditioned. All problems have small dimension; as we have already observed, direct search methods do not perform well when problems have a large number of variables.

We shall investigate two versions of the initial structure. Each version has the same basic "cross" form with points radiating from a central point along all the axes. The smaller structure S_A uses points

$$x^0, x^0 \pm \xi e^i, x^0 \pm 2\xi e^i.$$

Here e^i represents the ith unit vector in $\Re^n$and ξ is a scaling parameter initially set to 1. The larger structure S_B adds the points $x^0 \pm 3\xi e^i$ and $x^0 \pm 5\xi e^i$ to S_A.

We ran the algorithm from 200 randomly generated starting points whose coordinates were uniformly generated within $(-10, 10)$. In each case we continued until the size of the structure was reduced so that adjacent points in the structure were within a distance 0.0001 of each other. This roughly parallels the stopping criteria used by Barton and Ivey (1996).

The first set of runs is reported in Table 4.1. These describe the performance of the algorithm for each test problem when there is no noise. In the absence of noise, η_l is the only parameter to set. We choose $\eta_l = 10^{-8}$ at every step. Notice that in this case the algorithm is a pattern search method and the results of Torczon (1997) will imply convergence of the algorithm.

In the table we report, for both structures S_A and S_B, the average number of iterations, the average number of function evaluations required, the mean error in the solution value, and the average distance of the final point found to the solution. In addition we report the number of failures of the algorithm in the final column—the first figure is the number of runs (out of 200) in which both the error in the solution value and the distance to the known solution are greater than 10^{-2}. The second figure in parentheses is the number of major failures—identified as solutions in which both the error in the solution value and the distance to the solution are more than 10^{-1}. The problems *camel*, *ill*3*a*, *ill*3*b* all have multiple local optima. For these problems we take the closest local optima as the correct solution—this seems appropriate since the algorithm has not been designed to find a global optimum (even though in the early part of a run the method may well avoid being trapped in a local optimum in circumstances where more sophisticated techniques could fail).

It can be observed that the use of the larger structure S_B does not produce a significantly better-quality solution as measured by the distance to the solution or by the error in the function value. For *rosen*, and the ill-conditioned problems *quad*6*bad* and *quad*10*bad*, the larger structure produces faster convergence with fewer function evaluations, but for the other functions the average number of function evaluations is smaller using S_A. Overall there is no significant advantage in using a larger structure.

TABLE 4.1
No noise, no perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	409	6696	0.2	0.4	9(9)
	S_B	59	1934	0.14	0.17	4(4)
camel	S_A	20	337	2.9e-06	0.078	0(0)
	S_B	20	682	5.2e-06	0.086	0(0)
ill3a	S_A	20	341	1.3e-06	0.00037	0(0)
	S_B	20	672	8.8e-06	0.00099	0(0)
ill3b	S_A	19	325	0.0053	0.015	6(4)
	S_B	22	725	0.015	0.048	20(10)
powell	S_A	62	2024	0.00051	0.14	0(0)
	S_B	46	3021	0.00013	0.094	0(0)
sincusp	S_A	43	1739	0.041	0.0012	0(0)
	S_B	40	3295	0.065	0.003	0(0)
smalla	S_A	45	1845	1.1e-06	0.001	0(0)
	S_B	40	3254	6.5e-06	0.0025	0(0)
quad6	S_A	47	2293	2.7e-06	0.0012	0(0)
	S_B	46	4475	1.6e-05	0.003	0(0)
quad6bad	S_A	452	21888	0.00063	11	0(0)
	S_B	156	15091	0.00042	10	0(0)
quad10	S_A	75	6064	1.2e-05	0.002	0(0)
	S_B	71	11433	7.3e-05	0.0049	0(0)
quad10bad	S_A	2346	188487	0.0011	15	0(0)
	S_B	710	113956	0.00084	14	1(0)

The next set of results relates to the performance of the algorithm with random perturbations to the structure whenever a contraction is carried out. Thus, instead of using the structures outlined above, we use the following for S_A:

$$x^0, x^0 \pm \xi(e^i + p_1^i), x^0 \pm 2\xi(e^i + p_2^i), x^0 \pm 3\xi(e^i + p_3^i),$$

where each component of the perturbations p_j^i are selected from a uniform distribution on $(-0.5, 0.5)$. S_B is similar except that the points $x^0 \pm 5\xi(e^i + p_4^i)$ are added.

Whenever a contraction is carried out by the algorithm, the following steps are performed:

1. Remove the current perturbation;
2. contract towards the pivot point;
3. add a new randomly generated perturbation.

Note that the perturbations may re-align the structure, and this can be advantageous if it enables the structures to follow the geometry of the particular problem. Table 4.2 was generated using precisely the same mechanism as outlined for Table 4.1, except that random perturbations were incorporated.

The algorithm with perturbation often reaches a good solution more quickly than when there is no perturbation. This happens in 8 out of the 11 problems considered here. It is interesting that the speedup occurs on those problems where the algorithm without perturbation takes a large number of iterations; however, there are insufficient results here to draw strong statistical conclusions. Moreover, the solution quality with perturbations is often a little worse, so the superiority of perturbation is not clearly established. Just as in the case without perturbations, the advantage of using a larger structure is not significant given the greater number of function evaluations required.

TABLE 4.2
No noise, with perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	46	770	0.065	0.2	14(7)
	S_B	38	1238	0.037	0.11	6(4)
camel	S_A	20	350	1.6e-06	0.085	0(0)
	S_B	20	671	1.4e-06	0.097	0(0)
ill3a	S_A	21	361	6.3e-07	0.00028	0(0)
	S_B	20	674	6.9e-07	0.00024	0(0)
ill3b	S_A	24	403	0.0039	0.011	4(2)
	S_B	24	800	0.0023	0.0076	4(1)
powell	S_A	48	1558	0.00057	0.12	0(0)
	S_B	38	2462	0.00013	0.079	0(0)
sincusp	S_A	31	1284	0.048	0.0018	0(0)
	S_B	29	2358	0.04	0.0012	0(0)
smalla	S_A	31	1259	2e-06	0.0013	0(0)
	S_B	28	2320	1.1e-06	0.00097	0(0)
quad6	S_A	36	1754	1.1e-05	0.0024	0(0)
	S_B	32	3112	5e-06	0.0016	0(0)
quad6bad	S_A	57	2778	0.0011	12	1(0)
	S_B	74	7161	0.0011	12	4(0)
quad10	S_A	52	4234	7.4e-05	0.005	0(0)
	S_B	44	7200	3.8e-05	0.0035	0(0)
quad10bad	S_A	137	11045	0.0057	20	27(0)
	S_B	204	32792	0.0038	18	13(0)

Tables 4.3 and 4.4 repeat Tables 4.1 and 4.2 in the presence of noise. We deal with the case where the standard deviation of the noise is controlled. This is the situation for which the convergence of the algorithm has been established theoretically. We set $k_1 = 1$ and $k_2 = 0.1$ and hold other elements of the experiments as they were for Tables 4.1 and 4.2. The mean error is calculated from the underlying function value f at the final pivot point.

Observe that in Table 4.3 there are significant numbers of failures for *quad*6*bad* and *quad*10*bad*, but the overall performance of the algorithm in Tables 4.3 and 4.4 is reasonable. If Tables 4.3 and 4.4 are compared, one can see that, in almost all cases, using perturbation produces better-quality solutions with fewer function evaluations. For *rosen* the number of nonmajor failures is worse with perturbation. Nevertheless, it seems that perturbation offers an overall advantage on this set of test problems. It is possible that a similar perturbation approach could be beneficially applied to other direct search algorithms.

Using a larger structure gives better results, but there is some penalty in function evaluations. With both perturbation and the larger structures there are only 4 major failures in the complete set of 2200 starts, and in this case the performance of the algorithm is remarkably good. It is interesting that, with this setup, there is remarkably little degradation in performance due to the introduction of noise, as can be seen by a comparison of the relevant parts of Tables 4.2 and 4.4. Indeed, for some problems there appears to be an improvement in performance. Observe that for *rosen* there is one less major failure and smaller average errors than were achieved in any of the noiseless cases, and there are also substantially fewer function evaluations required for both *quad*6*bad* and *quad*10*bad*.

TABLE 4.3
Controlled noise, no perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	407	6658	0.2	0.43	11(10)
	S_B	62	2030	0.14	0.18	4(4)
camel	S_A	21	358	5.6e-06	0.074	0(0)
	S_B	22	723	5.1e-06	0.082	0(0)
ill3a	S_A	21	360	4.2e-06	0.00087	0(0)
	S_B	22	731	5.7e-06	0.00088	0(0)
ill3b	S_A	19	327	0.0053	0.015	6(4)
	S_B	22	730	0.016	0.051	21(11)
powell	S_A	60	1964	0.00075	0.16	0(0)
	S_B	44	2862	0.00026	0.11	0(0)
sincusp	S_A	42	1720	0.041	0.0012	0(0)
	S_B	40	3283	0.064	0.0029	0(0)
smalla	S_A	43	1766	4e-05	0.006	0(0)
	S_B	43	3459	1.6e-05	0.0037	0(0)
quad6	S_A	46	2272	3.9e-05	0.0049	0(0)
	S_B	48	4699	2.1e-05	0.0035	0(0)
quad6bad	S_A	105	5105	0.028	12	167(0)
	S_B	53	5111	0.0042	12	14(0)
quad10	S_A	76	6134	5.4e-05	0.0043	0(0)
	S_B	75	12077	5.6e-05	0.0043	0(0)
quad10bad	S_A	131	10554	0.037	17	198(2)
	S_B	63	10249	0.014	20	85(1)

TABLE 4.4
Controlled noise, with perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	45	744	0.1	0.3	25(10)
	S_B	37	1225	0.018	0.073	6(3)
camel	S_A	21	356	5.4e-06	0.086	0(0)
	S_B	20	670	3.9e-06	0.096	0(0)
ill3a	S_A	21	365	4.3e-06	0.00079	0(0)
	S_B	20	677	2.8e-06	0.00061	0(0)
ill3b	S_A	24	402	0.0039	0.011	4(2)
	S_B	24	812	0.0018	0.006	3(1)
powell	S_A	47	1538	0.00074	0.13	0(0)
	S_B	37	2405	0.00017	0.088	0(0)
sincusp	S_A	32	1315	0.046	0.0017	0(0)
	S_B	29	2368	0.04	0.0012	0(0)
smalla	S_A	31	1293	1.5e-05	0.0037	0(0)
	S_B	29	2402	1e-05	0.003	0(0)
quad6	S_A	36	1783	2.4e-05	0.0038	0(0)
	S_B	32	3134	1.5e-05	0.0029	0(0)
quad6bad	S_A	47	2277	0.0027	13	3(0)
	S_B	34	3357	0.0016	12	0(0)
quad10	S_A	52	4221	8.3e-05	0.0054	0(0)
	S_B	45	7234	4.9e-05	0.0041	0(0)
quad10bad	S_A	71	5723	0.0074	17	43(0)
	S_B	53	8581	0.0075	17	40(0)

TABLE 4.5
Uncontrolled noise, no perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	20	342	2.7	2.5	190(140)
	S_B	20	682	1.9	2.1	192(124)
camel	S_A	21	362	0.04	0.13	156(15)
	S_B	21	710	0.038	0.13	161(8)
ill3a	S_A	21	363	0.05	0.086	161(25)
	S_B	21	705	0.049	0.084	151(17)
ill3b	S_A	20	350	0.08	0.047	31(9)
	S_B	22	743	0.082	0.093	51(21)
powell	S_A	29	974	0.13	0.44	192(83)
	S_B	29	1932	0.11	0.4	191(64)
sincusp	S_A	31	1267	0.7	0.41	200(169)
	S_B	30	2458	0.64	0.38	200(160)
smalla	S_A	31	1277	0.13	0.34	200(112)
	S_B	30	2444	0.12	0.33	200(103)
quad6	S_A	33	1637	0.17	0.31	199(141)
	S_B	33	3221	0.16	0.3	200(146)
quad6bad	S_A	25	1223	0.2	13	194(114)
	S_B	24	2377	0.18	13	195(108)
quad10	S_A	47	3834	0.32	0.33	200(193)
	S_B	47	7645	0.31	0.32	200(191)
quad10bad	S_A	27	2242	0.56	18	200(188)
	S_B	26	4256	0.37	17	200(177)

TABLE 4.6
Uncontrolled noise, with perturbation, 200 *starts for each case.*

Problem	Structure	Average				Failures
		Iters	Evals	Obj error	Distance	(major)
rosen	S_A	22	372	1.7	1.8	188(119)
	S_B	22	722	1.3	1.7	178(117)
camel	S_A	21	362	0.041	0.13	155(20)
	S_B	21	700	0.031	0.13	145(10)
ill3a	S_A	22	370	0.038	0.075	145(14)
	S_B	21	703	0.033	0.066	149(11)
ill3b	S_A	23	390	0.054	0.046	32(7)
	S_B	23	754	0.056	0.057	35(11)
powell	S_A	30	995	0.15	0.46	196(95)
	S_B	28	1827	0.075	0.33	180(43)
sincusp	S_A	28	1157	0.61	0.36	200(159)
	S_B	28	2282	0.46	0.2	199(106)
smalla	S_A	27	1108	0.11	0.31	198(92)
	S_B	27	2241	0.079	0.26	193(63)
quad6	S_A	29	1451	0.13	0.27	198(108)
	S_B	28	2789	0.1	0.24	199(87)
quad6bad	S_A	24	1210	0.14	13	189(98)
	S_B	24	2331	0.13	13	186(80)
quad10	S_A	38	3136	0.28	0.31	200(185)
	S_B	36	5869	0.19	0.26	200(172)
quad10bad	S_A	27	2190	0.32	18	200(159)
	S_B	25	4139	0.31	17	200(162)

Finally, in Tables 4.5 and 4.6 a fixed standard deviation of the noise is used, with σ set to 0.1 at all levels. As before, the initial value of the scaling parameter is set to 1. These tables demonstrate how hard it is for direct search algorithms to perform well in the presence of uncontrolled noise. There are large numbers of major failures for all the functions tested.

5. Discussion and conclusions. We have presented an algorithm for the optimization of functions that are subject to stochastic error in their evaluations. The algorithm is shown to have a cluster point that is a stationary point of the function with probability 1. This is established using a novel proof technique.

The computational results show the method is effective in the presence of noise if this is controlled in the appropriate way. Our proof of convergence holds in the case where we allow a perturbation of the structure on contraction. Using this perturbation approach improves the computational performance of the algorithm on the test set we have considered when there is controlled noise. It is possible that the introduction of a perturbation of this kind could improve the performance of other direct search methods. A significant contribution of this paper is the introduction of this algorithmic innovation together with a proof that the method converges in this case.

However, the convergence result we establish here is weaker than would be desirable. In effect, we have arranged the test for expansion to ensure that with probability 1 there are only a finite number of expansion steps, and then we established convergence when no more expansions take place. Nevertheless there seems no good reason to suppose that convergence will fail when there is an infinite number of expansion steps, i.e., under conditions weaker than assumption A3. For example, we might allow η_i to tend to zero but ask that $\sum \eta_i$ diverge. However, a result of this kind seems hard to prove.

REFERENCES

F. Aluffi-Pentini, V. Parisi, and F. Zirilli (1988), *Algorithm* 667. *SIGMA—a stochastic-integration global minimization algorithm*, ACM Trans. Math. Software, 14, pp. 366–380.

R. Barton and J. S. Ivey (1996), *Nelder Mead simplex modifications for simulation optimization*, Management Sci., 42, pp. 954–973.

R. P. Brent (1973), *Algorithms for Minimization Without Derivatives*, Prentice-Hall, Englewood Cliffs, NJ.

A. R. Conn and P. L. Toint (1996), *An algorithm using quadratic interpolation for unconstrained derivative free optimization,* in Nonlinear Optimization and Applications, G. D. Pillo and F. Giannessi, eds., Plenum Press, New York, pp. 27–47.

M. del Valle, M. Poch, J. Alonso, and J. Bartroli (1990), *Comparison of the Powell and simplex methods in the optimization of flow-injection systems*, Analytica Chimica Acta, 241, pp. 31–42.

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

P. Dupuis and R. Simha (1991), *On sampling-controlled stochastic approximation*, IEEE Trans. Automat. Control, 36, pp. 915–924.

C. Elster and A. Neumaier (1995), *A grid algorithm for bound constrained optimization of noisy functions*, IMA J. Numer. Anal., 15, pp. 585–608.

W. Feller (1968), *An Introduction to Probability Theory and Its Applications*, John Wiley, New York.

P. E. Gill, W. Murray, M. A. Saunders, and M. H. Wright (1983), *Computing forward-difference intervals for numerical optimization*, SIAM J. Sci. Statist. Comput., 4, pp. 310–321.

T. Glad and A. Goldstein (1977), *Optimization of functions whose values are subject to small errors*, BIT, 17, pp. 160–169.

P. Hedlund and A. Gustavsson (1992), *Design and evaluation of modified simplex methods having enhanced convergence ability*, Analytica Chimica Acta, 259, pp. 243–256.

R. Hooke and T. A. Jeeves (1961), *Direct search solution of numerical and statistical problems*, J. ACM, 8, pp. 212–229.

A. I. Khuri and J. A. Cornell (1987), *Response Surfaces: Designs and Analyses*, Marcel Dekker, New York.

H. J. Kushner and D. S. Clark (1978), *Stochastic Approximation Methods for Constrained and Unconstrained Systems*, Springer-Verlag, New York.

J. C. Lagarias, J. A. Reeds, M. H. Wright, and P. E. Wright (1998), *Convergence properties of the Nelder–Mead simplex method in low dimensions*, SIAM J. Optim., 9, pp. 112–147.

K. I. M. McKinnon (1998), *Convergence of the Nelder–Mead simplex method to a nonstationary point*, SIAM J. Optim., 9, pp. 148–158.

J. J. Moré, B. S. Garbow, and K. E. Hillstrom (1981), *Testing unconstrained optimization software*, ACM Trans. Math. Software, 7, pp. 17–41.

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

L. R. Parker, M. R. Cave, and R. M. Barnes (1985), *Comparison of simplex algorithms*, Analytica Chimica Acta, 175, pp. 231–237.

B. T. Polyak (1987), *Introduction to Optimization*, Optimization Software, Inc., New York.

M. J. D. Powell (1964), *An efficient method for finding the minimum of a function of several variables without calculating derivatives*, Computer J., 17, pp. 155–162.

M. J. D. Powell (1970), *A hybrid method for nonlinear equations*, in Numerical Methods for Nonlinear Algebraic Equations, P. Rabinowitz, ed., Gordon and Breach, London, pp. 87–114.

M. J. D. Powell (1994), *A direct search optimization method that models the objective and constraint functions by linear interpolation*, in Advances in Optimization and Numerical Analysis, Proceedings of the Sixth Workshop on Optimization and Numerical Analysis, Oaxaca, Mexico, Vol. 275, Kluwer, Dordrecht, The Netherlands, pp. 51–67.

A. S. Rykov (1980), *Simplex direct search algorithms*, Automat. Remote Control, 41, pp. 784–793.

W. Spendley, G. R. Hext, and F. R. Himsworth (1962), *Sequential application of simplex designs in optimization and evolutionary operation*, Technometrics, 4, pp. 441–461.

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

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

P. Tseng (1999), *Fortified-descent simplicial search method: A general approach*, SIAM J. Optim., 10, pp. 269–288.

Y. Wardi (1988), *A stochastic steepest-descent algorithm*, J. Optim. Theory Appl., 59, pp. 307–323.

Y. Wardi (1990), *Stochastic algorithms for Armijo stepsizes for minimization of functions*, J. Optim. Theory Appl., 64, pp. 399–417.

W. C. Yu (1979), *The convergence property of the simplex evolutionary technique*, Sci. Sinica, Special Issue I on Math., 1, pp. 69–77.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 859–869

ON THE CONVERGENCE OF GRID-BASED METHODS FOR UNCONSTRAINED OPTIMIZATION*

I. D. COOPE† AND C. J. PRICE†

Abstract. The convergence of direct search methods for unconstrained minimization is examined in the case where the underlying method can be interpreted as a grid or pattern search over successively refined meshes. An important aspect of the main convergence result is that translation, rotation, scaling, and shearing of the successive grids are allowed.

Key words. grid-based optimization, derivative-free optimization, positive basis methods, convergence analysis, multidirectional search

AMS subject classifications. 49M30, 65K05

PII. S1052623499354989

1. Introduction.

Recent survey papers [1], [7], [10] report on significant renewed interest in algorithms for derivative-free unconstrained optimization. Much of this recent interest has been provoked by new convergence results (see, for example, [1], [6], [8], [9]). Most of the current derivative-free algorithms for which convergence results have been established belong to one or more of three categories: line search methods, trust region methods, or grid-based methods. In this paper, the convergence of derivative-free methods for unconstrained minimization is examined in the case where the underlying method can be interpreted as a grid or pattern search over successively refined meshes. Therefore, the methods discussed here are similar to those studied in [6], [8], [9], but permit greater freedom in the orientation and scaling of successive grids. Alternative approaches based on trust regions or line searches can be found in [1], [7], and the references therein.

The properties of grid-based methods are explored and it is shown that convergence can be achieved for a quite general class of algorithm. An important aspect of the main convergence result is that successive grids may be arbitrarily translated, rotated, and sheared relative to one another, and each grid axis may be rescaled independently of the others. This flexibility allows second-order information to be incorporated into the shape of successive grids, for example, by aligning grid axes along conjugate directions or the principal axes of an approximating quadratic. The hope is to construct nonderivative algorithms that possess useful properties of conjugate direction or quasi-Newton algorithms, thus exploiting curvature information without assuming the existence of second derivatives or the availability of first derivatives.

We present two optimization frameworks for unconstrained optimization of continuously differentiable functions that are bounded below. For the first framework, in which finite searches are conducted along grid directions of descent, we establish convergence of a subsequence of iterates to a stationary point of the objective function. For the second framework, under the stronger assumption that the algorithm searches the grid direction of locally greatest descent at every iterate, we show that

*Received by the editors April 26, 1999; accepted for publication (in revised form) December 14, 2000; published electronically March 15, 2001.
http://www.siam.org/journals/siopt/11-4/35498.html
†Department of Mathematics and Statistics, University of Canterbury, Private Bag 4800, Christchurch, New Zealand (i.coope@math.canterbury.ac.nz, c.price@math.canterbury.ac.nz).

the entire sequence of iterates converges to a stationary point.

The restrictions on the grids in our framework are much less severe than for the pattern search methods of [6], [9], where a single set of grid axes is used, only rational scalings of grids are permitted, and arbitrary translations are not allowed. Furthermore, the methods of [6], [9] do not allow scalings and realignments to reflect curvature information. A more complete discussion of differences from the pattern search methods of [6], [9] appears in section 3.

The nonderivative method of [4] for bound-constrained optimization includes curvature information through interpolating quadratics, but uses only nested grids that are aligned with one another.

The great flexibility of the algorithm framework means that there is much work to be done in determining the best algorithms which conform to this framework. Extensive results for a specific algorithm conforming to the framework are not presented in this paper—to do so would shift the focus of the paper away from the framework and onto that specific algorithm.

In the next section some properties of positive bases relevant to grid-based methods are introduced and in section 3 an algorithmic framework for grid-based methods is described which allows considerable flexibility in the design of algorithms of this type. The main convergence results are established in section 4 with further comments and discussion given in the final section.

2. Grid-based methods and positive bases. The algorithms under consideration seek a minimizer of the objective function $f : R^n \to R$ by examining f on a sequence of successively finer grids. Each grid $\mathcal{G}^{(m)}$ is defined by a set of n linearly independent basis vectors $\mathcal{V}^{(m)}$, where

$$\mathcal{V}^{(m)} = \left\{ v_i^{(m)} \in R^n \ : \ i = 1, \ldots, n \right\}.$$

The points on the grid $\mathcal{G}^{(m)}$ are

$$\mathcal{G}^{(m)} = \left\{ x \in R^n : x = x_o^{(m)} + h^{(m)} \sum_{i=1}^{n} \eta_i v_i^{(m)} \right\},$$

where $h^{(m)}$ is a positive scalar and each η_i is any integer. The parameter $h^{(m)}$ is referred to as the mesh size and is adjusted as m is increased in order to ensure that the meshes become finer in a manner needed to establish convergence. The point $x_o^{(m)}$ allows the grids to be offset relative to one another. The basis vectors in $\mathcal{V}^{(m)}$ are parallel to the axes of the grid $\mathcal{G}^{(m)}$.

The set $\mathcal{V}^{(m)}$ is used to form a positive basis $\mathcal{V}_+^{(m)}$. There are two requirements for a set $\mathcal{V}_+$ to form a positive basis:

(i) Every vector in R^n can be written as a nonnegative linear combination of the vectors in $\mathcal{V}_+$;

(ii) no member of $\mathcal{V}_+$ is expressible as a nonnegative linear combination of the remaining members of $\mathcal{V}_+$.

It is shown in [3] that the cardinality p of any positive basis for R^n satisfies $n + 1 \leq p \leq 2n$. For example, if $\{v_1, v_2, \ldots, v_n\}$ is a basis for R^n, then

$$\left\{ v_1, v_2, \ldots, v_n, -\sum_{i=1}^{n} v_i \right\} \tag{2.1}$$

is a positive basis with $n+1$ elements. At the other extreme,

$$\{v_1, v_2, \ldots, v_n, -v_1, -v_2, \ldots, -v_n\} \tag{2.2}$$

is a positive basis with $2n$ elements.

Let $p^{(m)}$ denote the cardinality of $\mathcal{V}_+^{(m)}$. We assume throughout that the first n members of $\mathcal{V}_+^{(m)}$ are those of $\mathcal{V}^{(m)}$ and that the remaining elements are given by an integer linear combination of the members of $\mathcal{V}^{(m)}$:

$$v_j^{(m)} = \sum_{i=1}^{n} \zeta_{ij}^{(m)} v_i^{(m)}, \quad j = n+1, \ldots, p^{(m)}, \tag{2.3}$$

where each $\zeta_{ij}^{(m)}$ must be an integer so that if $x \in \mathcal{G}^{(m)}$ and $v \in \mathcal{V}_+^{(m)}$, then $x + h^{(m)} v \in \mathcal{G}^{(m)}$. Equation (2.3) requires the members of $\mathcal{V}_+^{(m)}$ to assume a specific order, and positive bases satisfying (2.3) will be called *ordered* positive bases.

Use of ordered positive bases permits the formation of termination conditions for the search on each grid via the following theorem.

THEOREM 2.1. *If the set of vectors $\mathcal{V}_+$ is a positive basis, then*

$$g^T v \geq 0 \quad \forall v \in \mathcal{V}_+ \quad \Rightarrow \quad g = 0.$$

Proof. Let the members of $\mathcal{V}_+$ be v_i for $i = 1, \ldots, |\mathcal{V}_+|$. Then for any $g \in R^n$,

$$-g = \sum_{i=1}^{|\mathcal{V}_+|} \eta_i v_i, \quad \text{where} \quad \eta_i \geq 0 \quad \forall i.$$

Therefore, if $v_i^T g \geq 0$, for $i = 1, \ldots, |\mathcal{V}_+|$

$$0 \geq (-g)^T g = \sum_{i=1}^{|\mathcal{V}_+|} \eta_i v_i^T g \geq 0.$$

The only possibility is $g = 0$. ∎

This theorem motivates the following definition.

DEFINITION 2.2. Grid local minimizer. *A point x on the grid $\mathcal{G}^{(m)}$ is defined as a grid local minimizer with respect to the positive basis $\mathcal{V}_+^{(m)}$ if and only if*

$$f(x + h^{(m)} v_i) \geq f(x) \quad \forall v_i \in \mathcal{V}_+^{(m)}.$$

This definition is motivated by the fact that

$$v^T \nabla f \geq 0 \quad \forall v \in \mathcal{V}_+^{(m)} \quad \Rightarrow \quad \nabla f = 0.$$

The conditions which define a grid local minimizer are a finite difference approximation to this.

In order to establish convergence, some restrictions must be imposed on the form of the ordered positive bases used to define the grid local minimizers. The following definition allows these restrictions to be simply expressed as linear relationships between the members of each ordered positive basis.

DEFINITION 2.3. Structural equivalence. *Two ordered positive bases* $\{v_1, \ldots, v_p\}$ *and* $\{w_1, \ldots, w_p\}$ *are regarded as structurally equivalent if and only if*

$$\forall j > n, \quad v_j = \sum_{i=1}^{n} \zeta_{ij} v_i \quad \Longleftrightarrow \quad w_j = \sum_{i=1}^{n} \zeta_{ij} w_i.$$

Structurally equivalent positive bases necessarily have the same cardinality. As a simple example, the following two ordered positive bases for R^3 are structurally equivalent:

$$\{e_1, e_2, e_3, -e_1, -2(e_2 + e_3)\}, \quad \{e_3, e_1, e_2, -e_3, -2(e_1 + e_2)\},$$

where e_i is the ith unit coordinate vector.

An appropriate framework for optimization algorithms using ordered positive bases is described and analyzed in the following two sections.

3. The algorithm framework. The basic structure of the framework (listed immediately below) consists of two asynchronous loops. The outer loop (steps 1–3) selects each grid and checks the stopping conditions. The inner loop (step 2) conducts finite searches using each member of $\mathcal{V}_+^{(m)}$ in turn until $p^{(m)}$ consecutive searches fail to make progress. When this occurs, a grid local minimizer has been found; the inner loop then terminates and the outer loop selects the new grid.

ALGORITHM FRAMEWORK A.

Initialize $m = 1$, $k = 1$, and let $x_o^{(1)}$ be the initial point $x^{(1)}$.

while (stopping conditions do not hold) **do**

1. Choose $h^{(m)}$ and $\mathcal{V}_+^{(m)}$. Set $i = 1$ and $r = 0$. Set $p^{(m)} = \left|\mathcal{V}_+^{(m)}\right|$.
2. **while** $r < p^{(m)}$ **do**
   (a) Calculate f at a finite number of points on the grid $\mathcal{G}^{(m)}$, including $x^{(k)} + h^{(m)} v_i^{(m)}$. If any points lower than $x^{(k)}$ are found, set $x^{(k+1)}$ equal to the lowest of these points, increment k, and let $r = 0$. Otherwise increment r.
   (b) Set $i = i + 1$. If $i > p^{(m)}$, set $i = 1$.
   end
3. Set $\hat{x}^{(m)} = x^{(k)}$. Execute any finite process, and let $x_o^{(m+1)}$ be the lowest known point. If this finite process yields descent, set $x^{(k+1)} = x_o^{(m+1)}$ and increment k. Increment m.

end

In this framework, r is the number of consecutive failed finite searches using the members of $\mathcal{V}_+^{(m)}$. When $r = p^{(m)}$, a grid local minimizer $\hat{x}^{(m)}$ has been found and the algorithm terminates the search over the current grid $\mathcal{G}^{(m)}$. The next grid $\mathcal{G}^{(m+1)}$ has its origin $x_o^{(m+1)}$ positioned at the lowest known point. The algorithm generates a new iterate $x^{(k)}$ only when a new lowest point (one with a strictly lower function value) is found. In contrast, it generates a new grid local minimizer $\hat{x}^{(m)}$ every time step 2 is completed. The sequence $\{x^{(k)}\}$ may have finitely many members, whereas the sequence $\{\hat{x}^{(m)}\}$ will always have infinitely many members (ignoring stopping conditions) and may contain repetitions of some members of $\{x^{(k)}\}$. However, every member of the sequence $\{\hat{x}^{(m)}\}$ is also a member of the sequence of iterates $\{x^{(k)}\}$. If, for example, $x^{(1)}$ is a global minimizer of f, then the sequence of iterates

is the singleton set $\{x^{(1)}\}$, whereas the sequence of grid local minimizers is the set $\{x^{(1)}, x^{(1)}, x^{(1)}, \ldots\}$.

The finite process in step 3 is arbitrary. Many possible choices exist, including a null process, or a finite ray search along an estimate of the direction of steepest descent or along a quasi-Newton direction. In proving convergence, finiteness is the only requirement for this process.

It is only necessary that the finite search in step 2(a) inspect one point, namely, $x^{(k)}+h^{(m)}v_i^{(m)}$. No other point on $\mathcal{G}^{(m)}$ need be examined; however, it would normally be desirable to do so. For example, a search along the ray $x^{(k)} + \alpha h^{(m)}v_i^{(m)}$, $\alpha > 0$, could be implemented and the lowest grid point in that search taken as $x^{(k+1)}$, provided descent is obtained. The option of examining a finite number of other grid points in step 2(a) admits the possibility of an arbitrarily long step to a grid point at each iteration of the inner while loop. For example, this could be exploited by examining the grid point closest to a quasi-Newton step and accepting that grid point if it is sufficiently low. The intent behind such an approach would be to reduce the number of iterations of the while loop needed to locate a grid local minimum.

Framework A is not a special case of the analysis of pattern search methods in [6], [9]. In their notation, at each iteration [6], [9] examine a pattern of points $x^{(k)} + \Delta_k BC_k$, where Δ_k is a scale factor, B is a fixed matrix independent of the iteration number k, and C_k is an integer matrix. Three points should be noted:

(i) Because B is independent of k, all grids are aligned with one another, and the grid alignment must be chosen at the start of optimization, before information from function evaluations is available.

(ii) All grid axes at iteration k are scaled by the same factor Δ_k, and Δ_{k+1} must be a rational multiple of Δ_k.

(iii) The only way to scale directions is through the matrix C_k. But since the elements of C_k are integers, either the number of directions and scalings is small or else the grid may need to be much finer than the step sizes considered. Furthermore, pattern search methods cannot reach an arbitrary point or produce an arbitrary direction in a finite number of steps. For example, if the initial pattern is aligned with the x_1 and x_2 axes in two dimensions and $\Delta_0 = 1$, a pattern search method will require a very large number of iterations to reach a small neighborhood of the point $(0, \sqrt{2})$, and it cannot produce a direction at 30° to the x_1 axis. Thus, pattern search methods do not possess the property of finite termination on convex quadratics that is standard with conjugate direction methods. By contrast, algorithms conforming to Framework A can possess this finite termination property; see [2] for such a method.

The analogues of Δ_k, B, and C_k in our framework are less restricted than in pattern search methods. For example, $h^{(k)}$ can be irrational, unlike Δ_k; the bases $\mathcal{V}^{(m)}$ can be chosen independently from time to time, whereas B is fixed; and the ability to consider other grid points in step 2(a) of the framework offers the same level of freedom given by the matrices C_k. The techniques of [6], [9] are more general than Framework A in that they require C_k to contain only one of a finite number of integer positive bases, whereas we use exclusively the positive ordered basis $\mathcal{V}_+^{(m)}$. Framework A could be modified to use ordered positive bases other than $\mathcal{V}_+^{(m)}$ to explore the grid $\mathcal{G}^{(m)}$, but the flexibility in step 2(a), the choice of successive grids, and the arbitrary finite process in step 3 are likely to lessen the benefits of such an approach.

For Framework A, convergence can be shown only for subsequences of grid local minimizers. This is because the finite searches in step 2(a) are opportunistic; the first member encountered in $\mathcal{V}_+^{(m)}$ which gives descent leads to a new iterate. Convergence of the full sequence of iterates can be shown for a more restricted framework, Framework B (defined below), in which a thorough search is made along the member $s^{(k)}$ of $\mathcal{V}_+^{(m)}$ giving the "best drop" at $x^{(k)}$. The best drop member $s^{(k)}$ of $\mathcal{V}_+^{(m)}$ satisfies

$$f\left(x^{(k)} + h^{(m(k))}s^{(k)}\right) \leq f\left(x^{(k)} + h^{(m(k))}v\right) \quad \forall v \in \mathcal{V}_+^{(m(k))}, \tag{3.1}$$

where $m(k)$ is the number of the grid on which $x^{(k)}$ is placed. This determination of $s^{(k)}$ requires $p^{(m(k))}$ function evaluations. The search along each $s^{(k)}$ must evaluate f at a sequence of points

$$\tilde{x_i} = x^{(k)} + \alpha_i h^{(m(k))}s^{(k)}, \quad i = 0, \ldots, \tag{3.2}$$

where $\alpha_0 = 1$ and the integer sequence $\{\alpha_i\}, i \geq 1$, satisfies

$$\alpha_{i-1} + 1 \leq \alpha_i \leq \beta\alpha_{i-1} \quad \text{with} \quad \beta \geq 2.$$

The search may terminate only when an integer $\ell \geq 0$ is found such that $f(\tilde{x}_\ell) \leq f(\tilde{x}_{\ell+1})$.

ALGORITHM FRAMEWORK B.

Initialize $m = 1$, $k = 1$, and let $x_o^{(1)}$ be the initial point $x^{(1)}$.

while (stopping conditions do not hold) **do**

1. Choose $h^{(m)}$ and $\mathcal{V}_+^{(m)}$. Set $i = 1$ and $r = 0$. Set $p^{(m)} = \left|\mathcal{V}_+^{(m)}\right|$.
2. **while** $x^{(k)}$ is not a grid local minimizer **do**
   (a) Calculate the best drop direction, $s^{(k)}$, satisfying (3.1). If $f^{(k)} \leq f(x^{(k)} + h^{(m)}s^{(k)})$, then exit step 2, as $x^{(k)}$ is a grid local minimizer.
   (b) Starting with $\alpha_0 = 1$, choose successive integer values $\alpha_1, \alpha_2, \ldots$ until
   $$f\left(x^{(k)} + \alpha_{\ell+1}h^{(m)}s^{(k)}\right) \geq f\left(x^{(k)} + \alpha_\ell h^{(m)}s^{(k)}\right),$$
   where $\alpha_{\ell+1} \in [\alpha_\ell + 1, \beta\alpha_\ell]$.
   (c) Calculate f at a finite number of grid points, and choose $x^{(k+1)}$ to be the lowest of these points and $x^{(k)} + \alpha_\ell h^{(m)}s^{(k)}$. Increment k.
   end
3. Set $\hat{x}^{(m)} = x^{(k)}$. Execute any finite process, and let $x_o^{(m+1)}$ be the lowest known point. If this finite process yields descent, set $x^{(k+1)} = x_o^{(m+1)}$ and increment k. Increment m.

end

Framework B is a specialization of Framework A.

4. Convergence analysis. The convergence results for the methods discussed in this paper are similar to those in [9], but the method of analysis is sufficiently flexible to allow a myriad of other possibilities that may be more appropriate in other cases. The first theorem proves convergence of the subsequence of grid local minimizers for algorithms conforming to Framework A. This theorem is also applicable

to Framework B because any algorithm conforming to Framework B also conforms to Framework A.

For the purpose of establishing convergence, it is assumed in this section that the stopping conditions are never invoked. This permits examination of the full sequence of iterates and grid local minimizers and their asymptotic properties. From a practical point of view, stopping conditions are essential which is why they are incorporated into the general frameworks.

THEOREM 4.1. *For any optimization algorithm conforming to Framework A, assume that*

(a) *the sequence of iterates* $\{x^{(k)}\}$ *is bounded;*
(b) f *is continuously differentiable;*
(c) *there exist positive constants* K *and* κ *such that* $|\det(v_1^{(m)}, \ldots, v_n^{(m)})| \geq \kappa$ *and* $\|v_i^{(m)}\| \leq K$ *for all* m *and* i*;*
(d) $h^{(m)} \to 0$ *as* $m \to \infty$*; and*
(e) *there is a finite subset of* $\mathcal{B}$ *such that each member of* $\mathcal{B}$ *is structurally equivalent to some member of this finite subset, where* $\mathcal{B}$ *denotes the sequence of ordered positive bases* $\{\mathcal{V}_+^{(m)}\}_{m=1}^{\infty}$.

Then the sequence $\{\hat{x}^{(m)}\}$ *of grid local minimizers has infinitely many members, and each cluster point* $\hat{x}^{(\infty)}$ *of* $\{\hat{x}^{(m)}\}$ *is a stationary point of* f.

Proof. The proof is in two parts. The first part shows that step 2 terminates, and consequently the sequence of grid local minimizers, $\{\hat{x}^{(m)}\}$, is infinite. The main part of the theorem is then established.

First, by condition (a) of the theorem, there is a compact set $\mathcal{F}$ for which $\{x^{(k)}\} \subset \mathcal{F}$. Hence the set $\mathcal{F} \cap \mathcal{G}^{(m)}$ is finite. The sequence of function values is strictly decreasing so each iterate is distinct from all others, and so only a finite number of iterates is generated using $\mathcal{G}^{(m)}$. Hence the finiteness of step 2(a) means the algorithm generates only a finite number of points using each grid. The only way the algorithm can change from the grid $\mathcal{G}^{(m)}$ is if the last iterate generated using the grid $\mathcal{G}^{(m)}$ is a grid local minimizer. Hence $\{\hat{x}^{(m)}\}$ is infinite.

Next, choose a specific cluster point $\hat{x}^{(\infty)}$ of $\{\hat{x}^{(m)}\}$ and choose some $\mathcal{S} \subset \mathcal{B}$ such that $\mathcal{S}$ is an infinite subsequence of structurally equivalent bases and the corresponding subsequence of $\{\hat{x}^{(m)}\}$ converges to $\hat{x}^{(\infty)}$. Condition (e) ensures that one or more subsequences like $\mathcal{S}$ exist and that all but a finite number of members of $\mathcal{B}$ belong to a subsequence like $\mathcal{S}$. Now replace the sequence of iterates $\{x^{(k)}\}$ and all other sequences with the infinite subsequences of themselves which correspond to the subsequence $\mathcal{S}$. It then follows that

$$f\left(\hat{x}^{(m)} + h^{(m)} v_i^{(m)}\right) \geq f\left(\hat{x}^{(m)}\right) \quad \forall i \in 1, \ldots, p, \tag{4.1}$$

where $p = p^{(m)}$ for all m such that $\mathcal{V}_+^{(m)} \in \mathcal{S}$. Now

$$f\left(\hat{x}^{(m)} + h^{(m)} v_i^{(m)}\right) = f\left(\hat{x}^{(m)}\right) + \int_{t=0}^{h^{(m)}} \left[g\left(\hat{x}^{(m)} + t v_i^{(m)}\right) - \hat{g}^{(m)} + \hat{g}^{(m)}\right]^T v_i^{(m)}\, dt$$

$$= f\left(\hat{x}^{(m)}\right) + h^{(m)} \left(\hat{g}^{(m)}\right)^T v_i^{(m)} + E_i^{(m)},$$

where $g(x) \equiv \nabla f(x)$, where $\hat{g}^{(m)} = g(\hat{x}^{(m)})$, and where

$$E_i^{(m)} = \int_{t=0}^{h^{(m)}} \left[g\left(\hat{x}^{(m)} + t v_i^{(m)}\right) - \hat{g}^{(m)}\right]^T v_i^{(m)}\, dt\,.$$

The bound on $\|v_i^{(m)}\|$ in (c) yields

$$\left|E_i^{(m)}\right| \leq \int_{t=0}^{h^{(m)}} KM_i^{(m)}\, dt = h^{(m)} KM_i^{(m)},$$

where

$$M_i^{(m)} = \max\left\{\left\|g\left(\hat{x}^{(m)} + tv_i^{(m)}\right) - \hat{g}^{(m)}\right\| : t \in \left[0, h^{(m)}\right]\right\}.$$

The continuity of g and the compactness of $\mathcal{F}$ imply that g is also uniformly continuous on $\mathcal{F}$. The bound on $\|v_i^{(m)}\|$ in (c) then ensures that $M_i^{(m)} \to 0$ as $m \to \infty$ by condition (d).

Now $\hat{x}^{(\infty)}$ is a cluster point of the sequence $\{\hat{x}^{(m)}\}$ of grid local minimizers, and $\|v_i^{(m)}\| \leq K$ for all m and $i = 1, \ldots, n$, so there is a subsequence $\{\overline{x}^{(m)}\}$ of $\{\hat{x}^{(m)}\}$ and corresponding subsequences $\{\overline{v}_i^{(m)}\}$ of $\{v_i^{(m)}\}$ for $i = 1, \ldots, n$ which have unique limits $\hat{x}^{(\infty)}$ and $\overline{v}_i^{(\infty)}$ for $i = 1, \ldots, n$. The structural equivalence of all members of $\mathcal{S}$ implies that

$$\lim_{m \to \infty} \overline{v}_i^{(m)} = \overline{v}_i^{(\infty)} \quad \forall i = 1, \ldots, p. \tag{4.2}$$

Condition (c) implies that $\overline{v}_1^{(\infty)}, \ldots, \overline{v}_n^{(\infty)}$ are linearly independent and $\overline{v}_1^{(\infty)}, \ldots, \overline{v}_p^{(\infty)}$ are bounded in norm by K. Hence, $\{\overline{v}_1^{(\infty)}, \ldots, \overline{v}_p^{(\infty)}\}$ is an ordered positive basis which is structurally equivalent to all members of $\mathcal{S}$.

Now (4.1) implies

$$\overline{h}^{(m)} \left(\overline{g}^{(m)}\right)^T \overline{v}_i^{(m)} + \overline{h}^{(m)} K \overline{M}_i^{(m)} \geq 0 \quad \forall i = 1, \ldots, p.$$

In the limit as $m \to \infty$, condition (d) implies

$$\left(\hat{g}^{(\infty)}\right)^T \overline{v}_i^{(\infty)} \geq 0 \quad \forall i = 1, \ldots, p,$$

and so $\hat{g}^{(\infty)} = \nabla f(\hat{x}^{(\infty)}) = 0$ by Theorem 2.1. The choices of $\mathcal{S}$ and of the cluster point of the sequence of grid local minimizers were arbitrary, so every cluster point of the sequence of grid local minimizers is a stationary point of the objective function. □

Theorem 4.1 makes very few assumptions about how the sequence of grid local minimizers is generated; all that is required is that this sequence be bounded and have infinitely many members. Assumption (a) on the full sequence of iterates is only needed to establish these two properties. This assumption is automatically satisfied if, for example, the level set $\{x : f(x) \leq f(x^{(1)})\}$ is bounded, an assumption frequently made in convergence analyses; however, it may also be valid under much less restrictive conditions. Assumption (c) is easily satisfied by choosing each $\mathcal{V}_+^{(m)}$ appropriately. A very simple way to satisfy (d) is to halve h every time a grid local minimizer is found. An example of a more complex scheme is given in [2]. Assumption (e) is one of practicality and is easily enforced. We expect that most useful algorithms will use only one or two types of structurally equivalent bases (corresponding to ordered positive bases such as (2.1) or (2.2)); the proof is valid, however, for any finite number. Theorem 4.1 is, therefore, applicable to a wide range of algorithms.

If stopping conditions are never invoked and the sequence of iterates $x^{(k)}$ has finitely many members, then its last member is necessarily a stationary point of f. In the more usual case, when an infinite sequence of iterates is generated, we now show convergence of the full sequence of iterates under the stricter Framework B.

THEOREM 4.2. *If the conditions of Theorem* 4.1 *hold, then algorithms conforming to Framework B generate sequences of iterates which converge to one or more stationary points of* f.

Proof. The proof is by contradiction. Let $x^{(\infty)}$ be a cluster point of the sequence $\{x^{(k)}\}$ for which $g_\infty = \nabla f(x^{(\infty)})$ is nonzero. Following the proof of Theorem 4.1, replace $\{x^{(k)}\}$ by a subsequence of itself with $x^{(\infty)}$ as its unique limit, for which the corresponding subsequence of best drop directions $\{s^{(k)}\}$ (defined by (3.1)) has a unique limit $s^{(\infty)}$, and for which all ordered positive bases are structurally equivalent and have a unique limit $\mathcal{V}_+^{(\infty)}$. The structural equivalence of these ordered positive bases, condition (c) of Theorem 4.1, and (4.2) show that $\mathcal{V}_+^{(\infty)}$ is also an ordered positive basis.

First, it is shown that $g_\infty^T s^{(\infty)} < 0$. Now

$$\forall k, \forall v_i \in \mathcal{V}_+^{(m(k))}, \quad f\left(x^{(k)} + h^{(m(k))} v_i\right) - f^{(k)} \geq f\left(x^{(k)} + h^{(m(k))} s^{(k)}\right) - f^{(k)},$$

where $m(k)$ is the number of the grid on which $x^{(k)}$ is placed. Taylor's series expansions on both sides yield

$$v_i^T g^{(k)} + L_k \geq \left(g^{(k)}\right)^T s^{(k)} - L_k \quad \forall v_i \in \mathcal{V}_+^{(m(k))},$$

where L_k is defined as

$$L_k = K \max\left\{ \|g(x) - g^{(k)}\| \ : \ x \in \mathcal{F} \ \text{ and } \ \|x - x^{(k)}\| \leq K h^{(m(k))} \right\}.$$

In the limit as $k \to \infty$, $L_k \to 0$ by the uniform continuity of g on $\mathcal{F}$. Hence

$$g_\infty^T v_i^{(\infty)} \geq g_\infty^T s^{(\infty)} \quad \forall i = 1, \ldots, p. \tag{4.3}$$

However, $g_\infty \neq 0$ so there exists a $v \in \mathcal{V}_+^{(\infty)}$ such that $g_\infty^T v < 0$. Clearly, $g_\infty^T s^{(\infty)} < 0$ by inequality (4.3).

Next, define the closed ball B_ϵ about $x^{(\infty)}$ via

$$B_\epsilon = \left\{ x : \|x - x^{(\infty)}\| \leq \epsilon \right\}$$

and similarly for B_δ, where $\delta < \epsilon$. The continuity of $g(x)$ and the convergence of $s^{(k)}$ to $s^{(\infty)}$ imply

$$\exists N > 0, \ \exists \epsilon > 0 \text{ such that } \forall k > N, \ \forall x \in B_\epsilon,$$

$$g^T(x) s^{(k)} \leq \tfrac{1}{2} g_\infty^T s^{(\infty)} < 0 \quad \text{and} \quad \|s^{(k)}\| \leq 2\|s^{(\infty)}\|. \tag{4.4}$$

Now choose a specific $k > N$ such that $x^{(k)} \in B_\delta$. The C^1 continuity of f implies

$$f^{(k)} \leq f^{(\infty)} + M\delta, \tag{4.5}$$

where M is an upper bound for $\|g(x)\|$ over B_ϵ. The first inequality in (4.4) implies that f is strictly descending along the line segment

$$x^{(k)} + \alpha h^{(m(k))} s^{(k)} \quad \text{for} \quad 0 \leq \alpha \leq \frac{(\epsilon - \delta)}{h^{(m(k))} \left\| s^{(k)} \right\|}.$$

The restrictions on successive α values in the ray searches mean that the final α value for the ray $x^{(k)} + \alpha h^{(m(k))} s^{(k)}$, $\alpha > 0$, is at least $(\epsilon - \delta)/(\beta h^{(m(k))} \|s^{(k)}\|)$. Hence

$$\begin{aligned} f^{(k+1)} &= f^{(k)} + \int_{t=0}^{\alpha} h^{(m(k))} \left(s^{(k)} \right)^T g \left(x^{(k)} + t h^{(m(k))} s^{(k)} \right) dt \\ &\leq f^{(k)} + \frac{\alpha h^{(m(k))}}{2} g_\infty^T s^{(\infty)} \\ &\leq f^{(k)} - \frac{\epsilon - \delta}{2\beta \left\| s^{(k)} \right\|} \left| g_\infty^T s^{(\infty)} \right|. \end{aligned}$$

Then (4.5) and the last inequality in (4.4) imply

$$f^{(k+1)} \leq f^{(\infty)} + M\delta - \frac{\epsilon - \delta}{4\beta \left\| s^{(\infty)} \right\|} \left| g_\infty^T s^{(\infty)} \right|.$$

As $k \to \infty$, δ can be made arbitrarily small, implying that $f^{(k)} < f^{(\infty)}$ for k large. The continuity of f and the monotonicity of $\{f^{(k)}\}$ imply that $x^{(\infty)}$ cannot be a cluster point of the sequence of iterates. □

An example of an existing method which conforms to Framework A is that of Hooke and Jeeves [5]. Their method does not explicitly impose an upper bound on the step length, which appears to be at odds with Theorem 4.1. The applicability of Theorem 4.1 follows on noting that if condition (a) of the theorem holds (the sequence of iterates is bounded), then the maximum step length must also be bounded. Satisfaction of condition (a) of Theorem 4.1 is ensured if at least one iterate lies within a bounded level set. Although the method of Hooke and Jeeves conforms to Framework A, it makes little use of the flexibility afforded by that framework. An algorithm that exploits the flexibility allowed by Framework A is presented in [2], where numerical results are given for standard test functions; the authors expect further improvements to follow with more research into algorithms conforming to this framework.

5. Summary. We have presented two general algorithmic frameworks for unconstrained optimization methods based only on function values and have shown that, under mild conditions, such algorithms generate sequences of grid local minimizers that are guaranteed to converge to stationary points. There is much scope for improving efficiency within Framework A through the choice of the ordered positive basis $\mathcal{V}_+^{(m)}$ and the finite process in step 3, which could, for example, allow a quasi-Newton step or a step to the minimizer of an interpolating quadratic. An efficient algorithm that uses this flexibility to align grid axes along conjugate directions is described in [2].

The authors believe that Theorem 4.1 is applicable to many effective grid search methods, and much work remains to be done in examining the properties of these algorithms.

Acknowledgments. The authors would like to acknowledge the contribution of John Dennis, who provided the initial stimulus for this work, Margaret Wright for her many helpful comments and suggestions, and two anonymous referees.

REFERENCES

[1] A. Conn, K. Scheinberg, and P. L. Toint, *On the convergence of derivative-free methods for unconstrained optimization*, in Approximation Theory and Optimization, M. D. Buhmann and A. Iserles, eds., Cambridge University Press, Cambridge, 1997, pp. 83–108.

[2] I. D. Coope and C. J. Price, *A direct search conjugate directions algorithm for unconstrained minimization*, ANZIAM J., 42(E) 2000, pp. C478–498.

[3] C. Davis, *Theory of positive linear dependence*, Amer. J. Math., 76 (1954), pp. 733–746.

[4] C. Elster and A. Neumaier, *A grid algorithm for bound-constrained optimization of noisy functions*, IMA J. Numer. Anal., 15 (1995), pp. 585–608.

[5] R. Hooke and T. A. Jeeves, *Direct search solution of numerical and statistical problems*, J. Assoc. Comput. Mach., 8 (1961), pp. 212–219.

[6] R. M. Lewis and V. Torczon, *Rank ordering and positive bases in pattern search algorithms*, Math. Program., to appear.

[7] M. J. D. Powell, *Direct search algorithms for optimization calculations*, Acta Numer., 7 (1998), pp. 287–336.

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

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

[10] M. H. Wright, *Direct search methods: Once scorned now respectable*, in Proceedings of the 1995 Dundee Biennial Conference in Numerical Analysis, Longman, Harlow, UK, 1996, pp. 191–208.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 870–888

A SCALED GAUSS–NEWTON PRIMAL-DUAL SEARCH DIRECTION FOR SEMIDEFINITE OPTIMIZATION*

E. DE KLERK†, J. PENG†, C. ROOS†, AND T. TERLAKY‡

Abstract. Interior point methods for semidefinite optimization (SDO) have recently been studied intensively, due to their polynomial complexity and practical efficiency. Most of these methods are extensions of linear optimization (LO) algorithms. As opposed to the LO case, there are several different ways of constructing primal-dual search directions in SDO. The usual scheme is to apply linearization in conjunction with symmetrization to the perturbed optimality conditions of the SDO problem. Symmetrization is necessary since the linearized system is overdetermined. A way of avoiding symmetrization is to find a least squares solution of the overdetermined system. Such a "Gauss–Newton" direction was investigated by Kruk et al. [*The Gauss–Newton Direction in Semidefinite Programming*, Research report CORR 98-16, University of Waterloo, Waterloo, Canada, 1998] without giving any complexity analysis. In this paper we present a similar direction where a local norm is used in the least squares formulation, and we give a polynomial complexity analysis and computational evaluation of the resulting primal-dual algorithm.

Key words. semidefinite optimization, primal-dual search directions, interior point algorithms

AMS subject classification. 65K

PII. S1052623499352632

1. Introduction.

Interior point methods for semidefinite optimization (SDO) became a popular research area when it became clear that the algorithms for linear optimization (LO) can often be extended to the more general SDO case. Following the trend in LO, primal-dual algorithms soon enjoyed the most attention. Unlike the LO-case, however, there are many ways to obtain primal-dual search directions. Different directions arise when the perturbed optimality conditions are linearized and subsequently symmetrized (see section 1); a quite comprehensive survey of the search directions obtained this way may be found in [11]. The need for symmetrization arises from the fact that the system of linearized perturbed optimality conditions is overdetermined.

A recent idea by Kruk et al. [6] was to avoid symmetrization by solving a least squares problem by the Gauss–Newton method (see section 1). The authors obtained a numerically robust search direction in this way, but did not give convergence proofs for their search direction. The work in our paper was inspired by their approach: here we show that, by using scaling and a different (local) norm in the definition of the least squares problem, a direction is obtained which allows a polynomial time convergence analysis. We further show that the new direction is closely related to the well-known (primal) H..K..M and dual H..K..M directions (see the definitions in section 1); the primal part of the new direction coincides with the dual part of the (primal) H..K..M direction, and the dual part of the new direction is simply the primal

*Received by the editors February 22, 1999; accepted for publication (in revised form) August 14, 2000; published electronically March 15, 2001.

http://www.siam.org/journals/siopt/11-4/35263.html

†Faculty of Information Technology and Systems, Delft University of Technology, P.O. Box 5031, 2600 GA Delft, The Netherlands (E.deKlerk@ITS.TUDelft.nl, j.peng@ITS.TUDelft.nl, c.roos@ITS.TUDelft.nl).

‡Department of Computing and Software, McMaster University, Hamilton, ON, Canada (Terlaky@cas.mcmaster.ca). This research was done while the author was working at Delft University of Technology.

part of the dual H..K..M direction. Finally, we present some numerical experiments with the new direction.

Preliminaries. We consider the SDO problem in the standard form. Thus the primal problem (P) is given by

$$\text{(P)} \quad p^* = \inf \{ \operatorname{Tr} CX \ : \ \operatorname{Tr}(A_i X) = b_i \, (1 \le i \le m), \, X \succeq 0 \}$$

and its dual problem (D) is

$$\text{(D)} \quad d^* = \sup \left\{ b^T y \ : \ \sum_{i=1}^{m} y_i A_i + S = C, \, S \succeq 0 \right\},$$

where C and the A_i's are symmetric $n \times n$ matrices, $b, y \in \mathbb{R}^m$, and $X \succeq 0$ means that X is symmetric positive semidefinite. The matrices A_i are further assumed to be linearly independent. We will assume that a strictly feasible pair $(X \succ 0, S \succ 0)$ exists. This ensures the existence of an optimal primal-dual pair (X^*, S^*) with zero duality gap $(\operatorname{Tr}(X^* S^*) = 0)$.

The optimality conditions for the pair of problems are

$$\begin{aligned} \operatorname{Tr}(A_i X) &= b_i, \quad i = 1, \ldots, m, \quad X \succeq 0, \\ \sum_{i=1}^{m} y_i A_i + S &= C, \qquad\qquad\qquad S \succeq 0, \\ XS &= 0. \end{aligned}$$

If these conditions are perturbed to

$$\begin{aligned} \operatorname{Tr}(A_i X) &= b_i, \quad i = 1, \ldots, m, \quad X \succeq 0, \\ \sum_{i=1}^{m} y_i A_i + S &= C, \qquad\qquad\qquad S \succeq 0, \\ XS &= \mu I \end{aligned}$$

for some $\mu > 0$ where I denotes the identity matrix, then a unique solution of the perturbed system exists. This solution is denoted by $\{X(\mu), y(\mu), S(\mu)\}$. This solution may be seen as a parameterized curve in the Cartesian product of the primal and dual feasible regions,[1] called the *central path*, which converges to the analytic center of the optimal primal-dual set as $\mu \to 0$. The existence and uniqueness of the central path follow from the fact that $\{X(\mu), y(\mu), S(\mu)\}$ corresponds to the unique minimum of the strictly convex primal-dual barrier function

$$\Phi(X, S, \mu) = \frac{1}{\mu} \operatorname{Tr}(XS) - \log \det(XS) - n + n \log(\mu)$$

defined on the primal-dual feasible region. Because of the two different associations, the parameter μ is called either the *barrier parameter*, or the *centering* parameter.

Primal-dual interior point methods solve the system of perturbed optimality conditions approximately, followed by a reduction in μ. Ideally, the goal is to obtain primal and dual steps ΔX and ΔS, respectively, which satisfy $X + \Delta X \succeq 0$, $S + \Delta S \succeq 0$

[1] This Cartesian product of the primal and dual feasible sets will be called the *primal-dual feasible region*.

TABLE 1
Choices for the scaling matrix P.

P	Reference	Abbreviation
$\left[X^{\frac{1}{2}}\left(X^{\frac{1}{2}}SX^{\frac{1}{2}}\right)^{-\frac{1}{2}}X^{\frac{1}{2}}\right]^{-\frac{1}{2}}$	Nesterov and Todd [9]	NT
$X^{-\frac{1}{2}}$	Monteiro [7], Kojima, Shindoh, and Hara [5];	DH..K..M
$S^{\frac{1}{2}}$	Monteiro [7], Helmberg et al. [3], Kojima, Shindoh, and Hara [5];	PH..K..M
I	Alizadeh, Haeberly, and Overton [1]	AHO

and

$$(X+\Delta X)(S+\Delta S)=\mu I, \tag{1}$$

$$\mathrm{Tr}\,(A_i\Delta X)=0,\quad i=1,\ldots,m, \tag{2}$$

$$\sum_{i=1}^{m}\Delta y_i A_i+\Delta S=0, \tag{3}$$

$$(\Delta X)^T=\Delta X,\quad (\Delta S)^T=\Delta S. \tag{4}$$

Note that the requirement $\Delta S^T=\Delta S$ in (4) is redundant, due to the fact that the matrices A_i in (3) are symmetric. Furthermore, (1) is nonlinear, and primal-dual methods differ with regard to how it is linearized. Care must be taken to ensure that the resulting linear system is not overdetermined. Zhang [14] suggested discarding the symmetry requirements (4) and replacing the nonlinear equation by

$$H_P(XS+\Delta XS+X\Delta S-\mu I)=0,$$

where H_P is the linear transformation given by

$$H_P(M):=\frac{1}{2}\left[PMP^{-1}+P^{-T}M^TP^T\right]$$

for any matrix M and where the *scaling matrix* P determines the symmetrization strategy. Some popular choices for P are listed in Table 1. The resulting linear systems are now solvable (for the AHO direction ($P=I$) solvability is only guaranteed if (X,S) lies in a certain neighborhood of the central path), and the solution matrices $\Delta X$and ΔS are symmetric.

In the recent paper by Kruk et al. [6], the symmetrization operator H_P is not used, and the following least squares problem is solved instead:

$$\min\|XS+\Delta XS+X\Delta S-\mu I\|^2 \tag{5}$$

subject to (s.t.)

$$\begin{aligned}\mathrm{Tr}\,(A_i\Delta X) &= 0,\quad i=1,\ldots,m,\\ \sum_{i=1}^{m}\Delta y_iA_i+\Delta S &= 0,\\ \Delta X &= \Delta X^T,\end{aligned}$$

where the norm is the Frobenius norm. Note that the symmetry of ΔX is forced. The authors proved (among other things) the following about the resulting Gauss–Newton (GN) direction:

- its existence and uniqueness;
- it reduces to the familiar primal-dual direction in the special case of linear optimization;
- it coincides with all the other primal-dual directions from Table 1 if the least squares residual in (5) is zero at optimality.

The new direction we propose can be introduced in a similar way as the GN direction—as will be shown in the next section—and it shares all the above-mentioned features of the GN direction. Moreover, it allows a polynomial convergence analysis in the usual primal-dual algorithmic framework, as will become clear in section 4.

2. The new search direction. Using the well-known NT-scaling (see Table 1), we now reformulate the system (1)–(3). Defining

$$D = S^{-\frac{1}{2}}\left(S^{\frac{1}{2}}XS^{\frac{1}{2}}\right)^{\frac{1}{2}}S^{-\frac{1}{2}} = X^{\frac{1}{2}}\left(X^{\frac{1}{2}}SX^{\frac{1}{2}}\right)^{-\frac{1}{2}}X^{\frac{1}{2}},$$

one has $D^{-1}X = SD$. Using this, we introduce

$$V := D^{-\frac{1}{2}}XD^{-\frac{1}{2}} = D^{\frac{1}{2}}SD^{\frac{1}{2}}.$$

The matrices D and V are symmetric positive definite. We also introduce the scaled search directions $\hat{D}_X$ and $\hat{D}_S$:

$$\hat{D}_X := D^{-\frac{1}{2}}\Delta XD^{-\frac{1}{2}}, \quad \hat{D}_S := D^{\frac{1}{2}}\Delta SD^{\frac{1}{2}}.$$

Finally, scaling the data matrices A_i to

$$\tilde{A}_i := D^{\frac{1}{2}}A_iD^{\frac{1}{2}}, \quad 1 \le i \le m,$$

the system (1)–(4) can be reformulated as follows:

$$\left(V + \hat{D}_X\right)\left(V + \hat{D}_S\right) = \mu I, \tag{6}$$

$$\mathrm{Tr}\left(\tilde{A}_i\hat{D}_X\right) = 0, \quad i = 1, \ldots, m, \tag{7}$$

$$\sum_{i=1}^{m}\Delta y_i\tilde{A}_i + \hat{D}_S = 0, \tag{8}$$

$$\left(\hat{D}_X\right)^T = \hat{D}_X. \tag{9}$$

Equation (6) can be rewritten as

$$V^2 + V\hat{D}_S + \hat{D}_XV + \hat{D}_X\hat{D}_S - \mu I = 0.$$

Thus the desired scaled displacements are the (unique) solutions of the least squares problem

$$\min\left\|V^2 + V\hat{D}_S + \hat{D}_XV + \hat{D}_X\hat{D}_S - \mu I\right\|^2,$$

subject to the constraints (7)–(9), and the optimal value of this problem is zero. We now omit the nonlinear term $\hat{D}_X\hat{D}_S$ from the objective function of the least squares problem. This omission makes it important to specify which norm is used, since the

optimal solution to our new least squares problem will depend on the norm. The norm which we choose is the norm induced by the inner product:

$$\langle A, B\rangle := \mathrm{Tr}\,\left(V^{-1}AV^{-1}B^T\right) \quad \forall A \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times n}.$$

This can also be viewed as the local norm induced by the Hessian of the self-concordant barrier

$$f(V) = -\log\det(V),$$

since the Hessian of f evaluated at V is the linear operator

$$\nabla^2 f(V) : H \mapsto V^{-1}HV^{-1}.$$

Thus we obtain the least squares problem

$$\min \left\| V^{-\frac{1}{2}} \left(V^2 + V\hat{D}_S + \hat{D}_X V - \mu I \right) V^{-\frac{1}{2}} \right\|^2,$$

subject to the constraints (7)–(9) and where the norm now indicates the Frobenius norm. For convenience, we also introduce the notation

$$U := \frac{1}{\sqrt{\mu}}V, \quad D_X := \frac{1}{\sqrt{\mu}}\hat{D}_X, \quad D_S := \frac{1}{\sqrt{\mu}}\hat{D}_S, \quad \Delta\tilde{y} := \frac{1}{\sqrt{\mu}}\Delta y.$$

Using this notation, we can reformulate the above least squares problem as follows:

$$\text{(LQ)} \qquad \begin{cases} \min f\left(D_X, D_S\right) & := & \frac{1}{2}\left\| U + U^{\frac{1}{2}}D_S U^{-\frac{1}{2}} + U^{-\frac{1}{2}}D_X U^{\frac{1}{2}} - U^{-1} \right\|^2, \\ \text{s.t. } \mathrm{Tr}\left(\tilde{A}_i D_X\right) & = & 0, \quad i = 1\ldots, m, \\ D_X{}^T & = & D_X, \\ D_S & = & -\sum_{i=1}^{m} \Delta\tilde{y}_i \tilde{A}_i. \end{cases}$$

In what follows we will frequently use the notation

$$R := U + U^{\frac{1}{2}}D_S U^{-\frac{1}{2}} + U^{-\frac{1}{2}}D_X U^{\frac{1}{2}} - U^{-1}. \tag{10}$$

In other words, $f\left(D_X, D_S\right) = \frac{1}{2}\|R\|^2$ is the residual of the least squares problem (LQ). Note that the derivatives of f with respect to D_X and D_S are, respectively, given by

$$\nabla_{D_X} f(D_X, D_S) = U^{-\frac{1}{2}}RU^{\frac{1}{2}}, \quad \nabla_{D_S} f(D_X, D_S) = U^{\frac{1}{2}}RU^{-\frac{1}{2}}. \tag{11}$$

Optimality conditions for the least squares problem. We can formulate the optimality conditions for the least squares problem (LQ) by forming the Lagrangian:

$$L(D_X, D_S, \Delta\tilde{y}, \lambda, M_1, M_2) := f(D_X, D_S) - \sum_{i=1}^{m} \lambda_i \mathrm{Tr}\left(\tilde{A}_i D_X\right) + \mathrm{Tr}\left(\left(D_X - D_X^T\right) M_1\right)$$
$$+\mathrm{Tr}\left(M_2\left(D_S + \sum_{i=1}^{m} \Delta\tilde{y}_i \tilde{A}_i\right)\right),$$

where $\lambda \in \mathbb{R}^m$, $M_1 \in \mathbb{R}^{n\times n}$, and $M_2 \in \mathbb{R}^{n\times n}$ are Lagrange multipliers. Using the expressions in (11), one can easily rewrite the optimality condition

$$\nabla L(D_X, D_S, \Delta\tilde{y}, \lambda, M_1, M_2) = 0$$

as

$$U^{-\frac{1}{2}} R U^{\frac{1}{2}} = \sum_{i=1}^{m} \lambda_i \tilde{A}_i + M, \tag{12}$$

$$\text{Tr}\left(\tilde{A}_i U^{\frac{1}{2}} R U^{-\frac{1}{2}}\right) = 0, \quad i = 1, \ldots, m, \tag{13}$$

$$\text{Tr}\left(\tilde{A}_i D_X\right) = 0, \quad i = 1, \ldots, m, \tag{14}$$

$$D_X - D_X^T = 0, \tag{15}$$

$$D_S = -\sum_{i=1}^{m} \Delta\tilde{y}_i \tilde{A}_i, \tag{16}$$

where $M = M_1^T - M_1$ is a skew-symmetric matrix.

Existence and uniqueness of the new direction. We now state an existence and uniqueness result for the new search direction.

THEOREM 2.1 (existence and uniqueness of the new direction). *The problem* (LQ) *determines the displacements D_X, $\Delta\tilde{y}$, and D_S uniquely. Furthermore, one has $D_X = 0$ and $\Delta\tilde{y} = 0$ (whence $D_S = 0$), if and only if $U = I$ or, equivalently, $XS = \mu I$.*

This result can be proved by using the optimality conditions of (LQ). We omit such a proof here, since the theorem will follow from results in section 5, where we will explore the relation between the new direction and directions from literature.

3. Estimating the least squares residual. In the analysis of the new search direction it is essential to show that the residual of the least squares problem, $\|R\|$, is "small enough" at the optimal solution of (LQ) if the current iterate is close enough to the central path. The residual can be bounded from above in terms of the proximity to the target point μI, where the proximity is measured by

$$\delta(X, S, \mu) := \frac{1}{2}\left\|U - U^{-1}\right\|. \tag{17}$$

Note that $\delta(X, S, \mu) = 0$ if and only if $XS = \mu I$. In what follows, we will use the notation $\delta := \delta(X, S, \mu)$ if no confusion is possible.

Let us define $D_V := D_X + D_S$ and $Q_V := D_X - D_S$. Note that $\|D_V\| = \|Q_V\|$. We can now decompose $R := U^{-1} - U + U^{\frac{1}{2}} D_S U^{-\frac{1}{2}} + U^{-\frac{1}{2}} D_X U^{\frac{1}{2}}$ into a symmetric and skew-symmetric component, say

$$R := R_{sym} + R_{skew},$$

where

$$R_{sym} = U^{-1} - U + \frac{1}{2}\left(U^{\frac{1}{2}} D_V U^{-\frac{1}{2}} + U^{-\frac{1}{2}} D_V U^{\frac{1}{2}}\right)$$

and

$$R_{skew} = \frac{1}{2}\left(U^{-\frac{1}{2}} Q_V U^{\frac{1}{2}} - U^{\frac{1}{2}} Q_V U^{-\frac{1}{2}}\right).$$

By construction, one has

$$\|R\|^2 = \|R_{sym}\|^2 + \|R_{skew}\|^2.$$

The new direction $D_V \equiv D_X + D_S$ is chosen such that $\|R\|$ is minimized. In order to get an upper bound on the value $\|R\|$ for the new direction, we can consider the value of $\|R\|$ for a class of search directions where $U^{\frac{1}{2}} D_V U^{-\frac{1}{2}} = D_V$. In this way we obtain the bound

$$(18)\ \|R\|^2 \leq 4\delta^2 + 2\mathrm{Tr}\left((U^{-1} - U)D_V\right) + \|D_V\|^2 + \left\| \frac{1}{2}\left(U^{-\frac{1}{2}} Q_V U^{\frac{1}{2}} - U^{\frac{1}{2}} Q_V U^{-\frac{1}{2}}\right)\right\|^2,$$

where we have used $\left\|U - U^{-1}\right\|^2 = 4\delta^2$. In order to get an upper bound on $\|R_{skew}\|^2$ (the last term in (18)) we use the following lemma.

LEMMA 3.1. *Suppose that the $n \times n$ matrix A is symmetric positive definite and $\xi(A) = \mathrm{Tr}\left(A^2\right) - 2n + \mathrm{Tr}\left(A^{-2}\right)$. Then for any symmetric matrix $\bar{A}$, one has*

$$\|A\bar{A}A^{-1} - A^{-1}\bar{A}A\|^2 \leq \frac{\xi(A^2)}{2}\|\bar{A}\|^2.$$

Proof. Since A is symmetric positive definite, we can assume in general that A is a diagonal matrix with $a_i > 0$ on the ith diagonal position, by taking an orthogonal transformation if necessary. Denoting $\hat{A} = A\bar{A}A^{-1} - A^{-1}\bar{A}A$, one has

$$\hat{A}_{ii} = 0, \quad \hat{A}_{ij} = \left(\frac{a_i}{a_j} - \frac{a_j}{a_i}\right)\bar{A}_{ij} \quad (i \neq j).$$

The above relation means that

$$\begin{aligned}
\|\hat{A}\|^2 &\leq \max_{i,j}\left(\frac{a_i^2}{a_j^2} - 2 + \frac{a_j^2}{a_i^2}\right)\|\bar{A}\|^2 \\
&\leq \frac{1}{2}\max_{i,j}\left(a_i^4 + a_j^4 - 4 + \frac{1}{a_i^4} + \frac{1}{a_j^4}\right)\|\bar{A}\|^2 \\
&\leq \frac{\xi(A^2)}{2}\|\bar{A}\|^2,
\end{aligned}$$

where the second inequality can easily be verified by calculus, and the third inequality follows by noting that

$$\xi(A^2) = \sum_{i=1}^{n}\left(a_i^4 + \frac{1}{a_i^4} - 2\right). \qquad \square$$

The lemma implies that

$$\begin{aligned}
\|R_{skew}\|^2 &\equiv \left\| \frac{1}{2}\left(U^{-\frac{1}{2}} Q_V U^{\frac{1}{2}} - U^{\frac{1}{2}} Q_V U^{-\frac{1}{2}}\right)\right\|^2 \\
&\leq \frac{1}{8}\xi\left(U^{-1}\right)\|Q_V\|^2 \\
&= \frac{1}{2}\delta^2\|D_V\|^2,
\end{aligned}$$

where we have used $\|D_V\| = \|Q_V\|$ and $\xi(U^{-1}) = 4\delta^2$. Substituting the bound for $\|R_{skew}\|$ into (18) yields

$$\|R\|^2 \leq 4\delta^2 + 2\text{Tr}\,\left(D_V(U^{-1} - U)\right) + \left(1 + \frac{1}{2}\delta^2\right)\|D_V\|^2. \tag{19}$$

The right-hand side is a convex quadratic function of D_V and is minimized by

$$D_V^{NT} = -\frac{1}{1 + \frac{1}{2}\delta^2}\left(U^{-1} - U\right), \tag{20}$$

which happens to be a damped step along the Nesterov–Todd direction (see, e.g., de Klerk [2]). Substituting (20) into (19) yields

$$\begin{aligned}\|R\|^2 &\leq 4\delta^2 + 2\text{Tr}\,\left(D_V^{NT}(U - U^{-1})\right) + \left(1 + \frac{1}{2}\delta^2\right)\|D_V^{NT}\|^2 \\ &= 4\delta^2 - \frac{2}{1 + \frac{1}{2}\delta^2}(4\delta^2) + \frac{1}{1 + \frac{1}{2}\delta^2}(4\delta^2) \\ &= 4\delta^2\left(\frac{\delta^2}{2 + \delta^2}\right). \qquad (21)\end{aligned}$$

Let us now suppose that D_X, D_S are the solutions of (LQ), and denote

$$R_U := U^{\frac{1}{2}} D_S U^{-\frac{1}{2}} + U^{-\frac{1}{2}} D_X U^{\frac{1}{2}}.$$

Our main result in this section can be stated as follows.

LEMMA 3.2. *Let δ be defined by* (17). *One has*

$$\frac{2\delta}{\sqrt{1 + \frac{1}{2}\delta^2}} \leq \|R_U\| \leq 2\delta. \tag{22}$$

Proof. From the optimality conditions of (LQ) we immediately derive that

$$\text{Tr}\,\left(R^T R_U\right) = 0,$$

by noting that (12), (15), and (14) imply

$$\text{Tr}\left(R^T U^{-\frac{1}{2}} D_X U^{\frac{1}{2}}\right) = 0$$

and (13) and (16) imply

$$\text{Tr}\left(R^T U^{\frac{1}{2}} D_S U^{-\frac{1}{2}}\right) = 0.$$

Since $R = U^{-1} - U + R_U$ and R and R_U are orthogonal, we have

$$4\delta^2 \equiv \left\|U^{-1} - U\right\|^2 = \|R\|^2 + \|R_U\|^2 \leq 4\delta^2\left(\frac{\delta^2}{2 + \delta^2}\right) + \|R_U\|^2, \tag{23}$$

where the inequality follows from (21). The equations in (23) together with the nonnegativity of $\|R\|$ imply

$$4\delta^2 \equiv \left\|U^{-1} - U\right\|^2 \geq \|R_U\|^2,$$

and the inequality in (23) implies

$$\|R_U\|^2 \geq 4\delta^2 - 4\delta^2 \left(\frac{\delta^2}{2+\delta^2}\right) = 4\delta^2 \left(\frac{2}{2+\delta^2}\right).$$

Thus we have shown that

$$2\delta \geq \|R_U\| \geq \frac{2\delta}{\sqrt{1+\frac{1}{2}\delta^2}}. \qquad \square \tag{24}$$

4. Complexity analysis of a primal-dual method. In the present section, we will first propose a primal-dual path following method based on the new search direction, and we will subsequently perform a complexity analysis of the algorithm.

GENERIC PRIMAL-DUAL PATH FOLLOWING ALGORITHM.

Input

A strictly feasible starting pair (X^0, S^0), satisfying $\delta\left(X^0, S^0, \mu^0\right) \leq \tau$.

Parameters

A centering parameter $\tau > 0$;
An accuracy parameter $\epsilon > 0$;
An updating parameter $\theta < 1$;
An initial centering parameter $\mu^0 > 0$.

$X := X^0$; $S := S^0$;
while $\mathrm{Tr}(XS) > \epsilon$ **do**
  if $\delta(X, S, \mu) \leq \tau$ **do** (*outer iteration*)
    $\mu := (1-\theta)\mu$;
  else if $\delta(X, S, \mu) > \tau$ **do** (*inner iteration*)
    Compute $\Delta X, \Delta S$ by solving (LQ);
    Find α such that $\Phi(X, S, \mu) - \Phi(X + \alpha\Delta X, S + \alpha\Delta S, \mu)$ is sufficiently large;
    (A suitable default choice for α is given by (26).)
    $X := X + \alpha\Delta X$, $S := S + \alpha\Delta S$;
  end
end

Recall that

$$\Phi(X, S, \mu) = \frac{\mathrm{Tr}\,(XS)}{\mu} - n - \log\det(XS) + n\log\mu.$$

In the update of the iterate, we require that the step length α be chosen such that the barrier function $\Phi(X, S, \mu)$ decreases sufficiently. Lemma 4.2 will give a default value for α.

It is easy to verify that the barrier function can also be rewritten as

$$f(U) = \Phi(X, S, \mu) = \mathrm{Tr}\,\left(U^2\right) - n - \log\det\left(U^2\right).$$

Assuming that D_X, D_S are solutions of (LQ), we want to estimate the decreasing value of the barrier function, given by

$$\begin{aligned}\Delta\Phi(\alpha) &= f(U) - \left(\mathrm{Tr}\,((U+\alpha D_X)(U+\alpha D_S)) - n - \log\det\,(U+\alpha D_X)(U+\alpha D_S)\right)\\ &= -\alpha\mathrm{Tr}\,(UD_S + D_X U) + \log\det\,(I + \alpha U^{-\frac{1}{2}} D_X U^{-\frac{1}{2}})(I + \alpha U^{-\frac{1}{2}} D_S U^{-\frac{1}{2}}),\end{aligned}$$

where we have used the orthogonality of D_X and D_S. Now we have the following general bound on the reduction $\Delta\Phi(\alpha)$ which holds for any search direction. (For a

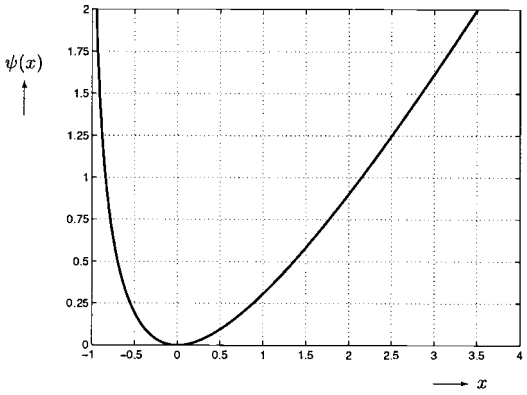


FIG. 1. *The graph of* ψ.

proof see, e.g., Jiang [4] and Roos, Terlaky, and Vial [10, Lemma II.69] for the linear optimization case.)

THEOREM 4.1. *Let* (X, S) *be a strictly feasible pair and let* (D_X, D_S) *be any feasible solution to problem (LQ); define* $D_V := D_X + D_S$. *Then*

$$\Delta\Phi(\alpha) \geq -\alpha \mathrm{Tr}\,(UD_V) + \alpha \mathrm{Tr}\,\left(U^{-1}D_V\right) - \psi(-\alpha h),$$

where $\psi(t) := t - \log(1+t)$ *(see Figure* 1*), and*

$$h^2 = \mathrm{Tr}\,\left(U^{-1}D_X U^{-1}D_X + U^{-1}D_S U^{-1}D_S\right).$$

Moreover any value of α *satisfying* $\alpha \leq \frac{1}{h}$ *is a feasible step length.*

COROLLARY 4.1. *Let* (D_X, D_S) *denote the optimal solution of* (LQ). *One has*

$$\Delta\Phi(\alpha) \geq \alpha\|R_U\|^2 - \psi(-\alpha h).$$

Proof. Using the definition of R in (10) and $\mathrm{Tr}\left(R^T R_U\right) = 0$ one has

$$-\mathrm{Tr}\,(UD_V) + \mathrm{Tr}\,\left(U^{-1}D_V\right) = \mathrm{Tr}\left(R_U^T\left(U^{-1} - U\right)\right) = \mathrm{Tr}\left(R_U^T\left(R_U - R\right)\right) = \|R_U\|^2. \tag{25}$$

The required result now follows from Theorem 4.1. ∎

All that remains is to give an upper bound for the term $-\psi(-\alpha h)$. This can be done by using the following lemma.

LEMMA 4.1. *Let* (D_X, D_S) *denote the optimal solution of* (LQ). *One has*

$$h \leq \rho(\delta)\|R_U\|,$$

where $\rho(\delta) := \delta + \sqrt{1+\delta^2}$.

Proof. By definition,

$$\begin{aligned} h^2 &= \mathrm{Tr}\,\left(U^{-1}D_X U^{-1}D_X + U^{-1}D_S U^{-1}D_S\right) \\ &= \mathrm{Tr}\,\left(U^{-2}\left(UD_X U^{-1}D_X + UD_S U^{-1}D_S\right)\right) \\ &\leq \lambda_{\max}\left(U^{-2}\right)\mathrm{Tr}\,\left(UD_X U^{-1}D_X + UD_S U^{-1}D_S\right) \\ &\leq \rho^2(\delta)\mathrm{Tr}\,\left(UD_X U^{-1}D_X + UD_S U^{-1}D_S\right) \\ &= \rho^2(\delta)\|R_U\|^2, \end{aligned}$$

where the last inequality is a result by Jiang [4]. (See Roos, Terlaky, and Vial [10, Lemma II.60] for the analogous result in the linear optimization case.) ∎

LEMMA 4.2. *Let* (D_X, D_S) *denote the optimal solution of* (LQ). *One has*

$$\Delta\Phi(\bar{\alpha}) \geq \psi\left(\frac{\|R_U\|}{\rho(\delta)}\right) \geq \psi\left(\frac{2\delta}{\rho(\delta)\sqrt{1+\frac{1}{2}\delta^2}}\right),$$

for

$$\bar{\alpha} := \frac{1}{h} - \frac{1}{\|R_U\|^2 + h}.$$

Proof. From Corollary 4.1 we have

$$\begin{aligned}\Delta\Phi(\alpha) &\geq \alpha\|R_U\|^2 - \psi(-\alpha h)\\ &\equiv \alpha\|R_U\|^2 + \alpha h + \log(1-\alpha h).\end{aligned}$$

The right-hand side of the inequality is maximized by

$$\bar{\alpha} = \frac{1}{h} - \frac{1}{\|R_U\|^2 + h}. \tag{26}$$

This maximizer yields the bound

$$\Delta\Phi(\bar{\alpha}) \geq \psi\left(\frac{\|R_U\|^2}{h}\right),$$

which, by Lemma 4.1, implies

$$\Delta\Phi(\bar{\alpha}) \geq \psi\left(\frac{\|R_U\|}{\rho(\delta)}\right).$$

Finally we use Lemma 3.2 to complete the proof. ∎

Now we show that δ is bounded in terms of the barrier function Φ, and vice versa. To this end, we use the following lemma which was proved for linear optimization by Roos, Terlaky, and Vial [10, Lemma II.67]. The extension of the proof to the SDO case is mechanical and is therefore omitted.

LEMMA 4.3. *Let* $\delta := \delta(X, S; \mu)$ *and* $\rho(\delta) := \delta + \sqrt{1+\delta^2}$. *Then*

$$\psi\left(\frac{-2\delta}{\rho(\delta)}\right) \leq \Phi(X, S, \mu) \leq \psi\left(2\delta\rho(\delta)\right).$$

The statement of the lemma is illustrated in Figure 2.

Small update methods. We are now in a position to perform the complexity analysis for a small update version of the algorithm. To fix our ideas, we choose the parameters

$$\tau = \frac{1}{2}, \quad \theta = \frac{1}{10\sqrt{n}}.$$

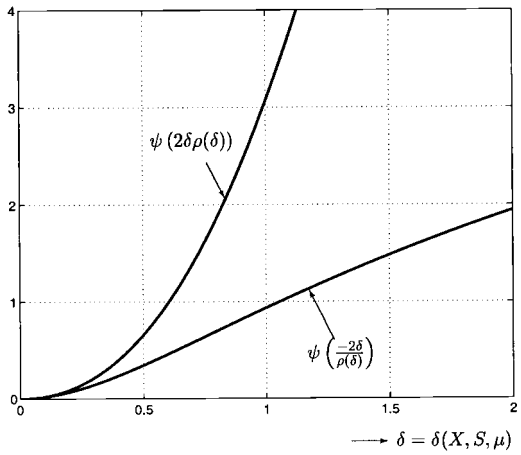


FIG. 2. *Bounds for* $\Phi(X, S, \mu)$.

We assume that at the current iterates (X, S) the proximity measure satisfies $\delta(X, S, \mu) \leq \tau = \frac{1}{2}$. In this situation, we perform the update $\mu^+ = (1-\theta)\mu$ (outer iteration). Analogously to the linear optimization case, one has (see Lemma IV. 36 in [10])

$$\delta(X, S, \mu^+) \leq \frac{2\delta + \theta\sqrt{n}}{2\sqrt{1-\theta}} \leq \frac{2\tau + \sqrt{n}\theta}{2\sqrt{1-\theta}} < 0.58.$$

This also means (by Lemma 4.3) that at the beginning of the inner iterative procedure, one has

$$\Phi(X, S, \mu^+) \leq \psi(2\delta\rho(\delta)) \leq 0.910.$$

This bound implies that the proximity $\delta(X, S, \mu^+)$ is also bounded from above by a constant during *all* inner iterations, by Lemma 4.3 (see Figure 2):

$$\delta \leq 0.98.$$

At each inner iteration one has $\delta \geq \frac{1}{2}$, which implies

$$\|R_U\| \geq \frac{2\delta}{\sqrt{1 + \frac{1}{2}\delta^2}} = \frac{1}{\sqrt{1.125}} \geq 0.9428$$

by Lemma 3.2. Lemma 4.2 shows that the reduction of the barrier function is at least

$$\psi\left(\frac{\|R_U\|}{\rho(\delta)}\right) \geq 0.062. \tag{27}$$

In order to guarantee that $\delta(X, S, \mu^+) \leq \frac{1}{2}$ at the end of the inner iteration phase, one must reduce the value of Φ to below 0.344 (see Figure 2). The bound in (27) implies that, after at most

$$\lceil (0.910 - 0.344)/0.062 \rceil = 10 \tag{28}$$

inner iterations, we have computed a pair (X, S) such that $\delta(X, S, \mu^+) \leq \frac{1}{2}$. Hence we have the following complexity bound for the algorithm.

THEOREM 4.2. *If $\tau = \frac{1}{2}$ and $\theta = \frac{1}{10\sqrt{n}}$, the total number of iterations required by the primal-dual path following algorithm is no more than*

$$\left\lceil 100\sqrt{n} \log \frac{2.5n\mu^0}{\epsilon} \right\rceil.$$

Proof. It can easily be shown that after

$$\left\lceil \frac{1}{\theta} \log \frac{n\mu^0}{\epsilon} \right\rceil \tag{29}$$

barrier parameter updates (outer iterations) one has $n\mu \leq \epsilon$ (cf. Lemma II.17 in [10]).

At the end of the inner iterations with respect to μ one has computed a pair (X, S) such that $\delta(X, S, \mu) \leq \frac{1}{2}$. Using the definition of δ, it is trivial to show that this implies

$$\operatorname{Tr}(XS) \leq 2.5n\mu,$$

and consequently $\operatorname{Tr}(XS) \leq 2.5\epsilon$.

Replacing ϵ by $\epsilon/2.5$ and multiplying the number of outer iterations in (29) by the bound (28) yields the theorem. □

REMARK 4.1. *We have only analyzed one special small update algorithm, but one can easily derive similar results for any fixed $\tau > 0$ and θ of the order $O(\frac{1}{\sqrt{n}})$.*

5. Relation to other search directions. In this section we show that the scaled Gauss–Newton (SGN) direction introduced in this paper is closely related to the primal and dual H..K..M directions (see Table 1). In particular, the ΔX part of the SGN direction is simply the ΔX part of the dual H..K..M direction, while the ΔS part of the SGN direction is the same as the ΔS part of the primal H..K..M direction. Note that this relationship implies Theorem 2.1.

The key in proving this is to decompose problem (LQ) into two independent subproblems. To this end, recall that for all feasible D_X and D_S, it holds that $\operatorname{Tr}(D_X D_S) = 0$. Using this fact, we can rewrite the objective of problem (LQ) as

$$\left\| U + U^{\frac{1}{2}} D_S U^{-\frac{1}{2}} - U^{-1} \right\|^2 + \left\| U + U^{-\frac{1}{2}} D_X U^{\frac{1}{2}} - U^{-1} \right\|^2 - \left\| U - U^{-1} \right\|^2. \tag{30}$$

Omitting the last (constant) term in the last expression, we can separate problem (LQ) into two subproblems,

$$\begin{aligned} \text{(SGN1)} \quad & \min_{D_X} \left\| U + U^{-\frac{1}{2}} D_X U^{\frac{1}{2}} - U^{-1} \right\|^2, \\ & \operatorname{Tr}\left(\tilde{A}_i D_X \right) = 0, \quad D_X = D_X^T; \end{aligned}$$

and

$$\begin{aligned} \text{(SGN2)} \quad & \min_{D_S} \left\| U + U^{\frac{1}{2}} D_S U^{-\frac{1}{2}} - U^{-1} \right\|^2, \\ & D_S = -\sum_{i=1}^{m} \Delta y_i \tilde{A}_i. \end{aligned}$$

To compute the SGN direction, one can solve the two independent subproblems (SGN1) and (SGN2). Now let us recall the definition of the primal H..K..M direction. As observed by Monteiro (see Lemma 2.1 in [7] and Kojima, Shindoh, and Hara [5]), the primal H..K..M direction is the unique solution of the following linear system:

$$\begin{aligned} XS + X\Delta S + (\Delta X + W)S &= \mu I, \\ \mathrm{Tr}(A_i \Delta X) &= 0; \quad i = 1, \ldots, m, \\ \sum_{i=1}^{m} \Delta y_i A_i + \Delta S = 0, \quad W + W^T = 0, \quad \Delta X &= \Delta X^T. \end{aligned}$$

Premultiplying the first equation in the above system by $D^{-1/2}$ and postmultiplying by $D^{1/2}$, and then dividing by μ, we can rewrite the above system in the scaled space as

$$U^2 + UD_S + (D_X + \tilde{W})U = I,$$

$$\mathrm{Tr}\Big(\tilde{A}_i D_X\Big) = 0, \quad i = 1, \ldots, m, \tag{31}$$

$$\sum_{i=1}^{m} \Delta\tilde{y}_i \tilde{A}_i + D_S = 0, \quad \tilde{W} + \tilde{W}^T = 0, \quad D_X = D_X^T, \tag{32}$$

where $\tilde{W} = \frac{1}{\mu} D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$ is skew symmetric and $\Delta\tilde{y}_i = \frac{1}{\mu}\Delta y_i$ as before. Again by pre- and postmultiplying the first equation by $U^{-1/2}$ we obtain

$$U - U^{-1} + U^{\frac{1}{2}} D_S U^{-\frac{1}{2}} + U^{-\frac{1}{2}}(D_X + \tilde{W})U^{\frac{1}{2}} = 0. \tag{33}$$

Now we state our main result in this section.

PROPOSITION 5.1. *Suppose that ΔS^* is the solution of the primal H..K..M direction. Then $D_S^* = \frac{1}{\mu} D^{\frac{1}{2}} \Delta S^* D^{\frac{1}{2}}$ is the unique solution of the problem* (SGN2).

Proof. The KKT system for problem (SGN2) can easily be written in the form (33), (31), and (32). ∎

We can approach the solution of problem (SGN1) in exactly the same way, by observing that the dual H..K..M direction is the unique solution of the following problem (see [5]):

$$\begin{aligned} XS + X(\Delta S + W) + \Delta X S &= \mu I, \\ \mathrm{Tr}(A_i \Delta X) &= 0, \qquad i = 1, \ldots, m, \\ \textstyle\sum_{i=1}^{m} \Delta y_i A_i + \Delta S &= 0, \\ \Delta X &= \Delta X^T, \quad W + W^T = 0. \end{aligned} \tag{34}$$

In the same way as before, one can now prove the following.

PROPOSITION 5.2. *Suppose that ΔX^* is the solution of the dual H..K..M directions. Then*

$$D_X^* = \frac{1}{\mu} D^{-\frac{1}{2}} \Delta X^* D^{-\frac{1}{2}}$$

is the unique solution of the problem (SGN1).

TABLE 2
Average number of iterations for the SDTP3 algorithm IPF using various search directions.

Test set	AHO	PH..K..M	NT	GT	SGN	DH..K..M
1: Random SDP $n = m = 50$	16.2	19.3	17.7	16.6	17.9	16.8
2: Norm min. problem $n = 100, m = 26$	18.9	21.3	20.4	19.9	21.1	19.5
3: Cheby. approx. in $\mathbb{R}^{n\times n}$ $n = 100, m = 26$	16.6	19.6	17.6	16.9	19.0	17.0
4: Max-cut $n = m = 50$	15.4	17.7	16.1	16.0	18.4	15.7
5: ETP $n = 100, m = 50$	29.5	34.0	31.2	30.7	32.1	30.4
6: Lovász θ function $n = 30, m = 220$	20.2	23.1	21.5	20.8	22.7	21.8
7: Log. Cheby. prob. $n = 300, m = 51$	21.5	22.7	22.6	21.1	23.8	21.2
8: Cheby. approx. on $\mathcal{C}$	16.1	16.6	16.3	16.1	19.0	16.1

REMARK 5.1. *The relation between the SGN direction and the H..K..M directions implies that the SGN direction shares the same scale-invariance properties as the H..K..M directions; see, e.g.,* [11] *for the definition of scale-invariance.*

In the appendix to this paper we show how the SGN direction can be computed via the solution of the primal and dual H..K..M directions. In particular, we show there that the computational complexity of the SGN direction is upper bounded by

$$2mn^3 + m^2n^2 + \frac{2}{3}m^3 + O\left(n^3 + mn^2 + m^2n\right)$$

flops.[2] In comparison, one has the bound

$$\frac{2}{3}mn^3 + \frac{1}{2}m^2n^2 + \frac{1}{3}m^3 + O\left(n^3 + mn^2 + m^2n\right)$$

flops for the NT direction, and

$$3\frac{2}{3}mn^3 + m^2n^2 + \frac{2}{3}m^3 + O\left(n^3 + mn^2 + m^2n\right)$$

flops for the AHO direction [8].

6. Numerical results. We have implemented two algorithms based on the SGN direction and the dual H..K..M direction by changing the main subroutine of SDPT3 (SDP.m in version 1.3) slightly to admit these two additional search directions. The algorithm we tested is the infeasible path following algorithm without second-order corrector (Algorithm IPF in [13]). Tables 2, 3, and 4 show the performance of this algorithm for various search directions. The test problems are taken from [13], and each test set consists of ten random instances generated by the subroutines in SDPT3. The convergence criterion was to reduce the initial duality gap by a factor of 10^{10}.

The tables show that the algorithm based on the SGN and the dual H..K..M directions are comparable to other search directions with respect to the number of iterations. As for the required CPU time, the SGN direction requires slightly less

[2]We follow the convention in, e.g., [8] that one flop is any floating point operation, i.e., addition and multiplication of two floating point numbers both constitute one flop.

TABLE 3
Average running time of the algorithms.

Test set	AHO	PH..K..M	NT	GT	SGN	DH..K..M
1	32.4	19.4	19.6	23.4	26.8	21.6
2	78.3	42.0	46.3	58.4	72.2	60.6
3	65.0	37.1	38.1	47.3	61.1	50.2
4	20.3	10.7	9.3	13.2	19.7	13.3
5	49.0	26.8	23.5	32.3	38.2	28.3
6	111.0	48.6	40.1	59.1	107.6	68.9
7	45.5	22.7	26.9	34.2	39.5	28.1
8	28.3	14.6	18.1	22.5	29.4	20.7

TABLE 4
Average absolute value of the logarithm of the duality gap at termination, i.e., $|\log_{10} \operatorname{Tr}(XS)|$ *where* (X, S) *are the final iterates.*

Test set	AHO	PH..K..M	NT	GT	SGN	DH..K..M
1	9.4	7.8	7.0	9.1	7.1	6.8
2	12.4	9.5	8.5	12.1	9.1	8.8
3	13.4	10.8	9.4	13.2	9.8	9.6
4	10.9	8.8	7.8	10.6	8.2	7.8
5	7.8	7.0	6.6	8.6	6.4	6.8
6	11.7	9.9	9.3	10.7	9.5	9.2
7	10.8	10.9	10.9	10.9	10.8	10.7
8	13.1	10.4	10.5	13.1	10.9	10.4

than the AHO direction and the dual H..K..M direction less than GT direction. As for the accuracy, both methods are comparable to the primal H..K..M and NT directions. Overall, the performance of the SGN method is somewhat disappointing. In particular, the method does not require fewer iterations than the related primal or dual H..K..M directions in general, even though it is more expensive to compute.

Note, however, that we used the default setting for all parameters in the SDPT3 algorithm IPF; it is reasonable to expect that the iteration count of the algorithms based on the SGN and dual H..K..M directions can be improved by implementing a different line search strategy. Also, the test problems used here are of moderate size. These computational results are therefore of a preliminary nature.

7. Conclusions. We have presented a primal-dual SGN direction for semidefinite optimization which allows polynomial worst-case iteration complexity analysis. This analysis was inspired by the Gauss–Newton direction of Kruk et al. [6], but the new direction seems much more amenable to complexity analysis, due to the use of scaling and a local norm in the definition of the least squares problem. In particular, the usual $O(\sqrt{n})$ iteration complexity was derived in this paper for the standard small update (short step) primal-dual path following algorithm. The complexity for methods using larger updates remains a topic for future research.

The new direction is closely related to the primal and dual H..K..M directions—it uses the ΔX part of the dual H..K..M direction and the ΔS part of the primal H..K..M direction. As a by-product, we have shown how the dual H..K..M direction can be computed at a cost of at most

$$2mn^3 + \frac{1}{2}m^2n^2 + \frac{1}{3}m^3 + O\left(n^3 + mn^2 + m^2n\right)$$

flops, and the SGN direction can subsequently be computed at a total cost of at most

$$2mn^3 + m^2n^2 + \frac{2}{3}m^3 + O\left(n^3 + mn^2 + m^2n\right)$$

flops.

A preliminary numerical evaluation of the performance of the SGN search direction is somewhat disappointing. The implementation was done using the infeasible path following algorithm in the Matlab code SDPT3. Since we used the default parameter settings for the step lengths and barrier parameter updates in SDTP3, we hope that these results can be improved by finding more suitable (dynamic) parameter settings for the new direction. This is a subject for future research.

Appendix. Computation of the SGN direction. In this appendix, we consider how to compute the SGN direction by first computing the dual H..K..M direction. To this end, we rewrite the linear system (34) (which yields the dual H..K..M direction) by using the Cholesky decompositions $X = L_X^T L_X$ and $S = L_S^T L_S$ as follows:

$$\begin{aligned} L_X L_S^T + L_X(\Delta S + W)L_S^{-1} + L_X^{-T}\Delta X L_S^T &= \mu L_X^{-T} L_S^{-1}, \\ \mathrm{Tr}(A_i \Delta X) &= 0, \quad i = 1, \ldots, m, \\ \sum_{i=1}^{m} \Delta y_i A_i + \Delta S &= 0, \\ \Delta X = \Delta X^T, \quad W + W^T &= 0. \end{aligned}$$

We wish to solve problem SGN1 which is equivalent to solving

$$\min_{\Delta X} \left\| L_X L_S^T + L_X^{-T}\Delta X L_S^T - \mu L_X^{-T} L_S^{-1} \right\|^2 \tag{35}$$

subject to

$$\mathrm{Tr}(A_i \Delta X) = 0, \quad i = 1, \ldots, m, \quad \Delta X = \Delta X^T.$$

We now perform a singular value decomposition of $L_X L_S^T$ or an eigenvalue decomposition of $L_X S L_X^T$ to obtain

$$Q^T L_X S L_X^T Q = \Lambda,$$

where Λ is a positive definite diagonal matrix and Q an orthonormal matrix. By defining

$$\Delta\bar{X} = Q^T L_X^{-T}\Delta X L_X^{-1} Q, \quad \bar{A}_i = Q^T L_X A_i L_X^T Q, \quad i = 1, \ldots, m,$$

we can rewrite problem (35) as

$$\begin{cases} \min_{\Delta\bar{X}} \left\| \Lambda^{1/2} + \Delta\bar{X}\Lambda^{1/2} - \mu\Lambda^{-1/2} \right\|^2, \\ \mathrm{Tr}\left(\bar{A}_i \Delta\bar{X}\right) = 0, \quad i = 1, \ldots, m, \quad \Delta\bar{X} = \Delta\bar{X}^T, \end{cases} \tag{36}$$

which is the same as

$$\begin{cases} \min_{\Delta\bar{X}} \frac{1}{2}\left\| \Lambda^{1/2} + \Delta\bar{X}\Lambda^{1/2} - \mu\Lambda^{-1/2} \right\|^2 + \frac{1}{2}\left\| \Lambda^{1/2} + \Lambda^{1/2}\Delta\bar{X} - \mu\Lambda^{-1/2} \right\|^2, \\ \mathrm{Tr}\left(\bar{A}_i \Delta\bar{X}\right) = 0, \quad i = 1, \ldots, m, \quad \Delta\bar{X} = \Delta\bar{X}^T. \end{cases} \tag{37}$$

since $\|A\| = \left\|A^T\right\|$.

In what follows, we will use this notation:

- $\mathbf{svec}\,(X) := \left(X_{11},\ \sqrt{2}X_{12},\ \ldots\ \sqrt{2}X_{1n},\ X_{22},\ \sqrt{2}X_{23},\ \ldots,\ X_{nn}\right)^T \quad \forall\ X = X^T$;
- The *symmetric Kronecker product* $G \otimes_s K$ of $G, K \in \mathbb{R}^{n\times n}$ is implicitly defined by

$$(G \otimes_s K)\,\mathbf{svec}\,(H) := \frac{1}{2}\mathbf{svec}\left(KHG^T + GHK^T\right) \quad \forall\ H = H^T.$$

Now let

$$G^T = (\mathbf{svec}\left(\bar{A}_1\right), \ldots, \mathbf{svec}\left(\bar{A}_m\right)), \quad d_x = \mathbf{svec}\left(\Delta\bar{X}\right). \tag{38}$$

The KKT system of (37) takes the form

$$\left\{\begin{array}{lcl} \mathcal{E}d_x + G^T v &=& -\mathbf{svec}\,(\Lambda - \mu I_n)\,, \\ Gd_x &=& 0, \end{array}\right. \tag{39}$$

where $\mathcal{E} = \Lambda \otimes_s I_n$ and v is a variable vector in the suitable space. Premultiplying the first equation in (39) by $G\mathcal{E}^{-1}$, we obtain a linear system in $\mathbb{R}^m$ such that

$$G\mathcal{E}^{-1}G^T v = -G\mathcal{E}^{-1}\mathbf{svec}\,(\Lambda - \mu I_n)\,. \tag{40}$$

Note that $\mathcal{E} = \Lambda \otimes_s I_n$ is a diagonal matrix (see, e.g., the appendix in [12]).

To compute the dual H..K..M direction, we therefore need only solve the system (40) first and then compute $\Delta X, \Delta S$ subsequently. In particular, Δy of the dual H..K..M direction is immediately available from the solution of (40).

PROPOSITION 7.1. *Suppose that $\Delta X, \Delta S$ are solutions of the dual H..K..M. direction and that*

$$\Delta S = -\sum_{i=1}^{m} \Delta y_i A_i.$$

Then $\Delta y = -v$ where v is the solution of the problem (40).

The proof of this proposition is straightforward and therefore omitted.

We can summarize the sequence of steps for the computation of the dual H..K..M direction as follows:

1. Compute G by computing $\bar{A}_i$ for $i = 1, \ldots, m$. Since all $\bar{A}_i$ are symmetric, the computation of all $\bar{A}_i$ requires at most $2mn^3 + O\left(n^3\right)$ flops (see Lemma A.10 in [8]);
2. compute $\mathcal{E}^{-1}G^T$ at a cost of $O(mn^2)$ flops;
3. form the Schur matrix $G\mathcal{E}^{-1}G^T$ ($\frac{1}{2}m^2n^2$ flops);
4. solve the linear system (40) ($\frac{1}{3}m^3$ flops).

Hence the total computation complexity for the dual H..K..M direction is upper bounded by $2mn^3 + \frac{1}{2}m^2n^2 + \frac{1}{3}m^3 + O\left(n^3 + n^2m + nm^2\right)$ flops.

Now recall that the SGN direction uses the ΔS part of the primal H..K..M direction. It is easy to show that the Schur matrix for the primal H..K..M direction has entries $\mathrm{Tr}\left(\bar{A}_i\Lambda^{-1}\bar{A}_j\right)$ $(i, j = 1, \ldots, m)$. We can therefore utilize the fact that G had already been computed when we formed the primal H..K..M Schur matrix. In other words, we only have to perform the analogous steps to steps 3 and 4 above. Hence, the total computational complexity for the SGN direction is $2mn^3 + m^2n^2 + \frac{2}{3}m^3 + O\left(n^3 + n^2m + nm^2\right)$ flops.

Acknowledgments. The authors would like to thank Prof. M. Todd, Prof. M. Kojima, Prof. H. Wolkowicz, and Dr. K. C. Toh for their generous help in the preparation of this work.

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] E. de Klerk, *Interior Point Methods for Semidefinite Programming*, Ph.D thesis, Delft University of Technology, Delft, The Netherlands, 1997.

[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] J. Jiang, *A long step primal–dual path following method for semidefinite programming*, Oper. Res. Lett., 23 (1998), pp. 53–62.

[5] 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.

[6] S. Kruk, M. Muramatsu, F. Rendl, R.J. Vanderbei, and H. Wolkowicz, *The Gauss–Newton Direction in Semidefinite Programming*, Research report CORR 98-16, University of Waterloo, Dept. Combinatorics and Optimization, Waterloo, Canada, 1998.

[7] R.D.C. Monteiro, *Primal-dual path-following algorithms for semidefinite programming*, SIAM J. Optim., 7 (1997), pp. 663–678.

[8] 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*, Optim. Methods Softw., 11 (1999), pp. 91–140.

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

[10] C. Roos, T. Terlaky, and J.-Ph. Vial, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley & Sons, New York, 1997.

[11] M.J. Todd, *A study of search directions in primal-dual interior-point methods for semidefinite programming*, Optim. Methods Softw., 11 (1999), pp. 1–46.

[12] 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.

[13] K.C. Toh, M.J. Todd, and R.H. Tütüncü, *SDPT3 — a Matlab software package for semidefinite programming*, Optim. Methods Softw., 11 (1999), pp. 545–581.

[14] 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. 11, No. 4, pp. 889–917

NONMONOTONE TRUST-REGION METHODS FOR BOUND-CONSTRAINED SEMISMOOTH EQUATIONS WITH APPLICATIONS TO NONLINEAR MIXED COMPLEMENTARITY PROBLEMS*

MICHAEL ULBRICH†

Abstract. We develop and analyze a class of trust-region methods for bound-constrained semismooth systems of equations. The algorithm is based on a simply constrained differentiable minimization reformulation. Our global convergence results are developed in a very general setting that allows for nonmonotonicity of the function values at subsequent iterates. We propose a way of computing trial steps by a semismooth Newton-like method that is augmented by a projection onto the feasible set. Under a Dennis–Moré-type condition we prove that close to a regular solution the trust-region algorithm turns into this projected Newton method, which is shown to converge locally q-superlinearly or quadratically, respectively, depending on the quality of the approximate subdifferentials used.

As an important application we discuss how the developed algorithm can be used to solve nonlinear mixed complementarity problems (MCPs). Hereby, the MCP is converted into a bound-constrained semismooth equation by means of an NCP-function. The efficiency of our algorithm is documented by numerical results for a subset of the MCPLIB problem collection.

Key words. semismooth equation, nonmonotone trust region method, nonlinear mixed complementarity problem, nonsmooth Newton method, global convergence, superlinear and quadratic convergence

AMS subject classifications. 90C30, 90C33, 49J40, 65H10, 65K05, 49M37

PII. S1052623499356344

1. Introduction.

In this paper we propose and analyze a class of trust-region methods for the solution of a simply constrained system of nonlinear nonsmooth equations

$$H(x) = 0, \quad x \in X. \tag{1.1}$$

Hereby, the function $H : \mathbb{R}^n \supset U \to \mathbb{R}^n$ is defined on the open set U containing the feasible set $X \stackrel{\text{def}}{=} [l, u] = \{x \in \mathbb{R}^n ;\ l_i \le x_i \le u_i, 1 \le i \le n\}$. The bounds $l_i \in \mathbb{R} \cup \{-\infty\}$ and $u_i \in \mathbb{R} \cup \{+\infty\}$ are assumed to satisfy $l_i < u_i$, $1 \le i \le n$. (Otherwise the variable $x_i = l_i = u_i$ could be eliminated.)

We require that (with the definition of semismoothness to follow)

(A1) the function H is semismooth or, stronger, p-order semismooth, $0 < p \le 1$;
(A2) each component function H_i of H is continuously differentiable on $U \setminus H_i^{-1}(0)$.

The locally q-superlinear/quadratic convergence of the algorithm to BD-regular (where "BD" stands for Bouligand differential) solutions of (1.1) will be achieved by a Newton-type method that is augmented by a projection onto X to maintain feasibility. Local convergence results for Newton's method without projection were established in [39, 40, 43]. Similar to [39], our local convergence results hold under a Dennis–Moré-type condition, thus allowing for inexactness in the computation of B-subdifferentials

*Received by the editors May 17, 1999; accepted for publication (in revised form) September 19, 2000; published electronically March 15, 2001.
http://www.siam.org/journals/siopt/11-4/35634.html
†Lehrstuhl für Angewandte Mathematik und Mathematische Statistik, Zentrum Mathematik, Technische Universität München, D-80290 München, Germany (mulbrich@ma.tum.de).

(where "B" stands for Bouligand) and in the solution of linear systems. We safeguard this locally convergent iteration by a nonmonotone trust-region globalization that is based on the minimization reformulation

$$\text{minimize } h(x) \quad \text{subject to} \quad x \in X, \tag{1.2}$$

where $h : U \to \mathbb{R}$, $h(x) \stackrel{\text{def}}{=} \|H(x)\|^2/2$. Here and throughout the paper, $\|\cdot\|$ denotes the Euclidean norm. Obviously, (1.1) and (1.2) are equivalent if (1.1) possesses a solution. As will be shown in Lemma 4.2, our assumptions on H imply that h is continuously differentiable on U. This enables us to invoke smooth proof techniques for the trust-region algorithm. We stress that our analysis is not based directly on (A2) but on the—in connection with (A1)—weaker assumption

(A2′) The function $h : U \to \mathbb{R}$, $h(x) = \|H(x)\|^2/2$ is continuously differentiable.

We believe, however, that (A2) is more concrete and easier to verify than (A2′) and thus decide to choose (A1) and (A2) as our working assumptions. An assumption of the form (A2′) was also used in [26]. Furthermore, the investigation of pseudosmooth equations in the recent work [42] are based on an assumption similar to (A2).

The need for efficient algorithms for the solution of (1.1) comes from the fact that very general classes of problems can be converted to this form. Of particular importance are semismooth reformulations of nonlinear mixed complementarity problems (MCPs). For the definition of the general MCP, which is a bound-constrained variational inequality problem, we refer to [12, 16] and section 8. To avoid notational overhead, we focus here on the following class of MCPs.

Find $x \in \mathbb{R}^n$ such that

$$\begin{aligned} &x_i \geq 0, \quad F_i(x) \geq 0, \quad x_i F_i(x) = 0, \quad && i = 1, \dots, m, \\ &F_i(x) = 0, && i = m+1, \dots, n, \end{aligned} \tag{1.3}$$

where the function $F : U \to \mathbb{R}^n$ is defined on the open set U containing $X = [0, \infty)^m \times \mathbb{R}^{n-m}$, and $0 \leq m \leq n$. In the case $m = n$ the MCP (1.3) reduces to a nonlinear complementarity problem, whereas for $m = 0$ we obtain a system of nonlinear equations. MCPs arise in a variety of areas, including computer sciences, economics, engineering, operations research, and mathematics. For a comprehensive discussion of applications, see [16]. Further applications can be derived from the fact that the Karush–Kuhn–Tucker (KKT) conditions of mathematical programs and, more general, of variational inequality problems are MCPs.

In order to apply our algorithm, we will reformulate the MCP (1.3) equivalently in the form (1.1), where $l_i = 0$ and $u_i = +\infty$ for $1 \leq i \leq m$, $l_i = -\infty$ and $u_i = +\infty$ for $m < i \leq n$, and $H : U \to \mathbb{R}^n$ is defined by

$$\begin{aligned} &H_i(x) \stackrel{\text{def}}{=} \phi(x_i, F_i(x)), && i = 1, \dots, m, \\ &H_i(x) \stackrel{\text{def}}{=} F_i(x), && i = m+1, \dots, n. \end{aligned} \tag{1.4}$$

Hereby, the function $\phi : \mathbb{R}^2 \to \mathbb{R}$ is an NCP-function, i.e., it satisfies

$$\phi(a, b) = 0 \qquad \text{if and only if} \qquad a \geq 0, \quad b \geq 0, \quad ab = 0. \tag{1.5}$$

Probably the most popular NCP-function is the Fischer–Burmeister function [17]

$$\phi_{\text{FB}}(a, b) \stackrel{\text{def}}{=} a + b - \sqrt{a^2 + b^2},$$

which is 1-order semismooth. We will show in section 7 that under mild assumptions on F and ϕ the function H defined in (1.4) satisfies (A1) and (A2) with $p = 1$. In particular, these assumptions hold if F is Lipschitz continuously differentiable and if $\phi = \phi_{\text{FB}}$ is chosen. For details on the variety of available NCP-functions, the reader is referred to [5, 46]. See also sections 7 and 8.

Most of the available literature on algorithms for problems of the form (1.1) focuses on special cases like nonsmooth reformulations of NCPs. Line search methods for problems (1.1) arising from reformulated NCPs and KKT-systems of variational inequality problems (VIPs) were analyzed in, e.g., [10, 15, 27]. Since x is a zero of the function H defined in (1.4) if and only if x solves the MCP (1.3), it is possible to omit the box-constraints in (1.1). Below, we discuss this approach and mention some of its potential drawbacks. Line search methods for these unconstrained reformulations are investigated by several authors; see [10, 13, 25, 50, 51].

For unconstrained semismooth equations arising from reformulations of NCPs, trust-region algorithms were analyzed in [24, 28]. A trust-region method for a box-constrained reformulation without NCP-function was investigated in [35] under a strict complementarity condition. This assumption is not needed for our analysis. Moreover, we stress that the results in [24, 28, 35] are established for monotone trust-region methods only. Other approaches for the solution of the NCP, MCP, or VIP can be found in [1, 3, 11, 37, 38]. For a survey, see [2, 14]. Algorithms for more general classes of nonsmooth equations are investigated in [20, 22, 39, 40, 41, 43, 50, 51].

Among the methods cited above, the trust-region algorithms in [24, 28] are probably the ones closest related to the class of methods proposed in this work. However, there are several important differences. In particular, we deal with a more general class of nonsmooth equations. Moreover, we allow for box-constraints and our algorithm generates feasible iterates with respect to these constraints. Numerical studies [5, 21, 48, 28] have shown that the performance of optimization methods for the solution of minimization problems can be significantly improved by using nonmonotone line search- or trust-region techniques. Especially for problems with least-squares objective functions like (1.2), nonmonotonicity helps to prevent convergence to local–nonglobal solutions of (1.2). In this paper we introduce a new nonmonotone trust-region technique and develop a global convergence theory that covers essentially all results that are known for monotone trust-region algorithms. These results appear to be new also for the special case of smooth problems (1.1). For other approaches to nonmonotone line search- and trust-region techniques we refer to [21, 48] and the references therein. Concerning literature on monotone trust-region methods for optimization problems with simple (or, more generally, convex) constraints we refer to [4, 7, 8, 19, 31, 47, 49].

Especially for problems (1.1) obtained from reformulating the MCP on the basis of (1.4) one might ask why we keep the box constraint although $H(x) \neq 0$ holds by definition for all $x \notin X$. More generally, if $\bar{x}$ is a solution to (1.1), then, obviously, it also solves the unconstrained semismooth equation

$$H(x) = 0. \tag{1.6}$$

The corresponding unconstrained counterpart to (1.2) is

$$\text{minimize} \quad h(x). \tag{1.7}$$

We stress, however, that (1.6) and (1.7) contain the implicit constraint $x \in U$, which can be quite unstructured and complicated. There are good reasons to prefer the

constrained formulations (1.1), (1.2) to the unconstrained ones (1.6), (1.7). For instance, there might exist solutions to (1.6) that do not solve (1.1) since they are not feasible with respect to X. This cannot occur if $H(x) \neq 0$ for all $x \notin X$; cf. (1.4). Sometimes H is known to have nice properties on X (like, e.g., positive definiteness of the Jacobian $H'(x)$), but not outside of X. When working with (1.2) instead of (1.7) this can help to reduce the risk of finding a local but nonglobal solution of the minimization problem only. Furthermore, in many applications the domain U of H is only implicitly available in the sense that H is numerically implemented as an oracle that returns the value $H(x)$ if $x \in U$ and an error message, otherwise. Therefore, algorithms for the solution of problem (1.7) can run into trouble by accidentally coming too close to the boundary of U. On the other hand, if a feasible point algorithm is applied to the reformulation (1.2), then it will never interfere with the boundary of U, since $X \subset U$ and U is open.

From the above discussion we conclude that algorithms that are based on the simply constrained problems (1.1) and (1.2) can, in fact, be more robust and efficient than those derived from the easier looking problems (1.6) and (1.7). This is confirmed by the numerical experiments in section 8.

The rest of this paper is organized in seven sections. In section 2 we collect important results of nonsmooth analysis needed for our investigations. The trust-region algorithm is developed in section 3. Hereby, we begin in section 3.1 with the underlying Newton-type method, which we augment by a projection to obtain feasible iterates. Then, in section 3.2 this iteration is embedded into a globally convergent nonmonotone trust-region method. We proceed in section 4 by proving the global convergence of this algorithm. In section 5, the locally q-superlinear convergence and convergence of q-order $1+p$, respectively, are established. A concrete implementation of the decrease condition by means of a Cauchy step is discussed in section 6. Section 7 is devoted to the application of the developed method to the nonlinear complementarity problem. Numerical results for a subset of the MCPLIB test set [12] are reported in section 8.

Notations. The ℓ_2-norm on $\mathbb{R}^n$ is denoted by $\|\cdot\|$. Given a set $M \subset \mathbb{R}^n$ and a point $x \in \mathbb{R}^n$, we set $M - x \stackrel{\text{def}}{=} \{y;\ x+y \in M\}$. $S^0(U,\mathbb{R}^n)$ denotes the set of all semismooth functions $f : \mathbb{R}^n \supset U \to \mathbb{R}^n$. $S^p(U,\mathbb{R}^n)$, $0 < p \leq 1$, is the set of all p-order semismooth functions. We write $f'(x,\cdot) : \mathbb{R}^n \to \mathbb{R}^m$ for the directional derivative, $f'(x) \in \mathbb{R}^{m\times n}$ for the Jacobian, $\partial_B f(x) \subset \mathbb{R}^{m\times n}$ for the B-subdifferential, and $\partial f(x) \in \mathbb{R}^{m\times n}$ for Clarke's generalized Jacobian of the function $f : \mathbb{R}^n \supset U \to \mathbb{R}^m$ at the point $x \in U$ (in case the respective objects exist). $\nabla f(x) = f'(x)^T$ denotes the gradient of the differentiable, real-valued function f at x.

2. Some notions of nonsmooth analysis. For convenience, we collect here all facts about nonsmooth analysis that are required for our investigations. Readers familiar with these concepts might want to skip this section.

Throughout, let $f : U \to \mathbb{R}^m$ be locally Lipschitz continuous on the nonempty open set $U \subset \mathbb{R}^n$. By D_f we denote the set of all $x \in U$ where f admits a (Fréchet) derivative $f'(x) \in \mathbb{R}^{m\times n}$. According to Rademacher's theorem [52], $U \setminus D_f$ has Lebesgue measure zero. Hence, the following constructions make sense.

DEFINITION 2.1 (see [6, 39, 43]). *The set*

$$\partial_B f(x) \stackrel{\text{def}}{=} \left\{V \in \mathbb{R}^{m\times n};\ \exists (x_k) \subset D_f :\ x_k \to x,\ f'(x_k) \to V\right\}$$

is called B-subdifferential of f at $x \in U$. Moreover, Clarke's generalized Jacobian of f at x is the set $\partial f(x) \stackrel{\text{def}}{=} \operatorname{conv}(\partial_B f(x))$. ∎

We collect some properties of $\partial_B f$ and ∂f.

PROPOSITION 2.2 (see [6, Prop. 2.6.2]). *For all $x \in U$ the following holds:*

(a) *$\partial_B f(x)$ is nonempty and compact.*

(b) *$\partial f(x)$ is nonempty, compact, and convex.*

(c) *The set-valued mappings $\partial_B f$ and ∂f, respectively, are locally bounded and upper semicontinuous.*

Next, we recall the directional derivative and define semismoothness.

DEFINITION 2.3 (see [34, 39, 43]).

(a) *f is directionally differentiable at $x \in U$ if the directional derivative*

$$f'(x,s) \stackrel{\text{def}}{=} \lim_{\tau \to 0^+} \frac{f(x+\tau s) - f(x)}{\tau}$$

exists for all $s \in \mathbb{R}^n$.

(b) *f is semismooth at $x \in U$ if it is locally Lipschitzian at x and the following limit exists for all $s \in \mathbb{R}^n$:*

$$\lim_{\substack{V \in \partial f(x+\tau d) \\ d \to s,\ \tau \to 0^+}} Vd.$$

(c) *By $S^0(U, \mathbb{R}^m)$ we denote the set of all functions $f : U \to \mathbb{R}^m$ that are semismooth on U.* □

Note that $f'(x, \cdot)$ is positive homogeneous. The following proposition gives an alternative definition of semismoothness.

PROPOSITION 2.4 (see [43, Thm. 2.3]). *For $x \in U$ the following statements are equivalent:*

(a) *f is semismooth at x,*

(b) *$f'(x, \cdot)$ exists, and*

$$\sup_{V \in \partial f(x+s)} \|Vs - f'(x,s)\| = o(\|s\|) \quad \text{as } s \to 0.$$

COROLLARY 2.5. *If f is semismooth at x, then for all $s \in \mathbb{R}^n$*

$$f'(x,s) = \lim_{\substack{V \in \partial f(x+\tau s) \\ \tau \to 0^+}} Vs.$$

Based on Proposition 2.4 (b), a semismooth relaxation of Hölder-continuous differentiability can be established.

DEFINITION 2.6 (see [43]). *Let $0 < p \leq 1$. The function f is called p-order semismooth at $x \in U$ if f is locally Lipschitz at x, $f'(x, \cdot)$ exists, and*

$$\sup_{V \in \partial f(x+s)} \|Vs - f'(x,s)\| = O(\|s\|^{1+p}) \quad \text{as } s \to 0.$$

By $S^p(U, \mathbb{R}^m)$ we denote the set of all functions $f : U \to \mathbb{R}^m$ that are p-order semismooth on U. □

PROPOSITION 2.7 (see [43]). *If f is semismooth at $x \in U$, then*

$$\|f(x+s) - f(x) - f'(x,s)\| = o(\|s\|) \quad \text{as } s \to 0.$$

If $0 < p \leq 1$ and f is p-order semismooth at $x \in U$, then

$$\|f(x+s) - f(x) - f'(x,s)\| = O(\|s\|^{1+p}) \quad \text{as } s \to 0.$$

PROPOSITION 2.8 (see [18, Lem. 18 and Thm. 21]). *Let $U_1 \subset \mathbb{R}^n$ and $U_2 \subset \mathbb{R}^l$ be open sets and $f_1 : U_1 \to U_2$, $f_2 : U_2 \to \mathbb{R}^m$ be locally Lipschitz mappings. Then, if f_1 is (p-order, $0 < p \leq 1$) semismooth at $x \in U_1$ and f_2 is (p-order) semismooth at $f_1(x)$, the composite map $f \stackrel{\text{def}}{=} f_2 \circ f_1 : U_1 \to \mathbb{R}^m$ is (p-order) semismooth at x. Moreover,*

$$f'(x, \cdot) = f_2'(f_1(x), f_1'(x, \cdot)).$$

The following is obvious.

PROPOSITION 2.9. *If f is continuously differentiable in a neighborhood of $x \in U$ (with p-Hölder continuous derivative, $0 < p \leq 1$), then f is (p-order) semismooth at x and $\partial f(x) = \partial_B f(x) = \{f'(x)\}$.*

The following regularity property is essential for fast local convergence of Newton-like methods.

DEFINITION 2.10 (see [39]). *The point $x \in U$ is called BD-regular for f if all elements in $\partial_B f(x)$ are nonsingular.* □

PROPOSITION 2.11 (see [39, Prop. 3]). *Let $x \in U$ be a BD-regular for f. Then there exist $\varepsilon > 0$ and $C > 0$ such that all $V \in \partial_B f(y)$, $\|y - x\| \leq \varepsilon$ are nonsingular with $\|V^{-1}\| \leq C$. If, in addition, f is semismooth at x, then there exist $\delta > 0$ and $\zeta > 0$ such that*

$$\|f(y) - f(x)\| \geq \zeta \|y - x\|$$

for all $y \in \mathbb{R}^n$, $\|y - x\| \leq \delta$.

3. Development of the algorithm. As motivated above, our algorithm for the solution of (1.1) will be based on the reformulation

$$\text{minimize } h(x) \quad \text{subject to} \quad l \leq x \leq u.$$

Basically, the concept of trust-region methods is to make Newton's method globally convergent while maintaining its excellent local convergence behavior. Therefore, we begin the description of our algorithm with its core, the underlying Newton-like iteration.

3.1. A Newton-like method with projection. In order to generate feasible iterates for (1.1) we introduce a Newton-like method that is augmented by the projection onto $X = [l, u]$. In what follows, given a nonempty closed convex set $C \subset \mathbb{R}^n$, the mapping $P_C : \mathbb{R}^n \to C$

$$P_C(x) \stackrel{\text{def}}{=} \operatorname*{argmin}_{y \in C} \|y - x\|$$

denotes the projection onto C. Since $X = [l, u]$ is a box, P_X can be easily computed:

$$P_X(x) = \max\{l, \min\{x, u\}\} \quad \text{(componentwise)}.$$

We now can formulate the algorithm.

ALGORITHM 3.1 (Newton-like iteration with projection).

1. Choose x_0 and set $k := 0$.
2. If $H(x_k) = 0$, then STOP.
3. Choose a nonsingular matrix $M_k \in \mathbb{R}^{n \times n}$ and compute the Newton-like step s_k^{N} by solving

$$M_k s_k^{\mathrm{N}} = -H(x_k).$$

4. Compute the projection of s_k^{N} onto $X - x_k$:

$$s_k^{\mathrm{PN}} := P_X(x_k + s_k^{\mathrm{N}}) - x_k.$$

5. Set $x_{k+1} := x_k + s_k^{\mathrm{PN}}$, $k := k + 1$, and go to Step 2.

Since $x_0 \in X$ and M_0 is invertible, the steps s_0^{N} and s_0^{PN} are well defined. Moreover, $x_1 = P_X(x_0 + s_0^{\mathrm{N}}) \in X$. By iterating this argument we obtain the well-definedness of Algorithm 3.1 and the feasibility of the iterates x_k with respect to X.

Without the augmentation by a projection the local convergence properties of Newton-like iterations for semismooth equations were investigated in, e.g., [39, 40, 43]. Newton methods for nonsmooth equations are also discussed in [29, 30, 36, 37, 45]. The additional projection does not affect the convergence speed since it is Lipschitzian of rank 1 [23, p. 118]. We will give a full proof of the local convergence result for Algorithm 3.1 since it is instructive to see how semismoothness, BD-regularity, and other concepts come into play, and, more importantly, because we will need all the estimates established in this proof for the more involved global-to-local analysis in Theorem 5.1.

THEOREM 3.2. *Assume that $H : U \to \mathbb{R}^n$ is locally Lipschitz continuous. Let $\bar{x} \in X$ be a BD-regular zero of H at which H is semismooth. Let $\|x_0 - \bar{x}\|$ and $\delta > 0$ be sufficiently small. Assume that Algorithm* 3.1 *generates infinitely many iterates, and that for all k holds*

$$\mu_k \stackrel{\mathrm{def}}{=} \min_{V \in \partial_B H(x_k)} \|(M_k - V)s_k^{\mathrm{N}}\| \le \delta \|s_k^{\mathrm{N}}\|. \tag{3.1}$$

Then (x_k) converges to $\bar{x}$.

If in addition

$$\lim_{k\to\infty} \frac{\mu_k}{\|s_k^{\mathrm{N}}\|} = 0, \tag{3.2}$$

then the sequence (x_k) converges q-superlinearly to $\bar{x}$.

If H is p-order semismooth at $\bar{x}$, $0 < p \le 1$, and

$$\limsup_{k\to\infty} \frac{\mu_k}{\|s_k^{\mathrm{N}}\|^{1+p}} < \infty, \tag{3.3}$$

then (x_k) converges with q-order $1 + p$ to $\bar{x}$.

Proof. By the local Lipschitz continuity, the BD-regularity of $\bar{x}$ and Proposition 2.11, there exist $\varepsilon > 0$, $L > 0$, and $C > 0$ such that H is Lipschitz continuous on $B_\varepsilon(\bar{x}) \stackrel{\mathrm{def}}{=} \{x \in \mathbb{R}^n ;\ \|x - \bar{x}\| \le \varepsilon\}$ of rank L, and

$$\|V^{-1}\| \le C \quad \forall\, V \in \partial_B H(x) \quad \forall\, x \in B_\varepsilon(\bar{x}). \tag{3.4}$$

Throughout the proof let $x_k \in B_\varepsilon(\bar{x})$ be arbitrary and let $V_k \in \partial_B H(x_k)$ be such that $\mu_k = \|(M_k - V_k)s_k^{\mathrm{N}}\|$. We define

$$d_k \stackrel{\mathrm{def}}{=} x_k - \bar{x}, \quad e_k \stackrel{\mathrm{def}}{=} x_k + s_k^{\mathrm{N}} - \bar{x}.$$

If $\delta \le 1/(2C)$, then

$$\begin{aligned} \|s_k^{\mathrm{N}}\| &\le \|{V_k}^{-1}\| \left(\|M_k s_k^{\mathrm{N}}\| + \|(V_k - M_k)s_k^{\mathrm{N}}\|\right) \le C\|H(x_k)\| + C\mu_k \\ &\le C\|H(x_k)\| + C\delta\|s_k^{\mathrm{N}}\| \le C\|H(x_k)\| + \frac{1}{2}\|s_k^{\mathrm{N}}\|. \end{aligned}$$

In particular,

$$\|s_k^{\mathrm{N}}\| \le 2C\|H(x_k)\| \le 2CL\|d_k\|. \tag{3.5}$$

Further,

$$\begin{aligned} V_k e_k &= V_k s_k^{\mathrm{N}} + V_k d_k = M_k s_k^{\mathrm{N}} + (V_k - M_k) s_k^{\mathrm{N}} + V_k d_k \\ &= -H(x_k) + V_k d_k + (V_k - M_k) s_k^{\mathrm{N}} \\ &= (V_k d_k - H'(\bar{x}, d_k)) + (-H(x_k) + H(\bar{x}) + H'(\bar{x}, d_k)) \\ &\quad + (V_k - M_k) s_k^{\mathrm{N}}. \end{aligned} \tag{3.6}$$

If δ and ε are sufficiently small, we obtain from (3.4), Proposition 2.4(b), Proposition 2.7, (3.1), and (3.5) that

$$\|e_k\| \le \frac{1}{2}\|d_k\|.$$

Since $\bar{x} \in X$ and P_X is Lipschitz continuous of rank 1, we thus have

$$\begin{aligned} \|d_{k+1}\| &= \|P_X(x_k + s_k^{\mathrm{N}}) - \bar{x}\| = \|P_X(x_k + s_k^{\mathrm{N}}) - P_X(\bar{x})\| \\ &\le \|x_k + s_k^{\mathrm{N}} - \bar{x}\| = \|e_k\| \le \frac{1}{2}\|d_k\|. \end{aligned} \tag{3.7}$$

Therefore, it follows inductively that the sequence (x_k) converges to $\bar{x}$ if $\delta > 0$ and $\|x_0 - \bar{x}\|$ are sufficiently small.

If (3.2) holds, we see from (3.5), Proposition 2.4(b), and Proposition 2.7, that the right-hand side of (3.6), and thus, by (3.4), also e_k is of the order $o(\|d_k\|)$. The q-superlinear convergence now follows from (3.7).

If H is p-order semismooth at $\bar{x}$ and if (3.3) holds, then the right-hand side of (3.6) is obviously of the order $O(\|d_k\|^{1+p})$. The proof is completed as before. ∎

3.2. The trust-region algorithm. We now wrap Algorithm 3.1 into a globally convergent trust-region method for problem (1.2). For the time being, let us take the continuous differentiability of the merit function h for granted. We return to this issue in Lemma 4.2. This amounts to us building the quadratic model

$$q_k(s) = g_k^T s + \frac{1}{2}\|M_k s\|^2$$

around the current iterate x_k, where $g_k \overset{\text{def}}{=} \nabla h(x_k)$. q_k is an at least first-order accurate approximation of $h(x_k + s) - h(x_k)$. The matrices $M_k \in \mathbb{R}^{n\times n}$ are the same as in Theorem 3.2. We stress, however, that the proposed trust-region method is globally convergent for much more general choices of M_k. Note hereby that, as we will show in Lemma 4.2 below, the computation of the gradient $g_k = \nabla h(x_k) = V_k^T H(x_k)$ requires us only to compute the action of V_k^T onto the vector $H(x_k)$, where $V_k \in \partial_B H(x_k)$ is arbitrary. This is usually much cheaper than the computation of the full matrix V_k. In the case of reformulations of MCPs with the Fischer–Burmeister function (or other suitable NCP-functions), it can be seen from the structure of $\partial_B H(x_k)$ (see [10]) that only $F'(x_k)^T v_k$, where the vector v_k is easily obtained from $F(x_k)$, has to be computed for the gradient. This can be done efficiently by, e.g., the reverse mode of automatic differentiation. Therefore, we will carry out our global analysis for general matrices M_k to allow the use of quasi-Newton or other approximations.

In each iteration of the trust-region algorithm, a trial step s_k is computed as approximate solution of the *trust-region subproblem*

$$\text{minimize } q_k(s) \quad \text{subject to} \quad l \le x_k + s \le u,\ \|s\|_\infty \le \Delta_k. \tag{3.8}$$

This is a convex box-constrained quadratic program (QP) with feasible set

$$X_k \stackrel{\text{def}}{=} [l - x_k, u - x_k] \cap [-\Delta_k, \Delta_k]^n.$$

We will assume that the trial steps meet the following two requirements:

feasibility condition

$$l \le x_k + s_k \le u \quad \text{and} \quad \|s_k\|_\infty \le \beta_1 \Delta_k, \quad \text{and} \tag{3.9}$$

reduction condition

$$\text{pred}_k(s_k) \stackrel{\text{def}}{=} -q_k(s_k) \ge \beta_2 \chi(x_k) \min\{1, \Delta_k, \chi(x_k)\} \tag{3.10}$$

with constants $\beta_1 \ge 1$ and $\beta_2 > 0$ independent of k. Hereby, χ is a suitably chosen *criticality measure*:

$$\begin{gathered} \chi : X \to \mathbb{R}_+ \text{ is continuous,} \\ \chi(x) = 0 \text{ if and only if } x \text{ is a KKT-point of problem (1.2).} \end{gathered} \tag{3.11}$$

A well-known criticality measure is the norm of the projected gradient

$$\chi(x) = \|x - P_X(x - \nabla h(x))\|.$$

We recall that $P_X(x) = \max\{l, \min\{x, u\}\}$ denotes the projection onto X. Usually, the update of the trust-region radius Δ_k is controlled by the ratio of *actual reduction*

$$\text{ared}_k(s) \stackrel{\text{def}}{=} h(x_k) - h(x_k + s)$$

and *predicted reduction* $\text{pred}_k \stackrel{\text{def}}{=} -q_k(s)$.

It has been observed [5, 21, 28, 48] that the performance of nonlinear programming algorithms can be significantly improved by using nonmonotone line search- or trust-region techniques. Hereby, in contrast to the traditional approach, the monotonicity $h(x_{k+1}) \le h(x_k)$ of the function values is not enforced in every iteration. We introduce a new nonmonotone trust-region technique for which all global convergence results for monotone methods remain valid. Hereby, the decrease requirement is significantly relaxed. Before we describe this approach and the corresponding reduction ratio $\rho_k(s)$ in detail, we first state the basic trust-region algorithm.

ALGORITHM 3.3 (trust-region algorithm).

1. Initialization: Choose $\eta_1 \in (0, 1)$, $\Delta_{\min} \ge 0$, and a criticality measure χ. Choose $x_0 \in X$, $\Delta_0 > \Delta_{\min}$, and a nonsingular matrix $M_0 \in \mathbb{R}^{n \times n}$. Choose an integer $m \ge 1$ and fix $\lambda \in (0, 1/m]$ for the computation of ρ_k. Set $k := 0$ and $i := -1$.
2. Compute $\chi_k := \chi(x_k)$. If $\chi_k = 0$, then STOP.
3. Compute a trial step s_k satisfying the conditions (3.9) and (3.10).
4. Compute the reduction ratio $\rho_k := \rho_k(s_k)$.
5. Compute the new trust-region radius Δ_{k+1} by invoking Algorithm 3.4.
6. If $\rho_k \le \eta_1$, then reject the step s_k, i.e., set $x_{k+1} := x_k$, $M_{k+1} := M_k$, increment k by 1, and go to Step 3.

7. Accept the step: Set $x_{k+1} := x_k + s_k$ and choose a nonsingular matrix $M_{k+1} \in \mathbb{R}^{n\times n}$. Set $j_{i+1} := k$, increment k and i by 1 and go to Step 2.

The increasing sequence $(j_i)_{i\geq 0}$ enumerates all indices of accepted steps. Moreover,

$$x_k = x_{j_i} \qquad \forall\, j_{i-1} < k \leq j_i \quad \forall\, i \geq 1. \tag{3.12}$$

Conversely, if $k \neq j_i$ for all i, then s_k was rejected. In the following we denote the set of all these "successful" indices j_i by $\mathcal{S}$:

$$\mathcal{S} \stackrel{\text{def}}{=} \{j_i\,;\ i \geq 0\} = \{k\,;\ \text{trial step } s_k \text{ is accepted}\}\,.$$

Sometimes, accepted steps will also be called successful. We will repeatedly use the fact that

$$\{x_k\,;\ k \geq 0\} = \{x_k\,;\ k \in \mathcal{S}\}\,.$$

The trust-region updates are implemented as usual. We deal with two different flavors of update rules simultaneously by introducing a nonnegative parameter $\Delta_{\min}$. We require that after successful steps $\Delta_{k+1} \geq \Delta_{\min}$. If $\Delta_{\min} = 0$ is chosen, this holds automatically. For $\Delta_{\min} > 0$, however, it is an additional feature that allows for special proof techniques.

ALGORITHM 3.4 (update of the trust-region radius).
Input: Δ_k, ρ_k. Output: Δ_{k+1}.
$\Delta_{\min} \geq 0$ and $\eta_1 \in (0, 1)$ are the constants defined in Step 1 of Algorithm 3.3.
Let $\eta_1 < \eta_2 < 1$, and $0 \leq \gamma_0 < \gamma_1 < 1 < \gamma_2$ be fixed.

1. If $\rho_k \leq \eta_1$, then choose
   $\Delta_{k+1} \in (\gamma_0\Delta_k, \gamma_1\Delta_k]$.
2. If $\rho_k \in (\eta_1, \eta_2)$, then choose
   $\Delta_{k+1} \in [\gamma_1\Delta_k, \max\{\Delta_{\min}, \Delta_k\}] \cap [\Delta_{\min}, \infty)$.
3. If $\rho_k \geq \eta_2$, then choose
   $\Delta_{k+1} \in (\Delta_k, \max\{\Delta_{\min}, \gamma_2\Delta_k\}] \cap [\Delta_{\min}, \infty)$.

We still have to describe how the reduction ratios $\rho_k(s)$ are defined. Here is a detailed description of Step 4:

4.1. Compute $m_k := \min\{i + 1, m\}$ and choose scalars

$$\lambda_{kr} \geq \lambda, \quad r = 0, \ldots, m_k - 1, \qquad \sum_{r=0}^{m_k-1} \lambda_{kr} = 1.$$

4.2. Compute the *relaxed actual reduction* $\text{rared}_k := \text{rared}_k(s_k)$, where

$$\text{rared}_k(s) \stackrel{\text{def}}{=} \max\left\{h(x_k),\ \sum_{r=0}^{m_k-1} \lambda_{kr} h(x_{j_{i-r}})\right\} - h(x_k + s). \tag{3.13}$$

4.3. Compute the reduction ratio $\rho_k := \rho_k(s_k)$ according to

$$\rho_k(s) \stackrel{\text{def}}{=} \frac{\text{rared}_k(s)}{\text{pred}_k(s)}. \qquad \square$$

Remark 3.5. At the very beginning of the iteration, Step 4 is encountered with $i = -1$. In this case the sum in (3.13) is empty and thus

$$\text{rared}_k(s) = \max\{h(x_k), 0\} - h(x_k + s) = h(x_k) - h(x_k + s) = \text{ared}_k(s). \qquad \square$$

The idea behind the above update rule is the following: Instead of requiring that $h(x_k + s_k)$ be smaller than $h(x_k)$, it is only required that $h(x_k + s_k)$ is either less than $h(x_k)$ or less than the weighted mean of the function values at the last $m_k = \min\{i+1, m\}$ successful iterates. Of course, if $m = 1$, then $\text{rared}_k(s) = \text{ared}_k(s)$ and the usual reduction ratio is recovered. Our approach is a slightly stronger requirement than the straightforward idea to replace ared_k with

$$\text{rared}_k^{\infty}(s) = \max_{0 \le r < m_k} h(x_{j_{i-r}}) - h(x_k + s).$$

Unfortunately, for this latter choice it does not seem to be possible to establish all the global convergence results that are available for the monotone case. For our approach, however, this is possible without making the theory substantially more difficult. Moreover, we can approximate rared_k^{∞} arbitrarily accurately by rared_k if we choose λ sufficiently small, in each iteration select $0 \le r_k < m_k$ satisfying $h(x_{j_{i-r_k}}) = \max_{0 \le r < m_k} h(x_{j_{i-r}})$, and set

$$\lambda_{kr} = \lambda \ \text{ if } r \ne r_k, \quad \lambda_{kr_k} = 1 - (m_k - 1)\lambda. \tag{3.14}$$

To obtain a globally *and* locally fast convergent algorithm, we will embed the Newton-like Algorithm 3.1 into the trust-region Algorithm 3.3 by using it to compute trial steps s_k^{P}. Since we need trial steps that satisfy the feasibility condition (3.9), we compute s_k^{P} slightly different than s_k^{PN}. Whereas s_k^{PN} is the projection of $s_k^{\text{N}} = -M_k^{-1}H(x_k)$ onto $X - x_k$, we obtain s_k^{P} by projection onto the feasible set X_k of the trust-region subproblem (3.8):

$$s_k^{\text{P}} := P_{X_k}(s_k^{\text{N}}).$$

Since we will show that $s_k^{\text{P}} = s_k^{\text{PN}}$ finally holds, this modification does not change the local convergence behavior. If s_k^{P} satisfies the decrease condition (3.10), we choose $s_k = s_k^{\text{P}}$ as trial step. Otherwise, a different trial step verifying (3.9) and (3.10) must be computed. In section 6 we propose a general way to do this. We thus arrive at the final version of the algorithm.

ALGORITHM 3.6 (trust-region projected-Newton algorithm).

As Algorithm 3.3, but with Step 3 implemented as follows:

3.1. If coming from Step 6, then set $s_k^{\text{N}} := s_{k-1}^{\text{N}}$. Otherwise compute the Newton-like step s_k^{N} by solving $M_k s_k^{\text{N}} = -H(x_k)$.

3.2. Compute the projected Newton step $s_k^{\text{P}} := P_{X_k}(s_k^{\text{N}})$.

3.3. If $s_k = s_k^{\text{P}}$ satisfies the decrease condition (3.10), then set $s_k := s_k^{\text{P}}$. Otherwise, compute a step s_k satisfying (3.9) and (3.10).

4. Global convergence. We first establish the continuous differentiability of the merit function h. The proof is based on the following lemma. A similar result can be found in [42, Proof of Thm. 3.1 (iii)].

LEMMA 4.1. *Let $f : U \to \mathbb{R}$ be locally Lipschitz continuous on the nonempty open set $U \subset \mathbb{R}^n$. Assume that f is continuously differentiable on $U \setminus f^{-1}(0)$. Then the function f^2 is continuously differentiable on U. Moreover, $\nabla f^2(x) = 2f(x)v^T$ for all $v \in \partial f(x)$ and all $x \in U$.*

Proof. By assumption, f is continuously differentiable on the open set $U \setminus f^{-1}(0)$. Therefore, f^2 is C^1 on $U \setminus f^{-1}(0)$ with gradient $\nabla f^2(x) = 2f(x)\nabla f(x)$. Moreover, we have $\partial f(x) = \{\nabla f(x)^T\}$. Further, it follows immediately from the local Lipschitz continuity of f that at all $x \in f^{-1}(0)$ the function f^2 is differentiable with $\nabla f^2(x) =$

$0 = 2f(x)v^T$ for all $v \in \partial f(x)$. To prove the continuity of ∇f^2 at $x \in f^{-1}(0)$, note that the local boundedness of ∂f implies $\nabla f^2(y) = 2f(y)v(y)^T \to 0$ as $y \to x$, where $v(y) \in \partial f(y)$. ∎

LEMMA 4.2. *Under the assumptions* (A1) *and* (A2) *on the mapping* H*, the function* $h(x) = \|H(x)\|^2/2$ *is continuously differentiable on* U *with gradient* $\nabla h(x) = V^T H(x)$*, where* $V \in \partial H(x)$ *is arbitrary. In particular,* (A1) *and* (A2) *imply* (A2′).

Proof. For $1 \le i \le n$, the component function H_i of H is semismooth and thus locally Lipschitz continuous on U. Moreover, it is C^1 on $U \setminus H_i^{-1}(0)$. Therefore, H_i^2 is C^1 on U by Lemma 4.1. The same then holds true for $h(x) = \frac{1}{2}\sum H_i^2(x)$. Furthermore, for all $V = (v_1, \dots, v_n)^T \in \partial H(x)$ holds $v_i \in \partial H_i(x)$, $1 \le i \le n$ (see [6, Prop. 2.6.2.(e)]) and thus by Lemma 4.1

$$\nabla h(x) = \sum_{i=1}^{n} \nabla H_i^2(x) = \sum_{i=1}^{n} H_i(x) v_i = V^T H(x). \qquad \blacksquare$$

In the next lemma an important decrease property of the function values $h(x_k)$ is established.

LEMMA 4.3. *Let* x_k*,* s_k*,* Δ_k*,* j_i*, etc., be generated by Algorithm* 3.3*. Then for all computed indices* $i \ge 1$ *holds*

$$h(x_{j_i}) < h(x_0) - \eta_1 \lambda \sum_{r=0}^{i-2} \mathrm{pred}_{j_r}(s_{j_r}) - \eta_1 \mathrm{pred}_{j_{i-1}}(s_{j_{i-1}}). \tag{4.1}$$

Proof. We will use the short notations $\mathrm{ared}_k = \mathrm{ared}_k(s_k)$, $\mathrm{rared}_k = \mathrm{rared}_k(s_k)$, and $\mathrm{pred}_k = \mathrm{pred}_k(s_k)$. First, let us note that (3.10) implies $\mathrm{pred}_k > 0$.

The rest of the proof is by induction. For $i = 1$ we have by (3.12) and using $\rho_{j_0}(s_{j_0}) > \eta_1$

$$h(x_{j_1}) = h(x_{j_0+1}) = h(x_{j_0}) - \mathrm{ared}_{j_0} < h(x_{j_0}) - \eta_1 \mathrm{pred}_{j_0} = h(x_0) - \eta_1 \mathrm{pred}_{j_0}.$$

Now assume that (4.1) holds for $1, \dots, i$.

If $\mathrm{rared}_{j_i} = \mathrm{ared}_{j_i}$ then, using (4.1) and $\lambda \le 1$,

$$\begin{aligned} h(x_{j_{i+1}}) = h(x_{j_i+1}) &= h(x_{j_i}) - \mathrm{ared}_{j_i} = h(x_{j_i}) - \mathrm{rared}_{j_i} \\ &< h(x_0) - \eta_1 \lambda \sum_{r=0}^{i-2} \mathrm{pred}_{j_r} - \eta_1 \mathrm{pred}_{j_{i-1}} - \eta_1 \mathrm{pred}_{j_i} \\ &\le h(x_0) - \eta_1 \lambda \sum_{r=0}^{i-1} \mathrm{pred}_{j_r} - \eta_1 \mathrm{pred}_{j_i}. \end{aligned}$$

If $\mathrm{rared}_{j_i} \ne \mathrm{ared}_{j_i}$ then $\mathrm{rared}_{j_i} > \mathrm{ared}_{j_i}$, and with $q = \min\{i, m-1\}$ we obtain

$$\begin{aligned} h(x_{j_{i+1}}) = h(x_{j_i+1}) &= \sum_{p=0}^{q} \lambda_{j_i p} h(x_{j_i - p}) - \mathrm{rared}_{j_i} \\ &< \sum_{p=0}^{q} \lambda_{j_i p} \left(h(x_0) - \eta_1 \lambda \sum_{r=0}^{i-p-2} \mathrm{pred}_{j_r} - \eta_1 \mathrm{pred}_{j_{i-p-1}} \right) - \eta_1 \mathrm{pred}_{j_i}. \end{aligned}$$

Using $\{0,\dots,q\}\times\{0,\dots,i-q-2\}\subset\{(p,r)\,;\ 0\le p\le q,\ 0\le r\le i-p-2\}$, $\lambda_{j_i0}+\cdots+\lambda_{j_iq}=1$, and $\lambda_{j_ip}\ge\lambda$, we can proceed as follows:

$$\begin{aligned}h(x_{j_{i+1}}) &< h(x_0)-\eta_1\lambda\sum_{r=0}^{i-q-2}\left(\sum_{p=0}^{q}\lambda_{j_ip}\right)\mathrm{pred}_{j_r}-\eta_1\lambda\sum_{p=0}^{q}\mathrm{pred}_{j_{i-p-1}}-\eta_1\mathrm{pred}_{j_i}\\ &\le h(x_0)-\eta_1\lambda\sum_{r=0}^{i-q-2}\mathrm{pred}_{j_r}-\eta_1\lambda\sum_{r=i-q-1}^{i-1}\mathrm{pred}_{j_r}-\eta_1\mathrm{pred}_{j_i}\\ &= h(x_0)-\eta_1\lambda\sum_{r=0}^{i-1}\mathrm{pred}_{j_r}-\eta_1\mathrm{pred}_{j_i}.\qquad\square\end{aligned}$$

LEMMA 4.4. *Let x_k, s_k, Δ_k, etc., be generated by Algorithm* 3.3. *Then for arbitrary $x\in X$ with $\chi(x)\ne0$ and $0<\eta<1$ there exist $\Delta>0$ and $\delta>0$ such that*

$$\rho_k\ge\eta$$

holds whenever $\|x_k-x\|\le\delta$ and $\Delta_k\le\Delta$ is satisfied.

Proof. Since $\chi(x)\ne0$, by continuity there exist $\delta>0$ and $\varepsilon>0$ such that $\chi(x_k)\ge\varepsilon$ for all k with $\|x_k-x\|\le\delta$. Now, for $0<\Delta\le\min\{1,\varepsilon\}$ and any k with $\|x_k-x\|\le\delta$ and $0<\Delta_k\le\Delta$, we obtain from the decrease condition (3.10)

$$\mathrm{pred}_k(s_k)=-q_k(s_k)\ge\beta_2\chi(x_k)\min\{1,\Delta_k,\chi(x_k)\}\ge\beta_2\varepsilon\Delta_k.$$

In particular, by (3.9)

$$\|s_k\|_\infty\le\beta_1\Delta_k\le\frac{\beta_1}{\beta_2\varepsilon}\mathrm{pred}_k(s_k).\tag{4.2}$$

Further, with appropriate $\tau_k\in[0,1]$

$$\begin{aligned}\mathrm{ared}_k(s_k)&=h(x_k)-h(x_k+s_k)=-\nabla h(x_k+\tau_ks_k)^Ts_k\\ &=-q_k(s)+(g_k-\nabla h(x_k+\tau_ks_k))^Ts_k+\frac12\|M_ks_k\|^2\\ &\ge\mathrm{pred}_k(s_k)+(g_k-\nabla h(x_k+\tau_ks_k))^Ts_k.\end{aligned}$$

Since ∇h is continuous, there exists $\delta'>0$ such that

$$\|\nabla h(x')-\nabla h(x)\|_1\le(1-\eta)\frac{\beta_2\varepsilon}{2\beta_1}$$

for all $x'\in X$ with $\|x'-x\|<\delta'$. By reducing Δ and δ, if necessary, such that $\delta+\sqrt{n}\beta_1\Delta<\delta'$ we achieve, using (3.9), that for all k with $\|x_k-x\|\le\delta$ and $0<\Delta_k\le\Delta$

$$\|x_k+\tau_ks_k-x\|\le\|x_k-x\|+\tau_k\|s_k\|\le\delta+\sqrt{n}\beta_1\Delta<\delta',\quad\|x_k-x\|\le\delta<\delta'.$$

Hence, for all these indices k,

$$\|g_k-\nabla h(x_k+\tau_ks_k)\|_1\le\|g_k-\nabla h(x)\|_1+\|\nabla h(x)-\nabla h(x_k+\tau_ks_k)\|_1\le(1-\eta)\frac{\beta_2\varepsilon}{\beta_1},$$

and thus by (4.2)

$$|(g_k - \nabla h(x_k + \tau_k s_k))^T s_k| \leq (1-\eta)\frac{\beta_2 \varepsilon}{\beta_1}\|s_k\|_\infty \leq (1-\eta)\mathrm{pred}_k(s_k).$$

This implies that for all these k there holds

$$\mathrm{rared}_k(s_k) \geq \mathrm{ared}_k(s_k) \geq \mathrm{pred}_k(s_k) - |(g_k - \nabla h(x_k + \tau_k s_k))^T s_k| \geq \eta \mathrm{pred}_k(s_k).$$

The proof is complete. ∎

LEMMA 4.5. *Algorithm* 3.3 *either terminates after finitely many steps with a KKT-point* x_k *of* (1.2) *or generates an infinite sequence* (s_{j_i}) *of accepted steps.*

Proof. Assume that Algorithm 3.3 neither terminates nor generates an infinite sequence (s_{j_i}) of accepted steps. Then there exists a smallest index k_0 such that all steps s_k are rejected for $k \geq k_0$. In particular, $x_k = x_{k_0}$, $k \geq k_0$, and the sequence of trust-region radii Δ_k tends to zero as $k \to \infty$, because

$$\Delta_{k_0+j} \leq \gamma_1^j \Delta_{k_0}.$$

Since the algorithm does not terminate, we know that $\chi(x_{k_0}) \neq 0$. But now Lemma 4.4 with $x = x_{k_0}$ yields that s_k is accepted as soon as Δ_k becomes sufficiently small. This contradicts our assumption. Therefore, the assertion of the lemma is true. ∎

LEMMA 4.6. *Assume that Algorithm* 3.3 *generates infinitely many successful steps* s_{j_i} *and that there exists* $\mathcal{S}' \subset \mathcal{S}$ *with*

$$\sum_{k \in \mathcal{S}'} \Delta_k = \infty. \tag{4.3}$$

Then $\liminf_{\mathcal{S}' \ni k \to \infty} \chi(x_k) = 0$.

Proof. Let the assumptions of the lemma hold and assume that the assertion is wrong. Then there exists $\varepsilon > 0$ such that $\chi(x_k) \geq \varepsilon$ for all $k \in \mathcal{S}' \subset \mathcal{S}$. From (4.3) follows that $\mathcal{S}'$ is not finite. For all $k \in \mathcal{S}'$

$$\mathrm{pred}_k(s_k) \geq \beta_2 \chi(x_k) \min\{1, \Delta_k, \chi(x_k)\} \geq \beta_2 \varepsilon \min\{1, \Delta_k, \varepsilon\}$$

holds by (3.10). From this estimate, the nonnegativity of h, and Lemma 4.3 we obtain for all $j \in \mathcal{S}'$, using $\lambda \leq 1$

$$\begin{aligned} h(x_0) \geq h(x_0) - h(x_j) &> \eta_1 \lambda \sum_{\substack{k \in \mathcal{S} \\ k<j}} \mathrm{pred}_k(s_k) \geq \eta_1 \lambda \sum_{\substack{k \in \mathcal{S}' \\ k<j}} \mathrm{pred}_k(s_k) \\ &\geq \eta_1 \lambda \beta_2 \varepsilon \sum_{\substack{k \in \mathcal{S}' \\ k<j}} \min\{1, \Delta_k, \varepsilon\} \to \infty \qquad (\text{as } j \to \infty). \end{aligned}$$

This is a contradiction. Therefore, the assumption was wrong and the lemma is proved. ∎

We now have everything at hand that we need to establish our first global convergence result. It is applicable in the case $\gamma_0 > 0$, $\Delta_{\min} > 0$ and says that accumulation points are KKT-points of (1.2).

THEOREM 4.7. *Let* $\gamma_0 > 0$ *and* $\Delta_{\min} > 0$. *Assume that Algorithm* 3.3 *does not terminate after finitely many steps with a KKT-point* x_k *of* (1.2). *Then the algorithm generates infinitely many accepted steps* (s_{j_i}). *Moreover, every accumulation point of* (x_k) *is a KKT-point of* (1.2).

Proof. Suppose that Algorithm 3.3 does not terminate after a finite number of steps. Then according to Lemma 4.5 infinitely many successful steps (s_{j_i}) are generated. Assume that $\bar{x}$ is an accumulation point of (x_k) that is not a KKT-point of (1.2). Since $\chi(\bar{x}) \neq 0$, invoking Lemma 4.4 with $x = \bar{x}$ yields $\Delta > 0$ and $\delta > 0$ such that $k \in \mathcal{S}$ holds for all k with $\|x_k - \bar{x}\| \leq \delta$ and $\Delta_k \leq \Delta$. Since $\bar{x}$ is an accumulation point, there exists an infinite increasing sequence $j_i' \in \mathcal{S}$, $i \geq 0$, of indices such that $\|x_{j_i'} - \bar{x}\| \leq \delta$ and $x_{j_i'} \to \bar{x}$.

If $(j_i' - 1) \in \mathcal{S}$, then $\Delta_{j_i'} \geq \Delta_{\min}$. Otherwise, $s_{j_i'-1}$ was rejected, which, since then $x_{j_i'-1} = x_{j_i'}$, is only possible if $\Delta_{j_i'-1} > \Delta$, and therefore $\Delta_{j_i'} \geq \gamma_0 \Delta_{j_i'-1} > \gamma_0 \Delta$. We conclude that for all i holds $\Delta_{j_i'} \geq \min\{\Delta_{\min}, \gamma_0 \Delta\}$. Now Lemma 4.6 is applicable with $\mathcal{S}' = \{j_i' ;\ i \geq 0\}$ and yields

$$0 \neq \chi(\bar{x}) = \lim_{i \to \infty} \chi(x_{j_i'}) = \liminf_{i \to \infty} \chi(x_{j_i'}) = 0,$$

where we have used the continuity of χ. This is a contradiction. Therefore, the assumption $\chi(\bar{x}) \neq 0$ was wrong. ∎

Next, we prove a result that holds also for $\Delta_{\min} = 0$. Moreover, the existence of accumulation points is not required.

THEOREM 4.8. *Let $\gamma_0 > 0$ or $\Delta_{\min} = 0$ hold. Assume that Algorithm* 3.3 *does not terminate after finitely many steps with a KKT-point x_k of* (1.2). *Then the algorithm generates infinitely many accepted steps (s_{j_i}). Moreover,*

$$\liminf_{k \to \infty} \chi(x_k) = 0. \tag{4.4}$$

In particular, if x_k converges to $\bar{x}$, then $\bar{x}$ is a KKT-point of (1.2).

Proof. By Lemma 4.5, infinitely many successful steps (s_{j_i}) are generated.

Now assume that (4.4) is wrong, i.e.,

$$\liminf_{k \to \infty} \chi(x_k) > 0. \tag{4.5}$$

Then we obtain from Lemma 4.6 that

$$\sum_{k \in \mathcal{S}} \Delta_k < \infty. \tag{4.6}$$

In particular, (x_{j_i}) is a Cauchy sequence by (3.9). Therefore, (x_k) converges to some limit $\bar{x}$, at which according to (4.5) and the continuity of χ holds $\chi(\bar{x}) \neq 0$.

Case 1. $\Delta_{\min} > 0$.

Then by assumption also $\gamma_0 > 0$, and Theorem 4.7 yields $\chi(\bar{x}) = 0$, which is a contradiction.

Case 2. $\Delta_{\min} = 0$.

Lemma 4.4 with $x = \bar{x}$ and $\eta = \eta_2$ yields $\Delta > 0$ and $\delta > 0$ such that $k \in \mathcal{S}$ and $\Delta_{k+1} \geq \Delta_k$ holds for all k with $\|x_k - \bar{x}\| \leq \delta$ and $\Delta_k \leq \Delta$. Since $x_k \to \bar{x}$, there exists $k' \geq 0$ with $\|x_k - \bar{x}\| \leq \delta$ for all $k \geq k'$.

Case 2.1. There exists $k'' \geq k'$ with $\Delta_k \leq \Delta$ for all $k \geq k''$.

Then $k \in \mathcal{S}$ and (inductively) $\Delta_k \geq \Delta_{k''}$ for all $k \geq k''$. This contradicts (4.6).

Case 2.2. For infinitely many k holds $\Delta_k > \Delta$.

By (4.6) there exists $k'' \geq k'$ with $\Delta_{j_i} \leq \Delta$ for all $j_i \geq k''$. Now, for each $j_i \geq k''$, there exists an index $k_i \geq j_i$ such that $\Delta_k \leq \Delta$, $j_i \leq k < k_i$, and $\Delta_{k_i} > \Delta$. If $k_i \in \mathcal{S}$, set $j_i' = k_i$, thus obtaining $j_i' \in \mathcal{S}$ with $\Delta_{j_i'} > \Delta$. If $k_i \notin \mathcal{S}$, we have

$j_i' \stackrel{\text{def}}{=} k_i - 1 \geq j_i \geq k'$, and thus $j_i' \in \mathcal{S}$, since by construction $\Delta_{j_i'} \leq \Delta$. Moreover, $\Delta < \Delta_{k_i} \leq \gamma_2 \Delta_{j_i'}$ (here $\Delta_{\min} = 0$ is used) implies that $\Delta_{j_i'} > \Delta/\gamma_2$. By this construction, we obtain an infinitely increasing sequence $(j_i') \subset \mathcal{S}$ with $\Delta_{j_i'} > \Delta/\gamma_2$. Again, this yields a contradiction to (4.6).

Therefore, in all cases we obtain a contradiction. Thus, the assumption was wrong and the proof of (4.4) is complete.

Finally, if $x_k \to \bar{x}$, the continuity of χ and (4.4) imply $\chi(\bar{x}) = 0$. Therefore, $\bar{x}$ is a KKT-point of (1.2). ∎

The next result shows that under appropriate assumptions the lim inf in (4.4) can be replaced by lim.

THEOREM 4.9. *Let $\gamma_0 > 0$ or $\Delta_{\min} = 0$ hold. Assume that Algorithm* 3.3 *does not terminate after finitely many steps with a KKT-point x_k of* (1.2). *Then the algorithm generates infinitely many accepted steps (s_{j_i}). Moreover, if there exists a set Ω that contains (x_k) and on which χ is uniformly continuous and bounded, then*

$$\lim_{k \to \infty} \chi(x_k) = 0. \tag{4.7}$$

Proof. In view of Theorem 4.8 we have only to prove (4.7). Thus, let us assume that (4.7) is not true. Then there exists $\varepsilon > 0$ such that $\chi(x_k) \geq 2\varepsilon$ for infinitely many $k \in \mathcal{S}$. Since (4.4) holds, we thus can find increasing sequences $(j_i')_{i \geq 0}$ and $(k_i')_{i \geq 0}$ with $j_i' < k_i' < j_{i+1}'$ and

$$\chi(x_{j_i'}) \geq 2\varepsilon, \qquad \chi(x_k) > \varepsilon \quad \forall\, k \in \mathcal{S} \text{ with } j_i' < k < k_i', \qquad \chi(x_{k_i'}) \leq \varepsilon.$$

Setting $\mathcal{S}' = \bigcup_{i=0}^{\infty} \mathcal{S}_i'$ with $\mathcal{S}_i' = \{k \in \mathcal{S}\,;\ j_i' \leq k < k_i'\}$, we have

$$\liminf_{\mathcal{S}' \ni k \to \infty} \chi(x_k) \geq \varepsilon.$$

Therefore, with Lemma 4.6

$$\sum_{k \in \mathcal{S}'} \Delta_k < \infty.$$

In particular, $\sum_{k \in \mathcal{S}_i'} \Delta_k \to 0$ as $i \to \infty$, and thus, using (3.9),

$$\|x_{k_i'} - x_{j_i'}\|_\infty \leq \sum_{k \in \mathcal{S}_i'} \|s_k\|_\infty \leq \beta_1 \sum_{k \in \mathcal{S}_i'} \Delta_k \to 0 \quad (\text{as } i \to \infty).$$

This is a contradiction to the uniform continuity of χ, since

$$\lim_{i \to \infty} (x_{k_i'} - x_{j_i'}) = 0, \quad \text{but} \quad |\chi(x_{k_i'}) - \chi(x_{j_i'})| \geq \varepsilon \quad \forall\, i \geq 0.$$

Therefore, the assumption was wrong and the assertion is proved. ∎

The above results establish global convergence to a KKT-point $\bar{x}$ of the minimization reformulation (1.2). Of course, it may happen that $\bar{x}$ fails to be a global solution, i.e., fails to satisfy $H(\bar{x}) = 0$ if (1.1) possesses a solution. However, as can be seen from our numerical results in section 8, our nonmonotone trust-region approach very successfully avoids convergence to local–nonglobal solutions. Moreover, in the context of semismooth reformulations of MCPs, conditions can be stated under which a local solution to (1.2) is a global solution. See [15, Thm. 2].

5. Local convergence. In this section we will prove under relatively weak assumptions that Algorithm 3.6 turns into Algorithm 3.1 as soon as x_k comes sufficiently close to a BD-regular solution $\bar{x}$ of (1.1). Of course, we have to take care that close to $\bar{x}$ the projected Newton step is used as trial step which requires (3.10) to hold. Note that (3.9) is automatically satisfied since $s_k^{\mathrm{P}} \in X_k$. We will see that it suffices to require that the following implication holds:

$$\begin{aligned} &\|x_k - \bar{x}\| < \beta_3, \quad \mathrm{pred}_k(s_k^{\mathrm{P}}) \geq \beta_4 h(x_k) \\ \Longrightarrow \quad &(3.10) \text{ is satisfied for } s_k = s_k^{\mathrm{P}}, \end{aligned} \tag{5.1}$$

where $\beta_3 > 0$ and $0 < \beta_4 < 1$ are constants independent of k. Since $h(\bar{x}) = 0$, this means that for x_k close to $\bar{x}$ the step $s_k = s_k^{\mathrm{P}}$ satisfies (3.10) if the predicted decrease is at least a fraction of the maximum possible actual decrease $h(x_k) - h(\bar{x}) = h(x_k)$. Therefore, (5.1) is certainly a reasonable requirement, which in section 6 will be shown to hold for decrease conditions that are implemented by means of a Cauchy step.

THEOREM 5.1. *Let the assumptions* (A1) *and* (A2) *hold. Assume* $\Delta_{\min} > 0$ *and that Algorithm* 3.6 *generates infinitely many iterates. Let* $\bar{x} \in X$ *be a BD-regular zero of* H *and let* (5.1) *be satisfied. Then there exist* $\delta > 0$ *and* $\varepsilon > 0$ *such that the following holds:*

If the index k' *satisfies* $(k'-1) \in \mathcal{S}$, $\|x_{k'} - \bar{x}\| \leq \varepsilon$, *and if* (3.1) *holds for all* $k \geq k'$, *then*

(a) *for all* $k \geq k'$ *the step* s_k^{P} *satisfies* $s_k^{\mathrm{P}} = s_k^{\mathrm{PN}} = P_X(x_k + s^{\mathrm{N}}) - x_k$, *is chosen as trial step, i.e.,* $s_k = s_k^{\mathrm{P}}$, *and is accepted, i.e.,* $k \in \mathcal{S}$;

(b) *the sequence* (x_k) *converges to* $\bar{x}$;

(c) *if* (3.2) *holds for all* $k \geq k'$, *then* (x_k) *converges q-superlinearly to* $\bar{x}$, *and moreover,*

$$\lim_{k\to\infty} \frac{q_k(s_k^{\mathrm{P}})}{h(x_k)} = \lim_{k\to\infty} \frac{q_k(s_k^{\mathrm{N}})}{h(x_k)} = -1; \tag{5.2}$$

(d) *if* $p > 0$ *and* (3.3) *holds for all* $k \geq k'$, *then* (x_k) *converges with q-order* $1+p$ *to* $\bar{x}$ *and* (5.2) *holds.*

Proof. Assume that

$$\delta > 0 \text{ and } \varepsilon > 0 \text{ are chosen sufficiently small} \tag{5.3}$$

with $\varepsilon < \beta_3$, the constant in (5.1). Then Proposition 2.11 yields a constant $\zeta > 0$ such that

$$\|H(x)\| \geq \zeta \|x - \bar{x}\| \qquad \forall\, B_\varepsilon(\bar{x}) \stackrel{\mathrm{def}}{=} \{x\,;\ \|x - \bar{x}\| \leq \varepsilon\}\,. \tag{5.4}$$

Now let k be arbitrary such that $(k-1) \in \mathcal{S}$, $x_k \in B_\varepsilon(\bar{x})$, and such that (3.1) is satisfied.

Let $d_k \stackrel{\mathrm{def}}{=} x_k - \bar{x}$, $e_k \stackrel{\mathrm{def}}{=} x_k + s_k^{\mathrm{N}} - \bar{x}$, and $V_k \in \partial_B H(x_k)$ such that $\mu_k = \|(M_k - V_k)s_k^{\mathrm{N}}\|$. We begin by copying the proof of Theorem 3.2 to derive the equations (3.4)–(3.6). From (3.5) and the fact that $(k-1) \in \mathcal{S}$ and $\Delta_{\min} > 0$ we conclude, using (5.3), that

$$\|s_k^{\mathrm{N}}\| \leq 2C\|H(x_k)\| = 2C\|H(x_k) - H(\bar{x})\| \leq 2CL\varepsilon \leq \Delta_{\min} \leq \Delta_k. \tag{5.5}$$

In particular,

$$s_k^{\mathrm{P}} = P_{X_k}(s_k^{\mathrm{N}}) = P_{X - x_k}(s_k^{\mathrm{N}}) = P_X(x_k + s_k^{\mathrm{N}}) - x_k = s_k^{\mathrm{PN}}.$$

Since $x_k \in X$ and P_X is Lipschitz continuous of rank 1 it follows that

$$\|s_k^{\mathrm{P}}\| = \|P_X(x_k + s_k^{\mathrm{N}}) - P_X(x_k)\| \leq \|s_k^{\mathrm{N}}\|. \tag{5.6}$$

Now, using $g_k = V_k^T H(x_k)$, (3.1), (3.5), and (5.5)

$$\begin{aligned} |q_k(s_k^{\mathrm{N}}) + h(x_k)| &= \left| g_k^T s_k^{\mathrm{N}} + \frac{1}{2}\|M_k s_k^{\mathrm{N}}\|^2 + h(x_k) \right| = \left| H(x_k)^T V_k s_k^{\mathrm{N}} + 2h(x_k) \right| \\ &\leq \left| H(x_k)^T M_k s_k^{\mathrm{N}} + 2h(x_k) \right| + \|H(x_k)\|\mu_k = \|H(x_k)\|\mu_k \\ &\leq \delta \|H(x_k)\| \|s_k^{\mathrm{N}}\| \leq 4C\delta h(x_k). \end{aligned} \tag{5.7}$$

This shows

$$-(1 + 4C\delta)h(x_k) \leq q_k(s_k^{\mathrm{N}}) \leq -(1 - 4C\delta)h(x_k). \tag{5.8}$$

Furthermore, since $\bar{x} \in X$, the properties of the projection P_X yield

$$\|s_k^{\mathrm{P}} - s_k^{\mathrm{N}}\| = \|P_X(x_k + s_k^{\mathrm{N}}) - (x_k + s_k^{\mathrm{N}})\| \leq \|\bar{x} - (x_k + s_k^{\mathrm{N}})\| = \|e_k\|. \tag{5.9}$$

Thus by (5.6), (5.9), and (3.5)

$$\begin{aligned} |q_k(s_k^{\mathrm{P}}) - q_k(s_k^{\mathrm{N}})| &= \left| g_k^T (s_k^{\mathrm{P}} - s_k^{\mathrm{N}}) + \frac{1}{2}(s_k^{\mathrm{P}} + s_k^{\mathrm{N}})^T M_k^T M_k (s_k^{\mathrm{P}} - s_k^{\mathrm{N}}) \right| \\ &\leq \left(\|V_k\| \|H(x_k)\| + \|M_k\|^2 \|s_k^{\mathrm{N}}\| \right) \|s_k^{\mathrm{P}} - s_k^{\mathrm{N}}\| \\ &\leq \left(\|V_k\| + 2C\|M_k\|^2 \right) \|H(x_k)\| \|e_k\|. \end{aligned} \tag{5.10}$$

From (3.6) and (5.4) we conclude as in the proof of Theorem 3.2 that

$$\frac{\|e_k\|}{\|H(x_k)\|} \leq \frac{1}{\zeta} \frac{\|e_k\|}{\|d_k\|} \to 0 \qquad (\text{as } (\delta, \varepsilon) \to 0). \tag{5.11}$$

Now let $\theta \in (0, 1)$ be arbitrary. Due to the upper semicontinuity of $\partial_B H$, cf. Proposition 2.2, the boundedness of $(\|M_k\|)$, and (5.11), we achieve by invoking (5.3) that

$$2 \left(\|V_k\| + 2C\|M_k\|^2 \right) \frac{\|e_k\|}{\|H(x_k)\|} + 4C\delta \leq \theta.$$

This in combination with (5.8) and (5.10) shows that

$$-(1 + \theta)h(x_k) \leq q_k(s_k^{\mathrm{P}}) \leq -(1 - \theta)h(x_k). \tag{5.12}$$

If θ was chosen $\leq 1 - \beta_4$, then

$$\mathrm{pred}_{\mathrm{k}}(s_k^{\mathrm{P}}) = -q_k(s_k^{\mathrm{P}}) \geq (1 - \theta)h(x_k) \geq \beta_4 h(x_k).$$

Therefore, (5.1) implies that $s_k = s_k^{\mathrm{P}} = s_k^{\mathrm{PN}}$. As in (3.7) we obtain

$$\|x_k + s_k^{\mathrm{P}} - \bar{x}\| \leq \|e_k\|, \tag{5.13}$$

and thus

$$\|H(x_k + s_k)\| = \|H(x_k + s_k^{\mathrm{P}})\| \leq L\|x_k + s_k^{\mathrm{P}} - \bar{x}\| \leq L\|e_k\|, \tag{5.14}$$

where L denotes the Lipschitz constant of H on $B_\varepsilon(\bar{x})$. By (5.3), (5.11), and (5.14) we thus may assume that

$$\text{rared}_k(s_k) \geq h(x_k) - h(x_k + s_k) \geq h(x_k) - \frac{L^2}{2}\|e_k\|^2 \geq (1-\theta)h(x_k). \tag{5.15}$$

If θ was chosen $< (1-\eta_1)/(1+\eta_1)$, then by (5.12) and (5.15)

$$\rho_k(s_k) = \frac{\text{rared}_k(s_k)}{\text{pred}_k(s_k)} \geq \frac{(1-\theta)h(x_k)}{(1+\theta)h(x_k)} > \eta_1.$$

Consequently, $s_k = s_k^{\text{P}}$ is accepted. Moreover, by (5.11), (5.13) and again invoking (5.3) we get $x_{k+1} \in B_\varepsilon(\bar{x})$.

We briefly resume what we have shown so far.

If $\delta > 0$ and $\varepsilon > 0$ are sufficiently small and k is such that $(k-1) \in \mathcal{S}$, $x_k \in B_\varepsilon(\bar{x})$, and (3.1) holds, then $s_k = s_k^{\text{P}} = s_k^{\text{PN}}$ is chosen as trial step and this step is accepted, i.e., $k \in \mathcal{S}$. Moreover, we have $x_{k+1} \in B_\varepsilon(\bar{x})$.

Consequently, $(k+1)$ satisfies again the requirements and we obtain inductively that assertion (a) holds.

Since by (a) we know that for $k \geq k'$ Algorithm 3.6 turns into Algorithm 3.1, the assertions (b)–(d), with the exception of (5.2), follow directly from Theorem 3.2. Finally, to prove (5.2) we observe that, if (3.2) holds, we can strengthen (5.7):

$$|q_k(s_k^{\text{N}}) + h(x_k)| \leq \|H(x_k)\|\mu_k = o(h(x_k)), \tag{5.16}$$

where we have used (3.2) and (5.5). From (5.10) and (5.11) we conclude

$$|q_k(s_k^{\text{P}}) - q_k(s_k^{\text{N}})| = O(\|H(x_k)\|\,\|e_k\|) = o(h(x_k)).$$

This and (5.16) imply (5.2). ∎

6. An implementable decrease condition. Our convergence analysis was carried out on the basis of the abstract condition (3.10) involving a criticality measure χ. For fast local convergence we also required (5.1). In this section we describe a concrete implementation of these conditions by means of a Cauchy step that is defined using an affinely scaled gradient. Similar approaches can be found in [7, 49].

We define the *Cauchy step* s_k^{C} as the solution of

$$\text{minimize } q_k(s) \quad \text{subject to} \quad s = -tD(x_k)^{2\gamma}g_k,\ t \geq 0,\ s \in X_k.$$

Hereby, $\gamma \geq 1$ is fixed and the diagonal affine-scaling matrix $D(x) \in \mathbb{R}^{n\times n}$ is defined by

$$D(x)_{ii} \stackrel{\text{def}}{=} \begin{cases} \min\{\kappa_D, x_i - l_i\} & \text{if } (\nabla h(x))_i > 0, \\ \min\{\kappa_D, u_i - x_i\} & \text{if } (\nabla h(x))_i < 0, \\ \min\{\kappa_D, x_i - l_i, u_i - x_i\} & \text{if } (\nabla h(x))_i = 0, \end{cases}$$

where $\kappa_D > 0$ is a constant.

The following condition will replace the abstract condition (3.10).

Fraction of Cauchy decrease condition.

$$q_k(s_k) \leq \alpha q_k(s_k^{\text{C}}), \tag{6.1}$$

where $\alpha \in (0,1)$ is a constant.

Let χ denote any criticality measure such that

$$\beta_5 \chi(x) \le \chi_{\text{AS}}(x) \stackrel{\text{def}}{=} \|D(x)^\gamma \nabla h(x)\| \tag{6.2}$$

holds on X for some $\beta_5 > 0$. We will show that there exists $\beta_2 > 0$ such that the validity of (6.1) implies (3.10). Certainly, a natural choice for χ satisfying (6.2) is $\chi = \chi_{\text{AS}}/\beta_5$. Therefore, we first show that χ_{AS} is a criticality measure.

LEMMA 6.1. *The function χ_{AS} defined in* (6.2) *is a criticality measure, i.e., satisfies* (3.11).

Proof. It is easily seen that for $x \in X$ $D(x)^\gamma \nabla h(x) = 0$ holds if and only if

$$(\nabla h(x))_i \begin{cases} \ge 0 & \text{if } x_i = l_i, \\ \le 0 & \text{if } x_i = u_i, \\ = 0 & \text{if } l_i < x_i < u_i, \end{cases}$$

which are the KKT-conditions of (1.2).

We still have to prove the continuity of χ_{AS}. Let $x \in X$ be arbitrary and set $I_+ = \{i\,;\ (\nabla h(x))_i > 0\}$, $I_- = \{i\,;\ (\nabla h(x))_i < 0\}$. Since ∇h is continuous, there exists $\delta > 0$ such that for all $y \in X$, $\|y - x\| \le \delta$, and all $i \in I_+ \cup I_-$ $(\nabla h(x))_i (\nabla h(y))_i > 0$ holds. Now let $y \in X$, $\|y - x\| \le \delta$, be arbitrary and set $r(x,y) = D(y)^\gamma \nabla h(y) - D(x)^\gamma \nabla h(x)$. For all $i \in I_+$

$$\begin{aligned} |r(x,y)_i| &\le (\nabla h(x))_i |(D(y)^\gamma - D(x)^\gamma)_{ii}| + D(y)_{ii}^\gamma |(\nabla h(y) - \nabla h(x))_i| \\ &\le (\nabla h(x))_i |\min\{\kappa_D, y_i - l_i\}^\gamma - \min\{\kappa_D, x_i - l_i\}^\gamma| + \kappa_D^\gamma |(\nabla h(y) - \nabla h(x))_i| \end{aligned}$$

holds. The same calculation yields for $i \in I_-$

$$|r(x,y)_i| \le |(\nabla h(x))_i| |\min\{\kappa_D, u_i - y_i\}^\gamma - \min\{\kappa_D, u_i - x_i\}^\gamma| + \kappa_D^\gamma |(\nabla h(y) - \nabla h(x))_i|.$$

For all $i \notin I_+ \cup I_-$, $(\nabla h(x))_i = 0$ holds and thus

$$|r(x,y)_i| = D(y)_{ii}^\gamma |(\nabla h(y) - \nabla h(x))_i| \le \kappa_D^\gamma |(\nabla h(y) - \nabla h(x))_i|.$$

In all three cases we see that $|r(x,y)_i| \to 0$ as $y \to x$. Hence, χ_{AS} is continuous. ∎

The next lemma gives a sufficient condition for χ_{AS} to be uniformly continuous. This is needed for Theorem 4.9.

LEMMA 6.2. *The criticality measure χ_{AS} defined in* (6.2) *is uniformly continuous on $\Omega \subset U$ if ∇h is bounded and uniformly continuous on Ω.*

Proof. Under the assumptions on ∇h it follows easily from the estimates of $|r(x,y)_i|$ in the proof of Lemma 6.1 that χ_{AS} is uniformly continuous on Ω. ∎

Remark 6.3. The assumptions of Lemma 6.2 are met if Ω is compact and h is continuously differentiable. According to Lemma 4.2, the latter is ensured by the assumptions (A1) and (A2). ∎

We have the following relation between (3.10) and (6.1).

LEMMA 6.4. *Let the criticality measure χ satisfy* (6.2). *Assume that the sequence $(\|M_k\|)$ is bounded above by a constant C_M. Then there exists a constant $\beta_2 > 0$ that only depends on α, β_5, γ, κ_D, and C_M such that the following holds: If $\chi(x_k) \ne 0$ and if the trial step s_k satisfies the fraction of Cauchy decrease condition* (6.1), *then* (3.10) *also holds.*

Proof. Set $D_k = D(x_k)$, $d_k = -D_k^{2\gamma} g_k$, and $\hat{g}_k = D_k^\gamma g_k$. We will derive an upper bound for $q_k(s_k^C) = q_k(t^* d_k) = \min\{q_k(t d_k)\,;\ t \ge 0,\ t d_k \in X_k\}$, and then apply $q_k(s_k) \le \alpha q_k(t^* d_k)$.

First, observe that d_k is a descent direction of q_k at 0, since by (6.2)

$$\nabla q_k(0)^T d_k = {g_k}^T d_k = -\chi_{\text{AS}}(x_k)^2 \le -\beta_5^2 \chi(x_k)^2 < 0.$$

The maximum stepsize allowed by the trust region constraint is

$$t_1 = \min\left\{ \frac{\Delta_k}{|(d_k)_i|} ;\ (d_k)_i \neq 0 \right\} = \frac{\Delta_k}{(D_k)_{ii}^{\gamma} |(\hat{g}_k)_i|} \ge \frac{\Delta_k}{\kappa_D^{\gamma} \chi_{\text{AS}}(x_k)}. \tag{6.3}$$

The maximum stepsize t_2 admitted by the lower bounds of the set $X - x_k$ is

$$\begin{aligned} t_2 &= \min\left\{ \frac{(x_k - l)_i}{|(d_k)_i|} ;\ (d_k)_i < 0 \right\} = \min\left\{ \frac{(x_k - l)_i}{(D_k)_{ii}^{\gamma} (\hat{g}_k)_i} ;\ (g_k)_i > 0,\ (x_k)_i > l_i \right\} \\ &\ge \min\left\{ \frac{(x_k - l)_i}{\min\{\kappa_D, (x_k - l)_i\}^{\gamma} \chi_{\text{AS}}(x_k)} ;\ (g_k)_i > 0,\ (x_k)_i > l_i \right\} \ge \frac{\kappa_D^{1-\gamma}}{\chi_{\text{AS}}(x_k)}. \end{aligned} \tag{6.4}$$

In the same way, the stepsize t_3 admitted by the upper bounds of the set $X - x_k$ can be estimated:

$$t_3 = \min\left\{ \frac{(u - x_k)_i}{(d_k)_i} ;\ (d_k)_i > 0 \right\} \ge \frac{\kappa_D^{1-\gamma}}{\chi_{\text{AS}}(x_k)}.$$

In the case $M_k d_k = 0$ we set $t_4 = +\infty$. Otherwise, the function $q_k(t d_k)$, $t \ge 0$, attains its global minimum at $t = t_4$, where

$$t_4 = \frac{-g_k^T d_k}{\|M_k d_k\|^2} = \frac{\|\hat{g}_k\|^2}{\|M_k d_k\|^2} \ge \frac{\|\hat{g}_k\|^2}{\|M_k\|^2 \|D_k^{\gamma}\|^2 \|\hat{g}_k\|^2} = \frac{1}{\|M_k\|^2 \|D_k^{\gamma}\|^2} \ge \frac{1}{C_M^2 \kappa_D^{2\gamma}}. \tag{6.5}$$

We have $t^* = \min\{t_1, t_2, t_3, t_4\}$. If $t^* < t_4$, then $\|\hat{g}_k\|^2 > t^* \|M_k d_k\|^2$ and

$$\begin{aligned} q_k(t^* d_k) &= -t^* \|\hat{g}_k\|^2 + \frac{1}{2}(t^*)^2 \|M_k d_k\|^2 < -\frac{t^*}{2} \chi_{\text{AS}}(x_k)^2 \\ &= -\frac{\min\{t_1, t_2, t_3\}}{2} \chi_{\text{AS}}(x_k)^2. \end{aligned} \tag{6.6}$$

If, on the other hand, $t^* = t_4$, then

$$q_k(t^* d_k) = -\frac{t_4}{2} \chi_{\text{AS}}(x_k)^2. \tag{6.7}$$

The proof is completed by combining (6.2) and the estimates (6.3)–(6.7). ∎

Therefore, for any criticality measure satisfying (6.2), we can replace the decrease condition (3.10) by (6.1). Moreover, the Cauchy step s_k^C, which is easy to compute as could be seen in the proof of Lemma 6.4, is always an admissible trial step. Also, the optimal solution of the trust-region subproblem satisfies (6.1). Therefore, the global convergence theory of section 4 is applicable. Conditions that imply the uniform continuity of $\chi = \chi_{\text{AS}}$ needed in Theorem 4.9 were established in Lemma 6.1.

On the basis of χ_{AS} and the fraction of Cauchy decrease condition (6.1) it is possible to state implementations for Step 3.3 of Algorithm 3.6 such that (a) the reduction condition (3.10) is easy to check, and (b) condition (5.1) is satisfied.

There are several ways to do this.

(I) We begin with a universally applicable approach. Certainly, the function

$$\chi_{\text{I}}(x) = \min\{\chi_{\text{AS}}(x), (\beta_4 h(x))^{1/2}\}$$

with $\beta_4 \in (0,1)$ is a criticality measure. Now let χ be any criticality measure verifying

$$\beta_5 \chi(x) \leq \chi_{\mathrm{I}}(x), \quad x \in X, \tag{6.8}$$

for some $\beta_5 > 0$. Then (6.2) holds. Let Step 3.3 of Algorithm 3.6 be implemented as follows:

3.3. If $s_k = s_k^{\mathrm{P}}$ satisfies

$$q_k(s_k) \leq \max\{\alpha q_k(s_k^{\mathrm{C}}), -\beta_4 h(x_k)\}, \tag{6.9}$$

then set $s_k := s_k^{\mathrm{P}}$. Otherwise, compute a step s_k satisfying (3.9) and (6.9).

Note that condition (6.9) is a relaxation of (6.1). Obviously, (5.1) holds for arbitrary $\beta_3 > 0$. Furthermore, under the assumptions of Lemma 6.4 there is $\beta_2 > 0$ such that all computed steps s_k satisfy (3.10). In fact, if $q_k(s_k) \leq -\beta_4 h(x_k)$, (3.10) immediately holds for any $\beta_2 \leq 1$. On the other hand, if $q_k(s_k) \leq -\alpha q_k(s_k^{\mathrm{C}})$, then Lemma 6.4 is applicable. Finally, it is easily seen that the particular choice $\chi = \chi_{\mathrm{I}}$ is uniformly continuous if both χ_{AS} and H are uniformly continuous on Ω. Lemma 6.2 provides a sufficient condition for the uniform continuity of χ_{AS}.

(II) We now discuss a situation in which it suffices for χ to obey (6.2). Assume that $\min_{V \in \partial_B H(x)} \|V\| \leq C_V < \infty$ on X. Then with $V(x) = \operatorname{argmin}_{V \in \partial_B H(x)} \|V\|$

$$\chi_{\mathrm{AS}}(x) = \|D(x)^\gamma \nabla h(x)\| \leq \kappa_D^\gamma \|V(x)\| \|H(x)\| \leq \kappa_D^\gamma C_V \|H(x)\|$$

holds. Therefore, if $\beta_4 \in (0,1)$ is fixed and the criticality measure χ satisfies (6.2) with β_5 replaced by β_5', then (6.8) holds for $0 < \beta_5 \leq \beta_5' \min\{1, (\beta_4/2)^{1/2}/(\kappa_D^\gamma C_V)\}$. Therefore, (I) applies. We stress that in this scenario we can choose $\chi = \chi_{\mathrm{AS}}$.

(III) Finally, we state conditions under which we can implement Step 3.3 of Algorithm 3.6 by means of the fraction of Cauchy decrease condition:

3.3. If $s_k = s_k^{\mathrm{P}}$ satisfies (6.1) then set $s_k := s_k^{\mathrm{P}}$. Otherwise, compute a step s_k satisfying (3.9) and (6.1).

Let the criticality measure χ satisfy (6.2). Assume that the sequences $(\|M_k\|)$ and $(\|M_k^{-1}\|)$ are bounded and that

$$\|M_k^T H(x_k) - g_k\| = o(\|H(x_k)\|) \quad (\text{as } k \to \infty).$$

Note that this holds true if, e.g., $M_k \in \partial H(x_k)$. By Lemma 6.4 there is $\beta_2 > 0$ such that (3.10) holds for all computed steps. Under the above assumptions we obtain

$$\begin{aligned}
q_k(s_k^{\mathrm{C}}) &\geq q_k(-(M_k^T M_k)^{-1} g_k) = -\frac{1}{2} g_k^T (M_k^T M_k)^{-1} g_k \\
&= -h(x_k) + \frac{1}{2}(M_k^T H(x_k) - g_k)^T (M_k^T M_k)^{-1} (M_k^T H(x_k) + g_k) \\
&= -h(x_k) - \frac{1}{2}(M_k^T H(x_k) - g_k)^T (M_k^T M_k)^{-1} \left((M_k^T H(x_k) - g_k) - 2 M_k^T H(x_k) \right) \\
&= -h(x_k) + o(\|H(x_k)\|^2) = -h(x_k) + o(h(x_k)).
\end{aligned}$$

Therefore, for arbitrary $\beta_4 \in (\alpha, 1)$ we can find $\beta_3 > 0$ such that for all k satisfying the left-hand side of (5.1) holds

$$q_k(s_k^{\mathrm{P}}) = -\mathrm{pred}_k(s_k^{\mathrm{P}}) \leq -\beta_4 h(x_k) \leq \alpha q_k(s_k^{\mathrm{C}}),$$

which implies that condition (5.1) is satisfied.

In all three scenarios the implementation of Step 3.3 yields a special case of Algorithm 3.6. Moreover, all global and local convergence results are applicable.

7. Application to nonlinear MCPs. In the introduction we showed how the nonlinear MCP (1.3) can be converted to an equivalent problem having the form (1.1). The reformulation is obtained by applying an NCP-function $\phi : \mathbb{R}^2 \to \mathbb{R}$, i.e., a function satisfying (1.5), to the pairs of components x_i and $F_i(x)$, $1 \le i \le m$, to obtain the operator H as defined in (1.4).

We will assume the following.

(A3) The function $F : \mathbb{R}^n \supset U \to \mathbb{R}^n$ is Lipschitz continuously differentiable.

(A4) ϕ is an NCP-function, i.e., satisfies (1.5), and is continuously differentiable on $\mathbb{R}^2 \setminus \phi^{-1}(0)$. Moreover, $\phi \in S^1(\mathbb{R}^2, \mathbb{R})$. For all $a, b \in \mathbb{R}$ with $\phi(a, b) = 0$ and all $v \in \partial\phi(a, b)$ holds:

$$\begin{aligned} v \ge 0,\ v \ne 0,& \\ v_1 = 0 \ \text{ if } a > 0 \text{ and } b = 0,& \\ v_2 = 0 \ \text{ if } a = 0 \text{ and } b > 0.& \end{aligned}$$

Assumption (A4) holds, e.g., for the Fischer–Burmeister function and for the penalized Fischer–Burmeister function, which is used in our numerical tests in section 8.

LEMMA 7.1. *Let the assumptions* (A3) *and* (A4) *hold. Then the function H defined in* (1.4) *satisfies* (A1) *with $p = 1$ and* (A2). *Moreover, for all $x \in U$ and all $V \in \partial H(x)$ holds with appropriate diagonal matrices $D_a, D_b \in \mathbb{R}^{n\times n}$:*

$$V = D_a + D_b F'(x),$$

where $(D_a)_{ii} = 0$ and $(D_b)_{ii} = 1$ for $i = m + 1, \dots, n$.

If x solves the MCP (1.3), *then $D_a + D_b$ is positive definite and for all $i = 1, \dots, m$ holds*

$$\begin{aligned} (D_a)_{ii} = 0 \quad &\text{if } x_i > 0 \text{ and } F_i(x) = 0, \\ (D_b)_{ii} = 0 \quad &\text{if } x_i = 0 \text{ and } F_i(x) > 0. \end{aligned}$$

Proof. The functions ϕ and F are 1-order semismooth by (A3) and (A4). Therefore, H is a composite of 1-order semismooth functions and thus, by Proposition 2.8, is 1-order semismooth itself. Thus, (A1) holds with $p = 1$.

The assertions on the lower $n - m$ rows of D_a, D_b, and V, respectively, are obvious. Now let $1 \le i \le m$ be fixed and $x \in U$ be arbitrary such that $H_i(x) \ne 0$. Then of course $\phi(x_i, F_i(x)) \ne 0$ and thus, by (A4), ϕ is continuously differentiable in a neighborhood of $(x_i, F_i(x))$. Therefore, using (A4), H_i is C^1 in a neighborhood of x. This implies (A2).

Now let $x \in U$ be arbitrary. By [6, Prop. 2.6.2.(e)] holds

$$\partial H(x) \subset \partial H_1(x) \times \cdots \times \partial H_n(x).$$

Furthermore, setting $f_i(x) = (x_i, F_i(x))^T$, we have by [6, Thm. 2.6.6]

$$\begin{aligned} \partial H_i(x) &= \operatorname{conv}\left(\partial\phi(f_i(x))\partial f_i(x)\right) = \partial\phi(f_i(x))\nabla f_i(x)^T \\ &= \left\{ v_1 e_i^T + v_2 \nabla F_i(x)^T \,;\ v \in \partial\phi(x_i, F_i(x)) \right\}. \end{aligned}$$

If x is a solution of the MCP, then $\phi(x_i, F_i(x)) = 0$ for all $i \le m$. Now the remaining assertions follow from (A4). ∎

We now introduce the notion of strong regularity. It plays an important role in the stability analysis of solutions to the MCP. For a comprehensive discussion of strong regularity see [32, 44].

DEFINITION 7.2 (see [44]). *A solution $\bar{x}$ of the MCP* (1.3) *is called* strongly regular *if there exist neighborhoods U_x of $\bar{x}$ and $U_y \subset \mathbb{R}^n$ of 0 such that for all $y \in U_y$ the perturbed linearized MCP obtained by replacing F in by* (1.3) *by $F^y(x) = F(\bar{x}) + F'(\bar{x})(x - \bar{x}) + y$, admits exactly one solution $x(y)$ satisfying $x(y) \in U_x$, and, moreover, the function $x(y)$ is Lipschitz continuous on U_y.* ∎

The following lemma gives an equivalent algebraic definition of strong regularity.

LEMMA 7.3 (see [13, Thm. 3.4]). *The solution $\bar{x}$ of the MCP* (1.3) *is strongly regular if and only if the submatrix $F'(\bar{x})_{\bar{I}\bar{I}} = \left(\frac{\partial F_i}{\partial x_j}(\bar{x})\right)_{i,j\in\bar{I}}$ is nonsingular and the Schur-complement*

$$F'(\bar{x})_{\bar{N}\bar{N}} - F'(\bar{x})_{\bar{N}\bar{I}} F'(\bar{x})^{-1}_{\bar{I}\bar{I}} F'(\bar{x})_{\bar{I}\bar{N}}$$

is a P-matrix, where $\bar{I} \stackrel{\text{def}}{=} \{i\,;\ i > m \text{ or } \bar{x}_i > 0\}$ and $\bar{N} \stackrel{\text{def}}{=} \{i \le m\,;\ \bar{x}_i = 0,\ F_i(\bar{x}) = 0\}$.

Using the Lemmas 7.1 and 7.3, the proof of [15, Thm. 1] can be easily modified to show the following result.

THEOREM 7.4. *If $\bar{x}$ is a strongly regular solution of the MCP* (1.3)*, then all elements of $\partial H(\bar{x})$ are nonsingular. In particular, $\bar{x}$ is BD-regular for H.*

8. Numerical results. In this section we present numerical results for Algorithm 3.6. As test problems we use a subset of the MCPLIB [12] collection of mixed complementarity problems. The problems in MCPLIB are represented as box-constrained variational inequality problems VIP(F, X), $X = [l, u]$.

Find $x \in X$ such that

$$F(x)^T(y - x) \ge 0 \quad \forall y \in X$$

with $F : U \to \mathbb{R}^n$ defined on $U \supset X$, and bounds $l_i \in \mathbb{R} \cup \{-\infty\}$, $u_i \in \mathbb{R} \cup \{+\infty\}$. Note that the MCP in the form (1.3) is equivalent to VIP(F, X) with $X = [0, \infty)^m \times \mathbb{R}^{n-m}$.

We selected all MCPs of size $n \le 150$ that are accessible from within MATLAB (some of the problems are available only as GAMS files) and that have at most one bound per variable, i.e., $u_i - l_i = +\infty$ for all i. It is obvious that these problems can be expressed in the form (1.3). The algorithm is applied to the reformulation (1.4) of the MCP. For the sake of comparison, test versions of four different algorithms were implemented in MATLAB. The difference between these methods consists of the choice of the NCP-function and the choice of the reformulation (constrained/unconstrained). As NCP-function we use the penalized Fischer–Burmeister function

$$\phi_\nu(a, b) \stackrel{\text{def}}{=} \nu\phi_{\text{FB}}(a, b) + (1 - \nu)a_+b_+,$$

where $0 < \nu < 1$ and $t_+ \stackrel{\text{def}}{=} \max\{0, t\}$. This function was recently introduced by Chen, Chen, and Kanzow [5]. Numerical tests indicate that the penalized Fischer–Burmeister function usually leads to better performance than the Fischer–Burmeister function [5]. The four algorithms are as follows:

[ALG1:] Algorithm 3.6 applied to (1.1) with H as in (1.4) and $\phi = \phi_\nu$ with $\nu = 0.7$.

[ALG2:] The function H is the same as in ALG1. However, we apply Algorithm 3.6 to the *unconstrained* reformulation (1.6).

[ALG3:] As ALG1, but with $\nu = 0.95$.

[ALG4:] The function H is the same as in ALG3. However, we apply Algorithm 3.6 to the *unconstrained* reformulation (1.6).

The MATLAB interface to MCPLIB provides an initialization routine that returns an initial point x_0. Moreover, functions for the computation of F and its Jacobian F' are

available. Therefore, it is easy to compute exact (up to roundoff errors) elements of the B-subdifferential of H; cf. [10]. Hence, we may assume that the matrices M_k are elements of $\partial_B H(x_k)$ up to machine accuracy. If M_k is too ill-conditioned, we add a small multiple of the identity to $M_k^T M_k$. We implemented Algorithm 3.6 on the basis of the fraction of Cauchy decrease condition (6.1) with Step 3.3 as described in (III) at the end of section 6. If $s_k = s_k^{\mathrm{p}}$ fails to satisfy (6.1), we compute s_k by solving the trust-region subproblem (3.8) exactly. Hereby, the QP-solver BPMPD [33] is used. The parameters in Algorithm 3.6 were chosen as follows:

- Termination criterion:
  Successful termination if

$$\max\Big(\{\min\{x_i, F_i(x)\}|\,;\ 1 \le i \le m\} \cup \{|F_i(x)|\,;\ m < i \le n\}\Big) \le 10^{-6}.$$

  Note that the left-hand side vanishes if and only if x solves the MCP (1.3).
  Unsuccessful termination if $\Delta_k \le 10^{-10}$ or $k \ge 200$.
- Trust-region parameters: $\Delta_{\min} = 1$, $\Delta_0 = 100$, $\eta_1 = 10^{-4}$, $\eta_2 = 0.75$, $\gamma_1 = 1/2$, $\gamma_2 = 2$,

$$\Delta_{k+1} = \begin{cases} \gamma_1 \Delta_k & \text{if } \rho_k \le \eta_1, \\ \max\{\Delta_{\min}, \Delta_k\} & \text{if } \eta_1 < \rho_k < \eta_2, \\ \max\{\Delta_{\min}, \gamma_2 \Delta_k\} & \text{if } \rho_k \ge \eta_2. \end{cases}$$

- Nonmonotonicity parameters: $m = 4$, $\lambda = 0.01$, and λ_{kr} as in (3.14).
- Affine-scaling parameter: $\gamma = 1$, $\kappa_D = 1$.
- Initial point: For the initial point computation we tried two variants. For each class of algorithms (constrained/unconstrained) we chose the one that yields the best overall results. The first variant projects $\hat{x}_0$, the initial point returned by the initialization routine, onto X to obtain x_0. The second variant chooses $x_0 = P_{[l+0.1, u-0.1]}(\hat{x}_0)$. (Note that we consider only problems with at most one bound per variable.) It turns out that the first choice is advantageous for the unconstrained versions (especially for ALG4, ALG2 behaves well for both variants), whereas the second choice is the preferable one for the constrained methods ALG1/ALG3. We think that interior starting points enable the constrained algorithms to identify the correct active constraints more efficiently than starting points close to the boundary. We stress, however, that the improvements achieved by the interior point modification are not significant.

The numerical results are shown in Table 8.1. For each of the four algorithms the number of major iterations (Maj. it), i.e., the value of $(i + 1)$ at termination, the number of iterations (It), i.e., the value of k at successful termination, and the number of QP-subproblems that had to be solved (QPs) are reported. Note that the number of evaluations of F equals (It + 1) and that the Jacobian of F is evaluated once per major iteration. The entry "−" is used to indicate that the algorithm terminated unsuccessfully. At the bottom we display the sum (Σ) over all column entries corresponding to problems that were successfully solved by all four methods. Moreover, we give an overall ranking of the algorithms for each of the three categories "Maj. It," "It," and "QPs." Hereby, in every category and for every problem we compute for each algorithm a rank between 1 and 5 as follows.

TABLE 8.1
Numerical results.

	Maj. it				It				QPs			
ALG:	1	2	3	4	1	2	3	4	1	2	3	4
badfree	4	5	4	5	4	5	4	5	0	0	0	0
bertsekas	19	19	20	20	43	48	40	39	27	31	21	23
billups	–	–	–	–	–	–	–	–	–	–	–	–
colvdual	9	12	21	12	9	12	30	14	0	0	9	0
colvnlp	8	12	11	12	8	12	12	14	0	0	0	0
cycle	5	5	7	7	5	5	7	7	0	0	0	0
degen	4	5	4	5	4	5	4	5	0	0	0	0
duopoly	11	–	12	–	11	–	12	–	9	–	10	–
ehl_k40	11	11	24	–	11	11	45	–	0	0	5	–
ehl_k60	11	13	32	31	11	13	57	65	2	0	8	12
ehl_k80	12	12	25	–	12	12	46	–	1	0	4	–
ehl_kost	12	13	–	–	31	13	–	–	22	0	–	–
freebert	8	10	12	22	8	17	23	83	0	0	0	66
games	12	–	–	20	22	–	–	40	13	–	–	21
hanskoop	9	10	11	10	24	19	31	19	6	2	9	2
hydroc06	12	12	7	7	22	19	7	7	8	5	1	1
hydroc20	12	12	11	11	13	15	12	12	1	2	0	0
jel	6	6	7	7	6	6	7	7	0	0	0	0
josephy	5	5	6	7	5	11	6	7	1	0	1	0
kojshin	6	6	7	6	12	12	7	11	0	0	0	0
mathinum	7	7	8	4	14	14	18	4	0	0	1	0
mathisum	5	6	5	5	5	6	12	12	0	0	0	0
methan08	4	4	4	4	4	4	4	4	0	0	0	0
nash	8	8	8	8	8	8	8	8	0	0	0	0
ne-hard	–	–	–	–	–	–	–	–	–	–	–	–
pgvon105	–	–	–	–	–	–	–	–	–	–	–	–
pgvon106	–	–	–	–	–	–	–	–	–	–	–	–
powell	7	9	7	9	7	16	7	16	0	0	0	0
powell_mcp	6	6	6	6	6	6	6	6	0	0	0	0
qp	4	5	3	5	4	5	3	5	0	0	0	0
scarfanum	9	12	9	11	9	12	15	19	2	0	1	1
scarfasum	8	9	7	7	8	9	18	18	1	0	0	0
scarfbnum	12	15	18	20	12	21	18	31	1	3	1	5
scarfbsum	11	15	14	14	11	22	14	24	4	8	4	7
shubik	16	25	25	32	31	62	69	82	23	51	59	70
simple-ex	–	–	–	–	–	–	–	–	–	–	–	–
simple-red	11	11	10	10	11	11	10	10	0	0	0	0
sppe	8	9	8	8	8	9	8	8	0	0	0	0
tinloi	9	7	8	7	9	7	27	7	0	0	3	0
tobin	8	10	11	14	11	16	15	17	0	0	3	0
Σ	251	290	311	326	332	427	499	566	76	102	121	187
Rank 1	28	11	14	12	25	10	12	11	25	27	21	23
Rank 2	2	9	4	4	3	9	7	2	3	1	2	3
Rank 3	4	8	10	10	6	10	10	10	4	2	5	1
Rank 4	1	5	5	5	1	4	4	8	3	3	5	4
Rank 5 (fails)	5	7	7	9	5	7	7	9	5	7	7	9

Let *cat* denote the category, *prob* the problem, and i_k the entries for ALGk, $k = 1, \ldots, 4$, where $i_k = +\infty$ for "–"-entries. We define the rank achieved by algorithm ALGk in category *cat* for problem *prob* by

$$\mathit{rank}(\text{ALG}k,\mathit{cat},\mathit{prob}) \stackrel{\text{def}}{=} \begin{cases} |\{j\,;\; i_j < i_k\}| + 1 & \text{if } i_k < +\infty, \\ 5 & \text{if } i_k = +\infty. \end{cases}$$

For example, in the category "Maj.It" and for problem "ehl_k40" we have $(i_1, \ldots, i_4) = (11, 11, 24, +\infty)$ and thus $\mathit{rank}(\text{ALG}k,\text{Maj.It},\text{ehl\_k40})_{1 \le k \le 4} = (1, 1, 3, 5)$.

We make a number of observations.

- Apparently, the algorithms ALG1 and ALG3, which use the proposed *box-constrained* reformulations (1.1) and (1.2), are more robust than their counterparts ALG2 and ALG4, which are based on the *unconstrained* reformulations (1.6) and (1.7).

- Comparing ALG1 with ALG3 and ALG2 with ALG4 shows that the usage of the parameter value $\nu = 0.7$ instead of $\nu = 0.95$ in the penalized Fischer–Burmeister function leads to an improvement in all three categories (major iterations, iterations, number of QPs). We leave the reasons for the positive effect of a stronger weighting of the term a_+b_+ as a topic for future research.
- ALG1, which combines the proposed Algorithm 3.6, the constrained reformulation, and the penalized Fischer–Burmeister function with the, compared to the choice $\nu = 0.95$ in [5], reduced value of $\nu = 0.7$ is the most robust and efficient method in the test. It fails to solve only 5 problems, followed by ALG2, ALG3, and ALG4 with 7, 7, and 9 fails. In addition, it needs less iterations—and thus function evaluations—and significantly less calls of the QP solver than the other three algorithms.
- To demonstrate the importance of nonmonotone trust-region techniques in this context, we ran a monotone version of our algorithm on the test set. If in algorithm ALG1 we set $m = 1$, i.e., $\text{rared}_k(s) = \text{ared}_k(s)$, then the resulting algorithm fails for 9 problems compared to only 5 fails with $m = 4$. This shows that the proposed nonmonotonicity technique increases the robustness of the algorithm considerably.

Overall, our numerical tests prove the viability and efficiency of the presented trust-region approach. In particular, we see that the constrained method, which is advantageous in its own right because it generates feasible iterates, not only performs comparably to the unconstrained method but even appears to be superior.

9. Conclusions. We have introduced a class of nonmonotone trust region methods for box-constrained semismooth equations. For these algorithms a comprehensive global convergence theory was established. The method remains feasible with respect to the box-constraints and is based on a reformulation as a simply constrained smooth minimization problem. A Newton-like method with projection was proposed for the computation of trial steps that converges under a Dennis–Moré-type condition locally q-superlinearly/quadratically to a BD-regular solution. We showed that close to the solution the trust-region algorithm turns into this Newton-like method. The convergence analysis was carried out on the general basis of a criticality measure and a sufficient decrease condition. As a concrete implementation we discussed a fraction of Cauchy decrease condition in which the negative affinely scaled gradient is used as Cauchy direction.

The developed algorithm was applied to the solution of the nonlinear mixed complementarity problem. To this end, an NCP-function was used to convert the MCP into an equivalent bound-constrained semismooth equation. Under appropriate assumptions on the NCP-function the resulting equation is 1-order semismooth and all strongly regular solutions are BD-regular. Therefore, the global and local convergence results for the developed algorithm are applicable for this broad class of problems. The numerical results presented for a subset of the MCPLIB test set show that—even in the case where H does not have a zero outside of X—incorporating the a priori knowledge $x \in X$ into the algorithm leads to more robustness and efficiency. This confirms the relevance of the problem class (1.1) and the need of algorithms for its solution together with the corresponding theory.

Acknowledgments. I would like to thank my brother Stefan, Technische Universität München, for several helpful discussions. Moreover, I am grateful to Csaba Mészáros, Computer and Automation Research Center, Budapest, for making available his QP-code BPMPD. Finally, I would like to thank the two referees for their

constructive comments that helped to improve the manuscript. I am also thankful to the associate editor, Danny Ralph, for handling the paper and for his helpful suggestions.

REFERENCES

[1] S. C. Billups, *Algorithms for Complementarity Problems and Generalized Equations*, Ph.D. thesis, Computer Sciences Department, University of Wisconsin, Madison, WI, 1995.
[2] S. C. Billups, S. P. Dirkse, and M. C. Ferris, *A comparison of large scale mixed complementarity problem solvers*, Comput. Optim. Appl., 7 (1997), pp. 3–25.
[3] S. C. Billups and M. C. Ferris, *QPCOMP: A quadratic program based solver for mixed complementarity problems*, Math. Programming, 76 (1997), pp. 533–562.
[4] J. V. Burke, J. J. Moré, and G. Toraldo, *Convergence properties of trust region methods for linear and convex constraints*, Math. Programming, 47 (1990), pp. 305–336.
[5] 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.
[6] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.
[7] T. F. Coleman and Y. Li, *An interior trust region approach for nonlinear minimization subject to bounds*, SIAM J. Optim., 6 (1996), pp. 418–445.
[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 [9].
[9] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *Correction to the paper on global convergence of a class of trust region algorithms for optimization with simple bounds*, SIAM J. Numer. Anal., 26 (1989), pp. 764–767.
[10] 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.
[11] S. P. Dirkse and M. C. Ferris, *The PATH solver: A non-monotone stabilization scheme for mixed comple-mentarity problems*, Optim. Methods Softw., 5 (1995), pp. 123–156.
[12] S. P. Dirkse and M. C. Ferris, *MCPLIB: A collection of nonlinear mixed complementarity problems*, Optim. Methods Softw., 5 (1995), pp. 319–345.
[13] F. Facchinei and C. Kanzow, *A nonsmooth inexact Newton method for the solution of large-scale nonlinear complementarity problems*, Math. Programming, 76 (1997), pp. 493–512.
[14] M. C. Ferris and C. Kanzow, *Complementarity and Related Problems: A Survey*, Mathematical Programming Technical Report 98-17, Computer Sciences Department, University of Wisconsin, Madison, WI, 1998.
[15] M. C. Ferris, C. Kanzow, and T. S. Munson, *Feasible descent algorithms for mixed complementarity problems*, Math. Programming, 86 (1999), pp. 475–497.
[16] M. C. Ferris and J.-S. Pang, *Engineering and economic applications of complementarity problems*, SIAM Rev., 39 (1997), pp. 669–713.
[17] A. Fischer, *A special Newton-type optimization method*, Optimization, 24 (1992), pp. 269–284.
[18] A. Fischer, *Solution of monotone complementarity problems with locally Lipschitzian functions*, Math. Programming, 76 (1997), pp. 513–532.
[19] A. Friedlander, J. M. Martínez, and A. Santos, *A new trust region algorithm for bound constrained minimization*, Appl. Math. Optim., 30 (1994), pp. 235–266.
[20] S. A. Gabriel and J.-S. Pang, *A trust region method for constrained nonsmooth equations*, in Large Scale Optimization: State of the Art, W. W. Hager, D. W. Hearn, and P. M. Pardalos, eds., Kluwer Academic Publishers, Boston, MA, 1994, pp. 159–186.
[21] 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.
[22] 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.
[23] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms. Part* I: *Fundamentals*, Springer, Berlin, 1993.
[24] H. Jiang, M. Fukushima, L. Qi, and D. Sun, *A trust region method for solving generalized complementarity problems*, SIAM J. Optim., 8 (1998), pp. 140–157.
[25] H. Jiang and L. Qi, *A new nonsmooth equations approach to nonlinear complementarity problems*, SIAM J. Control Optim., 35 (1997), pp. 178–193.
[26] 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, Dordrecht, 1999, pp. 181–209.

[27] C. Kanzow and H.-D. Qi, *A QP-free constrained Newton-type method for variational inequality problems*, Math. Programming, 85 (1999), pp. 81–106.

[28] C. Kanzow and M. Zupke, *Inexact trust-region methods for nonlinear complementarity problems*, in Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods, M. Fukushima and L. Qi, eds., Kluwer Academic Publishers, Dordrecht, 1999, pp. 211–233.

[29] B. Kummer, *Newton's method for non-differentiable functions*, in Advances in Mathematical Optimization, J. Guddat et al., eds., Akademie-Verlag, Berlin, 1988, pp. 114–125.

[30] B. Kummer, *Newton's method based on generalized derivatives for nonsmooth functions: Convergence analysis*, in Advances in Mathematical Optimization, Lecture Notes in Econom. and Math. Systems 382, W. Oettli and D. Pallaschke, eds., Springer-Verlag, Heidelberg, 1992, pp. 171–194.

[31] C.-J. Lin and J. J. Moré, *Newton's method for large bound-constrained optimization problems*, SIAM J. Optim., 9 (1999), pp. 1100–1127.

[32] J. Liu, *Strong stability in variational inequalities*, SIAM J. Control Optim., 33 (1995), pp. 725–749.

[33] Cs. Mészáros, *The BPMPD Interior Point Solver for Convex Quadratic Problems*, Working paper WP 98–8, Computer and Automation Research Institute, Hungarian Academy of Sciences, Budapest, Hungary, 1998.

[34] R. Mifflin, *Semismooth and semiconvex functions in constrained optimization*, SIAM J. Control Optim., 15 (1977), pp. 959–972.

[35] J. J. Moré, *Global methods for nonlinear complementarity problems*, Math. Oper. Res., 21 (1996), pp. 589–614.

[36] J.-S. Pang, *Newton's method for B-differentiable equations*, Math. Oper. Res., 15 (1990), pp. 311–341.

[37] J.-S. Pang, *A B-differentiable equation based, globally and locally quadratically convergent algorithm for nonlinear programs, complementarity and variational inequality problems*, Math. Programming, 51 (1991), pp. 101–132.

[38] J.-S. Pang and S. A. Gabriel, *NE/SQP: A robust algorithm for the nonlinear complementarity problem*, Math. Programming, 60 (1993), pp. 295–337.

[39] J.-S. Pang and L. Qi, *Nonsmooth equations: Motivation and algorithms*, SIAM J. Optim., 3 (1993), pp. 443–465.

[40] L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res., 18 (1993), pp. 227–244.

[41] L. Qi, *Trust region algorithms for solving nonsmooth equations*, SIAM J. Optim., 5 (1995), pp. 219–230.

[42] L. Qi, *Regular pseudo-smooth NCP and BVIP functions and globally and quadratically convergent generalized Newton methods for complementarity and variational inequality problems*, Math. Oper. Res., 24 (1999), pp. 440–471.

[43] L. Qi and J. Sun, *A nonsmooth version of Newton's method*, Math. Programming, 58 (1993), pp. 353–367.

[44] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

[45] S. M. Robinson, *Newton's method for a class of nonsmooth functions*, Set-Valued Anal., 2 (1994), pp. 291–305.

[46] D. Sun and L. Qi, *On NCP-functions*, Comput. Optim. Appl., 13 (1999), pp. 201–220.

[47] Ph. L. Toint, *Global convergence of a class of trust-region methods for nonconvex minimization in Hilbert space*, IMA J. Numer. Anal., 8 (1988), pp. 231–252.

[48] Ph. L. Toint, *Non-monotone trust-region algorithms for nonlinear optimization subject to convex constraints*, Math. Programming, 77 (1997), pp. 69–94.

[49] M. Ulbrich, S. Ulbrich, and M. Heinkenschloss, *Global convergence of trust-region interior-point algorithms for infinite-dimensional nonconvex minimization subject to pointwise bounds*, SIAM J. Control Optim., 37 (1999), pp. 731–764.

[50] B. Xiao and P. T. Harker, *A nonsmooth Newton method for variational inequalities:* I: *Theory*, Math. Programming, 65 (1994), pp. 151–194.

[51] B. Xiao and P. T. Harker, *A nonsmooth Newton method for variational inequalities:* II: *Numerical results*, Math. Programming, 65 (1994), pp. 195–216.

[52] W. P. Ziemer, *Weakly Differentiable Functions. Sobolev Spaces and Functions of Bounded Variation*, Springer-Verlag, Berlin, 1989.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 918–936

CONVERGENCE PROPERTIES OF A REGULARIZATION SCHEME FOR MATHEMATICAL PROGRAMS WITH COMPLEMENTARITY CONSTRAINTS*

STEFAN SCHOLTES†

Abstract. We study the convergence behavior of a sequence of stationary points of a parametric NLP which regularizes a mathematical program with equilibrium constraints (MPEC) in the form of complementarity conditions. Accumulation points are feasible points of the MPEC; they are C-stationary if the MPEC linear independence constraint qualification holds; they are M-stationary if, in addition, an approaching subsequence satisfies second order necessary conditions, and they are B-stationary if, in addition, an upper level strict complementarity condition holds. These results complement recent results of Fukushima and Pang [*Convergence of a smoothing continuation method for mathematical programs with equilibrium constraints,* in Ill-posed Variational Problems and Regularization Techniques, Springer-Verlag, New York, 1999]. We further show that every local minimizer of the MPEC which satisfies the linear independence, upper level strict complementarity, and a second order optimality condition can be embedded into a locally unique piecewise smooth curve of local minimizers of the parametric NLP.

Key words. complementarity constraints, regularization, B-stationarity

AMS subject classifications. 90C30, 90C33

PII. S1052623499361233

1. Introduction.

We consider parametric nonlinear programs NLP(t) of the form

$$\begin{array}{rl} \min\limits_{z} & f(z) \\ \text{subject to (s.t.)} & g(z) \le 0, \\ & h(z) = 0, \\ & G(z) \ge 0, \\ & H(z) \ge 0, \\ & G_i(z)H_i(z) \le t, \quad i = 1, \ldots, m, \end{array} \tag{1}$$

with twice continuously differentiable data $(f, g, h, G, H) : \mathbb{R}^n \to \mathbb{R}^{1+p+q+m+m}$. Our aim is to find a local minimizer of NLP(0) which is a mathematical program with equilibrium constraints (MPEC) in complementarity form [10, 11]. A direct application of a standard NLP code to NLP(0) is problematic since NLP(0) is inherently ill-posed. The main difficulty is that no feasible point satisfies the inequalities strictly. This implies that the Mangasarian–Fromovitz constraint qualification, a vital condition for the stability of the feasible set, is violated at every feasible point of NLP(0). As a consequence, NLP codes may have difficulties in finding a feasible or nearly feasible point. Moreover, if the iterates of an SQP method are feasible for NLP(0), then the QP subproblems are degenerate, too, in the sense that their feasible set has no strictly feasible point. This degeneracy or near degeneracy as the iterates approach a feasible point may cause problems for QP solvers; cf., e.g., [4]. Finally, even if an NLP code produces a feasible point, at present we do not know anything about its stationarity

*Received by the editors September 1, 1999; accepted for publication (in revised form) September 29, 2000; published electronically March 28, 2001.
http://www.siam.org/journals/siopt/11-4/36123.html
†Judge Institute of Management Studies and Department of Engineering, University of Cambridge, Cambridge CB2 1AG, England (s.scholtes@jims.cam.ac.uk).

properties since the standard NLP convergence analyses rely on conditions at limit points which comprise the Mangasarian–Fromovitz condition.

The situation is more favorable if the regularization parameter t is positive. Typically, the relaxed programs NLP(t) satisfy constraint qualifications and should therefore be easier to solve, in particular if care is taken to ensure that the inequalities $G_i(z) \geq 0$ and $H_i(z) \geq 0$ are treated as inactive whenever $H_i(z)G_i(z) \leq t$ is considered active and that the gradients of active functions $G_i(z)H_i(z)$ are properly scaled in subproblems if G_i and H_i are both close to zero. An immediate question is whether solutions of NLP(t) for small positive t approximate solutions of NLP(0). To investigate this problem we consider a sequence of positive numbers t_n converging to zero and a corresponding sequence of stationary points z_n of NLP(t_n). While every accumulation point of the sequence is obviously feasible for NLP(0), it not clear if it is stationary for NLP(0) in one sense or another. The aim of this paper is to characterize conditions which guarantee that accumulation points of the sequence $\{z_n\}$ are B-stationary points of NLP(0) as defined in [14]. This complements results obtained recently by Fukushima and Pang [5] for a related smoothing method. Our convergence assumptions are similar but our approach comprises a discussion of situations where some of the assumptions for convergence to B-stationary points fail. We further provide conditions which guarantee that a local minimizer of NLP(0) is a limit point of a curve of stationary points of the programs NLP(t) as t tends to zero.

Before we present the technical analysis, we provide some intuition for our line of arguments by discussing the simple problem of minimizing the Euclidean distance from a point $(a, b) \in \mathbb{R}^2$ to the boundary of the positive orthant. The corresponding parametric program NLP(t) is of the form

$$\begin{array}{ll} \min & \frac{1}{2}[(x-a)^2 + (y-b)^2] \\ \text{s.t.} & x \geq 0, \\ & y \geq 0, \\ & xy \leq t. \end{array} \tag{2}$$

If $a \leq 0$ or $b \leq 0$, then the unique solution for every $t \geq 0$ is the Euclidean projection of (a, b) onto the positive orthant. There are no other local minimizers or stationary points. If $(a, b) > 0$, then there are two local minimizers, $(a, 0)$ and $(0, b)$, of NLP(0) which are both B-stationary points in the sense of [14]. NLP(t), however, has three stationary points on the curve $xy = t$ for every sufficiently small t, two local minima and between them a stationary point with a second order descent direction tangential to the manifold $xy = t$. If, e.g., $(a, b) = (1, 1)$, then the two local minima are $(1/2, 1/2) \pm (\sqrt{1/4 - t})(-1, 1)$ and the other stationary point is $(\sqrt{t}, \sqrt{t})$. While the two local minima converge to the minima of the MPEC, the other stationary point converges to the origin which is not a B-stationary point since there are two feasible first order descent directions $(1, 0)$ and $(0, 1)$. The geometry suggests that failing B-stationarity of a limit point may be connected to failing second order necessary conditions along an approaching path of stationary points $z(t)$ for NLP(t) as observed by Fukushima and Pang [5]. Indeed, the second derivative of the constraint function $(x, y) \to xy$ is negative definite on the tangent spaces of the equation $xy = t$ and appears, weighted with the unbounded multiplier $\delta(t) = (1 - \sqrt{t})/\sqrt{t}$, in the Hessian of the Lagrangian. However, care must be taken since the tangent spaces may approach the x- or y-axis where the second derivative of the constraint function vanishes. Elementary geometric insight shows that this will only be the case if either $a = 0$ or $b = 0$. Hence, if $(a, b) > (0, 0)$ and the solutions $(x(t), y(t))$ are guaranteed to

satisfy second order necessary conditions for NLP(t), then every accumulation point of a solution curve as $t \to 0$ is a B-stationary point which is equivalent to a local minimizer in this example.

2. Preliminaries: Linear independence, stationarity, and upper level strict complementarity. In this preliminary section we recall some notions from MPEC theory which we will use later in the text. For convenience we define the index sets

$$\begin{array}{rcl} I_g(z) & = & \{i \mid g_i(z) = 0\}, \\ I_h(z) & = & \{j \mid h_j(z) = 0\}, \\ I_G(z) & = & \{k \mid G_k(z) = 0\}, \\ I_H(z) & = & \{l \mid H_l(z) = 0\}, \\ I_{GH}(z,t) & = & \{m \mid G_m(z)H_m(z) = t\}. \end{array} \tag{3}$$

Throughout the paper we denote by ∇f the $m \times n$ Jacobian matrix of a smooth function $f : \mathbb{R}^n \to \mathbb{R}^m$. In particular, gradients of real-valued functions are row vectors. We will make use of the following linear independence constraint qualification for MPECs.

MPEC linear independence constraint qualification (MPEC-LICQ). *The MPEC-LICQ is said to hold at a feasible point $\bar{z}$ of the MPEC NLP(0) if the gradients*

$$\begin{array}{ll} \nabla g_i(\bar{z}), & i \in I_g(\bar{z}), \\ \nabla h_j(\bar{z}), & j \in I_h(\bar{z}), \\ \nabla G_k(\bar{z}), & k \in I_G(\bar{z}), \\ \nabla H_l(\bar{z}), & l \in I_H(\bar{z}), \end{array} \tag{4}$$

are linearly independent.

Notice that this definition differs from the standard definition of the linear independence constraint qualification in nonlinear programming (NLP-LICQ) since it discards the gradients of the constraints $G_i(z)H_i(z) \leq 0$ which, in the MPEC setting, merely serve a combinatorial purpose. Indeed, NLP(0) does not satisfy NLP-LICQ at any feasible point. The severity of the MPEC-LICQ condition has been investigated in [16], where it is argued that for a typical MPEC the condition is satisfied at every feasible point.

The following elementary lemma is crucial for our analysis. It states that the MPEC-LICQ condition carries over to the NLP-LICQ condition for the relaxed problems NLP(t) if $t > 0$ is sufficiently small. A similar result can be found in [5] in the context of a smoothing method.

LEMMA 2.1. *If MPEC-LICQ holds at the feasible point $\bar{z}$ of the MPEC NLP(0), then there exists a neighborhood U of $\bar{z}$ and a scalar $\bar{t} > 0$ such that for every $t \in (0, \bar{t})$ NLP-LICQ holds at every feasible point $z \in U$ of NLP(t).*

Proof. The lemma is an immediate consequence of the relations

$$\begin{array}{rcl} I_g(z) & \subseteq & I_g(\bar{z}), \\ I_h(z) & \subseteq & I_h(\bar{z}), \\ I_G(z) \cup I_H(z) \cup I_{GH}(z,t) & \subseteq & I_G(\bar{z}) \cup I_H(\bar{z}), \\ I_G(z) \cap I_{GH}(z,t) & = & \emptyset, \\ I_H(z) \cap I_{GH}(z,t) & = & \emptyset \end{array} \tag{5}$$

which hold for all z in a sufficiently small neighborhood U of $\bar{z}$ and all $t \in (0, \bar{t})$ for sufficiently small $\bar{t} > 0$. Indeed, for such t, the active gradients of NLP(t) at a feasible

point $z \in U$ are

$$\begin{array}{rl} \nabla g_i(z), & i \in I_g(z), \\ \nabla h_j(z), & j \in I_h(z), \\ \nabla G_k(z), & k \in I_G(z), \\ \nabla H_l(z), & l \in I_H(z), \\ G_m(z)\nabla H_m(z) + H_m(z)\nabla G_m(z), & m \in I_{GH}(z,t). \end{array}$$

In view of the MPEC-LICQ assumption and (5), the equation

$$\sum_{i\in I_g(z)} \lambda_i \nabla g_i(z) + \sum_{j\in I_h(z)} \mu_j \nabla h_j(z) + \sum_{k\in I_G(z)} \gamma_k \nabla G_k(z) + \sum_{l\in I_H(z)} \nu_l \nabla H_l(z) + \sum_{m\in I_{GH}(z,t)} (\delta_m G_m(z)\nabla H_m(z) + \delta_m H_m(z)\nabla G_m(z)) = 0$$

implies that $\lambda_i = \mu_j = \gamma_k = \nu_l = \delta_m G_m(z) = \delta_m H_m(z) = 0$. This proves the lemma since $G_m(z) > 0$ for every $m \in I_{GH}(z,t)$ and therefore $\delta_m G_m(z) = 0$ implies $\delta_m = 0$. □

We will use the standard definition of stationarity for NLP(t) if $t > 0$, i.e., z is a stationary point of NLP(t) if there exist NLP multipliers λ_i, μ_j, γ_k, ν_l, and δ_m such that

$$\begin{array}{rcl} \nabla f(z) + \sum_{i\in I_g(z)} \lambda_i \nabla g_i(z) + \sum_{j\in I_h(z)} \mu_j \nabla h_j(z) - \sum_{k\in I_G(z)} \gamma_k \nabla G_k(z) & & \\ - \sum_{l\in I_H(z)} \nu_l \nabla H_l(z) + \sum_{m\in I_{GH}(z,t)} \delta_m [H_m(z)\nabla G_m(z) + G_m(z)\nabla H_m(z)] & = & 0, \\ \lambda_i, \gamma_k, \nu_l, \delta_m & \geq & 0. \end{array} \tag{6}$$

We will find it convenient to distinguish various degrees of stationarity for $t = 0$. Following [14], a feasible point $\bar{z}$ of the MPEC NLP(0) is called *weakly stationary* if there exist MPEC multipliers $\bar{\lambda}_i \geq 0, \bar{\mu}_j, \bar{\gamma}_k, \bar{\nu}_l$ satisfying

$$\nabla f(\bar{z}) + \sum_{i\in I_g(\bar{z})} \bar{\lambda}_i \nabla g_i(\bar{z}) + \sum_{j\in I_h(\bar{z})} \bar{\mu}_j \nabla h_j(\bar{z}) - \sum_{k\in I_G(\bar{z})} \bar{\gamma}_k \nabla G_k(\bar{z}) - \sum_{l\in I_H(\bar{z})} \bar{\nu}_l \nabla H_l(\bar{z}) = 0. \tag{7}$$

We will use three stationarity concepts of increasing strength which we define by imposing additional sign constraints on the multipliers corresponding to the complementarity terms:

C-stationarity: $\bar{\gamma}_m \bar{\nu}_m \geq 0$ *for all* $m \in I_G(\bar{z}) \cap I_H(\bar{z})$.

M-stationarity: *for all* $m \in I_G(\bar{z}) \cap I_H(\bar{z})$, *either* $\bar{\gamma}_m, \bar{\nu}_m > 0$ *or* $\bar{\gamma}_m \bar{\nu}_m = 0$.

Strong stationarity: $\bar{\gamma}_m, \bar{\nu}_m \geq 0$ *for all* $m \in I_G(\bar{z}) \cap I_H(\bar{z})$.

The first two concepts arise from nonsmooth reformulations of MPECs. C-stationarity relates to Clark's calculus as explained in [14], while M-stationarity is derived from Morukhovich's calculus; cf., e.g., [12]. The strongest of these stationarity concepts, strong stationarity, is equivalent to stationarity of $\bar{z}$ in the standard sense for the following relaxed NLP:

$$\begin{array}{ll} \min & f(z) \\ \text{s.t.} & g(z) \leq 0, \\ & h(z) = 0, \\ & G_i(z) = 0, \quad i \notin I_H(\bar{z}), \\ & H_j(z) = 0, \quad j \notin I_G(\bar{z}), \\ & G_k(z) \geq 0, \quad k \in I_G(\bar{z}) \cap I_H(\bar{z}), \\ & H_k(z) \geq 0, \quad k \in I_G(\bar{z}) \cap I_H(\bar{z}). \end{array} \tag{8}$$

It implies B-stationarity in the sense of [14] and is, in fact, equivalent to B-stationarity in the presence of MPEC-LICQ; see also [10]. Strong stationarity is sufficient for local optimality, without constraint qualification, provided f and all component functions of g are convex and h, G, H are affine mappings. We will not distinguish between B-stationarity and strong stationarity in what follows as we will assume MPEC-LICQ throughout. The reason for considering the weaker stationarity concepts is that such points are potential attractors of methods based on (7) or variants thereof. We will see that this is indeed the case for our regularization method.

At places we will use the upper level strict complementarity condition which was used in [14] in the context of sensitivity analysis for MPECs.

Upper level strict complementarity (ULSC). *A weakly stationary point $\bar{z}$ is said to satisfy ULSC if there exists MPEC multipliers satisfying* (7) *with $\bar{\gamma}_m \bar{\nu}_m \neq 0$ for all $m \in I_G(\bar{z}) \cap I_H(\bar{z})$.*

ULSC holds in particular if lower level strict complementarity holds, i.e., if $G(\bar{z}) + H(\bar{z}) > 0$, but is considerably weaker than the latter. In problem (2) lower level strict complementarity is violated at a local minimizer if and only if $(a, b) \leq 0$, while ULSC is violated if and only if $(a, b) \leq 0$ and $ab = 0$. In fact, ULSC is close in spirit to strict complementarity in the standard NLP sense, in particular in connection with strong stationarity. Indeed, as mentioned above, a feasible point $\bar{z}$ of NLP(0) is strongly stationary if and only if it is stationary in the standard sense for the relaxed NLP (8). If, in addition, strict complementarity in the standard sense holds for (8), then ULSC holds at $\bar{z}$.

3. Convergence results. Our first result relates MPEC multipliers at $\bar{z}$ to the NLP multipliers of an approaching sequence of stationary points of NLP(t_n). This allows us to characterize B-stationarity of limit points in the presence of MPEC-LICQ.

THEOREM 3.1. *Let $\{t_n\}$ be a sequence of positive scalars tending to zero, let z_n be a stationary point of NLP(t_n) tending to $\bar{z}$ and suppose MPEC-LICQ holds at $\bar{z}$. Let*

$$I_0 = \{m \mid m \in I_{GH}(z_n, t_n) \ \text{ for infinitely many } n\}.$$

Then the following statements hold:

1. *For every sufficiently large n NLP(t_n) has unique multipliers $\lambda_{i,n}, \mu_{j,n}, \gamma_{k,n}, \nu_{l,n}$, $\delta_{m,n}$ at z_n* (see (6)).
2. *The point $\bar{z}$ is a C-stationary point of NLP(0) with unique multipliers $\bar{\lambda}, \bar{\mu}, \bar{\gamma}, \bar{\nu}$ which satisfy*

$$\begin{aligned}
\bar{\lambda}_i &= \lim_{n\to\infty} \lambda_{i,n} \geq 0, \\
\bar{\mu}_j &= \lim_{n\to\infty} \mu_{j,n}, \\
\bar{\gamma}_k &= \lim_{n\to\infty} \gamma_{k,n} \geq 0, \quad k \notin I_0, \\
\bar{\nu}_k &= \lim_{n\to\infty} \nu_{k,n} \geq 0, \quad k \notin I_0, \\
\bar{\gamma}_m &= -\lim_{n\to\infty} \delta_{m,n} H_m(z_n) \leq 0, \quad m \in I_0, \\
\bar{\nu}_m &= -\lim_{n\to\infty} \delta_{m,n} G_m(z_n) \leq 0, \quad m \in I_0.
\end{aligned}$$

3. *The point $\bar{z}$ is B-stationary if and only if $\bar{\gamma}_m = \bar{\nu}_m = 0$ for all $m \in I_G(\bar{z}) \cap I_H(\bar{z}) \cap I_0$.*

Proof. The first statement is an immediate consequence of Lemma 2.1. To see the second statement, note that the stationarity conditions (6) for NLP(t_n) at z_n can be rewritten as

$$\begin{aligned}
-\nabla f(z_n) = &\sum_{i\in I_g(z_n)} \lambda_{i,n}\nabla g_i(z_n) + \sum_{j\in I_h(z_n)} \mu_{j,n}\nabla h_j(z_n)\\
&- \sum_{k\in I_G(z_n)} \gamma_{k,n}\nabla G_k(z_n) - \sum_{l\in I_H(z_n)} \nu_{l,n}\nabla H_l(z_n)\\
&- \sum_{\substack{m\in I_{GH}(z_n,t_n)\\ m\notin I_G(\bar z)}} \nu_{m,n}\left[\nabla H_m(z_n) + \frac{H_m(z_n)}{G_m(z_n)}\nabla G_m(z_n)\right]\\
&- \sum_{\substack{m\in I_{GH}(z_n,t_n)\\ m\notin I_H(\bar z)}} \gamma_{m,n}\left[\nabla G_m(z_n) + \frac{G_m(z_n)}{H_m(z_n)}\nabla H_m(z_n)\right]\\
&- \sum_{\substack{m\in I_{GH}(z_n,t_n)\\ m\in I_G(\bar z)\cap I_H(\bar z)}} [\gamma_{m,n}\nabla G_m(z_n) + \nu_{m,n}\nabla H_m(z_n)]
\end{aligned}$$

with

$$\gamma_{m,n} = -\delta_{m,n}H_m(z_n), \quad \nu_{m,n} = -\delta_{m,n}G_m(z_n)$$

for $m \in I_{GH}(z_n, t_n)$. For sufficiently large n we construct a matrix $A(z_n)$ from the matrix with rows

$$\begin{array}{ll}
\nabla g_i(z_n), & i \in I_g(\bar z),\\
\nabla h_j(z_n), & j \in I_h(\bar z),\\
-\nabla G_k(z_n), & k \in I_G(\bar z),\\
-\nabla H_l(z_n), & l \in I_H(\bar z),
\end{array}$$

by replacing the row $-\nabla G_m(z_n)$ by $-\nabla G_m(z_n) - \frac{G_m(z_n)}{H_m(z_n)}\nabla H_m(z_n)$ if $m \in I_{GH}(z_n, t_n)$ and $m \notin I_H(\bar z)$ and replacing the row $-\nabla H_m(z_n)$ by $-\nabla H_m(z_n) - \frac{H_m(z_n)}{G_m(z_n)}\nabla G_m(z_n)$ if $m \in I_{GH}(z_n, t_n)$ and $m \notin I_G(\bar z)$. The above system is of the form $A(z_n)^\top x_n = -\nabla f(z_n)$ with some components of the multiplier vector x_n set to zero. The matrix $A(z_n)$ converges to the matrix $A(\bar z)$ with rows

$$\begin{array}{ll}
\nabla g_i(\bar z), & i \in I_g(\bar z),\\
\nabla h_j(\bar z), & j \in I_h(\bar z),\\
-\nabla G_k(\bar z), & k \in I_G(\bar z),\\
-\nabla H_l(\bar z), & l \in I_H(\bar z),
\end{array}$$

which has full row rank by assumption. It follows that the multipliers $\lambda_n, \mu_n, \gamma_n, \nu_n$ converge to the unique MPEC multipliers $\bar\lambda, \bar\mu, \bar\gamma, \bar\nu$ at $\bar z$ and that the limiting expressions hold. The last statement follows immediately from the limiting expressions and the fact that B-stationarity is equivalent to strong stationarity in the presence of MPEC-LICQ; cf., e.g., [14]. ∎

Notice that the characteristic condition for B-stationarity given in the theorem is obviously satisfied if $\bar z$ is nondegenerate on the lower level, i.e., $G(\bar z) + H(\bar z) > 0$, or if the multiplier sequences $\delta_{m,n}$ are bounded for all $m \in I_G(\bar z) \cap I_H(\bar z) \cap I_0$.

The following corollary is an immediate consequence of Theorem 3.1.

COROLLARY 3.2. *If, in the setting of Theorem* 3.1*, n is sufficiently large, then*

1. $G_k(z_n) = 0$ *for all* k *with* $\bar{\gamma}_k > 0$;
2. $H_k(z_n) = 0$ *for all* k *with* $\bar{\nu}_k > 0$;
3. $G_k(z_n)H_k(z_n) = t_n$ *for all* k *with* $\bar{\gamma}_k + \bar{\nu}_k < 0$.

We next analyze to what extent the statement of Theorem 3.1 can be improved if, in addition to MPEC-LICQ at $\bar{z}$, second order necessary conditions hold at each z_n, i.e., the Hessian with respect to z of the Lagrangian

$$\begin{aligned}&L_{t_n}(z, \lambda, \mu, \gamma, \nu, \delta)\\ &= f(z) + \lambda^\top g(z) + \mu^\top h(z) - \gamma^\top G(z) - \nu^\top H(z) + \sum \delta_i (G_i(z)H_i(z) - t_n)\end{aligned}$$

is positive semidefinite on the cone of critical directions at z_n; cf. Theorem 9.3.1 of [3].

THEOREM 3.3. *If, in addition to the assumptions of Theorem* 3.1*, second order necessary optimality conditions hold at each* z_n*, then* $\bar{z}$ *is M-stationary.*

Proof. Suppose $\bar{z}$ is not M-stationary. Then, by Theorem 3.1, there exists an index $m \in I_G(\bar{z}) \cap I_H(\bar{z}) \cap I_0$ such that

$$\begin{aligned}\bar{\gamma}_m &= -\lim_{n\to\infty} \delta_{m,n} H_m(z_n) < 0,\\ \bar{\nu}_m &= -\lim_{n\to\infty} \delta_{m,n} G_m(z_n) < 0.\end{aligned}$$

Since $m \in I_0$ we may assume, by passing to a subsequence, that $H_m(z_n)G_m(z_n) = t_n$ for every sufficiently large n. Hence the Hessian of $\phi_m(z) = G_m(z)H_m(z)$ appears weighted with the corresponding multiplier $\delta_{m,n}$ in the Hessian of the Lagrangian for every sufficiently large n. Notice that gradient and Hessian of $\phi_m(z)$ are of the form

$$\begin{aligned}\nabla\phi_m(z_n) &= G_m(z)\nabla H_m(z_n) + H_m(z)\nabla G_m(z_n),\\ \nabla^2\phi_m(z_n) &= \nabla G_m(z_n)^\top \nabla H_m(z_n) + \nabla H_m(z_n)^\top \nabla G_m(z_n)\\ &\quad + G_m(z_n)\nabla^2 H_m(z_n) + H_m(z_n)\nabla^2 G_m(z_n).\end{aligned}$$

Because of the MPEC-LICQ assumption we can thus choose, for sufficiently large n, a sequence of directions d_n with

$$\begin{aligned}\nabla g_i(z_n)d_n &= 0, \quad i:\ g_i(\bar{z}) = 0,\\ \nabla h(z_n)d_n &= 0,\\ \nabla G_i(z_n)d_n &= 0, \quad i:\ G_i(\bar{z}) = 0, i \neq m,\\ \nabla H_i(z_n)d_n &= 0, \quad i:\ H_i(\bar{z}) = 0, i \neq m,\\ \nabla G_m(z_n)d_n &= 1,\\ \nabla H_m(z_n)d_n &= -H_m(z_n)/G_m(z_n).\end{aligned}$$

We may assume that the sequence $\{d_n\}$ is bounded since $H_m(z_n)/G_m(z_n)$ converges to $\bar{\gamma}_m/\bar{\nu}_m$. Notice that $\nabla\phi_m(z_n)d_n = 0$ and that d_n is a critical direction of NLP(t_n) at z_n for every sufficiently large n. Moreover,

$$\begin{aligned}&\delta_{m,n} d_n^\top \nabla^2\phi_m(z_n)d_n\\ &= \delta_{m,n}H_m(z_n)d_n^\top\nabla^2 G_m(z_n)d_n + \delta_{m,n}G_m(z_n)d_n^\top\nabla^2 H_m(z_n)d_n - \delta_{m,n}H_m(z_n)\frac{2}{G_m(z_n)},\end{aligned}$$

which tends to $-\infty$ since the first two terms are bounded and $\delta_{m,n}H_m(z_n)$ tends to $-\bar{\gamma}_m > 0$ while $G_m(z_n)$ is a positive null-sequence. Since all other terms in

$$d_n^\top \nabla_z^2 L_{t_n}(z_n, \lambda_n, \mu_n, \gamma_n, \nu_n, \delta_n)d_n$$

are bounded, second order necessary optimality conditions fail for sufficiently large n. □

The foregoing theorem complements a recent result of Fukushima and Pang [5], who proved convergence of a smoothing procedure to B-stationary points under MPEC-LICQ, second order necessary optimality conditions, and an asymptotic weak nondegeneracy assumption. We obtain this result in our setting with the asymptotic weak nondegeneracy assumption replaced by ULSC.

COROLLARY 3.4. *Let $\{t_n\}$ be a positive null-sequence, $\{z_n\}$ be a sequence of feasible points of NLP(t_n) satisfying second order necessary conditions, and $\bar{z}$ be an accumulation point of $\{z_n\}$. If MPEC-LICQ and ULSC are satisfied at $\bar{z}$, then $\bar{z}$ is B-stationary.*

4. Characterization of attractors. In this section we study the question whether a local minimizer of an MPEC is a limit of a sequence of stationary points of the regularized programs. This is not necessarily the case. Consider the example

$$\begin{array}{ll} \min & -x \\ \text{s.t.} & xy \le t, \\ & x, y \ge 0. \end{array}$$

Every point on the positive y-axis is a local minimizer of NLP(0) but the regularized program NLP(t), $t > 0$, has no local minimizers nor stationary points. If the objective function is replaced by $x(1-y^2)$, then the set of local, and global, minimizers of NLP(0) is the nonnegative y-axis. The relaxed problem, however, has no local minimizers at all. This shows that further conditions, in addition to MPEC-LICQ, are necessary to guarantee that a local minimizer is a limit point of a sequence of local minimizers z_n of NLP(t_n) with t_n tending to zero. To this end, we use the following second order condition based on the MPEC-Lagrangian:

$$L(z, \lambda, \mu, \gamma, \nu) = f(z) + g(z)^\top \lambda + h(z)^\top \mu - G(z)^\top \gamma - H(z)^\top \nu$$

of NLP(0); cf. [14].

Strong second order sufficient condition (SSOSC). *Let $\bar{z}$ be a B-stationary point of NLP(0). Suppose MPEC-LICQ holds and $(\bar{\lambda}, \bar{\mu}, \bar{\gamma}, \bar{\nu})$ are the unique MPEC multipliers at $\bar{z}$. We say that the strong second order sufficient condition holds at $\bar{z}$ if*

$$d^\top \nabla_z^2 L(\bar{z}, \bar{\lambda}, \bar{\mu}, \bar{\gamma}, \bar{\nu}) d > 0$$

for every nonvanishing d with

$$\begin{aligned} \nabla g_i(\bar{z}) d = 0, &\quad i : \ \bar{\lambda}_i > 0, \\ \nabla h(\bar{z}) d = 0, & \\ \nabla G_j(\bar{z}) d = 0, &\quad j : \ \bar{\gamma}_j \ne 0, \\ \nabla H_k(\bar{z}) d = 0, &\quad k : \ \bar{\nu}_k \ne 0. \end{aligned}$$

The following theorem follows from Corollary 3.2 through an application of standard NLP stability theory. We confine ourselves to a sketch of the proof.

THEOREM 4.1. *Suppose*

1. *$\bar{z}$ is a B-stationary point of NLP(0) and MPEC-LICQ as well as SSOSC hold at $\bar{z}$;*
2. *$\bar{\gamma}_i \ne 0$ if $G_i(\bar{z}) = 0$, and $\bar{\nu}_i \ne 0$ if $H_i(\bar{z}) = 0$ for every $i = 1, \ldots, m$.*

Then there exists an open neighborhood U of $\bar{z}$, a scalar $\bar{t} > 0$, and a piecewise smooth function $z : (-\bar{t}, \bar{t}) \to U$ such that $z(t)$ is the unique stationary point of NLP(t) for every $0 < t < \bar{t}$. Moreover, $z(t)$ satisfies second order sufficient optimality conditions.

Proof. Consider the parametric nonlinear program $\mathcal{P}(t)$ defined by

$$
\begin{array}{ll}
\min & f(z) \\
\text{s.t.} & g(z) \leq 0, \\
 & h(z) = 0, \\
 & G_i(z) \geq 0 \ \text{ if } \bar{\gamma}_i > 0, \\
 & H_j(z) \geq 0 \ \text{ if } \bar{\nu}_j > 0, \\
 & G_k(z)H_k(z) \leq t \ \text{ if } \bar{\gamma}_k < 0, \\
 & G_l(z)H_l(z) \leq t \ \text{ if } \bar{\nu}_l < 0.
\end{array} \tag{9}
$$

Let us denote the multipliers of (9) by $\tilde{\lambda}$, $\tilde{\mu}$, $\tilde{\gamma}_i$, $\tilde{\nu}_j$, $\tilde{\delta}_k$, and $\tilde{\delta}_l$. B-stationarity and the MPEC-LICQ assumption imply that NLP-LICQ holds at the feasible point $\bar{z}$ of $\mathcal{P}(0)$ and that $\bar{z}$ is a stationary point of $\mathcal{P}(0)$ with unique multipliers

$$
\tilde{\lambda} = \bar{\lambda}, \quad \tilde{\mu} = \bar{\mu}, \quad \tilde{\gamma}_i = \bar{\gamma}_i, \quad \tilde{\nu}_j = \bar{\nu}_j, \quad \tilde{\delta}_k = -\bar{\gamma}_k / H_k(\bar{z}), \quad \tilde{\delta}_l = -\bar{\nu}_l / G_l(\bar{z}).
$$

SSOSC ensures that the program $\mathcal{P}(0)$ is stable in the sense of Kojima [9] and Robinson [13]. Hence there exist a locally unique and piecewise smooth stationary point function $z(t)$ and a unique and piecewise smooth multiplier function $(\tilde{\lambda}, \tilde{\mu}, \tilde{\gamma}_i, \tilde{\nu}_j, \tilde{\delta}_k, \tilde{\delta}_l)(t)$. Moreover, $z(t)$ satisfies second order sufficient conditions for $\mathcal{P}(t)$ for sufficiently small t. Since the feasible region of NLP(t) is contained in the feasible region of $\mathcal{P}(t)$, it only remains to be shown that the minimizer is feasible for NLP(t). If $\bar{\gamma}_i > 0$ or $\bar{\nu}_j > 0$, then, by continuity of the multipliers, the respective inequalities $G_i(z) \geq 0$ or $H_j(z) \geq 0$ remain active for sufficiently small $t > 0$ and therefore $H_i(z(t))G_i(z(t)) = H_j(z(t))G_j(z(t)) = 0 < t$. If $\bar{\gamma}_k < 0$, then the inequality $G_k(z)H_k(z) \leq t$ will be active for small $t > 0$ since the multiplier function is continuous. Moreover, $\bar{\gamma}_k < 0$ implies $H_k(\bar{z}) > 0$, in view of the equivalence of B-stationarity and strong stationarity under MPEC-LICQ; therefore $G_k(z(t)) > 0$ for sufficiently small $t > 0$. The same argument shows that $H_l(z(t)) > 0$ for sufficiently small $t > 0$ if $\bar{\nu}_l < 0$. We conclude that, for sufficiently small $t > 0$, $z(t)$ is a local minimizer of NLP(t) which satisfies second order sufficient conditions since it satisfies these conditions for $\mathcal{P}(t)$ which is obtained from NLP(t) by deleting some constraints. Corollary 3.2 implies that $z(t)$ is the unique B-stationary point of NLP(t) in a neighborhood of $\bar{z}$. ∎

Notice that the strict complementarity assumption of the latter theorem is stronger than ULSC as it requires *all* multipliers corresponding to vanishing components of (G, H) to be nonvanishing, not just the ones corresponding to components with $G_k(\bar{z}) = H_k(\bar{z}) = 0$. The curve $z(t)$ is smooth in a neighborhood of $t = 0$ if, in addition to the assumptions of the theorem, strict complementarity holds with respect to the remaining inequalities $g(z) \leq 0$, i.e., $\bar{\lambda} > g(\bar{z})$.

5. Extensions.

5.1. An alternative regularization. The last m inequalities in problem (1) can be replaced by a single inequality $G(z)^\top H(z) \le t$, resulting in the program

$$\begin{array}{ll} \min & f(z) \\ \text{s.t.} & g(z) \le 0, \\ & h(z) = 0, \\ & G(z) \ge 0, \\ & H(z) \ge 0, \\ & G(z)^\top H(z) \le t. \end{array} \tag{10}$$

This alternative is certainly appealing from a numerical point of view since this NLP has fewer inequality constraints. We show in the appendix that the convergence results of section 3 remain valid for this regularization. Hu [6] has announced an example which shows that the attractor results of section 4 do not hold in this case.

5.2. Mixed complementarity constraints. The suggested regularization scheme extends to mixed complementarity constraints. In a general framework, mixed complementarity constraints involve three mappings F, G, H and are naturally embedded in a parametric system of the form

$$F(z) \ge 0, \quad G(z) \ge 0, \quad F_i(z)H_i(z) \ge -t, \quad G_i(z)H_i(z) \le t, \tag{11}$$

where t is a nonnegative regularization parameter which has to be set to zero to obtain the original mixed complementarity constraints. If $z = (x, y)$, $F(x, y) = b - y$, and $G(x, y) = y - a$ where $a \le b$, then the constraints (11) correspond to a parametric variational inequality induced by a parametric vector field $H(x, .)$ over the box $a \le y \le b$. Notice that we recover the standard complementarity constraints if we set $F(z) = H(z)$ and treat the third inequality in (11) as redundant. An alternative is to allow components of $F(z)$ or $G(z)$ to have infinite values in which case $G_i(z)H_i(z) \le t$ is interpreted as $H_i(z) \le 0$ if $G_i(z) = \infty$, while $F_i(z)H_i(z) \ge -t$ means $H_i(z) \ge 0$ if $F_i(z) = \infty$. We can recover standard complementarity constraints by setting $F_i(z) = \infty$ for all i.

The convergence results given in the foregoing sections extend, mutatis mutandis, to programs NLP(t) of the form

$$\begin{array}{ll} \min & f(z) \\ \text{s.t.} & g(z) \le 0, \\ & h(z) = 0, \\ & F(z) \ge 0, \\ & G(z) \ge 0, \\ & G_i(z)H_i(z) \le t \quad \text{for all } i, \\ & F_i(z)H_i(z) \ge -t \quad \text{for all } i, \end{array}$$

provided $F(\bar{z}) + G(\bar{z}) > 0$, which is obviously the case if $F_i(z) = \infty$ for all i or if $a < b$ in the above formulation of the box constrained variational inequality. To see this, suppose $F_i(\bar{z}) > 0$. Then the constraint $F_i(z) \ge 0$ is locally inactive and the perturbed mixed complementarity constraint

$$F_i(z) \ge 0, \quad G_i(z) \ge 0, \quad F_i(z)H_i(z) \ge -t, \quad G_i(z)H_i(z) \le t$$

turns locally into the perturbed standard complementarity constraint

$$G_i(z) \geq 0, \quad H_i(z) + t/F_i(z) \geq 0, \quad G_i(z)H_i(z) \leq t.$$

Notice that for $\rho_i(z) = H_i(z) + t/F_i(z)$ we have

$$\begin{aligned}\nabla\rho_i(z) &= \nabla H_i(z) - (t/F_i(z)^2)\nabla F_i(z),\\ \nabla^2\rho_i(z) &= \nabla^2 H_i(z) + (2t/F_i(z)^3)\nabla F_i(z)^\top \nabla F_i(z) - (t/F_i(z)^2)\nabla^2 F_i(z)\end{aligned}$$

which tend to $\nabla H_i(z)$ and $\nabla^2 H_i(z)$, respectively, uniformly in a neighborhood of $\bar{z}$ if $F_i(\bar{z}) > 0$. Since similar approximations hold with suitable sign changes for the case $G_i(\bar{z}) > 0$, the first and second order arguments that we used in the case of standard complementarity constraints carry through for mixed complementarity constraints if $F(\bar{z}) + G(\bar{z}) > 0$.

6. Preliminary numerical experience. The proposed scheme is only conceptual since it assumes nonlinear programs to be solved in each iteration. It is therefore not sensible to make extensive numerical tests. Nevertheless, the approach was tested in an ad hoc way on a variety of small and medium-size problems and the implemented method showed a credible performance.

We used the MATLAB 5.3 built-in solver function *fmincon* with gradient evaluations and otherwise default settings to solve the nonlinear programs for positive t-values. It is generally acknowledged that this solver is inferior with regard to robustness to some other available NLP codes. For our purpose, this deficiency is quite desirable since our approach is meant to enhance the robustness of an NLP solver when applied to MPECs.

The tests were run on randomly generated problems with convex quadratic objective and linear constraint functions. The parametric NLPs are thus of the form

$$\begin{array}{ll}\min & \frac{1}{2}z^\top Qz + q^\top z\\ \text{s.t.} & Az \leq a,\\ & Cz \geq c,\\ & Dz \geq d,\\ & (Cz-c)_i(Dz-d)_i \leq t, \quad i = 1,\ldots,m.\end{array}$$

We chose $d = 0$ and $D = [I, 0]$, where I is a unit matrix and 0 is a zero matrix of appropriate size, so that the complementarity constraint for $t = 0$ is a standard parametrized linear complementarity constraint. The objective function was chosen to be of the form $0.5z^\top z + q^\top z$; i.e., we tried to find the feasible point that is closest to the vector $-q$. The remaining data was randomly generated in the sequence $A, a, C, -c$ using the MATLAB *rand* function. First we attempted to solve the problem for $t = 10^{-16}$. We then started with $t = 1$ and performed 17 iterations where t was reduced after each iteration to $t/10$ so that the t-value in the final iteration was again $t = 10^{-16}$. If at the end of an iteration the exit flag of the MATLAB *fmincon* function indicated that a stationary point was found, then this stationary point was used as the starting point for the next iteration.

For illustration we report the results of a typical test run of 100 random problems with 50 variables, where the matrices C and D had 25 rows, the matrix A had 50 rows, and q was chosen to be the vector of all -1's. For this run, we used the vector of all 1's as the starting point and seeded the *rand* function at 1. For fixed $t = 10^{-16}$ the solver was able to solve all but one of the problems and needed on average 8.2

QP iterations with a standard deviation of 1.8 iterations. The regularization method with decreasing t-values, starting at $t = 1$ and iterating down to $t = 10^{-16}$, solved all problems for all t-values. The method found an initial stationary point for $t = 1$ for all problems after two QP iterations and very few QP iterations were needed to solve the NLPs in each of the following iterations. Table 1 shows the average number of QP iterations for each t-value.

TABLE 1
Number of QP iterations of the regularization method.

t	1	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	10^{-14}	10^{-16}
Mean	2	1	5.6	2.5	1.9	1	1	1	1
Stdev	0	0	2.2	0.9	0.3	0	0	0	0

On average, 32% of the complementarity constraints were both active at the optimal solution.

For comparison we changed the inequalities $(C_i z - c_i)(D_i z - d) \leq t$ to equations, which is geometrically equivalent to a smoothing of the feasible set as discussed in [2, 5, 8]. The iterative approach with t decreased from $t = 1$ down to $t = 10^{-16}$ worked for 85% of the problems in the sense that the solver produced a solution for the final value $t = 10^{-16}$. Problems were, however, often encountered along the way for some larger values of t for which the solver reported that it could not find a feasible solution or that the maximum number of function evaluations was exceeded. Indeed, the solver produced solutions for all t for only 34 of the 100 problems. Interestingly, the smoothing approach was not only less robust but also less efficient on our set of test problems. Even for the 34 successful problems the smoothing approach needed significantly more QP iterations per NLP iteration, as indicated in Table 2.

TABLE 2
Number of QP iterations of the smoothing method for successful problems.

t	1	10^{-2}	10^{-4}	10^{-6}	10^{-8}	10^{-10}	10^{-12}	10^{-14}	10^{-16}
Mean	34	25	22.9	23.3	22.1	15.1	4.8	1.9	1
Stdev	11.2	8.2	3.5	7.2	5.9	8.6	2.6	0.8	0

The path-following property, characterized by a single QP per NLP iteration, appeared typically only for very small t-values in the smoothing approach. Starting at small t-values, however, often resulted in failures.

It is interesting to compare the quality of the local solutions produced by the two methods. In our tests we found that the regularization method often produced lower objective function values if the two methods produced different local solutions. For the particular test run discussed here, both approaches produced the same solution for 47 of the 85 problems on which the smoothing approach was eventually successful. For two of these problems the smoothing approach produced a solution that improved on the objective function produced by the regularization approach, albeit by not more than 0.6%. For the remaining 36 problems the regularization approach produced better local solutions. For 7 problems the improvement was more than 5% with a maximal improvement of 12%.

Our numerical experiments are insufficient to claim that one of the approaches is superior to the other. They do indicate, however, that the perturbation approach is a sensible way of stabilizing a standard NLP code for the MPEC NLP(0) if the code struggles to produce a solution if applied to the MPEC directly. Tests based on the

alternative regularization (10) showed a similar performance to the regularization (1) for the small and medium-size problems that we solved.

The smoothing approach we used for our numerical comparison is based directly on constraints of the form $G_i(z) \geq 0, H_i(z) \geq 0$, $G_i(z)H_i(z) = t$. Facchinei, Jiang, and Qi [2], Fukushima and Pang [5], and Jiang and Ralph [8] allow for alternative algebraic descriptions of these constraints, such as

$$\frac{1}{2}\left[G_i(z) + H_i(z) - \sqrt{(G_i(z) - H_i(z))^2 + 4t}\right] = 0$$

or

$$H_i(z) + G_i(z) - \sqrt{G_i(z)^2 + H_i(z)^2 + 2t} = 0.$$

It is possible that such alternative representations improve the performance of the smoothing approach. However, one advantage of the regularization approach $(G_i(z)H_i(z) \leq t)$ over the smoothing approach $(G_i(z)H_i(z) = t)$ or its alternatives is that the former encompasses the feasible region of the MPEC NLP(0) and may therefore identify active parts of the complementarity constraints early on. For example, if we choose $(a, b) = (-1, -1)$ in problem (2), then the regularized problem identifies the correct solution for any $t > 0$, while the smoothed problem gives estimates $(\sqrt{t}, \sqrt{t})$ and needs t to converge to zero to produce the solution. The regularization approach can also be more robust if constraint qualifications are violated as illustrated by the example

$$\begin{array}{ll} \min & \frac{1}{2}[(x-1)^2 + (y-2)^2 + (z+1)^2] \\ \text{s.t.} & x, y, z \geq 0, \\ & xz = 0, \\ & yz = 0. \end{array}$$

Notice that a sequence of points with $x, y, z > 0$ and $xz = t$ $yz = t$ for $t > 0$ can only converge to a point on either the nonnegative z-axis or the diagonal in the nonnegative orthant of the (x, y) plane. Hence the smoothing method cannot detect the minimizer $(1, 2, 0)$. The reason is that MPEC-LICQ is violated at every feasible point in the (x, y) plane. Nevertheless, the minimizer will be obtained for any $t > 0$ as a solution of the regularized program and these programs satisfy NLP-LICQ.

7. Conclusions. We have studied a regularization scheme for MPECs with complementarity constraints. Accumulation points of a regularization sequence were shown to be B-stationary points of the MPEC, provided the iterates satisfy second order necessary conditions and linear independence and upper level strict complementarity conditions hold at the accumulation point. This complements results of Fukushima and Pang [5] for a smoothing scheme. Hu [7] has recently derived similar convergence properties for a penalty method. In addition to the convergence results, we have shown that local minimizers of the MPEC can be embedded into a smooth curve of locally unique local minimizers of the regularized program, provided linear independence and suitable second order sufficient conditions hold, and all MPEC multipliers corresponding to vanishing components of constraint functions g_i, G_k, H_k are nonvanishing.

The proposed scheme is only conceptual since it assumes nonlinear programs to be solved in each iteration. Nevertheless, the latter mentioned embedding result gives rise to the hope that a practical method can be devised which eventually follows the

smooth path of local minimizers of the regularized program if iterates approach a well-behaved local minimizer. At this stage, however, the suggested approach cannot yet be compared with alternative methods [10, 11, 15]. At present, the most satisfying theoretical convergence results seem to be available for the MPEC modification of the trust region SL_1QP method of Fletcher [3] which is based on exact penalization and sequential quadratic programming. The method has been introduced in [15] and is further modified and analyzed in [1, 17]. It converges globally and locally superlinearly to B-stationary points of complementarity constrained MPECs under assumptions similar to those for standard nonlinear programs.

8. Appendix: An alternative regularization. In this appendix we show that the results of section 3 remain true if (1) is replaced by the parametric problem

$$\begin{array}{ll} \min & f(z) \\ \text{s.t.} & g(z) \le 0, \\ & h(z) = 0, \\ & G(z) \ge 0, \\ & H(z) \ge 0, \\ & G(z)^\top H(z) \le t \end{array} \tag{12}$$

which has fewer inequalities. In this appendix, NLP(t) refers to (12). We first show that, in analogy to Lemma 2.1, the linear independence condition is passed on to the regularized programs.

LEMMA 8.1. *If MPEC-LICQ holds at the feasible point $\bar{z}$ of the MPEC NLP(0), then there exists a neighborhood U of $\bar{z}$ and a scalar $\bar{t} > 0$ such that for every $t \in (0, \bar{t})$ NLP-LICQ holds at every feasible point $z \in U$ of NLP(t).*

Proof. Renumbering components if necessary, we may assume that

$$\begin{array}{lll} G_1(\bar{z}) > 0, & G_2(\bar{z}) = 0, & G_3(\bar{z}) = 0, \\ H_1(\bar{z}) = 0, & H_2(\bar{z}) > 0, & H_3(\bar{z}) = 0. \end{array}$$

If we assume that g_1 contains the components of g that vanish at $\bar{z}$, then the assumption is that the matrix with row blocks

$$\nabla g_1(\bar{z}), \nabla h(\bar{z}), \nabla G_2(\bar{z}), \nabla G_3(\bar{z}), \nabla H_1(\bar{z}), \nabla H_3(\bar{z})$$

has full row rank. Now let us assume that the assertion of the lemma does not hold. Then there exists a positive sequence $t_n \to 0$ and a sequence $z_n \to \bar{z}$ such that z_n is feasible for NLP(t_n) and NLP-LICQ does not hold at z_n. Since $z_n \to \bar{z}$ we have $G_1(z_n) > 0$ and $H_2(z_n) > 0$ for sufficiently large n and we may partition G_i and H_i further into G_{ij} and H_{ij}, respectively, such that

$$\begin{array}{llll} G_{11}(z_n) > 0, & G_{12}(z_n) > 0, & G_{21}(z_n) > 0, & G_{22}(z_n) = 0, \\ H_{11}(z_n) > 0, & H_{12}(z_n) = 0, & H_{21}(z_n) > 0, & H_{22}(z_n) > 0, \\ G_{31}(z_n) > 0, & G_{32}(z_n) > 0, & G_{33}(z_n) = 0, & G_{34}(z_n) = 0, \\ H_{31}(z_n) = 0, & H_{32}(z_n) > 0, & H_{33}(z_n) > 0, & H_{34}(z_n) = 0. \end{array} \tag{13}$$

This partitioning may depend on n for a general sequence z_n but, passing to a subsequence if necessary, we may assume that it is constant for all n since there are only finitely many such partitions. We may also assume that $G(z_n)^\top H(z_n) = t > 0$ for otherwise the assertion of the lemma holds trivially. It suffices to show that the matrix with row blocks

$$\begin{array}{l} \nabla g_1(z_n), \nabla h(z_n), \nabla G_{22}(z_n), \nabla G_{33}(z_n), \nabla G_{34}(z_n), \\ \nabla H_{12}(z_n), \nabla H_{31}(z_n), \nabla H_{34}(z_n), H(z_n)^\top \nabla G(z_n) + G(z_n)^\top \nabla H(z_n) \end{array}$$

has full row rank. The MPEC-LICQ assumption implies that, for sufficiently large n, the matrix turns into a matrix with full row rank if the last row is removed. Hence the full matrix fails to have full row rank if and only if there are multipliers $\lambda_n, \mu_n, \gamma_n, \nu_n$ such that

$$\begin{aligned}&\nabla G(z_n)^\top H(z_n) + \nabla H(z_n)^\top G(z_n)\\ &\quad = \nabla g_1(z_n)^\top \lambda_{1,n} + \nabla h(z_n)^\top \mu_n\\ &\quad +\nabla G_{22}(z_n)^\top \gamma_{22,n} + \nabla G_{33}(z_n)^\top \gamma_{33,n} + \nabla G_{34}(z_n)^\top \gamma_{34,n}\\ &\quad +\nabla H_{12}(z_n)^\top \nu_{12,n} + \nabla H_{31}(z_n)^\top \nu_{31,n} + \nabla H_{34}(z_n)^\top \nu_{34,n}.\end{aligned}$$

Rearranging terms in the equation yields

$$\begin{aligned}&\nabla G_{11}(z_n)^\top H_{11}(z_n) + \nabla H_{21}(z_n)^\top G_{21}(z_n)\\ &= \nabla g_1(z_n)^\top \lambda_{1,n} + \nabla h(z_n)^\top \mu_n\\ &-\nabla G_{21}(z_n)^\top H_{21}(z_n) + \nabla G_{22}(z_n)^\top (\gamma_{22,n} - H_{22}(z_n)) - \nabla G_{32}(z_n)^\top H_{32}(z_n)\\ &+\nabla G_{33}(z_n)^\top (\gamma_{33,n} - H_{33}(z_n)) + \nabla G_{34}(z_n)^\top \gamma_{34,n}\\ &-\nabla H_{11}(z_n) G_{11}(z_n) + \nabla H_{12}(z_n)^\top (\nu_{12} - G_{12}(z)) + \nabla H_{31}(z)^\top (\nu_{31,n} - G_{31}(z_n))\\ &-\nabla H_{32}(z_n)^\top G_{32}(z_n) + \nabla H_{34}(z_n)^\top \nu_{34,n}.\end{aligned}$$

In view of the MPEC-LICQ assumption the gradients on the right-hand side are linearly independent for sufficiently large n. Moreover, the left-hand side of the equation tends to zeros as n tends to ∞. Therefore, if n is large, the augmented multipliers on the right-hand side must be arbitrarily close to zero. In particular $G_{11}(z_n), H_{21}(z_n)$ must be close to zero which contradicts the assumptions $G_{11}(\bar{z}) > 0$ and $H_{21}(\bar{z}) > 0$. Therefore we can simplify (13) to

$$\begin{aligned}&G_1(z_n) > 0, \quad G_2(z_n) = 0, \quad G_{31}(z_n) > 0, \quad G_{32}(z_n) > 0, \quad G_{33}(z_n) = 0, \quad G_{34}(z_n) = 0,\\ &H_1(z_n) = 0, \quad H_2(z_n) > 0, \quad H_{31}(z_n) = 0, \quad H_{32}(z_n) > 0, \quad H_{33}(z_n) > 0, \quad H_{34}(z_n) = 0.\end{aligned}$$

This implies that the left-hand side of the above equation vanishes and therefore that $H_{32}(z_n) = 0$ and $G_{32}(z_n) = 0$. However, if this is the case, then $G(z_n)^\top H(z_n) = 0$ which contradicts our assumption that $G(z_n)^\top H(z_n) = t > 0$. □

The next result is analogous to Theorem 3.1 and relates MPEC multipliers at $\bar{z}$ to the NLP multipliers of an approaching sequence of stationary points of NLP(t_n).

THEOREM 8.2. *Let $\{t_n\}$ be a sequence of positive scalars tending to zero, let z_n be a stationary point of NLP(t_n) tending to $\bar{z}$ and suppose MPEC-LICQ holds at $\bar{z}$. Then for every sufficiently large n NLP(t_n) has unique multipliers* (6) *$\lambda_{i,n}, \mu_{j,n}, \gamma_{k,n}, \nu_{l,n}, \delta_n$ at z_n. Moreover, the point $\bar{z}$ is a C-stationary point of NLP(0) with unique multipliers $\bar{\lambda}, \bar{\mu}, \bar{\gamma}, \bar{\nu}$ which satisfy*

$$\begin{aligned}\bar{\lambda}_i &= \lim_{n\to\infty} \lambda_{i,n} \geq 0,\\ \bar{\mu}_j &= \lim_{n\to\infty} \mu_{j,n},\\ \bar{\gamma}_k &= \lim_{n\to\infty} (\gamma_{k,n} - \delta_n H_k(z_n)),\\ \bar{\nu}_k &= \lim_{n\to\infty} (\nu_{k,n} - \delta_n G_k(z_n)).\end{aligned}$$

The point $\bar{z}$ is B-stationary for NLP(0) if and only if $\bar{\gamma}_k \geq 0$ and $\bar{\nu}_k \geq 0$ for every k with $G_k(\bar{z}) = H_k(\bar{z}) = 0$.

Proof. Renumbering components if necessary, we may assume that

$$\begin{array}{lll} G_1(\bar{z}) > 0, & G_2(\bar{z}) = 0, & G_3(\bar{z}) = 0, \\ H_1(\bar{z}) = 0, & H_2(\bar{z}) > 0, & H_3(\bar{z}) = 0. \end{array}$$

We further assume that g_1 contains the components of g that vanish at $\bar{z}$. The stationarity conditions for NLP(t) consist of the Lagrangian equation which, after a suitable rearrangement of terms, is of the form

$$\begin{aligned} -\nabla f(z)^\top - \delta(\nabla G_1(z)^\top H_1(z) + \nabla H_2(z)^\top G_2(z)) = {} & \nabla g_1(z)^\top \lambda_1 + \nabla h(z)^\top \mu \\ & -\nabla G_2(z)^\top(\gamma_2 - \delta H_2(z)) \\ & -\nabla G_3(z)^\top(\gamma_3 - \delta H_3(z)) \\ & -\nabla H_1(z)^\top(\nu_1 - \delta G_1(z)) \\ & -\nabla H_3(z)^\top(\nu_3 - \delta G_3(z)), \end{aligned}$$

the sign constraints

$$\lambda_1, \gamma_2, \gamma_3, \nu_1, \nu_3, \delta \geq 0,$$

and the complementarity conditions

$$0 = g_1(z)^\top \lambda_1 = G_i(z)^\top \gamma_i = H_j(z)^\top \nu_j = \delta(G(z)^\top H(z) - t),$$

$i = 2, 3, j = 1, 3$. Notice that, in view of the MPEC-LICQ assumption, the gradients on the right-hand side of the Lagrangian equation are linearly independent for sufficiently small t and z close to $\bar{z}$. If, on the one hand, δ is bounded as t tends to zero then the left-hand side of the equation tends to $-\nabla f(\bar{z})^\top$ and therefore the augmented multipliers converge to multipliers for the MPEC NLP(0) as indicated in the statement of the theorem. If, on the other hand, δ is unbounded then we can find a subsequence such that the left-hand side of the equation, after division by δ, tends to zero. Hence, due to the linear independence of the gradients, the correspondingly scaled multipliers on the right-hand side tend to zero as well. If $H_i(z_n) > 0$ for an infinite subsequence, then $\nu_{i,n}/\delta = 0$ since the corresponding inequality is inactive and therefore $G_i(z_n)$ tends to zero. We thus conclude that $H_i(z_n) = 0$ for every sufficiently large n and every i with $G_i(\bar{z}) > 0$ and, by symmetry, $G_j(z_n) = 0$ for all sufficiently large n and all j such that $H_j(\bar{z}) > 0$. This implies, however, that the term $\nabla G_1(z)^\top H_1(z) + \nabla H_2(z)^\top G_2(z)$ vanishes eventually and the Lagrangian equation simplifies to

$$\begin{aligned} -\nabla f(z)^\top = {} & \nabla g_1(z)^\top \lambda_1 + \nabla h(z)^\top \mu \\ & -\nabla G_2(z)^\top(\gamma_2 - \delta H_2(z)) \\ & -\nabla G_3(z)^\top(\gamma_3 - \delta H_3(z)) \\ & -\nabla H_1(z)^\top(\nu_1 - \delta G_1(z)) \\ & -\nabla H_3(z)^\top(\nu_3 - \delta G_3(z)). \end{aligned}$$

Using again the fact that the gradients on the right-hand side are linearly independent for z close to $\bar{z}$, we conclude that the limiting expressions for the multipliers also hold in the case of unbounded δ. The inequality $\bar{\gamma}_k \bar{\nu}_k \geq 0$ for every k with $G_k(\bar{z}) = H_k(\bar{z}) = 0$ is a direct consequence of the complementarity conditions $H_k(z_n)\nu_{k,n} =$

$G_k(z_n)\gamma_{k,n} = 0$ because $\bar{\gamma}_k\bar{\nu}_k$ is the limit of

$$\begin{aligned}(\gamma_{k,n} - \delta_n H_k(z_n))(\nu_{k,n} - \delta_n G_k(z_n)) &= \gamma_{k,n}\nu_{k,n} - \delta_n(H_k(z_n)\nu_{k,n} + G_k(z_n)\gamma_{k,n}) \\ &\quad +\delta_n^2 H_k(z_n)G_k(z_n) \\ &= \gamma_{k,n}\nu_{k,n} + \delta_n^2 H_k(z_n)G_k(z_n) \geq 0.\end{aligned}$$

The final statement is a direct consequence of the limiting expressions and the fact that B-stationarity is equivalent to strong stationarity under MPEC-LICQ. □

Notice that the characteristic condition for B-stationarity given in the theorem is obviously satisfied if $\bar{z}$ is nondegenerate on the lower level, i.e., $G(\bar{z}) + H(\bar{z}) > 0$, or if the multiplier sequence δ_n is bounded.

The following identification result is analogous to Corollary 3.2.

COROLLARY 8.3. *If, in the setting of Theorem* 8.2*, n is sufficiently large, then*

1. $G_k(z_n) = 0$ *for all* k *with* $\bar{\gamma}_k > 0$*;*
2. $H_k(z_n) = 0$ *for all* k *with* $\bar{\nu}_k > 0$.

Proof. If $G_k(z_n) > 0$ for an infinite subsequence, then $\gamma_{k,n} = 0$ due to complementarity and thus $\bar{\gamma}_k \leq 0$ since $H_k(z_n), \delta_n \geq 0$. The same argument proves the second statement. □

We finally give the analogous result to Theorem 3.3 which implies B-stationarity of the limit points under MPEC-LICQ, ULSC and second order necessary optimality conditions at z_n.

THEOREM 8.4. *If, in addition to the assumptions of Theorem* 8.2*, second order necessary optimality conditions hold at each z_n, then $\bar{z}$ is M-stationary.*

Proof. Suppose $\bar{z}$ is not M-stationary. Since, by Theorem 8.2, $\bar{z}$ is C-stationary this implies that there exists an index k such that $G_k(\bar{z}) = H_k(\bar{z}) = 0$ and

$$\begin{aligned}\bar{\gamma}_k &= \lim_{n\to\infty}(\gamma_{k,n} - \delta_n H_k(z_n)) < 0, \\ \bar{\nu}_k &= \lim_{n\to\infty}(\nu_{k,n} - \delta_n G_k(z_n)) < 0.\end{aligned} \tag{14}$$

Hence for sufficiently large n

$$\begin{aligned}\delta_n H_k(z_n) > \gamma_{k,n} \geq 0, \\ \delta_n G_k(z_n) > \nu_{k,n} \geq 0.\end{aligned}$$

Therefore δ_n, $H_k(z_n)$ and $G_k(z_n)$ are positive and thus $\nu_{k,n} = \gamma_{k,n} = 0$ for all sufficiently large n. The relations (14) therefore simplify to

$$\begin{aligned}\bar{\gamma}_k &= -\lim_{n\to\infty}\delta_n H_k(z_n) < 0, \\ \bar{\nu}_k &= -\lim_{n\to\infty}\delta_n G_k(z_n) < 0.\end{aligned} \tag{15}$$

Let us now focus our attention to the Hessian of the term $\phi_k(z) = H_k(z)G_k(z)$ at z_n, which is of the form

$$\begin{aligned}\nabla^2\phi_k(z_n) = \nabla G_k(z_n)^\top \nabla H_k(z_n) + \nabla H_k(z_n)^\top \nabla G_k(z_n) \\ +G_k(z_n)\nabla^2 H_k(z_n) + H_k(z_n)\nabla^2 G_k(z_n)\end{aligned}$$

and appears weighted with the multiplier δ_n in the Hessian of the Lagrangian

$$L_{t_n}(z,\lambda,\mu,\gamma,\nu,\delta) = f(z)+g(z)^\top\lambda+h(z)^\top\mu-G(z)^\top\gamma-H(z)^\top\nu+\delta(G(z)^\top H(z)-t_n)$$

at z_n. Because of the MPEC-LICQ assumption we can choose, for sufficiently large n, a sequence of directions d_n with

$$\begin{aligned}
\nabla g_i(z_n)d_n &= 0, \quad i:\ g_i(\bar{z}) = 0,\\
\nabla h(z_n)d_n &= 0,\\
\nabla G_i(z_n)d_n &= 0, \quad i:\ G_i(\bar{z}) = 0, i \neq k,\\
\nabla H_i(z_n)d_n &= 0, \quad i:\ H_i(\bar{z}) = 0, i \neq k,\\
\nabla G_k(z_n)d_n &= 1,\\
\nabla H_k(z_n)d_n &= -H_k(z_n)/G_k(z_n).
\end{aligned}$$

We may assume that the sequence $\{d_n\}$ is bounded since, in view of (15), $H_k(z_n)/G_k(z_n)$ converges to $\bar{\gamma}_k/\bar{\nu}_k$. Notice that

$$\nabla\phi_k(z_n)d_n = G_k(z_n)\nabla H_k(z_n)d_n + H_k(z_n)\nabla G_k(z_n)d_n = 0.$$

Hence d_n is a critical direction of NLP(t_n) at z_n for every sufficiently large n. Moreover,

$$\begin{aligned}
&\delta_n d_n^\top \nabla^2\phi_k(z_n)d_n\\
&= \delta_n H_k(z_n)d_n^\top\nabla^2 G_k(z_n)d_n + \delta_n G_k(z_n)d_n^\top\nabla^2 H_k(z_n)d_n - \frac{2\delta_n H_k(z_n)}{G_k(z_n)},
\end{aligned}$$

which tends to $-\infty$ since the first two terms are bounded and $\delta_n H_k(z_n)$ tends to $-\bar{\gamma}_k > 0$, while $G_k(z_n)$ is a positive null-sequence. Since all other terms in

$$d_n^\top\nabla_z^2 L_{t_n}(z_n, \lambda_n, \mu_n, \gamma_n, \nu_n, \delta_n)d_n$$

are bounded, second order necessary optimality conditions fail for sufficiently large n. □

Acknowledgment. I am most grateful to the referees for their careful reading of the manuscript. The paper has benefited from their valuable suggestions and detailed comments.

REFERENCES

[1] A. Ehrenmann, *Exakte Bestrafung von Gleichgewichtsrestriktionen*, Diploma thesis, Institut für Statistik und Mathematische Wirtschaftstheorie, University of Karlsruhe, Karlsruhe, Germany, 2000.

[2] F. Facchinei, H. Jiang, and L. Qi, *A smoothing method for mathematical programs with equilibrium constraints*, Math. Program., 85 (1999), pp. 107–134.

[3] R. Fletcher, *Practical Methods of Optimization*, John Wiley, New York, 1987.

[4] R. Fletcher, *Resolving degeneracy in quadratic programming*, Ann. Oper. Res., 47 (1993), pp. 307–334.

[5] M. Fukushima and J. S. Pang, *Convergence of a smoothing continuation method for mathematical programs with equilibrium constraints*, in Ill-posed Variational Problems and Regularization Techniques, M. Théra and R. Tichatschke, eds., Springer-Verlag, New York, 1999, pp. 99–110.

[6] X. Hu, *private communication,* 2000.

[7] X. Hu, *A Penalty Method for Mathematical Programs with Complementarity Constraints*, Department of Mathematics and Statistics, University of Melbourne, Australia, 2000.

[8] H. Jiang and D. Ralph, *Smooth SQP methods for mathematical programs with nonlinear complementarity constraints*, SIAM J. Optim., 10 (2000), pp. 779–808.

[9] 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.
[10] Z. Q. Luo, J. S. Pang, and D. Ralph, *Mathematical Programs with Equilibrium Constraints*, Cambridge University Press, Cambridge, UK, 1996.
[11] J. Outrata, M. Kocvara, and J. Zowe, *Nonsmooth Approach to Optimization Problems with Equilibrium Constraints: Theory, Applications and Numerical Results*, Kluwer Academic Publishers, Dordrecht, the Netherlands, 1998.
[12] J. Outrata, *Optimality conditions for a class of mathematical programs with equilibrium constraints*, Math. Oper. Res., 24 (1999), pp. 627–644.
[13] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.
[14] H. Scheel and S. Scholtes, *Mathematical programs with complementarity constraints: Stationarity, optimality, and sensitivity*, Math. Oper. Res., 25 (2000), pp. 1–22.
[15] S. Scholtes and M. Stöhr, *Exact penalization of mathematical programs with equilibrium constraints*, SIAM J. Control Optim., 37 (1999), pp. 617–652.
[16] S. Scholtes and M. Stöhr, *How Stringent is the Linear Independence Assumption for Mathematical Programs with Stationarity Constraints?*, Working Paper, Judge Institute of Management Studies, University of Cambridge, UK, 2000.
[17] M. Stöhr, *Nonsmooth Trust Region Methods and Their Applications to Mathematical Programs with Equilibrium Constraints*, Shaker-Verlag, Aachen, Germany, 1999.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 937–961

SECOND-ORDER ALGORITHMS FOR GENERALIZED FINITE AND SEMI-INFINITE MIN-MAX PROBLEMS*

ELIJAH POLAK†, LIQUN QI‡, AND DEFENG SUN§

Abstract. We present two second-order algorithms, one for solving a class of finite generalized min-max problems and one for solving semi-infinite generalized min-max problems. Our algorithms make use of optimality functions based on second-order approximations to the cost function and of corresponding search direction functions. Under reasonable assumptions we prove that both of these algorithms converge Q-superlinearly, with rate at least 3/2.

This paper is a continuation of [E. Polak, L. Qi, and D. Sun, *Comput. Optim. Appl.*, 13 (1999), pp. 137–161].

Key words. generalized min-max problems, consistent approximations, optimality functions, second-order methods, superlinear convergence

AMS subject classifications. 65K05, 90C34, 90C47

PII. S1052623499358951

1. Introduction.

As is also the case with ordinary min-max problems, generalized min-max problems can be either finite or semi-infinite. Both are of the form

$$\mathbf{P} \qquad \min_{x\in\Re^n} f^0(x), \tag{1.1}$$

where

$$f^0(x) = F(\psi(x)) \ , \tag{1.2}$$

with $F : \Re^m \to \Re$ is a smooth function and $\psi : \Re^n \to \Re^m$ is a nonsmooth, vector-valued function. In the case of generalized finite min-max problems, the components of $\psi(\cdot)$ are of the form[1]

$$\psi^j(x) = \max_{k\in\mathbf{q_j}} \ f^{j,k}(x) \ , \tag{1.3}$$

where the functions $f^{j,k} : \Re^n \to \Re$, $j \in \mathbf{m}$ and $k \in \mathbf{q_j}$, are continuously differentiable and the sets $\mathbf{q_j} := \{1, 2, ..., q_j\}$ are of finite cardinality.[2]

In semi-infinite generalized min-max problems the components of $\psi(\cdot)$ are of the form

$$\psi^j(x) = \max_{y_j\in Y_j} \phi^j(x, y_j) \ , \tag{1.4}$$

*Received by the editors July 12, 1999; accepted for publication (in revised form) September 18, 2000; published electronically March 28, 2001. The research of this paper was supported by the National Science Foundation under grants NSF-INT-9725220 and ECS-9900985 and the Australian Research Council.

http://www.siam.org/journals/siopt/11-4/35895.html

†Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720 (polak@optimum.eecs.berkeley.edu).

‡Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (maqilq@polyu.edu.hk), and School of Mathematics, The University of New South Wales, Sydney, NSW 2052, Australia (L.Qi@unsw.edu.au).

§School of Mathematics, The University of New South Wales, Sydney, NSW 2052, Australia (sun@maths.unsw.edu.au).

[1]We denote components of a vector by superscripts and elements of a sequence or a set by subscripts.

[2]Given any positive integer q, we use the notation $\mathbf{q} := \{1, 2, \ldots, q\}$.

where the functions $\phi^j : \Re^n \times \Re^{m_j} \to \Re$, $j \in \mathbf{m}$, and $Y_j \subset \Re^{m_j}$, $j \in \mathbf{m}$.

Finite generalized min-max problems are obviously a special case of semi-infinite generalized min-max problems, since when the sets

$$Y_j = \{y_{j,k}\}_{k \in \mathbf{q_j}} \ , \tag{1.5}$$

we can define the functions $f^{j,k}(x)$ by

$$f^{j,k}(x) := \phi^j(x, y_{j,k}) \ . \tag{1.6}$$

The best known generalized min-max problem occurs when an optimization problem with a max function cost and equality and inequality constraints is set up for solution using exact penalty functions, which results in an unconstrained optimization problem with $f^0(x)$ in (1.1) of the form

$$f^0(x) = \max_{i \in \mathbf{p}} c^i(x) + \pi_e \sum_{j=1}^{q} |g^j(x)| + \pi_i \sum_{k=1}^{r} \max\{0, f^k(x)\} \ , \tag{1.7}$$

where π_e and π_i are two positive penalty parameters.

Another simple example occurs in a least squares problem involving max functions, in which case

$$f^0(x) = \sum_{j=1}^{q} \psi^j(x)^2 \ , \tag{1.8}$$

where each $\psi^j(x)$ is as in (1.3).

As a last example, in trying to approximate a structural optimization problem, the aim of which was to minimize the sum of the probability of failure[3] plus the cost of the steel in the structure, using linearizations of a state-limit function, we obtained a cost function of the form

$$f^0(x) = F(-a/(\psi(x) + b)), \tag{1.9}$$

where $F'(y) > 0$, $a > 0$,

$$\psi(x) = \max_{u \in B_\rho} g(x, u) \ , \tag{1.10}$$

B_ρ is a ball of radius ρ, centered at the origin in the space of the random variables u, and $g(x, u)$ is a smooth state-limit function which defined the boundary between outcomes that result in structural failure from those that do not [4].

Functions of the form $f^0(x) = F(\psi(x))$, with $\psi(\cdot)$ as in (1.4), are the best known examples of quasi-differentiable functions and are treated in depth in [3]. Hence generalized min-max problems can be solved using algorithms developed for quasi-differentiable functions; see, e.g., [3, 6, 7, 8]. Under the additional assumption that $\partial F(y)/\partial y^j > 0$ for all $y \in \Re^m$ and $j = 1, \ldots, m$, finite generalized min-max problems

[3]The probability of failure was given by $\int_{g(x,u) \geq 0} \phi(u) du$, with $\phi(\cdot)$ the normal probability density function.

can be solved using transformations[4] into a smooth, constrained nonlinear programming problem (see, e.g., [1, 5, 9]). Direct methods that depend on the assumption that $\partial F(y)/\partial y^j > 0$ for all $y \in \Re^m$ and $j = 1, \ldots, m$ can be found, for example, in [6, 8] and in [17].

We will consider semi-infinite generalized min-max problems under the following hypotheses.

Assumption 1.1. We will assume that

(a) the functions $F(\cdot)$ and $\phi^j(\cdot, y)$, $j \in \mathbf{m}$, $y \in Y_j$, are at least once continuously differentiable;

(b) there exists a positive number $c_F > 0$ such that $\partial F(y)/\partial y^j \geq c_F$ for all $y \in \Re^m$ and $j \in \mathbf{m}$;

(c) the sets Y_j are either compact sets of infinite cardinality, or sets of finite cardinality, of the form given in (1.5).

Parts (a) and (b) of Assumption 1.1 ensure that when both the $F(\cdot)$ and the $\psi^j(\cdot)$ are convex, the function $f^0(\cdot)$ is also convex. In addition, as we will see, when all parts of Assumption 1.1 hold, the function $f^0(\cdot)$ has a subgradient. In [17], this fact was used in defining an optimality function and an associated descent direction for the problem **P** and in extending the Pshenichnyi–Pironneau–Polak (PPP) Algorithm 4.1 in [13] (see also [18, 10, 11]) to finite generalized min-max problems and the Polak–He PPP Rate-Preserving Algorithm 3.4.9 in [13] (see also [14]) to semi-infinite generalized min-max problems.

In this paper we make use of the following observations, described in section 3.3 of [13] and also used in [15] and [16], for constructing Q-superlinearly converging algorithms for solving finite and semi-infinite min-max problems, of the form (1.1) and (1.2).

First, suppose that the sets Y_j, $j \in \mathbf{m}$, are as in (1.5), i.e., they are of finite cardinality; that the cost function $f^0(\cdot)$ is strongly convex at the minimizer $\hat{x}$, i.e., there exist $\alpha < \infty$ such that

$$f^0(x_i) - f^0(\hat{x}) \geq \alpha \|x_i - \hat{x}\|^2 \ ; \tag{1.11}$$

and that we have a local model $\hat{f}^0(x_i, x - x_i)$ for the cost function at x_i, with the property that for some $\kappa < \infty$,

$$|f^0(x) - \hat{f}^0(x_i, x - x_i)| \leq \kappa \|x - x_i\|^3 \ . \tag{1.12}$$

Then, a local algorithm of the form

$$x_{i+1} \in \arg \min_{x \in \Re^n} \hat{f}^0(x_i, x - x_i) \tag{1.13}$$

converges superlinearly, and, in particular, there exists a $\kappa' < \infty$ such that

$$\|x_{i+1} - \hat{x}\| \leq \kappa' \|x_i - \hat{x}\|^{3/2} \ . \tag{1.14}$$

[4]These transformations result in a smooth problem with more variables than in the nonsmooth problem. There is a fair bit of anecdotal evidence that they can induce considerable ill-conditioning in the smooth problem because they introduce arbitrary scaling. In particular, all methods based on the smooth transformations require the linear independent constraint qualification (LICQ) to be satisfied, which is unlikely to be true for the problem considered here, and some of these methods also require the strict complementarity condition to hold. Instead of using smooth transformations, we directly exploit the problem structure to avoid assuming either LICQ or the strict complementarity condition. However, we do need to solve a slightly more complicated subproblem at each iteration than methods based on smooth transformations.

Next, consider a problem $\mathbf{P}$, with a unique solution $\hat{x}$, and a sequence of approximating problems $\mathbf{P}_N$, with unique solutions $\hat{x}_N$, such that $\hat{x}_N \to \hat{x}$, as $N \to \infty$. Suppose we have an algorithm for solving the problems $\mathbf{P}_N$ such that the iterates that it constructs satisfy the relation

$$\|x_{i+1} - \hat{x}_N\| \leq \gamma \|x_i - \hat{x}_N\|^\tau \tag{1.15}$$

for some $\gamma < \infty$ and $\tau > 1$. If we choose N at each iteration so that

$$\|\hat{x}_N - \hat{x}\| \leq \gamma' \|x_i - \hat{x}_N\|^\sigma \tag{1.16}$$

for some $\gamma' < \infty$ and any $1 < \tau < \sigma$, then there exists a γ'' such that

$$\|x_{i+1} - \hat{x}\| \leq \gamma'' \|x_i - \hat{x}\|^\tau. \tag{1.17}$$

Note that our convergence analysis is heavily dependent on Assumption 2.4, to be introduced in section 2, and hence our results are valid only for convex problems.

In section 2, we present a continuous optimality function and its associated search direction function which, together with a backstepping rule, constitute the backbone of our algorithms. In section 3, we extend the Polak–Mayne–Higgins Newton's method [15], for solving finite min-max problems, to generalized finite min-max problems. We prove the Q-superlinear convergence of this extension in section 4. In section 5, we make use of the theory of consistent approximations developed in [13] and the algorithm presented in section 3 to develop an algorithm for solving generalized semi-infinite min-max problems and prove its convergence and Q-superlinear convergence. Section 6 is devoted to some numerical results to demonstrate the behavior of the proposed algorithms. We sum up in the concluding section 7.

2. Optimality conditions. We will now present optimality conditions for the semi-infinite generalized min-max problem, defined in (1.1), (1.2), (1.4), both in "classical" form and in terms of an optimality function which leads to a superlinearly converging second-order algorithm.

LEMMA 2.1 (see [17]). *Suppose that $F : \Re^m \to \Re$ is continuously differentiable and that $\psi : \Re^n \to \Re^m$ is a locally Lipschitz continuous function that has directional derivatives at every $x \in \Re^n$. Let $f^0 : \Re^n \to \Re$ be defined by*

$$f^0(x) = F(\psi(x)) \ . \tag{2.1}$$

Then, given any $x \in \Re^n$, and any direction vector $h \in \Re^n$, the function $f^0(\cdot)$ has a directional derivative $df^0(x;h)$ which is given by

$$df^0(x;h) = \langle \nabla F(\psi(x)), d\psi(x;h) \rangle \ . \tag{2.2}$$

Suppose that Assumption 1.1 is satisfied. Then it follows from Lemma 2.1 that the directional derivative of $f^0(\cdot)$, at a point $x \in \Re^n$ in the direction h, is given by

$$\left\{ \begin{array}{ll} df^0(x;h) & = \displaystyle\sum_{j \in \mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(x)) d\psi^j(x;h) \\ & \\ & = \displaystyle\sum_{j \in \mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(x)) \max_{y_j \in \hat{Y}_j(x)} \langle \nabla_x \phi^j(x, y_j), h \rangle, \end{array} \right. \tag{2.3}$$

where

$$\hat{Y}_j(x) := \{y_j \in Y_j \mid \phi^j(x, y_j) = \psi^j(x)\}. \tag{2.4}$$

When all the sets Y_j are as in (1.5), (2.3) assumes the form

$$df^0(x;h) = \sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(x)) \max_{k\in\hat{\mathbf{q}}_\mathbf{j}(\mathbf{x})} \langle \nabla f^{j,k}(x), h\rangle \ , \tag{2.5}$$

where the functions $f^{j,k}(\cdot)$ are defined by

$$f^{j,k}(x) := \phi^j(x, y_{j,k}),\ k \in \mathbf{q_j}, \tag{2.6}$$

and the sets $\hat{\mathbf{q}}_\mathbf{j}(x)$ by

$$\hat{\mathbf{q}}_\mathbf{j}(x) := \{k \in \mathbf{q_j} \mid f^{j,k}(x) = \psi^j(x)\} \ . \tag{2.7}$$

Hence the following result is obvious.

THEOREM 2.2. *Suppose that $\hat{x}$ is a local minimizer for the problem* (1.1), (1.2), (1.4). *Then for all $h \in \Re^n$,*

$$\left\{ \begin{array}{ll} df^0(\hat{x};h) & = \displaystyle\sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(\hat{x}))d\psi^j(\hat{x};h) \\ & \\ & = \displaystyle\sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(\hat{x})) \max_{y_j\in\hat{Y}_j(\hat{x})} \langle \nabla_x \phi^j(\hat{x}, y_j), h\rangle \geq 0 \ . \end{array} \right. \tag{2.8}$$

Furthermore, (2.8) *holds if and only if $0 \in \partial f^0(\hat{x})$, where the subgradient $\partial f^0(\hat{x})$ is given by*

$$\partial f^0(\hat{x}) = \sum_{j\in\mathbf{m}} \left\{ \mathrm{conv}_{y_j\in\hat{Y}_j(\hat{x})} \left\{ \frac{\partial F}{\partial y^j}(\psi(\hat{x}))\nabla_x \phi^j(\hat{x}, y_j) \right\} \right\} \ . \tag{2.9}$$

Since (2.8) is a necessary condition of optimality, any point $\hat{x} \in \Re^n$ that satisfies (2.8) will be called *stationary*.

When all the sets Y_j are of the form (1.5), the expressions (2.8) and (2.9) assume the following form:

$$df^0(\hat{x};h) = \sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(\hat{x})) \max_{k\in\hat{\mathbf{q}}_\mathbf{j}(\hat{x})} \langle \nabla f^{j,k}(\hat{x}), h\rangle \geq 0 \quad \forall\ h \in \Re^n, \tag{2.10}$$

$$\partial f^0(\hat{x}) = \sum_{j\in\mathbf{m}} \mathrm{conv}_{k\in\hat{\mathbf{q}}_\mathbf{j}(\hat{x})} \left\{ \frac{\partial F}{\partial y^j}(\psi(\hat{x}))\nabla f^{j,k}(\hat{x}) \right\} \ . \tag{2.11}$$

DEFINITION 2.3. *We will say that $\theta : \Re^n \to \Re$ is an* optimality function *for problem* (1.1), (1.2), (1.4) *if*

(a) *$\theta(\cdot)$ is upper semicontinuous,*

(b) *$\theta(x) \leq 0$ for all $x \in \Re^n$, and*

(c) *for any $\hat{x} \in \Re^n$,* (2.8) *holds if and only if $\theta(\hat{x}) = 0$.*

Assumption 2.4. We will assume that

(a) the functions $\phi^j(\cdot, y_j)$, $j \in \mathbf{m}$, $y_j \in Y_j$, and $F(\cdot)$, in (1.1), (1.2), (1.4), are twice Lipschitz continuously differentiable on bounded sets,

(b) the functions $\phi^j(\cdot, y_j)$, $\nabla_x \phi^j(\cdot, y_j)$, and $\nabla_x^2 \phi^j(\cdot, y_j)$ are locally Lipschitz continuous, $j \in \mathbf{m}$, $y_j \in Y_j$, and

(c) there exist constants $0 < c \leq C < \infty$, such that for all $j \in \mathbf{m}$, $y_j \in Y_j$, $x \in \Re^n$, $h \in \Re^n$, and $w \in \Re^m$,

$$c\|h\|^2 \leq \langle h, \nabla_x^2 \phi^j(x, y_j)h\rangle \leq C\|h\|^2 \tag{2.12}$$

and

$$0 \leq \langle w, \nabla^2 F(\psi(x))w\rangle \leq C\|w\|^2. \tag{2.13}$$

For the sake of convenience, for any $x, h \in \Re^n$ and $w \in \Re^m$, we define

$$u(x, h, w) := \langle \nabla F(\psi(x)), \hat{\psi}(x, h) - \psi(x) + w\rangle \tag{2.14}$$

and

$$v(x, h, w) := \tfrac{1}{2}\langle \hat{\psi}(x, h) - \psi(x) + w, \nabla^2 F(\psi(x))(\hat{\psi}(x, h) - \psi(x) + w)\rangle, \tag{2.15}$$

where $\hat{\psi}(x, h) = (\hat{\psi}^1(x, h), \ldots, \hat{\psi}^m(x, h))$, and

$$\hat{\psi}^j(x, h) := \max_{y_j \in Y_j} \{\phi^j(x, y_j) + \langle \nabla_x \phi^j(x, y_j), h\rangle + \tfrac{1}{2}\langle h, \nabla_x^2 \phi^j(x, y_j)h\rangle\}. \tag{2.16}$$

The reason for the introduction of the artificial variable w is as follows. The function

$$\tilde{f}^0(x, h) := F(\psi(x)) + u(x, h, 0) + v(x, h, 0) \tag{2.17}$$

is a perfectly good second-order approximation to $F(\psi(x+h))$, but unfortunately, it is not always convex and hence leads to problems in developing an algorithm for solving semi-infinite generalized min-max problems. By introducing the artificial variable w, we can define the function

$$\hat{f}^0(x, h) := \min_{w \in \Re^m_+} \{F(\psi(x)) + u(x, h, w) + v(x, h, w)\} \tag{2.18}$$

which, as we will later see, is a convex second-order approximation to $F(\psi(x + h))$ and hence much more useful in algorithm construction.

We define the function $\theta : \Re^n \to \Re$ and the associated search direction function $H : \Re^n \to \Re^n$ by

$$\theta(x) := \min_{h \in \Re^n} \{ \min_{w \in \Re^m_+} [u(x, h, w) + v(x, h, w)]\} \tag{2.19}$$

and

$$H(x) := \arg\min_{h \in \Re^n} \{ \min_{w \in \Re^m_+} [u(x, h, w) + v(x, h, w)]\} \ . \tag{2.20}$$

Note that

$$\theta(x) = \min_{h \in \Re^n} \{\hat{f}^0(x, h) - f^0(x)\}. \tag{2.21}$$

We will shortly see that the function $\theta(\cdot)$ is an optimality function for the problem (1.1), (1.2), (1.4). For any $y, \delta y \in \Re^m$, let

$$\hat{F}(y,\delta y) := \min_{w \in \Re^m_+} \{F(y) + \langle \nabla F(y), \delta y + w\rangle + \tfrac{1}{2}\langle \delta y + w, \nabla^2 F(y)(\delta y + w)\rangle\}. \tag{2.22}$$

LEMMA 2.5. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. For any* $y, \delta y \in \Re^m$, *let* $\Omega^*(y,\delta y) \subset \Re^m_+$ *be the solution set of* (2.22). *Then* $\Omega^*(y,\delta y)$ *is nonempty and compact and for any* $w^* \in \Omega^*(y,\delta y)$, *we have*

$$\nabla F(y) + \nabla^2 F(y)\delta y + \nabla^2 F(y) w^* \geq 0. \tag{2.23}$$

Proof. Since $\nabla F(y) > 0$ and $\nabla^2 F(y)$ is positive semidefinite, for any $w \in \Re^m_+$ and $\|w\| \to \infty$ we have

$$F(y) + \langle \nabla F(y), \delta y + w\rangle + \tfrac{1}{2}\langle \delta y + w, \nabla^2 F(y)(\delta y + w)\rangle \to +\infty. \tag{2.24}$$

Thus, $\Omega^*(y,\delta y)$ is nonempty and compact.

Suppose that $w^* \in \Omega^*(y,\delta y)$. Then w^* satisfies the following first-order optimality conditions which follow directly from (and are equivalent to) the KKT conditions:

$$\begin{cases} \nabla F(y) + \nabla^2 F(y)(\delta y + w^*) - \lambda^* = 0, \\ \\ w^* \geq 0,\ \lambda^* \geq 0,\ \langle w^*, \lambda^*\rangle = 0, \end{cases} \tag{2.25}$$

i.e.,

$$\begin{cases} \nabla F(y) + \nabla^2 F(y)(\delta y + w^*) \geq 0, \\ \\ w^* \geq 0, \\ \\ \langle w^*, \nabla F(y) + \nabla^2 F(y)(\delta y + w^*)\rangle = 0. \end{cases} \tag{2.26}$$

Clearly, (2.26) implies that for any $w^* \in \Omega^*(y,\delta y)$, we have

$$\nabla F(y) + \nabla^2 F(y)\delta y + \nabla^2 F(y) w^* \geq 0. \qquad \square \tag{2.27}$$

LEMMA 2.6. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. Then for any* $z \in \Re^n$ *there exists an* $\varepsilon > 0$ *such that for all* $h \in \Re^n$ *with* $\|h\| \leq \varepsilon$ *and for all* $x \in \Re^n$ *with* $\|x - z\| \leq \varepsilon$ *we have*

$$\hat{f}^0(x,h) = F(\psi(x)) + u(x,h,0) + v(x,h,0), \tag{2.28}$$

i.e.,

$$\hat{f}^0(x,h) = \tilde{f}^0(x,h), \tag{2.29}$$

where $\tilde{f}^0(\cdot,\cdot)$ *is defined by* (2.17).

Proof. Since $F(\psi(x)) + u(x,h,\cdot) + v(x,h,\cdot)$ is a convex quadratic function, any $w \in \Re^m$ satisfying the first-order conditions

$$\begin{cases} w \geq 0, \\ \\ \nabla F(\psi(x)) + \nabla^2 F(\psi(x))(\hat{\psi}(x,h) - \psi(x) + w) \geq 0, \\ \\ \langle w, \nabla F(\psi(x)) + \nabla^2 F(\psi(x))(\hat{\psi}(x,h) - \psi(x) + w)\rangle = 0 \end{cases} \tag{2.30}$$

is a solution of (2.18). Then, because $\partial F(y)/\partial y_j \geq c_F$, for every $j \in \mathbf{m}$ and $y \in \Re^m$, and $\hat{\psi}(\cdot,\cdot)$ is uniformly continuous on any compact set and $\hat{\psi}(x,0) = \psi(x)$, we see that for any $z \in \Re^n$ there exists an $\varepsilon > 0$ such that for all $h \in \Re^n$ with $\|h\| \leq \varepsilon$ and for all $x \in \Re^n$ with $\|x - z\| \leq \varepsilon$, $w = 0$ satisfies (2.30). This implies that for all those h and x, we have

$$\hat{f}^0(x,h) = F(\psi(x)) + u(x,h,0) + v(x,h,0) = \tilde{f}^0(x,h). \tag{2.31}$$

Hence our proof is complete. □

The above lemma shows that $\hat{f}^0(x,h)$ is identical to $\tilde{f}^0(x,h)$ for all h sufficiently small. This fact will be used in proving our superlinear convergence results.

In general, $\tilde{f}^0(x,h)$ is not convex in h. We will now show that $\hat{f}^0(x,h)$ is convex in h.

LEMMA 2.7. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. Then for any fixed $x \in \Re^n$, $\hat{f}^0(x,\cdot)$ is a convex function. Moreover, $\hat{f}^0(\cdot,\cdot)$ is continuous.*

Proof. First we will show that $\hat{f}^0(x,\cdot)$ is a convex function. For any $y \in \Re^m$ and $\delta y \in \Re^m$, we have

$$\hat{F}(y,\delta y) = F(y) + \langle \nabla F(y), \delta y\rangle + \tfrac{1}{2}\langle \delta y, \nabla^2 F(y)\delta y\rangle + S(\delta y), \tag{2.32}$$

where

$$S(\delta y) = \min_{w \in \Re^m_+} \langle \nabla F(y) + \nabla^2 F(y)\delta y, w\rangle + \tfrac{1}{2}\langle w, \nabla^2 F(y) w\rangle \ . \tag{2.33}$$

It is easy to verify that $S(\delta y)$ is a concave function and that its subgradient is given by

$$\partial S(\delta y) = \operatorname{conv}\{\nabla^2 F(y) w^* : w^* \in \Omega^*(y,\delta y)\}, \tag{2.34}$$

where $\Omega^*(y,\delta y) \subset \Re^m_+$ is the solution set of (2.33). It now follows from (2.32) that for any $y \in \Re^m$, $\hat{F}(y,\cdot)$ is locally Lipschitz continuous and that its generalized gradient at δy in the sense of Clarke [2] is given by

$$\partial_{\delta y}\hat{F}(y,\delta y) = \operatorname{conv}\{\nabla F(y) + \nabla^2 F(y)\delta y + \nabla^2 F(y) w^* : w^* \in \Omega^*(y,\delta y)\}. \tag{2.35}$$

Since, by Lemma 2.5, for any $w^* \in \Omega^*(y,\delta y)$,

$$\nabla F(y) + \nabla^2 F(y)\delta y + \nabla^2 F(y) w^* \geq 0, \tag{2.36}$$

we conclude that $s \geq 0$ for any $s \in \partial_{\delta y}\hat{F}(y,\delta y)$. Hence, since $\hat{\psi}^j(x,\cdot)$ is convex for every $j \in \{1,\ldots,m\}$, it follows that $\hat{f}^0(x,h) = \hat{F}(\psi(x), \hat{\psi}(x,h) - \psi(x))$ is convex in $h \in \Re^n$ (because it is the composition of a convex function with positive elements in the generalized gradient and a vector function whose components are convex).

Next, we will prove that $\hat{f}^0(x,h)$ is continuous. First, since $\partial F(y)/\partial y_j \geq c_F > 0$ and $\nabla^2 F(y)$ is positive semidefinite for all $j \in \{1,\ldots,m\}$ and $y \in \Re^m$, it follows from (2.22) that $\Omega^*(y,\delta y)$ is uniformly bounded in a neighborhood of given point $(z,\delta z) \in \Re^m \times \Re^m$. It now follows from Corollary 5.4.2 in [13] that $\hat{F}(\cdot,\cdot)$ is continuous. Hence

$$\hat{F}(y,\delta y) \to \hat{F}(z,\delta z) \quad \text{as} \quad y \to z, \delta y \to \delta z \ , \tag{2.37}$$

which implies that $\hat{f}^0(x,h)$ is continuous on $\Re^n \times \Re^n$ because

$$\hat{f}^0(x,h) = \hat{F}(\psi(x), \hat{\psi}(x,h) - \psi(x)) \tag{2.38}$$

with $y := \psi(x)$ and $\delta y := \hat{\psi}(x,h) - \psi(x)$. □

The following theorem shows that $\theta(\cdot)$ is indeed an optimality function for the problem (1.1), (1.2), (1.4) and that the set-valued function $H(\cdot)$ is a descent direction function for $f^0(\cdot)$.

THEOREM 2.8. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. Consider the functions $\theta(\cdot)$ and $H(\cdot)$ defined by* (2.19) *and* (2.20)*, respectively. Then the following hold:*

(i) *For all $x \in \Re^n$,*

$$\theta(x) \leq 0 \ . \tag{2.39}$$

(ii) *For all $x \in \Re^n$,*

$$df^0(x;h) \leq \theta(x) - \gamma\|h\|^2 \ \forall \ h \in H(x), \tag{2.40}$$

where $df^0(x;h)$ is the directional derivative of f^0 at x in the direction h and $\gamma = \frac{1}{2}mc_Fc$.

(iii) *For any $x \in \Re^n$, $0 \in \partial f^0(x)$ if and only if $\theta(x) = 0$, where $\partial f^0(x)$ is the subgradient of $f^0(\cdot)$ at x, defined in* (2.9)*. Moreover, for any $x \in \Re^n$ such that $\theta(x) = 0$ we have $H(x) = \{0\}$.*

(iv) *The set-valued map $H(\cdot)$ is* (a) *bounded on bounded sets,* (b) *compact valued, and* (c) *outer-semicontinuous, i.e., for any $x \in \Re^n$, $H(x)$ is closed and, for every compact set S such that $H(x) \cap S = \emptyset$, there exists a $\rho > 0$ such that $H(z) \cap S = \emptyset$ for all $z \in B(x,\rho) := \{y \in \Re^n | \|y - x\| \ \leq \rho\}$.*

(v) *The function $\theta(\cdot)$ is continuous.*

Proof. (i) Since $h = 0$ is admissible in (2.19) that $\theta(x) \leq 0$ for all $x \in \Re^n$.

(ii) Since $\hat{Y}_j(x) \subset Y_j$, it follows directly from the definition of $\theta(x)$ in (2.19) that for any $h \in H(x)$,

$$\left\{ \begin{array}{ll} \theta(x) & \geq \min\limits_{w \in \Re^m_+} \langle \nabla F(\psi(x)), \hat{\psi}(x,h) - \psi(x) + w \rangle \\ & \\ & = \langle \nabla F(\psi(x)), \hat{\psi}(x,h) - \psi(x) \rangle \\ & \\ & \geq \sum\limits_{j \in \mathbf{m}} \dfrac{\partial F}{\partial y^j}(\psi(x)) \max\limits_{y_j \in \hat{Y}_j(x)} \{\phi^j(x,y_j) - \psi^j(x) \\ & \qquad\qquad + \langle \nabla_x \phi^j(x,y_j), h \rangle + \frac{1}{2}c\|h\|^2\} \\ & \\ & \geq df^0(x,h) \ + \ \frac{1}{2}mc_Fc\|h\|^2. \end{array} \right. \tag{2.41}$$

Thus we have shown that (2.40) holds.

(iii) For any $x \in \Re^n$, let

$$\eta(x) := \min_{h \in \Re^n} \min_{w \in \Re^m_+} u(x,h,w) = \min_{h \in \Re^n} u(x,h,0) \ . \tag{2.42}$$

We will first prove that

$$\theta(x) = 0 \Longleftrightarrow \eta(x) = 0. \tag{2.43}$$

It is easy to see that $\eta(x) = 0 \Rightarrow \theta(x) = 0$ because $\theta(x) \geq \eta(x)$ and $\theta(x) \leq 0$. Hence we only need to show that $\theta(x) = 0 \Rightarrow \eta(x) = 0$.

Suppose that $\theta(x) = 0$ but $\eta(x) < 0$. Then, there exists an $h' \in \Re^n$ such that $\eta(x) = u(x, h', 0) < 0$.

For any $j \in \{1, \ldots, m\}$, we have

$$\left\{\begin{array}{l} \hat{\psi}^j(x,h) - \psi^j(x) \\ \\ = \max\limits_{y_j \in Y_j} \{\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \frac{1}{2}\langle h, \nabla_{xx}\phi^j(x,y_j)h\rangle\} - \psi^j(x) \\ \\ \leq \max\limits_{y_j \in Y_j} \{\langle \nabla_x \phi^j(x,y_j), h\rangle + \frac{1}{2}\langle h, \nabla_{xx}\phi^j(x,y_j)h\rangle\} \end{array}\right. \tag{2.44}$$

and

$$\left\{\begin{array}{l} \hat{\psi}^j(x,h) - \psi^j(x) \\ \\ \geq \max\limits_{y_j \in \hat{Y}_j(x)} \{\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \frac{1}{2}\langle h, \nabla_{xx}\phi^j(x,y_j)h\rangle\} - \psi^j(x) \\ \\ = \max\limits_{y_j \in \hat{Y}_j(x)} \{\langle \nabla_x \phi^j(x,y_j), h\rangle + \frac{1}{2}\langle h, \nabla_{xx}\phi^j(x,y_j)h\rangle\} \\ \\ \geq \min\limits_{y_j \in Y_j} \{\langle \nabla_x \phi^j(x,y_j), h\rangle + \frac{1}{2}\langle h, \nabla_{xx}\phi^j(x,y_j)h\rangle\} \ . \end{array}\right. \tag{2.45}$$

Thus, there exists a constant C_0 such that

$$\|\hat{\psi}(x,h) - \psi(x)\| \leq C_0 \max\{\|h\|, \|h\|^2\} \quad \forall\ h \in \Re^n, \tag{2.46}$$

which further implies that there exists a constant C_1 such that

$$0 \leq v(x,h,0) \leq C_1 \max\{\|h\|^2, \|h\|^4\} \quad \forall\ h \in \Re^n. \tag{2.47}$$

Since $u(x,\cdot,0)$ is a convex function and $u(x,0,0) = 0$, for $\lambda > 0$ sufficiently small we have

$$\left\{\begin{array}{lcl} u(x,\lambda h',0) + v(x,\lambda h',0) & \leq & \lambda u(x,h',0) + \lambda^2 C_1 \|h'\|^2 \\ & = & \lambda\eta(x) + \lambda^2 C_1 \|h'\|^2 \\ & < & 0, \end{array}\right. \tag{2.48}$$

which contradicts that $\theta(x) = 0$. Hence $\theta(x) = 0 \Rightarrow \eta(x) = 0$.

Next, with $\partial f^0(x)$ the subgradient of $f^0(\cdot)$ at x, defined in (2.9), by emulating the proof of Lemma 2.5.5 in [13], we can prove that for any $x \in \Re^n$, $0 \in \partial f^0(x)$ if and only if $\eta(x) = 0$, and therefore if and only if $\theta(x) = 0$.

Finally we will show that for any $x \in \Re^n$ such that $\theta(x) = 0$ we have $H(x) = \{0\}$. For the sake of contradiction, suppose that there exists an $x \in \Re^n$ such that $\theta(x) = 0$ but $H(x) \neq \{0\}$. Then there exist $0 \neq h \in \Re^n$ and $w \in \Re^m_+$ such that

$$u(x,h,w) + v(x,h,w) = \theta(x) = 0, \tag{2.49}$$

which, together with the fact that $v(x,h,w) \geq 0$ implies that $u(x,h,w) \leq 0$. Hence we conclude that both $u(x,h,w) = 0$ and $w = 0$ because otherwise $\eta(x) \leq u(x,h,0) \leq u(x,h,w) < 0$, which contradicts (2.43). However, $\eta(x) = u(x,h,0) = 0$ implies that $h = 0$ because $u(x,h,0)$ is strongly convex in h and $u(x,0,0) = 0$.

(iv) According to our definition, for each $h \in \Re^n$ there exists a $w(h) \in \Re^m_+$ such that

$$\hat{f}^0(x,h) = F(\psi(x)) + u(x,h,w(h)) + v(x,h,w(h)), \tag{2.50}$$

which, together with the fact that $\nabla F(y) > 0$, $y \in \Re^m$ and $v(x,h,w(h)) \geq 0$, implies that

$$\hat{f}^0(x,h) \geq F(\psi(x)) + u(x,h,w(h)) \geq F(\psi(x)) + u(x,h,0). \tag{2.51}$$

Since for each $j \in \{1,\ldots,m\}$ and $h \in \Re^n$,

$$\hat{\psi}^j(x,h) \geq \max_{y_j \in Y_j} \{\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \tfrac{1}{2} c\langle h,h\rangle\}\ , \tag{2.52}$$

it follows from (2.51) that for all s in any bounded neighborhood of x,

$$\hat{f}^0(s,h) \to \infty \quad \text{as} \quad \|h\| \to \infty. \tag{2.53}$$

Consequently, for any $x \in \Re^n$, $H(x)$ is nonempty and bounded and $H(\cdot)$ is bounded on bounded sets. Since $\hat{f}^0(x,h)$ is continuous (Lemma 2.7), it follows that $H(x)$ is closed. Next we will prove that for every $x \in \Re^n$ and every compact set S such that $H(x) \cap S = \emptyset$, there exists a $\rho > 0$ such that $H(z) \cap S = \emptyset$ for all $z \in B(x,\rho)$. Suppose not; then there exists an $x \in \Re^n$ and a compact set S such that $H(x) \cap S = \emptyset$ and a sequence $\{x_i\}$ converging to x such that $H(x_i) \cap S \neq \emptyset$. Hence there exists a sequence $\{h_i\}$ such that $h_i \in H(x_i) \cap S$. Since S is a compact set, without loss of generality, we can assume that

$$h_i \to \bar{h} \in S. \tag{2.54}$$

By definition of $H(x_i)$,

$$\hat{f}^0(x_i,h_i) \leq \hat{f}^0(x_i,h) \quad \forall\ h \in \Re^n. \tag{2.55}$$

Since $\hat{f}^0(\cdot,\cdot)$ is continuous, it follows from (2.55) that

$$\hat{f}^0(x,\bar{h}) \leq \hat{f}^0(x,h) \quad \forall\ h \in \Re^n, \tag{2.56}$$

which implies that $\bar{h} \in H(x)$. This contradicts that $H(x) \cap S = \emptyset$. Thus, we have shown that $H(\cdot)$ is outer-semicontinuous.

(v) Finally, it follows from Corollary 5.4.2 in Polak [13] that θ is continuous. □

By introducing an additional variable, we can rewrite the expression for $\theta(x)$, defined in (2.19), as follows:

$$\begin{cases} \theta(x) = \min\limits_{(p,h)} & \{\langle \nabla F(\psi(x)), p\rangle + \frac{1}{2}\langle p, \nabla^2 F(\psi(x))p\rangle\} \\ \text{s.t.} & p - \hat{\psi}(x,h) + \psi(x) \geq 0. \end{cases} \tag{2.57}$$

The constraints in (2.57) involve maximum functions, and hence (2.57) appears to be a nonsmooth problem. However, (2.57) can be reformulated as a smooth problem with quadratic cost and quadratic constraints, as follows:

$$\begin{cases} \theta(x) = \min\limits_{(p,h)} & \{\langle \nabla F(\psi(x)), p\rangle + \frac{1}{2}\langle p, \nabla^2 F(\psi(x))p\rangle\} \\ & \text{s.t.} \quad p^j - \phi^j(x, y_j) - \langle \nabla_x \phi^j(x, y_j), h\rangle - \frac{1}{2}\langle h, \nabla_x^2 \phi^j(x, y_j)h\rangle + \psi^j(x) \geq 0, \\ & \quad j \in \mathbf{m}, y_j \in Y_j. \end{cases} \tag{2.58}$$

Under the assumptions in this paper, (2.58) is convex, and hence can be solved using the smoothing Newton method in [19] (see [19, 21] for the details of the implementation of the smoothing Newton method as well as section 6 for numerical results). Alternatively, one can use primal-dual interior point methods, described in [20] and references therein.

THEOREM 2.9. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied and the sets* Y_j *are as in* (1.5). *For any* $x \in \Re^n$, *let* $\Gamma(x)$ *be the solution set of* (2.57), *i.e., any* $(p, h) \in \Gamma(x)$ *solves* (2.57). *Then*

(i) *problem* (2.57) *is a convex quadratic problem with convex quadratic constraints;*

(ii) *for* $x \in \Re^n$, $\Gamma(x)$ *is nonempty and compact and* $\Gamma(\cdot)$ *is outer-semicontinuous and bounded on bounded sets;*

(iii) *if* $z \in \Re^n$ *is such that* $\theta(z) = 0$, *then* $\Gamma(z) = \{(0, 0)\}$ *and there exist a neighborhood* $N(z)$ *of* z *and an* $\varepsilon > 0$ *such that for any* $(p, h) \in \Gamma(x)$, $x \in N(z)$, *we have*

$$\theta(x) \leq -\varepsilon \|h\|^2. \tag{2.59}$$

Proof. (i) Under the conditions of Assumptions 1.1 and 2.4, $\nabla^2 F(\psi(x))$ is positive semidefinite and for each $j \in \{1, 2, \ldots, m\}$, $\hat{\psi}^j(x, \cdot)$ is strongly convex. Hence (2.57) is a convex quadratic problem with convex quadratic constraints.

(ii) Since for all z in a bounded neighborhood $N(x)$ of x and $j \in \{1, 2, \ldots, m\}$,

$$\hat{\psi}^j(z, h) - \psi^j(z) \to +\infty \quad \text{as} \quad \|h\| \to \infty, \tag{2.60}$$

it follows that for all $z \in N(x)$ and $(p, h) \in \Re^m \times \Re^n$ satisfying

$$p \geq \hat{\psi}(z, h) - \psi(z), \tag{2.61}$$

we have

$$\langle \nabla F(\psi(z)), p\rangle + \tfrac{1}{2}\langle p, \nabla^2 F(\psi(z))p\rangle \geq c_F \sum_{j \in \mathbf{m}} p^j \to \infty \text{ as } \|(p, h)\| \to \infty. \tag{2.62}$$

Hence, for all $x \in \Re^n$, $\Gamma(x)$ is nonempty and compact, and $\Gamma(\cdot)$ is bounded on bounded sets.

The outer-semicontinuity of $\Gamma(\cdot)$ follows from the fact that $\theta(\cdot)$ is continuous and the constraint set in (2.57) is outer-semicontinuous.

(iii) Since $z \in \Re^n$ is such that $\theta(z) = 0$, $(0, 0) \in \Gamma(z)$. For any $x \in \Re^n$, the KKT conditions for (2.57) are

$$\begin{cases} \nabla F(\psi(x)) + \nabla^2 F(\psi(x))p = \lambda, \\ 0 \in \sum\limits_{j \in \mathbf{m}} \lambda^j \partial_h \hat{\psi}^j(x, h), \\ \lambda \geq 0,\ p - \hat{\psi}(x, h) + \psi(x) \geq 0,\ \lambda^T(p - \hat{\psi}(x, h) + \psi(x)) = 0, \end{cases} \tag{2.63}$$

where $\partial_h \hat{\psi}^j(x,h)$ is the subgradient of $\hat{\psi}^j(x,h)$ with respect to h.

Suppose that $(p,h) \in \Gamma(z)$. By (iii) of Theorem 2.8, we have $h = 0$. Hence it follows from (2.63) and the fact that $\hat{\psi}(z,0) = \psi(z)$ that

$$\langle p, \nabla F(\psi(z))\rangle + \langle p, \nabla^2 F(\psi(z))p\rangle = 0, \tag{2.64}$$

which implies that $p = 0$ because $p \geq 0$, $\nabla F(\psi(z)) > 0$ and $\nabla^2 F(\psi(z))$ is positive semidefinite. Thus, we have proved that $\Gamma(z) = \{(0,0)\}$. Hence, since $\Gamma(\cdot)$ is outer-semicontinuous, it follows that if $x \to z$ and $(p,h) \in \Gamma(x)$, then

$$(p,h) \to (0,0)\ . \tag{2.65}$$

It now follows from (2.63), (2.65), and the fact that for any $y \in \Re^m$, $\partial F(y)/\partial y^j \geq c_F > 0$ for $j \in \{1,2,\ldots,m\}$ that there exists a neighborhood $N(z)$ of z such that for all $x \in N(z)$, the multiplier λ in the KKT (2.63) must have all components positive and hence for all $x \in N(z)$, the KKT conditions for (2.57) become

$$\begin{cases} \nabla F(\psi(x)) + \nabla^2 F(\psi(x))p = \lambda, \\ 0 \in \displaystyle\sum_{j\in\mathbf{m}} \lambda^j \partial_h \hat{\psi}^j(x,h), \\ \lambda > 0,\ p - \hat{\psi}(x,h) + \psi(x) = 0. \end{cases} \tag{2.66}$$

Thus, for any $x \in N(z)$ and $j \in \{1,2,\ldots,m\}$, there exist nonnegative numbers $\mu^{j,k} \in [0,1]$ satisfying $\sum_{k\in\mathbf{q_j}} \mu^{j,k} = 1$ such that for any $(p,h) \in \Gamma(x)$

$$\sum_{j\in\mathbf{m}} \lambda^j \sum_{k\in\mathbf{q_j}} \mu^{j,k}(\nabla f^{j,k}(x) + \nabla^2 f^{j,k}(x)h) = 0, \tag{2.67}$$

where

$$\lambda = \nabla F(\psi(x)) + \nabla^2 F(\psi(x))p > 0, \tag{2.68}$$

and for any $k \in \mathbf{q_j}$ such that

$$\hat{\psi}^j(x,h) > f^{j,k}(x) + \langle \nabla f^{j,k}(x), h\rangle + \tfrac{1}{2}\langle h, \nabla^2 f^{j,k}(x)h\rangle, \tag{2.69}$$

we have

$$\mu^{j,k} = 0. \tag{2.70}$$

We conclude from (2.66), (2.67), and (2.68) that for all $x \in N(z)$ and $(p,h) \in \Gamma(x)$,

$$\left\{\begin{array}{rcl}
\theta(x) & = & \langle \nabla F(\psi(x)), p\rangle + \frac{1}{2}\langle p, \nabla^2 F(\psi(x))p\rangle \\
& = & \langle \lambda, p\rangle - \frac{1}{2}\langle p, \nabla^2 F(\psi(x))p\rangle \\
& \leq & \langle \lambda, p\rangle \\
& = & \langle \lambda, \hat{\psi}(x,h) - \psi(x)\rangle \\
& = & \displaystyle\sum_{j\in\mathbf{m}} \lambda^j \sum_{k\in\mathbf{q_j}} \mu^{j,k}[(f^{j,k}(x) - \psi^j(x)) + \langle \nabla f^{j,k}(x), h\rangle + \tfrac{1}{2}\langle h, \nabla^2 f^{j,k}(x)h\rangle] \\
& = & \displaystyle\sum_{j\in\mathbf{m}} \lambda^j \sum_{k\in\mathbf{q_j}} \mu^{j,k}[(f^{j,k}(x) - \psi^j(x)) - \tfrac{1}{2}\langle h, \nabla^2 f^{j,k}(x)h\rangle] \\
& \leq & \displaystyle\sum_{j\in\mathbf{m}} \lambda^j \sum_{k\in\mathbf{q_j}} \mu^{j,k}[(f^{j,k}(x) - \psi^j(x)) - \tfrac{1}{2}c\langle h, h\rangle] \\
& \leq & \displaystyle -\tfrac{1}{2}c\|h\|^2 \sum_{j\in\mathbf{m}} \lambda^j,
\end{array}\right. \tag{2.71}$$

where the last inequality follows from the fact that $f^{j,k}(x) \leq \psi^j(x)$ for all $k \in \mathbf{q_j}$ and $j \in \mathbf{m}$. By shrinking $N(z)$ if necessary, we conclude from (2.68), (2.71), and Assumptions 1.1 and 2.4 that there exists a positive number $\varepsilon > 0$ such that for all $x \in N(z)$ and $(p,h) \in \Gamma(x)$, $\theta(x) \leq -\varepsilon\|h\|^2$. ∎

3. An algorithm for solving generalized finite min-max problems. An algorithm for solving generalized finite min-max problems is obviously of interest in its own right. However, we will also need it as a subroutine for our algorithms for solving generalized semi-infinite min-max problems. Hence, for the time being, we will assume that the sets Y_j are of the form (1.5) and that the functions $f^{j,k}(\cdot)$ are as in (2.6). As a result, our generalized finite min-max problem assumes the form (1.1), (1.2), (1.4), with

$$\left\{\begin{array}{c}
\displaystyle\min_{x\in\Re^n} \; f^0(x) \\
f^0(x) = F(\psi(x)), \\
\psi(x) = (\psi^1(x), \ldots, \psi^m(x)), \\
\psi^j(x) = \displaystyle\max_{k\in\mathbf{q_j}} \; f^{j,k}(x), \quad j \in \mathbf{m}\ ,
\end{array}\right. \tag{3.1}$$

where, in view of Assumption 1.1, the functions $F(\cdot)$ and $f^{j,k}(\cdot)$, $j \in \mathbf{m}$, $k \in \mathbf{q_j}$ are all continuously differentiable, where $f^{j,k}(\cdot)$ are defined by (2.6).

We are now ready to state an algorithm for solving generalized finite min-max problems. This algorithm is a generalization of the Polak–Mayne–Higgins Newton's algorithm for solving finite min-max problems [15].

ALGORITHM 3.1 (solves problem (3.1)).
Parameters. $\alpha \in (0,1)$, $\beta \in (0,1)$, *and* $\delta > 0$.

Data. $x_0 \in \Re^n$.
Step 0. *Set* $i = 0$.
Step 1. *Compute the optimality function value* $\theta_i := \theta(x_i)$ *and a search direction* $h_i \in H(x_i)$ *according to the formulae* (2.19) *and* (2.20).
Step 2. *If* $\theta_i = 0$, *stop. Else, compute the step-size*

$$\lambda_i = \lambda(x_i) := \max_{k \in \mathcal{N}} \{\beta^k |\ f^0(x_i + \beta^k h_i) - f^0(x_i) - \beta^k \alpha \theta_i \leq 0\}, \tag{3.2}$$

where $\mathcal{N} := \{0, 1, 2, \ldots\}$.
Step 3. *Set*

$$x_{i+1} = x_i + \lambda_i h_i, \tag{3.3}$$

replace i *by* $i + 1$, *and go to Step* 1.

LEMMA 3.2 (see [17]). *Suppose that Assumption* 1.1 *holds. Then for any* $y, y' \in \Re^m$ *such that* $y' \geq y$,

$$F(y') - F(y) \geq c_F \sum_{j \in \mathbf{m}} (y'_j - y_j). \tag{3.4}$$

LEMMA 3.3 (see [17]). *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. Then there exists a constant* $\tau > 0$ *such that for all* $x, x' \in \Re^n$ *and* $\lambda \in [0, 1]$,

$$f^0(\lambda x + (1 - \lambda)x') \leq \lambda f^0(x) + (1 - \lambda) f^0(x') - \tfrac{1}{2}\tau\lambda(1 - \lambda)\|x - x'\|^2. \tag{3.5}$$

THEOREM 3.4. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied and that all the* Y_j, $j \in \mathbf{m}$, *are of the form* (1.5), *so that problem* (1.1), (1.2), (1.4) *reduces to problem* (3.1). *If* $\{x_i\}_{i=0}^{\infty}$ *is an infinite sequence generated by Algorithm* 3.1 *and* x^* *is the unique solution of* (3.1), *then* $\{x_i\}_{i=0}^{\infty}$ *converges to* x^*.

Proof. Suppose that $\{x_i\}_{i=0}^{\infty}$ is an infinite sequence generated by Algorithm 3.1. Since $f(\cdot)$ is strongly convex by Lemma 3.3, the sequence $\{x_i\}_{i=0}^{\infty}$ is bounded. Suppose that $\hat{x}$ is an accumulation point of this sequence. Since the cost function $f^0(\cdot)$ is continuous, $f^0(\hat{x})$ is an accumulation point of the cost sequence. Hence, since, by construction, the cost sequence $\{f^0(x_i)\}_{i=0}^{\infty}$ is monotone decreasing, it follows that $f^0(x_i) \to f^0(\hat{x})$, as $i \to \infty$.

Now, for the sake of contradiction, suppose that $\theta(\hat{x}) < 0$. Since for any $x \in \Re^n$, $H(x)$ is compact, and $H(\cdot)$ is bounded on bounded sets and is outer-semicontinuous ((iv) of Theorem 2.8), it follows from Theorem 5.3.7 (b) in Polak [13] that there exists a subsequence $\{j_i\}_{i=0}^{\infty}$ of the integers such that $x_{j_i} \to \hat{x}$ and $h_{j_i} \to \hat{h} \in H(\hat{x})$, as $i \to \infty$. It follows from (ii) and (iii) of Theorem 2.8 that $\hat{h} \neq 0$ and

$$df^0(\hat{x}; \hat{h}) \leq \theta(\hat{x}) - \gamma\|\hat{h}\|^2. \tag{3.6}$$

Let $\varepsilon > 0$ be such that $0 < \alpha - \varepsilon < 1$. Then it follows from the definition of the directional derivative of $f^0(\cdot)$ that there exists a $k_\varepsilon \in \mathcal{N}$ such that

$$\begin{cases} f^0(\hat{x} + \beta^{k_\varepsilon}\hat{h}) - f^0(\hat{x}) & \leq \beta^{k_\varepsilon}(\alpha - \varepsilon) df^0(\hat{x}; \hat{h}) \\ & \leq \beta^{k_\varepsilon}(\alpha - \varepsilon)[\theta(\hat{x}) - \gamma\|\hat{h}\|^2]. \end{cases} \tag{3.7}$$

Hence,

$$f^0(\hat{x}+\beta^{k_\varepsilon}\hat{h})-f^0(\hat{x})-\beta^{k_\varepsilon}\alpha\theta(\hat{x})\le-\beta^{k_\varepsilon}[\varepsilon\theta(\hat{x})+(\alpha-\varepsilon)\gamma\|\hat{h}\|^2]. \tag{3.8}$$

Now,

$$\varepsilon\theta(\hat{x})+(\alpha-\varepsilon)\gamma\|\hat{h}\|^2>0 \tag{3.9}$$

for all $\varepsilon>0$ such that

$$\varepsilon<\varepsilon':=\frac{\alpha\gamma\|\hat{h}\|^2}{-\theta(\hat{x})+\gamma\|\hat{h}\|^2}. \tag{3.10}$$

Let $\hat{\varepsilon}:=\frac{1}{2}\varepsilon'$. Then, since $f^0(\cdot)$ and $\theta(\cdot)$ are continuous and $h_{j_i}\to\hat{h}$, as $i\to\infty$, there exists a $\rho>0$ such that for all $x_{j_i}\in B(\hat{x};\rho)$,

$$f^0(x_{j_i}+\beta^{k_{\hat{\varepsilon}}}h_{j_i})-f^0(x_{j_i})-\beta^{k_{\hat{\varepsilon}}}\alpha\theta(x_{j_i})<0, \tag{3.11}$$

which shows that for all $x_{j_i}\in B(\hat{x};\rho)$, $\lambda(x_{j_i})\ge\beta^{k_{\hat{\varepsilon}}}$. Next, since $\theta(\cdot)$ is continuous, there exists $\hat{\rho}\in(0,\rho)$ such that for all $x_{j_i}\in B(\hat{x};\hat{\rho})$, $\theta(x_{j_i})\le\frac{1}{2}\theta(\hat{x})$. It therefore follows from the step-size rule (3.2) that for all $x_{j_i}\in B(\hat{x};\hat{\rho})$,

$$f^0(x_{j_i+1})-f^0(x_{j_i})\le\beta^{k_{\hat{\varepsilon}}}\alpha\theta(x_{j_i})\le\tfrac{1}{2}\beta^{k_{\hat{\varepsilon}}}\alpha\theta(\hat{x})\ . \tag{3.12}$$

Since $\{f^0(x_i)\}_{i=0}^\infty$ is monotone decreasing, (3.12) implies that $f^0(x_i)\to-\infty$, as $i\to\infty$, contradicting the fact that $f^0(x_i)\to f^0(\hat{x})$, as $i\to\infty$. Hence we conclude that $\theta(\hat{x})=0$, and therefore that $\hat{x}=x^*$. Since by Lemma 3.3, $f^0(\cdot)$ is strongly convex, the whole sequence $\{x_i\}$ converges to x^*. ∎

4. Rate of convergence of Algorithm 3.1. We will now show that (1.11)–(1.13) hold for Algorithm 3.1.

PROPOSITION 4.1. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied and that* $\hat{x}$ *is the unique minimizer of* $f^0(\cdot)$. *Then for all* $x\in\Re^n$,

$$f^0(x)-f^0(\hat{x})\ge\tfrac{1}{2}cc_Fm\|x-\hat{x}\|^2. \tag{4.1}$$

Proof. By Lemma 3.3, $f^0(\cdot)$ is a strongly convex function. Hence, for any $x\in\Re^n$ we have

$$\begin{cases} F(\psi(x))-F(\psi(\hat{x})) \\ \quad\ge\displaystyle\sum_{j\in\mathbf{m}}\frac{\partial F}{\partial y^j}(\psi(\hat{x}))(\psi^j(x)-\psi^j(\hat{x})) \\ \quad\ge\displaystyle\sum_{j\in\mathbf{m}}\frac{\partial F}{\partial y^j}(\psi(\hat{x}))\max_{k\in\mathbf{q_j}(x)}\{f^{j,k}(\hat{x})-\psi^j(\hat{x}) \\ \qquad\qquad +\langle\nabla f^{j,k}(\hat{x}),x-\hat{x}\rangle+\tfrac{c}{2}\|x-\hat{x}\|^2\} \\ \quad\ge\displaystyle\sum_{j\in\mathbf{m}}\frac{\partial F}{\partial y^j}(\psi(\hat{x}))\max_{k\in\hat{\mathbf{q}}_\mathbf{j}(x)}\{\langle\nabla f^{j,k}(\hat{x}),x-\hat{x}\rangle+\tfrac{c}{2}\|x-\hat{x}\|^2\}, \end{cases} \tag{4.2}$$

where $\hat{\mathbf{q}}_{\mathbf{j}}(x)$ is defined by (2.7). It now follows from (2.5) and (4.2) that

$$F(\psi(x)) - F(\psi(\hat{x})) \geq df^0(\hat{x}, x - \hat{x}) + \frac{mc_F c}{2}\|x - \hat{x}\|^2, \tag{4.3}$$

Since $df^0(\hat{x}, x - \hat{x}) \geq 0$, (4.1) follows. □

PROPOSITION 4.2. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied. Then for any compact convex set S there exists a $\kappa > 0$ such that for any $x, z \in S$,*

$$|f^0(x) - \tilde{f}^0(z, x - z)| \leq \kappa\|x - z\|^3, \tag{4.4}$$

where $\tilde{f}^0(z, x - z)$ was defined in (2.17).

Proof. First, it follows from Polak [13, Lemma 2.5.4] or [15] that there exists a constant $L_1 < \infty$ such that for any $x, z \in \Re^n$,

$$|\psi^j(x) - \hat{\psi}^j(z, x - z)| \leq \frac{L_1}{6}\|x - z\|^3, j \in \mathbf{m}. \tag{4.5}$$

Let $S \subset \Re^n$ be a compact set, and let $L_2(\geq C) < \infty$ be a constant associated with S, such that for any $z \in S$,

$$\|\nabla F(\psi(z))\| \leq L_2. \tag{4.6}$$

Then for all $x, z \in S$, by the mean-value theorem, it holds that

$$\begin{cases} f^0(x) = F(\psi(x)) \\ \quad = F(\psi(z)) + \langle \nabla F(\psi(z)), \psi(x) - \psi(z)\rangle \\ \quad +\frac{1}{2}\langle \psi(x) - \psi(z), \nabla^2 F(\psi(z))(\psi(x) - \psi(z))\rangle \\ \quad + \int_0^1 t \int_0^1 \langle \psi(x) - \psi(z), [\nabla^2 F(\psi(z) + st(\psi(x) - \psi(z))) \\ \qquad\qquad -\nabla^2 F(\psi(z))](\psi(x) - \psi(z))\rangle ds dt \\ \quad \leq F(\psi(z)) + \langle \nabla F(\psi(z)), \psi(x) - \psi(z)\rangle \\ \quad +\frac{1}{2}\langle \psi(x) - \psi(z), \nabla^2 F(\psi(z))(\psi(x) - \psi(z))\rangle + \frac{L_2}{6}\|\psi(x) - \psi(z)\|^3 \\ \quad = \tilde{f}^0(z, x - z) + \langle \nabla F(\psi(z)), \psi(x) - \hat{\psi}(z, x - z)\rangle \\ \quad +\frac{1}{2}\langle \psi(x) - \psi(z), \nabla^2 F(\psi(z))(\psi(x) - \psi(z))\rangle \\ \quad -\frac{1}{2}\langle \hat{\psi}(z, x - z) - \psi(z), \nabla^2 F(\psi(z))(\hat{\psi}(z, x - z) - \psi(z))\rangle \\ \quad +\frac{L_2}{6}\|\psi(x) - \psi(z)\|^3 \ . \end{cases} \tag{4.7}$$

Thus, according to (4.5) and (4.7), we have

$$\begin{cases} f^0(x) \quad \leq \tilde{f}^0(z, x - z) + L_3\|x - z\|^3 + \frac{L_2}{6}\|\psi(x) - \psi(z)\|^3 \\ \qquad\qquad +\frac{1}{2}\langle \psi(x) - \psi(z), \nabla^2 F(\psi(z))(\psi(x) - \hat{\psi}(z, x - z))\rangle \\ \qquad\qquad +\frac{1}{2}\langle \psi(x) - \hat{\psi}(z, x - z), \nabla^2 F(\psi(z))(\hat{\psi}(z, x - z) - \psi(z))\rangle, \end{cases} \tag{4.8}$$

where $L_3 := mL_2L_1/6$. For $x, z \in S$ and $j \in \mathbf{m}$, by the definition of $\hat{\psi}^j(\cdot,\cdot)$ (see (2.16)), it holds that

$$\begin{aligned}
&\hat{\psi}^j(z, x-z) - \psi^j(z) \\
&\quad = \max_{k\in\mathbf{q_j}}\{f^{j,k}(z) + \langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\} - \psi^j(z) \\
&\quad \le \max_{k\in\mathbf{q_j}} f^{j,k}(z) + \max_{k\in\mathbf{q_j}}\{\langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\} - \psi^j(z) \\
&\quad = \max_{k\in\mathbf{q_j}}\{\langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\}
\end{aligned}$$

and, on the other hand,

$$\begin{aligned}
&\hat{\psi}^j(z, x-z) - \psi^j(z) \\
&\quad = \max_{k\in\mathbf{q_j}}\{f^{j,k}(z) + \langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\} - \psi^j(z) \\
&\quad \ge \max_{k\in\hat{\mathbf{q}}_\mathbf{j}(z)} f^{j,k}(z) + \langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\} - \psi^j(z) \\
&\quad = \max_{k\in\hat{\mathbf{q}}_\mathbf{j}(z)} \{\langle \nabla f^{j,k}(z), x-z\rangle + \tfrac{1}{2}\langle x-z, \nabla^2 f^{j,k}(z)(x-z)\rangle\} \ ,
\end{aligned}$$

where the definition of $\hat{\mathbf{q}}_\mathbf{j}(z)$ can be found in (2.7). Thus, since S is compact, there exists a positive number L_4 such that for all $x, z \in S$,

$$\|\hat{\psi}(z, x-z) - \psi(z)\| \le L_4\|x-z\|. \tag{4.9}$$

By the Lipschitzian property of ψ and (4.5) it follows that there exists a positive number $L_5(\ge L_4)$ such that for all $x, z \in S$,

$$\|\psi(x) - \psi(z)\| \le L_5\|x-z\| \tag{4.10}$$

and

$$\|\psi(x) - \hat{\psi}(z, x-z)\| \le L_5\|x-z\|^2. \tag{4.11}$$

Hence for all $x, z \in S$,

$$f^0(x) - \tilde{f}^0(z, x-z) \le \kappa\|x-z\|^3 \tag{4.12}$$

with

$$\kappa := L_3 + \frac{L_2L_5}{6} + L_2L_5^2. \tag{4.13}$$

The other half of the inequality of (4.4) follows similarly (with κ as defined in (4.13)). □

THEOREM 4.3. *Suppose that Assumptions* 1.1 *and* 2.4 *are satisfied, that all the* Y_j, $j \in \mathbf{m}$ *are of the form* (1.5), *so that problem* (1.1), (1.2), (1.4) *reduces to problem* (3.1). *If* $\{x_i\}_{i=0}^\infty$ *is a sequence constructed by Algorithm* 3.1, *in solving problem* (3.1), *then,* $\{x_i\}_{i=0}^\infty$ *converges superlinearly with Q-order at least* 3/2.

Proof. First we will prove that after a finite number of iterations, the step-size λ_i stabilizes to 1, so that eventually $x_{i+1} = x_i + h_i$ holds for the sequence $\{x_i\}_{i=0}^{\infty}$. We will then complete our proof by making use of results in [13, Corollary 2.5.8].

It follows from Theorem 3.4 that the sequence $\{x_i\}_{i=0}^{\infty}$ converges to the unique minimizer $\hat{x}$ of $f^0(\cdot)$. Hence we conclude from Theorem 2.8 that

$$h_i \to 0 \quad \text{as } i \to \infty. \tag{4.14}$$

In view of this, we conclude from Lemma 2.6 that there exist a positive number $\varepsilon > 0$ and a nonnegative integer i_0 such that for all $i \geq i_0$,

$$\hat{f}^0(x_i, h_i) = u(x_i, h_i, 0) + v(x_i, h_i, 0) = \tilde{f}^0(x_i, h_i) = \min_{h \in \Re^n, \|h\| \leq \varepsilon} \tilde{f}^0(x_i, h). \tag{4.15}$$

Suppose that i_0 is sufficiently large to ensure that for all $i \geq i_0$,

$$\|h_i\| \leq \varepsilon, \quad \|x_i - \hat{x}\| \leq \varepsilon\ . \tag{4.16}$$

Then, making use of (4.1), we find that, for $i = i_0, i_0 + 1, i_0 + 2, \ldots,$

$$\begin{cases} \frac{1}{2} c c_F m \|x_i + h_i - \hat{x}\|^2 \\ \quad \leq f^0(x_i + h_i) - f^0(\hat{x}) \\ \quad = f^0(x_i + h_i) - \tilde{f}^0(x_i, h_i) + \tilde{f}^0(x_i, h_i) - f^0(\hat{x}) \\ \quad \leq f^0(x_i + h_i) - \tilde{f}^0(x_i, h_i) + \tilde{f}^0(x_i, \hat{x} - x_i) - f^0(\hat{x}), \end{cases} \tag{4.17}$$

because $\tilde{f}^0(x_i, h_i) \leq \tilde{f}^0(x_i, \hat{x} - x_i)$, by (4.15). It now follows from Proposition 4.2 that there exists a $\kappa > 0$ such that for all $i \geq i_0$,

$$\begin{cases} \frac{1}{2} c c_F m \|x_i + h_i - \hat{x}\|^2 \\ \quad \leq \kappa(\|x_i + h_i - x_i\|^3 + \|x_i - \hat{x}\|^3) \\ \quad \leq \kappa[(\|x_i + h_i - \hat{x}\| + \|x_i - \hat{x}\|)^3 + \|x_i - \hat{x}\|^3]. \end{cases} \tag{4.18}$$

Now, by Theorem 2.9, there exist a positive integer $i_1 \geq i_0$ and an $\varepsilon_1 > 0$ such that for all $i \geq i_1$,

$$\theta(x_i) \leq -\varepsilon_1 \|h_i\|^2. \tag{4.19}$$

Next, Proposition 4.2 and (4.15) imply that for all $i \geq i_1$,

$$\begin{cases} \theta(x_i) &= \hat{f}^0(x_i, h_i) - f^0(x) \\ &= \tilde{f}^0(x_i, h_i) - f^0(x) \\ &= \tilde{f}^0(x_i, h_i) - f^0(x_i + h_i) + f^0(x_i + h_i) - f^0(x_i) \\ &\geq -\kappa \|h_i\|^3 + f^0(x_i + h_i) - f^0(x_i). \end{cases} \tag{4.20}$$

Hence, from (4.20) and (4.19), we have

$$\left\{\begin{array}{lll} f^0(x_i+h_i)-f^0(x_i)-\alpha\theta(x_i) & \leq & (1-\alpha)\theta(x_i)+\kappa\|h_i\|^3 \\ & \leq & -(1-\alpha)\varepsilon_1\|h_i\|^2+\kappa\|h_i\|^3. \end{array}\right. \tag{4.21}$$

It now follows from (4.21) and the fact that $h_i \to 0$ as $i \to \infty$ that for all i sufficiently large,

$$x_{i+1} = x_i + h_i. \tag{4.22}$$

We therefore conclude from [13, Corollary 2.5.8] or [15], (4.18), and (4.19) that $\{x_i\}_{i=0}^{\infty}$ converges to $\hat{x}$ superlinearly with Q-order at least 3/2. ∎

5. An algorithm for solving generalized semi-infinite min-max problems. We are now ready to tackle the generalized semi-infinite min-max problems defined in (1.1), (1.2), (1.4). Such problems can be solved only by discretization techniques. We will use discretizations that result in consistent approximations (as defined in section 3.3 of [13]) and use them in conjunction with a master algorithm that calls Algorithm 3.1 as a subroutine. We will see that under a reasonable assumption, the resulting algorithm retains the rate of convergence of Algorithm 3.1.

5.1. Consistent approximations. Let N_0 be a strictly positive integer, and, for $N \in \mathcal{N}_0 := \{N_0, N_0+1, N_0+2, \ldots\}$, let $Y_{j,N}$ be finite cardinality subsets of Y_j, $j \in \mathbf{m}$, such that $Y_{j,N} \subset Y_{j,N+1}$ for all N and the closure of the set $\lim Y_{j,N}$ is equal to Y_j, $j \in \mathbf{m}$. Then we define the family of approximating problems $\mathbf{P_N}$, $N \in \mathcal{N}_0$, as follows:

$$\mathbf{P_N} \qquad \min_{x\in\Re^n} f_N^0(x), \tag{5.1}$$

where

$$f_N^0(x) := F(\psi_N(x)), \tag{5.2}$$

$\psi_N(x) = (\psi_N^1(x), \ldots, \psi_N^m(x))$, and for $j \in \mathbf{m}$,

$$\psi_N^j(x) = \max_{y_j\in Y_{j,N}} \phi^j(x, y_j)\ . \tag{5.3}$$

It should be clear that the approximating problems $\mathbf{P_N}$ are of the form (3.1) and that one can define optimality functions $\theta_N(\cdot)$ for them of the form (2.19). We will refer to the original problem (1.1), (1.2), (1.4) as $\mathbf{P}$.

DEFINITION 5.1 (see [13]). *We will say that the pairs* $(\mathbf{P_N}, \theta_\mathbf{N})$ *in the sequence* $\{(\mathbf{P_N}, \theta_\mathbf{N})\}_{\mathbf{N}\in\mathcal{N}_\mathbf{0}}$ *are consistent approximations to the pair* $(\mathbf{P}, \theta)$ *if the problems* $\mathbf{P_N}$ *epi-converge to* $\mathbf{P}$ *(i.e., the epigraphs of the* $f_N^0(\cdot)$ *converge to the epigraph of* $f^0(\cdot)$ *in the sense defined in Definition* 5.3.6 *in* [13]*) and for any infinite sequence* $\{x_N\}_{N\in K}$, $K \subset \mathcal{N}_0$, *such that* $x_N \to^K x$, $\overline{\lim}_{N\in K}\theta_N(x_N) \leq \theta(x)$.

Assumption 5.2. We will assume as follows:

(a) For every $N \in \mathcal{N}_0$, the problem (5.1) has a solution.

(b) There exists a strictly positive valued, strictly monotone decreasing function $\Delta : \mathcal{N} \to \Re$, such that $\Delta(N) \to 0$, as $N \to \infty$, and a $L < \infty$, such that for every $N \geq N_0$, $j \in \mathbf{m}$, and $y \in Y_j$, there exists a $y' \in Y_{j,N}$ such that

$$\|y - y'\| \leq L\Delta(N). \tag{5.4}$$

For example, if for all $j \in \mathbf{m}$, Y_j is the unit cube in $\Re^{m_j}$, i.e., $Y_j = I^{m_j}$, with $I := [0,1]$, then we can define $Y_{j,N} = I_N^{m_j}$, where

$$I_N = \{0, 1/a(N), 2/a(N), \ldots, (a(N)-1)/a(N), 1\},$$

with $a(N) := 2^{N-N_0}$. In this case it is easy to see that $\Delta(N) = 1/a(N)$ and $L = \frac{1}{2}\max_{j\in\mathbf{m}}\{{m_j}^{(1/m_j)}\}$. Similar constructions can be obtained for other polyhedral sets.

For any $x, h \in \Re^n$ and $w \in \Re^m$, we define

$$u_N(x,h,w) := \langle \nabla F(\psi_N(x)), \hat{\psi}_N(x,h) - \psi_N(x) + w\rangle \tag{5.5}$$

and

$$\begin{cases} v_N(x,h,w) \\ \quad = \frac{1}{2}\langle \hat{\psi}_N(x,h) - \psi_N(x) + w, \nabla^2 F(\psi_N(x))(\hat{\psi}_N(x,h) - \psi_N(x) + w)\rangle, \end{cases} \tag{5.6}$$

where

$$\hat{\psi}_N(x,h) = (\hat{\psi}_N^1(x,h), \ldots, \hat{\psi}_N^m(x,h)) \tag{5.7}$$

and

$$\hat{\psi}_N^j(x,h) = \max_{y_j \in Y_{j,N}} \{\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \tfrac{1}{2}\langle h, \nabla_x^2 \phi^j(x,y_j)h\rangle\}\ . \tag{5.8}$$

We infer from (2.19) that the optimality functions $\theta_N(\cdot)$, for the problems $\mathbf{P_N}$ have the following form:

$$\theta_N(x) := \min_{h\in\Re^n}\{\min_{w\in\Re_+^m}(u_N(x,h,w) + v_N(x,h,w))\}. \tag{5.9}$$

Since the cardinality of the sets $Y_{j,N}$ is finite, it is obvious that the $\theta_N(x)$ can be evaluated.

As was also done in the Polak–Mayne–Higgins rate-preserving method [16] (see also [17]), we use an alternative optimality function for the problems $\mathbf{P_N}$ for precision adjustment in our algorithm. This optimality function is defined by

$$\bar{\theta}_N(x) := \min_{h\in\Re^n} \bar{f}_N^0(x,h) - \sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi_N(x))\psi_N^j(x), \tag{5.10}$$

where

$$\begin{cases} \bar{f}_N^0(x,h) \\ \quad = \displaystyle\sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi_N(x)) \max_{y_j\in Y_{j,N}} [\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \tfrac{1}{2}\delta\|h\|^2], \end{cases} \tag{5.11}$$

with $\delta > 0$, a constant.

Similarly (as in [17]), we define an alternative optimality function for the problem $\mathbf{P}$ by

$$\bar{\theta}(x) := \min_{h\in\Re^n} \bar{f}^0(x,h) - \sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(x))\psi^j(x), \tag{5.12}$$

where

$$\bar{f}^0(x,h) = \sum_{j\in\mathbf{m}} \frac{\partial F}{\partial y^j}(\psi(x)) \max_{y_j\in Y_j}[\phi^j(x,y_j) + \langle \nabla_x \phi^j(x,y_j), h\rangle + \tfrac{1}{2}\delta\|h\|^2], \tag{5.13}$$

with $\delta > 0$ the same constant as in (5.11).

PROPOSITION 5.3 (see [17]). *Suppose that Assumptions* 1.1 *and* 5.2 *are satisfied and that for all* $N \in \mathcal{N}_0$, $f_N^0(\cdot)$ *is defined by* (5.2) *and* $\bar{\theta}_N(\cdot)$ *by* (5.10). *Let* $S \subset \Re^n$ *be a bounded subset and let* $L < \infty$ *be a Lipschitz constant valid for the functions* $\phi^j(\cdot,\cdot)$ *and* $\nabla_x\phi^j(\cdot,\cdot)$ *on* $S \times Y_j$, $j \in \mathbf{q}$. *Then there exists a constant* $C_S < \infty$ *such that for all* $x \in S, N \in \mathcal{N}_0$,

$$|f_N^0(x) - f^0(x)| \le C_S \Delta(N), \tag{5.14}$$

and

$$|\bar{\theta}_N(x) - \bar{\theta}(x)| \le C_S \Delta(N). \tag{5.15}$$

5.2. The superlinear rate-preserving algorithm.

ALGORITHM 5.4 (solves problem (1.1), (1.2), (1.4)).
Parameters. $\alpha, \beta \in (0,1)$, $\delta > 0$, $D > 0$, $\sigma \ge 3$.
Data. $x_0 \in \Re^n$, $N_0 \in \mathcal{N}$.
Step 0. *Set* $i = 0$, $N = N_0$.
Step 1. *Compute the optimality function value* $\bar{\theta}_N(x_i)$ *according to* (5.10) *and* (5.11).
Step 2. *If*

$$D\Delta(N) \le |\bar{\theta}_N(x_i)|^\sigma, \tag{5.16}$$

go to Step 3. *Else, replace* N *by* $N+1$, *and go to Step* 1.
Step 3. *Compute the second* optimality function value $\theta_N(x_i)$ *according to* (5.9), *i.e.,*

$$\theta_N(x_i) = \min_{h\in\Re^n}\{\min_{w\in\Re^m_+}(u_N(x_i,h,w) + v_N(x_i,h,w))\} \tag{5.17}$$

and the corresponding search direction h_i *according to*

$$h_i \in \arg\min_{h\in\Re^n}\{\min_{w\in\Re^m_+}(u_N(x_i,h,w) + v_N(x_i,h,w))\}. \tag{5.18}$$

Step 4. *Compute the step-size*

$$\lambda_i = \max_{k\in\mathcal{N}}\{\beta^k |\ f_N^0(x_i + \beta^k h_i) - f_N^0(x_i) - \beta^k\alpha\theta_N(x_i) \le 0\}, \tag{5.19}$$

and go to Step 5.
Step 5. *Set*

$$x_{i+1} = x_i + \lambda_i h_i. \tag{5.20}$$

Set $N_i = N$, *replace* i *by* $i+1$, *and go to Step* 1.

Remark.

(a) It follows from Proposition 5.3 that $\bar{\theta}_N(x_i) \to \bar{\theta}(x_i)$, as $N \to \infty$. Hence, whenever $\bar{\theta}(x_i) \ne 0$, the loop consisting of Step 1 and Step 2 of Algorithm 5.4 yields

a finite discretization parameter N_i. For simplicity, we will assume that Algorithm 5.4 does not produce an iterate x_i such that $\bar{\theta}(x_i) = 0$.

(b) Note that the work needed to compute x_i by Algorithm 5.4 increases with the iteration number i.

The purpose of the following results is to show that the relations (1.15)–(1.17) hold for Algorithm (5.4)

LEMMA 5.5. *Suppose that Assumptions* 1.1*,* 2.4*, and* 5.2 *are satisfied and that Algorithm* 5.4 *has constructed a sequence* $\{x_i\}_{i=0}^{\infty}$ *together with the corresponding sequence of discretization parameters* $\{N_i\}_{i=0}^{\infty}$*. If the sequence* $\{x_i\}_{i=0}^{\infty}$ *has at least one accumulation point, then* $N_i \to \infty$ *as* $i \to \infty$*.*

Proof. For the sake of contradiction, suppose that the sequence $\{x_i\}_{i=0}^{\infty}$ has an accumulation point $\hat{x}$ and that the sequence $\{N_i\}_{i=0}^{\infty}$ is bounded. Then, because $\{N_i\}_{i=0}^{\infty}$ is a monotonically increasing sequence of integers, there exists an $i_0 \in \mathcal{N}$, such that for all $i \geq i_0$, $N_i = N_{i_0} =: N^*$. Hence for $i \geq i_0$, the construction of the sequence $\{x_i\}_{i=0}^{\infty}$ is carried out by Algorithm 3.1 applied to problem (5.1) with $N = N^*$. Furthermore, it follows from (5.16) that there exists an $\varepsilon > 0$, such that $\bar{\theta}_i = \bar{\theta}_{N^*}(x_i) \leq -\varepsilon$ for all $i \geq i_0$. However, it follows from Theorem 3.4 that $\theta_{N^*}(\hat{x}) = 0$. Thus, by (iii) of Theorem 2.8, $0 \in \partial f^0_{N^*}(x_i)$. By [17, Theorem 2], $0 \in \partial f^0_{N^*}(x_i)$ implies $\bar{\theta}_{N^*}(x_i) = 0$. Then, from the continuity of $\bar{\theta}_{N^*}(\cdot)$ [17, Theorem 2], it holds that $\bar{\theta}_{N^*}(x_i) \to \bar{\theta}_{N^*}(\hat{x}) = 0$ as $i \to \infty, i \in K$, where the infinite subsequence $\{x_i\}_{i\in K}$, $K \subset \mathcal{N}$, converges to $\hat{x}$, which contradicts the previous finding, and hence completes our proof. ∎

THEOREM 5.6. *Suppose that Assumptions* 1.1*,* 2.4*, and* 5.2 *are satisfied and that Algorithm* 5.4 *has constructed a bounded sequence* $\{x_i\}_{i=0}^{\infty}$*. Then every accumulation point* $\hat{x}$ *of* $\{x_i\}_{i=0}^{\infty}$ *satisfies* $\bar{\theta}(\hat{x}) = 0$*.*

Proof. By applying Theorem 3.3.23 of [13] or theorems in section 5 of [12] and Lemma 5.5 to Algorithm 5.4, we obtain the desired result. ∎

THEOREM 5.7. *Suppose that Assumptions* 1.1*,* 2.4*, and* 5.2 *are satisfied and that Algorithm* 5.4 *has constructed a bounded sequence* $\{x_i\}_{i=0}^{\infty}$*. Then* $\{x_i\}$ *converges to the unique minimizer* $\hat{x}$ *of* $f^0(\cdot)$ *with Q-order* 3/2*.*

Proof. First, by Theorem 5.6 and the fact that $f^0(\cdot)$ has a unique minimizer $\hat{x}$, the whole sequence $\{x_i\}$ converges to $\hat{x}$. Hence one can deduce from Theorem 4.3 and the proof of [13, Theorem 3.4.20], that $\{x_i\}$ converges to $\hat{x}$ with Q-order 3/2. Since the derivation is straightforward, we omit the details here. ∎

6. Some numerical results. We now present some numerical results that illustrate the behavior of the algorithm proposed in section 5 for generalized semi-infinite programming problems. The algorithm was implemented in Matlab. Throughout the computational experiments, the parameters used in the algorithm were $\alpha = 0.05, \beta = 0.5, \delta = 1.0, D = 10^{-10}$, and $\sigma = 3.1$. For both examples, we used the starting point $(1, 1)$. The iteration of the algorithm is stopped at x_i if for some N the meshsize $\Delta(N) < 0.005$ and $|\theta_N(x_i)| \leq 10^{-8}$. A Matlab code developed in [21], which was based on a smoothing Newton method [19] for variational inequalities, was used to solve our search direction finding subproblem (2.57).

Example 1. In this case, $f^0(x) = F(\psi^1(x), \psi^2(x))$, with $x = (x^1, x^2) \in \Re^2$, $F(z) = z^1 + z^2$, with $z = (z^1, z^2) \in \Re^2$, and

$$\psi^1(x) = \max_{t\in Y_1}\{t^2 - (tx^1 + e^t x^2) + (x^1 + x^2)^2 + (x^1)^2 + (x^2)^2 + e^{(x^1+x^2)}\}$$

TABLE 6.1
Numerical results for Example 1.

Iteration i	0	1	2	3	4
$\|\|x_i - \hat{x}\|\|$	1.6×10^0	5.7×10^{-1}	5.1×10^{-2}	2.9×10^{-4}	0.0
Discretization level	1	1	1	1	9

TABLE 6.2
Numerical results for Example 2.

Iteration i	0	1	2	3	4
$\|\|x_i - \hat{x}\|\|$	1.7×10^0	6.4×10^{-1}	5.0×10^{-2}	1.9×10^{-4}	0.0
Discretization level	1	1	1	1	9

and

$$\psi^2(x) = \max_{t \in Y_2} \{(t-1)^2 + 0.5(x^1 + x^2)^2 - 2t(x^1 + x^2) + 0.5[(x^1)^2 + (x^2)^2]\},$$

where $Y_1 = [0, 1]$ and $Y_2 = [-1, 0]$.

Example 2. In this case, the functions $f^0(\cdot)$, $\psi^1(\cdot)$, and $\psi^2(\cdot)$ are also defined as in Example 1, but $F(\cdot)$ is defined by

$$F(z) = 0.5(z^1 + \sqrt{(z^1)^2 + 4}) + \ln(1 + e^{z^2}) + 0.5((z^1)^2 + (z^2)^2), \quad z = (z^1, z^2) \in \Re^2.$$

The numerical results are summarized in Table 6.1 and Table 6.2. In these two tables the first row represents the iteration number, the second row is the residue $||x_i - \hat{x}||$ (we used the last iterate as a substitute for $\hat{x}$) and the third row shows the discretization level (the meshsize at the present level is decreased to half of the previous one) refined by the master algorithm at the i-th step. It is clear from the numerical results that the rate of convergence is superlinear.

7. Conclusion. We have presented two superlinearly converging algorithms, one for solving finite generalized min-max problems of the form (1.1), (1.2), (1.3) and one for solving generalized semi-infinite min-max problems of the form (1.1), (1.2), (1.4). These algorithms were obtained by making use of the concepts underlying the construction of the Polak–Mayne–Higgins Newton's method [15] and the Polak–Mayne–Higgins rate-preserving method [16], respectively. The construction of the algorithms depends on the cost function having a subgradient and their rate of convergence depends on convexity and second order smoothness, and hence Assumption 2.4 is essential.

Our numerical results are consistent with our theoretical prediction that the algorithms converge Q-superlinearly.

Acknowledgments. The authors wish to thank Prof. R. T. Rockafellar for suggesting the function $\hat{f}^0(x, h)$ as a way to get around the possible nonconvexity of the function $\tilde{f}^0(x, h)$ in h, as well as for the formula (2.57) which shows that our optimality function is defined by a quadratically constrained quadratic programming problem. They also thank two anonymous referees for their helpful comments and suggestions.

REFERENCES

[1] D. P. Bertsekas, *Nondifferentiable optimization via approximation*, Math. Programming Study, 3 (1975), pp. 1–25.

[2] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley, New York, 1983.

[3] V. F. Dem'yanov and A. M. Rubinov, *Quasidifferential Calculus*, Optimization Software Inc., New York, 1986.

[4] A. der Kiureghian, *private communication*, Department of Civil Engineering, University of California at Berkeley, 1998.

[5] G. Di Pillo, L. Grippo, and S. Lucidi, *A smooth transformation of the generalized minimax problem*, J. Optim. Theory Appl., 95 (1997), pp. 1–24.

[6] K. C. Kiwiel, *A quadratic approximation method for minimizing a class of quasidifferentiable functions*, Numer. Math., 45 (1984), pp. 411–430.

[7] K. C. Kiwiel, *A linearization method for minimizing certain quasidifferentiable functions*, Math. Programming Study, 29 (1986), pp. 85–94.

[8] K. C. Kiwiel, *Descent methods for quasidifferentiable minimization*, Appl. Math. Optim., 18 (1988), pp. 163–180.

[9] G. Papavassilopoulos, *Algorithms for a class of nondifferentiable problems*, J. Optim. Theory Appl., 34 (1981), pp. 41–82.

[10] O. Pironneau and E. Polak, *On the rate of convergence of certain methods of centers*, Math. Programming, 2 (1972), pp. 230–258.

[11] E. Polak, *Basics of minimax algorithms*, in Nonsmooth Optimization and Related Topics, F. H. Clarke, V. F. Dem'yanov, and F. Giannessi, eds., Plenum Press, New York, 1989, pp. 343–367.

[12] E. Polak, *On the use of consistent approximations in the solution of semi-infinite optimization and optimal control problems*, Math. Programming, 62 (1993), pp. 385–414.

[13] E. Polak, *Optimization: Algorithms and Consistent Approximations*, Springer, New York, 1997.

[14] E. Polak and L. He, *Rate-preserving discretization strategies for semi-infinite programming and optimal control*, SIAM J. Control Optim., 30 (1992), pp. 548–572.

[15] E. Polak, D. Q. Mayne, and J. Higgins, *A superlinearly convergent algorithm for min-max problems*, J. Optim. Theory Appl., 89 (1991), pp. 407–439.

[16] E. Polak, D. Q. Mayne and J. Higgins, *On the extension of Newton's method to semi-infinite for minimax problems*, SIAM J. Control Optim., 30 (1992), pp. 367–389.

[17] E. Polak, L. Qi, and D. Sun, *First-Order algorithms for generalized finite and semi-infinite min-max problems*, Comput. Optim. Appl., 13 (1999), pp. 137–161.

[18] B. N. Pshenichnyi and Yu. M. Danilin, *Numerical Methods in Extremal Problems*, Nauka, Moscow, 1975 (in Russian).

[19] 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, 87 (2000), pp. 1–35.

[20] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, PA, 1996.

[21] 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, 1999, pp. 421–441.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 962–973

MONOTONICITY OF FIXED POINT AND NORMAL MAPPINGS ASSOCIATED WITH VARIATIONAL INEQUALITY AND ITS APPLICATION*

YUN-BIN ZHAO†‡ AND DUAN LI‡

Abstract. We prove sufficient conditions for the monotonicity and the strong monotonicity of fixed point and normal maps associated with variational inequality problems over a general closed convex set. Sufficient conditions for the strong monotonicity of their perturbed versions are also shown. These results include some well known in the literature as particular instances. Inspired by these results, we propose a modified Solodov and Svaiter iterative algorithm for the variational inequality problem whose fixed point map or normal map is monotone.

Key words. variational inequalities, cocoercive maps, (strongly) monotone maps, fixed point and normal maps, iterative algorithm

AMS subject classifications. 90C30, 90C33, 90C25

PII. S1052623499357957

1. Introduction. Given a continuous function $f : R^n \to R^n$ and a closed convex set K in R^n, the well-known finite-dimensional variational inequality, denoted by $\mathrm{VI}(K, f)$, is to find an element $x^* \in K$ such that

$$(x - x^*)^T f(x^*) \geq 0 \quad \text{for all } x \in K.$$

It is well known that the above problem can be reformulated as nonsmooth equations such as the fixed point and normal equations (see, e.g., [9, 18]). The fixed point equation is defined by

$$\pi_\alpha(x) = x - \Pi_K(x - \alpha f(x)) = 0, \tag{1}$$

and the normal equation is defined by

$$\Phi_\alpha(x) = f(\Pi_K(x)) + \alpha(x - \Pi_K(x)) = 0, \tag{2}$$

where $\alpha > 0$ is a positive scalar and $\Pi_K(\cdot)$ denotes the projection operator on the convex set K, i.e.,

$$\Pi_K(x) = \arg\min\{\|z - x\| : z \in K\}.$$

Throughout the paper, $\|\cdot\|$ denotes the 2-norm (Euclidean norm) of the vector in R^n. It turns out that x^* solves $\mathrm{VI}(K, f)$ if and only if $\pi_\alpha(x^*) = 0$ and that if x^* solves $\mathrm{VI}(K, f)$, then $x^* - \frac{1}{\alpha} f(x^*)$ is a solution to $\Phi_\alpha(x) = 0$; conversely, if $\Phi_\alpha(u^*) = 0$, then $\Pi_K(u^*)$ is a solution to $\mathrm{VI}(K, f)$.

Recently, several authors studied the P_0 property of fixed point and normal maps when K is a rectangular box in R^n, i.e., the Cartesian product of n one-dimensional

*Received by the editors June 17, 1999; accepted for publication (in revised form) November 8, 2000; published electronically April 26, 2001. This work was partially supported by Research Grants Council of Hong Kong under grant CUHK4392/99E.
http://www.siam.org/journals/siopt/11-4/35795.html
†Institute of Applied Mathematics, Chinese Academy of Sciences, Beijing, China.
‡Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Shatin, N. T., Hong Kong (ybzhao@se.cuhk.edu.hk, dli@se.cuhk.edu.hk).

intervals. For such a K, Ravindran and Gowda [17] (respectively, Gowda and Tawhid [8]) showed that $\pi_\alpha(x)$ (respectively, $\Phi_\alpha(x)$) is a P_0-function if f is. Notice that the monotone maps are very important special cases of the class of P_0-functions. It is worth considering the problem:

(P) *When are the mappings $\pi_\alpha(x)$ and $\Phi_\alpha(x)$ monotone if K is a general closed convex set?*

Intuitively, we may conjecture that the fixed point map and the normal map are monotone if f is. However, this conjecture is not true. The following example shows that for a given $\alpha > 0$ the monotonicity of f, in general, does not imply the monotonicity of the fixed point map $\pi_\alpha(x)$ and the normal map $\Phi_\alpha(x)$.

Example 1.1. Let K be a closed convex set given by

$$K = \{x \in R^2 : \ x_1 \geq 0, \ x_2 = 0\}$$

and

$$f(x) = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} -x_2 \\ x_1 \end{pmatrix}.$$

For any $x, y \in R^2$, we have that $(x-y)^T(f(x)-f(y)) = 0$. Hence the function f is monotone on R^2. We now show that for an arbitrary scalar $\alpha > 0$ the fixed point mapping $\pi_\alpha(x) = x - \Pi_K(x - \alpha f(x))$ is not monotone in R^2. Indeed, let $u = (0,0)^T$ and $y = (1, \alpha/2)^T$. It is easy to verify that $\pi_\alpha(u) = (0,0)^T$ and $\pi_\alpha(y) = (-\alpha^2/2, \alpha/2)^T$. Thus, we have

$$(u-y)^T(\pi_\alpha(u) - \pi_\alpha(y)) = -\alpha^2/2 < 0,$$

which implies that $\pi_\alpha(\cdot)$ is not monotone on R^n.

Example 1.2. Let K be a closed convex set given by

$$K = \{x \in R^2 : \ x_1 \leq 0, \ x_2 = 0\}$$

and $f(x) : R^2 \to R^2$ be given as in Example 1.1. We now show that for an arbitrary $\alpha > 0$ the normal mapping $\Phi_\alpha(x) = f(\Pi_K(x)) + \alpha(x - \Pi_K(x))$ is not monotone in R^2. Indeed, let $u = (0,0)^T$ and $y = (-2\alpha^2, \alpha)^T$. We have that $\Phi_\alpha(u) = (0,0)^T$ and $\Phi_\alpha(y) = (0, -\alpha^2)^T$. Thus, we have

$$(u-y)^T(\Phi_\alpha(u) - \Phi_\alpha(y)) = -\alpha^3 < 0,$$

which implies that $\Phi_\alpha(\cdot)$ is not monotone on R^n.

From the above examples, we conclude that a certain condition stronger than the monotonicity of f is required to guarantee the monotonicity of $\pi_\alpha(x)$ and $\Phi_\alpha(x)$. One such condition is the so-called cocoercivity condition. We recall that f is said to be cocoercive with modulus $\beta > 0$ on a set $S \subset R^n$ if there exists a constant $\beta > 0$ such that

$$(x-y)^T(f(x)-f(y)) \geq \beta \|f(x) - f(y)\|^2 \ \text{ for all } x, y \in S.$$

The cocoercivity condition was used in several works, such as Bruck [1], Gabay [7] (in which this condition is used implicitly), Tseng [25], Marcotte and Wu [15], Magnanti and Perakis [13, 14], and Zhu and Marcotte [29, 30]. It is also used to study the strict feasibility of complementarity problems [27]. It is interesting to note that in an

affine case the cocoercivity has a close relation to the property of positive semidefinite (psd)-plus matrices [12, 30]. A special case of the cocoercive map is the strongly monotone and Lipschitzian map. We recall that a mapping f is said to be strongly monotone with modulus $c > 0$ on the set S if there is a scalar $c > 0$ such that

$$(x - y)^T (f(x) - f(y)) \geq c\|x - y\|^2 \quad \text{for all } x, y \in S.$$

It is evident that any cocoercive map on the set S must be monotone and Lipschitz continuous (with constant $L = 1/\beta$), but not necessarily strongly monotone (for instance, the constant mapping) on the same set.

In fact, the aforementioned problem (P) is not completely unknown. By using the cocoercivity condition implicitly and using properties of nonexpansive maps, Gabay [7] actually showed (but did not explicitly state) that $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ are monotone if the scalar α is chosen such that the map $I - \alpha f$ is nonexpansive. Furthermore, for strongly monotone and Lipschitzian map f, Gabay [7] and Sibony [20] actually showed that $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ are strongly monotone if the scalar α is chosen such that the map $I - \alpha f$ is contractive. Throughout this paper, we use the standard concept "nonexpansive" map and "contractive" map in the literature to mean a Lipschitzian map with constant $L = 1$ and $L < 1$, respectively.

However, it is easy to give an example to show that $\pi_\alpha(x)$ and $\Phi_\alpha(x)$ are still monotone (strongly monotone) even when α is chosen such that $I - \alpha f$ is not nonexpansive (contractive). For instance, let $K = R^n_+$ and $f(x) = x$. We see that the function f is cocoercive with modulus $\beta = 1$. While $I - \alpha f$ is not nonexpansive for $\alpha > 2$, the map $\pi_\alpha(x)$ remains monotone. As a result, the main purpose of this paper is to expand the results of Sibony [20] and Gabay [7]. We show that if f is cocoercive (strongly monotone and Lipschitz continuous, respectively), the monotonicity (strong monotonicity, respectively) of the maps $\pi_\alpha(x)$ and $\Phi_\alpha(x)$ can be ensured when α lies in a larger interval in which the map $I - \alpha f$ may not be nonexpansive (contractive, respectively). The results derived in this paper are not obtainable by the proof based on the nonexpansiveness and contractiveness of maps.

The other purpose of the paper is to introduce an application of the monotonicity of $\pi_\alpha(x)$ and $\Phi_\alpha(x)$. This application (see section 3) is motivated by the globally convergent inexact Newton method for the system of monotone equations proposed by Solodov and Svaiter [21]. See also [22, 23, 24]. We propose a modified Solodov and Svaiter method to solve the monotone equations $\pi_\alpha(x) = 0$ or $\Phi_\alpha(x) = 0$. This modified algorithm requires no projection operations in the line-search step.

2. Monotonicity of $\pi_\alpha(x)$ and $\Phi_\alpha(x)$. It is known (see Sibony [20] and Gabay [7]) that if f is strongly monotone with modulus $c > 0$ and Lipschitz continuous with constant $L > 0$, then $I - \alpha f$ is contractive when $0 < \alpha < 2c/L^2$. Since Π_K is nonexpansive, this in turn implies that $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ are both strongly monotone for $0 < \alpha < 2c/L^2$. Similarly, it follows from Gabay [7] (see Theorem 6.1 therein) that if f is cocoercive with modulus $\beta > 0$, then $I - \alpha f$ is nonexpansive for $0 < \alpha \leq 2\beta$, and thus we can easily verify that $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ are monotone for $0 < \alpha \leq 2\beta$.

In this section, we prove an improved version of the above-mentioned results. We prove that (i) when α lies outside of the interval $(0, 2c/L^2)$, for instance, $2c/L^2 \leq \alpha \leq 4c/L^2$, $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ are still strongly monotone although $I - \alpha f$, in this case, is not contractive, and (ii) when α lies outside of the interval $(0, 2\beta]$, for instance, $2\beta < \alpha \leq 4\beta$, $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ remain monotone although $I - \alpha f$ is not nonexpansive. This new result on monotonicity (strong monotonicity) of $\pi_\alpha(x)$ and $\Phi_{1/\alpha}(x)$ for $\alpha > 2\beta$ ($\alpha \geq 2c/L^2$) is not obtainable by using the nonexpansive (contractive) property

of $I - \alpha f$. The reason is as follows: Let f be cocoercive with modulus $\beta > 0$ on the set $S \subseteq R^n$, where

$$\beta = \sup\{\gamma > 0 : (x-y)^T(f(x)-f(y)) \geq \gamma\|f(x)-f(y)\|^2 \quad \text{for all } x, y \in S\}.$$

Clearly, such a scalar β is unique and $0 < \beta < \infty$ provided that f is not a constant mapping. We now verify that $I - \alpha f$ is nonexpansive on S if and only if $0 < \alpha \leq 2\beta$. It is sufficient to show that if $\alpha > 0$ is chosen such that $I - \alpha f$ is nonexpansive on S, then we must have $\alpha \leq 2\beta$. In fact, if $I - \alpha f$ is nonexpansive, then for any x, y in S we have

$$\begin{aligned}\|x-y\|^2 &\geq \|(I-\alpha f)(x) - (I-\alpha f)(y)\|^2 \\ &= \|x-y\|^2 - 2\alpha(x-y)^T(f(x)-f(y)) + \alpha^2\|f(x)-f(y)\|^2,\end{aligned}$$

which implies that

$$(x-y)^T(f(x)-f(y)) \geq (\alpha/2)\|f(x)-f(y)\|^2.$$

By the definition of β, we deduce that $\alpha/2 \leq \beta$, the desired consequence. Similarly, let f be strongly monotone with modulus $c > 0$ and Lipschitz continuous with constant $L > 0$ on the set S, where

$$c = \sup\{\gamma > 0 : (x-y)^T(f(x)-f(y)) \geq \gamma\|x-y\|^2 \quad \text{for all } x, y \in S\}$$

and

$$L = \inf\{\gamma > 0 : \|f(x)-f(y)\| \leq \gamma\|x-y\| \quad \text{for all } x, y \in S\}.$$

We can easily see that $0 < c < \infty$ and $L > 0$ provided that S is not a single point set. It is also easy to show that $I - \alpha f$ is contractive if and only if $0 < \alpha < 2c/L^2$.

Since the map $I - \alpha f$ is not contractive (nonexpansive, respectively) for $\alpha \geq 2c/L^2$ ($\alpha > 2\beta$, respectively), our result established in this section cannot follow directly from the proof of Sibony [20] and Gabay [7].

We also study the strong monotonicity of the perturbed fixed point and normal maps defined by

$$\pi_{\alpha,\varepsilon}(x) := x - \Pi_K(x - \alpha(f(x) + \varepsilon x)),$$

and

$$\Phi_{\alpha,\varepsilon}(x) := f(\Pi_K(x)) + \varepsilon\Pi_K(x) + \alpha(x - \Pi_K(x)),$$

respectively. This is motivated by the well-known Tikhonov regularization method for complementarity problems and variational inequalities. See, for example, Isac [10, 11], Venkateswaran [26], Facchinei [3], Facchinei and Kanzow [4], Facchinei and Pang [5], Gowda and Tawhid [8], Qi [16], Ravindran and Gowda [17], Zhao and Li [28], etc. It is worth mentioning that Gowda and Tawhid [8] showed that when $\alpha = 1$ the perturbed mapping $\Phi_{1,\varepsilon}(x)$ is a P-function if f is a P_0-function and K is a rectangular set. We show in this paper a sufficient condition for the strong monotonicity of $\pi_{\alpha,\varepsilon}(x)$ and $\Phi_{\alpha,\varepsilon}(x)$. The following lemma is helpful.

LEMMA 2.1. (i) *Denote*

$$u_z = z - \Pi_K(z) \quad \textit{for all } z \in R^n. \tag{3}$$

Then

$$(z-w)^T(u_z-u_w) \geq \|u_z-u_w\|^2.$$

(ii) *For any $\alpha > 0$ and vector $b \in R^n$, the following inequality holds for all $v \in R^n$:*

$$\alpha\|v\|^2 + v^T b \geq -\frac{\|b\|^2}{4\alpha}.$$

Proof. By the property of projection operator, we have

$$(\Pi_K(z) - \Pi_K(w))^T(\Pi_K(w)) - w) \geq 0 \text{ for all } z, w \in R^n,$$

$$(\Pi_K(w) - \Pi_K(z))^T(\Pi_K(z)) - z) \geq 0 \text{ for all } z, w \in R^n.$$

Adding the above two inequalities leads to

$$(\Pi_K(z) - \Pi_K(w))^T(z - \Pi_K(z) - (w - \Pi_K(w))) \geq 0 \text{ for all } z, w \in R^n,$$

i.e.,

$$[z - u_z - (w - u_w)]^T(u_z - u_w) \geq 0 \text{ for all } z, w \in R^n.$$

This proves the result (i).

Given $\alpha > 0$ and $b \in R^n$, it is easy to check that the minimum value of $\alpha\|v\|^2 + v^T b$ is $-\|b\|^2/(4\alpha)$. This proves the result (ii). □

We are ready to prove the main result in this section.

THEOREM 2.1. *Let K be an arbitrary closed convex set in R^n and $K \subseteq S \subseteq R^n$. Let $f : R^n \to R^n$ be a function.*

(i) *If f is cocoercive with modulus $\beta > 0$ on the set S, then for any fixed scalar α satisfying $0 < \alpha \leq 4\beta$, the fixed point map $\pi_\alpha(x)$ defined by (1) is monotone on the set S.*

(ii) *If f is strongly monotone with modulus $c > 0$ on the set S, and f is Lipschitz continuous with constant $L > 0$ on S, then for any fixed scalar α satisfying $0 < \alpha < 4c/L^2$, the fixed point map $\pi_\alpha(x)$ is strongly monotone on the set S.*

(iii) *If f is cocoercive with modulus $\beta > 0$ on the set S, then for any $0 < \alpha < 4\beta$ and $0 < \varepsilon \leq 2(\frac{1}{\alpha} - \frac{1}{4\beta})$ the perturbed map $\pi_{\alpha,\varepsilon}(x)$ is strongly monotone in x on the set S.*

Proof. Let $\alpha > 0$ and $0 \leq \varepsilon \leq 2/\alpha$ be two scalars. For any vector x, y in S, denote

$$z = x - \alpha(f(x) + \varepsilon x),\ w = y - \alpha(f(y) + \varepsilon y).$$

By using the notation of (3) and Lemma 2.1, we have

$$\begin{aligned}
&(x-y)^T(\pi_{\alpha,\varepsilon}(x) - \pi_{\alpha,\varepsilon}(y)) \\
&= (x-y)^T[(z - \Pi_K(z) - (w - \Pi_K(w)) + \alpha(f(x) + \varepsilon x) \\
&\qquad - \ \alpha(f(y) + \varepsilon y))] \\
&= (x-y)^T(u_z - u_w) + \alpha\varepsilon\|x-y\|^2 + \alpha(x-y)^T(f(x) - f(y)) \\
&= [z + \alpha(f(x) + \varepsilon x) - (w + \alpha(f(y) + \varepsilon y))]^T(u_z - u_w) \\
&\quad + \alpha\varepsilon\|x-y\|^2 + \alpha(x-y)^T(f(x) - f(y)) \\
&= (z-w)^T(u_z - u_w) + \alpha[f(x) + \varepsilon x - (f(y) + \varepsilon y)]^T(u_z - u_w)
\end{aligned}$$

$$
\begin{aligned}
&\quad + \alpha\varepsilon\|x-y\|^2 + \alpha(x-y)^T(f(x)-f(y)) \\
&\geq \|u_z-u_w\|^2 + \alpha[f(x)+\varepsilon x-(f(y)+\varepsilon y)]^T(u_z-u_w) \\
&\quad + \alpha\varepsilon\|x-y\|^2 + \alpha(x-y)^T(f(x)-f(y)) \\
&\geq -(\alpha^2/4)\|f(x)+\varepsilon x-(f(y)+\varepsilon y)\|^2 + \alpha\varepsilon\|x-y\|^2 \\
&\quad + \alpha(x-y)^T(f(x)-f(y)) \\
&= \left(\alpha\varepsilon-\alpha^2\varepsilon^2/4\right)\|x-y\|^2 - (\alpha^2/4)\|f(x)-f(y)\|^2 \\
&\quad + (\alpha-\alpha^2\varepsilon/2)(x-y)^T(f(x)-f(y)). \qquad (4)
\end{aligned}
$$

If f is cocoercive with modulus $\beta > 0$, using $\varepsilon \leq 2/\alpha$ we see from the above that

$$
\begin{aligned}
&(x-y)^T(\pi_{\alpha,\varepsilon}(x)-\pi_{\alpha,\varepsilon}(y)) \\
&\geq \left(\alpha\varepsilon-\alpha^2\varepsilon^2/4\right)\|x-y\|^2 - (\alpha^2/4)\|f(x)-f(y)\|^2 \\
&\quad + (\alpha-\alpha^2\varepsilon/2)\beta\|f(x)-f(y)\|^2 \\
&= \alpha\varepsilon(1-\alpha\varepsilon/4)\|x-y\|^2 + \alpha^2\beta\left(\frac{1}{\alpha}-\frac{1}{4\beta}-\frac{\varepsilon}{2}\right)\|f(x)-f(y)\|^2.
\end{aligned}
$$

Setting $\varepsilon = 0$ in the above inequality, we see that for $0 < \alpha \leq 4\beta$ the right-hand side is nonnegative, showing that π_α is monotone on the set S. This proves the result (i). Also, if $\alpha < 4\beta$ and $0 < \varepsilon \leq 2(\frac{1}{\alpha} - \frac{1}{4\beta})$, the right-hand side of the above inequality is greater than or equal to $r\|x-y\|^2$, where $r = \alpha\varepsilon(1-\alpha\varepsilon/4) > 0$, showing that $\pi_{\alpha,\varepsilon}$ is strongly monotone on the set S. The proof of the result (iii) is complete.

Assume that f is strongly monotone with modulus $c > 0$ and Lipschitz continuous with constant $L > 0$. We now prove the result (ii). For this case, setting $\varepsilon = 0$ in (4), we have that

$$
\begin{aligned}
&(x-y)^T(\pi_\alpha(x)-\pi_\alpha(y)) \\
&\geq -(\alpha^2/4)\|f(x)-f(y)\|^2 + \alpha(x-y)^T(f(x)-f(y)) \\
&\geq -(\alpha^2L^2/4)\|x-y\|^2 + \alpha c\|x-y\|^2 \\
&= (\alpha c - \alpha^2L^2/4)\|x-y\|^2.
\end{aligned}
$$

For $\alpha < 4c/L^2$, it is evident that the scalar

$$
r = \alpha c - \frac{\alpha^2L^2}{4} = \frac{\alpha L^2}{4}\left(\frac{4c}{L^2}-\alpha\right) > 0.
$$

Result (ii) is proved. □

Similarly, we have the following result for $\Phi_\alpha(x)$.

THEOREM 2.2. *Let f be a function from R^n into itself and K be a closed convex set and $K \subseteq S \subseteq R^n$.*

(i) *If f is cocoercive with modulus $\beta > 0$ on the set S, then for any constant α such that $\alpha > 1/(4\beta)$, the normal map $\Phi_\alpha(x)$ given by (2) is monotone on the set S.*

(ii) *If f is strongly monotone with modulus $c > 0$ and Lipschitz continuous with constant $L > 0$ on the set S, then for any α satisfying $\alpha > L^2/(4c)$, the normal map $\Phi_\alpha(x)$ given by (2) is strongly monotone on the set S.*

(iii) *If f is cocoercive with modulus $\beta > 0$ on the set S, then for any constant $\alpha > 1/(4\beta)$, the perturbed normal map $\Phi_{\alpha,\varepsilon}(x)$, where $0 < \varepsilon < \alpha$, is strongly monotone in x on the set S.*

Proof. Let α, ε, r be given such that $\alpha > \varepsilon \geq r \geq 0$. For any vector x, y in S, let u_x and u_y be defined by (3) with $z = x$ and $z = y$, respectively. Then, by Lemma 2.1 we have

$$\begin{aligned}&(\alpha - r)\|u_x - u_y\|^2 + (u_x - u_y)^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &\geq -\frac{1}{4(\alpha - r)}\|f(\Pi_K(x)) - f(\Pi_K(y))\|^2 \qquad (5)\end{aligned}$$

and

$$(x - y)^T(u_x - u_y) \geq \|u_x - u_y\|^2, \qquad (6)$$

which further implies

$$\|x - y\| \geq \|u_x - u_y\|.$$

By using the above three inequalities, we have

$$\begin{aligned}&(x - y)^T(\Phi_{\alpha,\varepsilon}(x) - \Phi_{\alpha,\varepsilon}(y)) - r\|x - y\|^2\\ &= (x - y)^T[f(\Pi_K(x)) + \varepsilon\Pi_K(x) + \alpha u_x - f(\Pi_K(y)) - \varepsilon\Pi_K(y) - \alpha u_y]\\ &\quad -r\|x - y\|^2\\ &= \alpha(x - y)^T(u_x - u_y) + \varepsilon(x - y)^T(\Pi_K(x) - \Pi_K(y)) - r\|x - y\|^2\\ &\quad + (x - y)^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &= (\alpha - \varepsilon)(x - y)^T(u_x - u_y) + (\varepsilon - r)\|x - y\|^2\\ &\quad + (x - y)^T(f(\Pi_K(x)) - f(\Pi_K(y))) \qquad (7)\\ &= (\alpha - \varepsilon)(x - y)^T(u_x - u_y) + (\varepsilon - r)\|x - y\|^2 + (u_x - u_y)^T(f(\Pi_K(x))\\ &\quad -f(\Pi_K(y))) + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &\geq (\alpha - \varepsilon)\|u_x - u_y\|^2 + (\varepsilon - r)\|u_x - u_y\|^2 + (u_x - u_y)^T(f(\Pi_K(x))\\ &\quad -f(\Pi_K(y))) + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &= (\alpha - r)\|u_x - u_y\|^2 + (u_x - u_y)^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &\quad + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &\geq -\frac{1}{4(\alpha - r)}\|f(\Pi_K(x)) - f(\Pi_K(y))\|^2\\ &\quad + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y))). \qquad (8)\end{aligned}$$

Let f be cocoercive with modulus $\beta > 0$ on the set S. Setting $\varepsilon = r = 0$ in the above inequality, and using the cocoercivity of f, we have

$$\begin{aligned}(x - y)^T(\Phi_\alpha(x) - \Phi_\alpha(y)) &\geq -\frac{1}{4\alpha}\|f(\Pi_K(x)) - f(\Pi_K(y))\|^2 +\\ &\quad (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y)))\\ &\geq \left(\beta - \frac{1}{4\alpha}\right)\|f(\Pi_K(x)) - f(\Pi_K(y))\|^2.\end{aligned}$$

For $\alpha > 1/(4\beta)$, the right-hand side is nonnegative, and hence the map Φ_α is monotone on the set S. This proves the result (i).

Let $\alpha > 1/(4\beta)$, $0 < \varepsilon < \alpha$, and $0 < r < \min\{\varepsilon,\ \alpha - 1/(4\beta)\}$. By the cocoercivity of f, the inequality (8) can be further written as

$$\begin{aligned}&(x - y)^T(\Phi_{\alpha,\varepsilon}(x) - \Phi_{\alpha,\varepsilon}(y)) - r\|x - y\|^2\\ &\geq \left(\beta - \frac{1}{4(\alpha - r)}\right)\|f(\Pi_K(x)) - f(\Pi_K(y)\|^2.\end{aligned}$$

Since $0 < r < \alpha - 1/(4\beta)$, the right-hand side of the above is nonnegative, and thus the map $\Phi_{\alpha,\varepsilon}$ is strongly monotone on the set S. Result (iii) is proved.

Finally, we prove result (ii). Assume that f is strongly monotone with modulus $c > 0$ and Lipschitz continuous with constant $L > 0$. For any vector x, y in S, we note that (7) holds for any $\alpha > 0, \varepsilon \geq 0$ and $r \geq 0$. Setting $\varepsilon = 0$, (7) reduces to

$$\begin{aligned}
&(x-y)^T(\Phi_\alpha(x) - \Phi_\alpha(y)) - r\|x-y\|^2 \\
&= \alpha(x-y)^T(u_x - u_y) - r\|x-y\|^2 \\
&\quad + (x-y)^T(f(\Pi_K(x)) - f(\Pi_K(y))). \qquad (9)
\end{aligned}$$

Given $\alpha > L^2/(4c)$, let r be a scalar such that $0 < r < \alpha/2$ and $2r + \frac{L^2}{4(\alpha-2r)} < c$. Notice that

$$\begin{aligned}
\|x-y\|^2 &= \|\Pi_K(x) - \Pi_K(y) + u_x - u_y\|^2 \\
&\leq 2(\|\Pi_K(x) - \Pi_K(y)\|^2 + \|u_x - u_y\|^2).
\end{aligned}$$

Substituting the above into (9) and using inequalities (5) and (6), we have

$$\begin{aligned}
&(x-y)^T(\Phi_\alpha(x) - \Phi_\alpha(y)) - r\|x-y\|^2 \\
&\geq \alpha\|u_x - u_y\|^2 - 2r(\|\Pi_K(x) - \Pi_K(y)\|^2 + \|u_x - u_y\|^2) \\
&\quad + (x-y)^T(f(\Pi_K(x)) - f(\Pi_K(y)) \\
&= (\alpha - 2r)\|u_x - u_y\|^2 + (u_x - u_y)^T(f(\Pi_K(x)) - f(\Pi_K(y))) \\
&\quad + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y))) - 2r\|\Pi_K(x) - \Pi_K(y)\|^2 \\
&\geq -\frac{1}{4(\alpha-2r)}\|f(\Pi_K(x)) - f(\Pi_K(y))\|^2 - 2r\|\Pi_K(x) - \Pi_K(y)\|^2 \\
&\quad + (\Pi_K(x) - \Pi_K(y))^T(f(\Pi_K(x)) - f(\Pi_K(y))) \\
&\geq \left(-\frac{L^2}{4(\alpha-2r)} - 2r + c\right)\|\Pi_K(x) - \Pi_K(y)\|^2,
\end{aligned}$$

where the last inequality follows from the Lipschitz continuity and strong monotonicity of f. The right-hand side of the above is nonnegative. Thus, the map Φ_α is strongly monotone on the set S. This proves result (ii). ∎

The following result is an immediate consequence of Theorems 2.1 and 2.2.

COROLLARY 2.1. *Assume that f is monotone and Lipschitz continuous with constant $L > 0$ on a set $S \supseteq K$.*

(i) *If $0 < \varepsilon < \infty$ and $0 < \alpha < \frac{4\varepsilon}{(L+\varepsilon)^2}$, then the perturbed map $\pi_{\alpha,\varepsilon}(x)$ is strongly monotone in x on the set S.*

(ii) *If $0 < \varepsilon < \infty$ and $\alpha > \frac{(L+\varepsilon)^2}{4\varepsilon}$, then the perturbed normal map $\Phi_{\alpha,\varepsilon}(x)$ is strongly monotone in x on the set S.*

Proof. Let $\varepsilon \in (0, \infty)$ be a fixed scalar. It is evident that under the condition of the corollary, the function $F(x) = f(x) + \varepsilon x$ is strongly monotone with modulus $\varepsilon > 0$ and Lipschitz continuous with constant $L + \varepsilon$. Therefore, from Theorem 2.1(ii) we deduce that if $0 < \alpha < 4\varepsilon/(L+\varepsilon)^2$, the map $\pi_{\alpha,\varepsilon}(x)$ is strongly monotone on S. Similarly, the strong monotonicity of $\Phi_{\alpha,\varepsilon}(x)$ follows from Theorem 2.2(ii). ∎

Items (iii) in both Theorem 2.1 and Theorem 2.2 show that for any sufficiently small parameter ε, the perturbed fixed point and normal maps are strongly monotone. This result is quite different from Corollary 2.1. When α is a fixed constant, Corollary 2.1 does not cover the case where ε can be sufficiently small. Indeed, for a fixed $\alpha > 0$, the inequalities $0 < \alpha < \frac{4\varepsilon}{(L+\varepsilon)^2}$ and $\alpha > \frac{(L+\varepsilon)^2}{4\varepsilon}$ fail to hold when $\varepsilon \to 0$.

Up to now, we have shown that the fixed point map $\pi_\alpha(x)$ (respectively, the normal map $\Phi_\alpha(x)$) is monotone if f is cocoercive with modulus $\beta > 0$ and $\alpha \in (0, 4\beta]$ (respectively, $\alpha \in (1/(4\beta), \infty)$). This result includes those known from Sibony [20] and Gabay [7] as special cases. Under the same assumption on f and α, we deduce from items (iii) of Theorems 2.1 and 2.2 that the perturbed forms $\pi_{\alpha,\varepsilon}$ and $\Phi_{\alpha,\varepsilon}$ are strongly monotone provided that the scalar ε is sufficiently small. In the succeeding sections, we will introduce an application of the above results on globally convergent iterative algorithms for VI(K, f) whose fixed point map or normal map is monotone.

3. Application: Iterative algorithm for VI(K, f). Since $\pi_\alpha(x)$ and $\Phi_\alpha(x)$ are monotone if the function f is cocoercive and α lies in a certain interval, we can solve the cocoercive variational inequity problems via solving the system of monotone equation $\pi_\alpha(x) = 0$ or $\Phi_\alpha(x) = 0$. Recently, Solodov and Svaiter [21] (see also [22, 23, 24]) proposed a class of inexact Newton methods for monotone equations. Let $\mathcal{F}(x)$ be a monotone mapping from R^n into R^n. The Solodov and Svaiter algorithm for the equation $\mathcal{F}(x) = 0$ proceeds as follows.

ALGORITHM SS (see [21]). Choose any $x^0 \in R^n, t \in (0, 1)$, and $\lambda \in (0, 1)$. Set $k := 0$.

Inexact Newton step. Choose a psd matrix G_k. Choose $\mu_k > 0$ and $\gamma_k \in [0, 1)$. Compute $d^k \in R^n$ such that

$$0 = \mathcal{F}(x^k) + (G_k + \mu_k I)d^k + e^k,$$

where $\|e^k\| \le \gamma_k \mu_k \|d^k\|$. Stop if $d^k = 0$. Otherwise,

Line-search step. Find $y^k = x^k + \alpha_k d^k$, where $\alpha_k = t^{m_k}$ with m_k being the smallest nonnegative integer m such that

$$-\mathcal{F}(x^k + t^m d^k)^T d^k \ge \lambda(1 - \gamma_k)\mu_k \|d^k\|^2.$$

Projection step. Compute

$$x^{k+1} = x^k - \frac{\mathcal{F}(y^k)^T(x^k - y^k)}{\|\mathcal{F}(y^k)\|^2}\mathcal{F}(y^k).$$

Set $k := k + 1$, and repeat.

As pointed out in [21], the above inexact Newton step is motivated by the idea of the proximal point algorithm [2, 6, 19]. Algorithm SS has an advantage over other Newton methods in that the whole iteration sequence is globally convergent to a solution of the system of equations, provided a solution exists, under no assumption on $\mathcal{F}$ other than continuity and monotonicity. Setting $\mathcal{F}(x) = \pi_\alpha(x)$ or $\Phi_\alpha(x)$, from Theorems 2.1 and 2.2 in this paper and Theorem 2.1 in [21], we have the following result.

THEOREM 3.1. *Let f be a cocoercive map with constant $\beta > 0$. Substitute $\mathcal{F}(x)$ in Algorithm SS by $\pi_\alpha(x)$ (respectively, $\Phi_\alpha(x)$) where $0 < \alpha \le 4\beta$ (respectively, $\alpha > 1/4\beta$). If μ_k is chosen such that $C_2 \ge \mu_k \ge C_1\|\mathcal{F}(x^k)\|$, where C_1 and C_2 are two constants, then Algorithm SS converges to a solution of the variational inequality provided that a solution exists.*

While Algorithm SS can be used to solve the monotone equations $\pi_\alpha(x) = 0$ and $\Phi_\alpha(x) = 0$, each line-search step needs to compute the values of $\pi_\alpha(x^k + \beta^m d^k)$ and $\Phi_\alpha(x^k + \beta^m d^k)$, which represents a major cost of the algorithm in calculating projection operations. Hence, in general cases, Algorithm SS has high computational cost per iteration when applied to solve $\Phi_\alpha(x) = 0$ or $\pi_\alpha(x) = 0$. To reduce this major

computational burden, we propose the following algorithm which needs no projection operations other than the evaluation of the function f in line-search steps.

ALGORITHM 3.1. Choose $x^0 \in R^n, t \in (0,1)$, and $\gamma \in [0,1)$. Set $k := 0$.

Inexact Newton Step: Choose a positive semidefinite matrix G_k. Choose $\mu_k > 0$. Compute $d^k \in R^n$ such that

$$0 = \pi_\alpha(x^k) + (G_k + \mu_k I)d^k + e^k, \tag{10}$$

where $\|e^k\| \le \gamma\mu_k\|d^k\|$. Stop if $d^k = 0$. Otherwise,

Line-search step. Find $y^k = x^k + s_k d^k$, where $s_k = t^{m_k}$ with m_k being the smallest nonnegative integer m such that

$$\|f(x^k + t^m d^k) - f(x^k)\| < \frac{(1-\gamma)\mu_k - 4t^m}{2\alpha}\|d^k\|. \tag{11}$$

Projection step. Compute

$$x^{k+1} = x^k - \frac{\pi_\alpha(y^k)^T(x^k - y^k)}{\|\pi_\alpha(y^k)\|^2}\pi_\alpha(y^k).$$

Set $k := k + 1$. Return.

The above algorithm has the following property.

LEMMA 3.1. *Let $\pi_\alpha(x)$ be given as* (1). *At kth iteration, if m_k is the smallest nonnegative integer such that* (11) *holds, then $y^k = x^k + t^{m_k}d^k$ satisfies the following estimation:*

$$-\pi_\alpha(y^k)^T d^k \ge \frac{1}{2}(1-\gamma)\mu_k\|d^k\|^2.$$

Proof. By the definition of $\pi_\alpha(x)$, the nonexpansiveness of the projection operator, and (11), we have

$$\begin{aligned}
&\|\pi_\alpha(x^k + t^{m_k}d^k) - \pi_\alpha(x^k)\| \\
&= \|x^k + t^{m_k}d^k - \Pi_K(x^k + t^{m_k}d^k - \alpha f(x^k + t^{m_k}d^k)) \\
&\quad -(x^k - \Pi_K(x^k - \alpha f(x^k)))\| \\
&\le t^{m_k}\|d^k\| + \|\Pi_K(x^k + t^{m_k}d^k - \alpha f(x^k + t^{m_k}d^k)) \\
&\quad -\Pi_K(x^k - \alpha f(x^k))\| \\
&\le t^{m_k}\|d^k\| + \|x^k + t^{m_k}d^k - \alpha f(x^k + t^{m_k}d^k) \\
&\quad -(x^k - \alpha f(x^k))\| \\
&\le 2t^{m_k}\|d^k\| + \alpha\|f(x^k + t^{m_k}d^k) - f(x^k)\| \\
&\le \frac{1}{2}(1-\gamma)\mu_k\|d^k\|. \qquad (12)
\end{aligned}$$

Also,

$$\begin{aligned}
&-\pi_\alpha(x^k + t^{m_k}d^k)^T d^k \\
&= -[\pi_\alpha(x^k + t^{m_k}d^k) - \pi_\alpha(x^k)]^T d^k - \pi_\alpha(x^k)^T d^k \\
&\ge -\|\pi_\alpha(x^k + t^{m_k}d^k) - \pi_\alpha(x^k)\|\|d^k\| - \pi_\alpha(x^k)^T d^k. \qquad (13)
\end{aligned}$$

By (10) and positive semidefiniteness of G_k, we have

$$\begin{aligned}
-\pi_\alpha(x^k)^T d^k &= (d^k)^T(G_k + \mu_k I)d^k + (e^k)^T d^k \\
&\ge \mu_k\|d^k\|^2 - \gamma\mu_k\|d^k\|^2 \\
&= (1-\gamma)\mu_k\|d^k\|^2. \qquad (14)
\end{aligned}$$

Combining (12), (13), and (14) yields

$$-\pi_\alpha(x^k + t^{m_k} d^k)^T d^k \geq \frac{1}{2}(1-\gamma)\mu_k \|d^k\|^2.$$

The proof is complete. □

Using Lemma 3.1 and following the line of the proof of Theorem 2.1 in [21], it is not difficult to prove the following convergence result.

THEOREM 3.2. *Let $f : R^n \to R^n$ be a continuous function such that there exists a constant $\alpha > 0$ such that $\pi_\alpha(x)$ defined by (1) is monotone. Choose G_k and μ_k such that $\|G_k\| \leq C'$ and $\mu_k = C\|\pi_\alpha(x^k)\|^p$, where C', C and p are three fixed positive numbers and $p \in (0, 1]$. Then the sequence $\{x^k\}$ generated by Algorithm* 3.1 *converges to a solution of the variational inequality provided that a solution exists.*

Algorithm 3.1 can solve the variational inequality whose fixed point mapping $\pi_\alpha(x)$ is monotone for some $\alpha > 0$. Since the cocoercivity of f implies the monotonicity of the functions $\pi_\alpha(x)$ and $\Phi_\alpha(x)$ for suitable choices of the value of α, Algorithm 3.1 can locate a solution of any solvable cocoercive variational inequality problem. This algorithm has an advantage over Algorithm SS in that it does not carry out any projection operation in the line-search step and hence the computational cost is significantly reduced.

4. Conclusions. In this paper, we show some sufficient conditions for the monotonicity (strong monotonicity) of the fixed point and normal maps associated with the variational inequality problem. The results proved in the paper encompass some known results as particular cases. Based on these results, an iterative algorithm for a class of variational inequalities is proposed. This algorithm can be viewed as a modification of Solodov and Svaiter's method but has lower computational cost than the latter.

Acknowledgments. The authors would like to thank two anonymous referees for their incisive comments and helpful suggestions which helped us improve many aspects of the paper. The authors also thank Professor O. L. Mangasarian for encouragement and one referee for pointing out [7, 20].

REFERENCES

[1] R. E. BRUCK, JR., *An iterative solution of a variational inequality for certain monotone operators in Hilbert space,* Bull. Amer. Math. Soc., 81 (1975), pp. 890–892.

[2] J. ECKSTEIN AND D. P. BERTSEKAS, *On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators,* Math. Program., 55 (1992), pp. 293–318.

[3] F. FACCHINEI, *Structural and stability properties of P_0 nonlinear complementarity problems,* Math. Oper. Res., 23 (1998), pp. 735–749.

[4] F. FACCHINEI AND C. KANZOW, *Beyond monotonicity in regularization methods for nonlinear complementarity problems,* SIAM J. Control Optim., 37 (1999), pp. 1150–1161.

[5] F. FACCHINEI AND J. S. PANG, *Total Stability of Variational Inequalities,* Dipartimento di Informatica e Sistemistica, Università di Roma, Via Buonarroti, Roma, 1998.

[6] M. C. FERRIS, *Finite termination of the proximal point algorithm,* Math. Program., 50 (1991), pp. 359–366.

[7] D. GABAY, *Applications of the method of multipliers to variational inequalities,* Augmented Lagrangian Methods: Application to the Numerical Solution of Boundary-Value Problems, M. Fortin and R. Glowinski, eds., North–Holland, Amsterdam, 1983, pp. 229–332.

[8] M. S. GOWDA AND M. A. TAWHID, *Existence and limiting behavior of trajectories associated with P_0 equations,* Comput. Optim. Appl., 12 (1999), pp. 229–251.

[9] P. T. Harker and J. S. Pang, *Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications,* Math. Program., 48 (1990), pp. 161–220.
[10] G. Isac, *Tikhonov's regularization and the complementarity problem in Hilbert spaces,* J. Math. Anal. Appl., 174 (1993), pp. 53–66.
[11] G. Isac, *A generalization of Karamardian's condition in complementarity theory,* Nonlinear Anal. Forum, 4 (1999), pp. 49–63.
[12] Z. Q. Luo and P. Tseng, *A decomposition property for a class of square matrices,* Appl. Math. Lett., 4 (1991), pp. 67–69.
[13] T. L. Magnanti and G. Perakis, *A unifying geometric solution framework and complexity analysis for variational inequalities,* Math. Program., 71 (1996), pp. 327–352.
[14] T. L. Magnanti and G. Perakis, *The orthogonality theorem and the strong-f-monotonicity condition for variational inequality algorithms,* SIAM J. Optim., 7 (1997), pp. 248–273.
[15] P. Marcotte and J. H. Wu, *On the convergence of projection methods: Application to the decomposition of affine variational inequalities,* J. Optim. Theory Appl., 85 (1995), pp. 347–362.
[16] H. D. Qi, *Tihonov regularization methods for variational inequality problems,* J. Optim. Theory Appl., 102 (1999), pp. 193–201.
[17] G. Ravindran and M. S. Gowda, *Regularization of P_0-functions in box variational inequality problems,* SIAM J. Optim., 11 (2000), pp. 748–760.
[18] S. M. Robinson, *Normal maps induced by linear transformations,* Math. Oper. Res., 16 (1992), pp. 292–309.
[19] R. T. Rockafellar, *Monotone operators and the proximal point algorithm,* SIAM J. Control Optim., 14 (1976), pp. 877–898.
[20] M. Sibony, *Méthods itératives pour les équations et inéquations aus dérivées partielles nonlinéaries de type monotone,* Calcolo, 7 (1970), pp. 65–183.
[21] 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.
[22] 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.
[23] M. V. Solodov and B. F. Svaiter, *A hybrid projection-proximal point algorithm,* J. Convex Anal., 6 (1999), pp. 59–70.
[24] M. V. Solodov and B. F. Svaiter, *A new projection method for variational inequality problems,* SIAM J. Control Optim., 37 (1999), pp. 765–776.
[25] P. Tseng, *Further applications of a matrix splitting algorithm to decomposition in variational inequalities and convex programming,* Math. Program., 48 (1990), pp. 249–264.
[26] V. Venkateswaran, *An algorithm for the linear complementarity problem with a P_0-matrix,* SIAM J. Matrix Anal. Appl., 14 (1993), pp. 967–977.
[27] Y. B. Zhao and D. Li, *Strict feasibility condition in nonlinear complementarity problems,* J. Optim. Theory Appl., 107 (2000), pp. 641–664.
[28] Y. B. Zhao and D. Li, *On a new homotopy continuation trajectory for nonlinear complementarity problems,* Math. Oper. Res., 26 (2001), pp. 119–146.
[29] D. Zhu and P. Marcotte, *New classes of generalized monotonicity,* J. Optim. Theory Appl., 87 (1995), pp. 457–471.
[30] D. Zhu and P. Marcotte, *Co-coercivity and its role in the convergence of iterative schemes for solving variational inequalities,* SIAM J. Optim., 6 (1996), pp. 714–726.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 974–1002

SUPERLINEAR CONVERGENCE OF PRIMAL-DUAL INTERIOR POINT ALGORITHMS FOR NONLINEAR PROGRAMMING*

NICHOLAS I. M. GOULD†, DOMINIQUE ORBAN‡, ANNICK SARTENAER§, AND PHILIPPE L. TOINT¶

Abstract. The local convergence properties of a class of primal-dual interior point methods are analyzed. These methods are designed to minimize a nonlinear, nonconvex, objective function subject to linear equality constraints and general inequalities. They involve an inner iteration in which the log-barrier merit function is approximately minimized subject to satisfying the linear equality constraints, and an outer iteration that specifies both the decrease in the barrier parameter and the level of accuracy for the inner minimization. Under nondegeneracy assumptions, it is shown that, asymptotically, for each value of the barrier parameter, solving a single primal-dual linear system is enough to produce an iterate that already matches the barrier subproblem accuracy requirements. The asymptotic rate of convergence of the resulting algorithm is Q-superlinear and may be chosen arbitrarily close to quadratic. Furthermore, this rate applies componentwise. These results hold in particular for the method described in [A. R. Conn, N. I. M. Gould, D. Orban, and P. L. Toint, *Math. Program. Ser. B*, 87 (2000), pp. 215–249] and indicate that the details of its inner minimization are irrelevant in the asymptotics, except for its accuracy requirements.

Key words. primal-dual interior point method, componentwise Q-superlinear convergence

AMS subject classifications. 65K05, 90C26, 90C30, 90C51

PII. S1052623400370515

1. Introduction.

In this paper, we aim to provide insight on the local behavior of a class of primal-dual interior point algorithms. The class of algorithms is intended to solve nonconvex nonlinear programs that involve linear equality and nonlinear inequality constraints using a barrier-type method. More specifically, we consider the following problem:

$$\text{NLP} \equiv \left\{ \begin{array}{ll} \min & f(x) \\ \text{s.t.} & Ax = b, \\ & c(x) \geq 0, \end{array} \right. \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $c : \mathbb{R}^n \to \mathbb{R}^p$ are assumed to be twice continuously differentiable, the matrix $A \in \mathbb{R}^{m \times n}$ $(m \leq n)$ has full rank and $b \in \mathbb{R}^m$. The method starts with a strictly feasible initial point, and, rather than solving NLP directly, instead approximately solves a sequence of barrier subproblems of the form

$$\text{BS}(\mu) \equiv \left\{ \begin{array}{ll} \min & \phi(x, \mu) \\ \text{s.t.} & Ax = b \end{array} \right. \tag{1.2}$$

*Received by the editors April 5, 2000; accepted for publication (in revised form) December 6, 2000; published electronically April 26, 2001. This work was supported by the MNRT grant for joint Ph.D. Support.

http://www.siam.org/journals/siopt/11-4/37051.html

†Rutherford Appleton Laboratory, Chilton, Oxfordshire OX11 0QX, UK (N.I.M.Gould@ rl.ac.uk).

‡CERFACS, 42, av. Gaspard Coriolis, Toulouse 31057, France (Dominique.Orban@cerfacs.fr).

§Research Associate of the Belgian National Fund for Scientific Research. Department of Mathematics, Facultés Universitaires Notre-Dame de la Paix, Namur, Belgium (Annick.Sartenaer@fundp.ac.be).

¶Department of Mathematics, Facultés Universitaires Notre-Dame de la Paix, Namur, Belgium (Philippe.Toint@fundp.ac.be).

for a decreasing sequence $\{\mu_k\}$ of positive barrier parameters. Here the barrier function is

$$\phi(x,\mu) \stackrel{\text{def}}{=} f(x) - \mu \sum_{i=1}^{p} \log c_i(x), \tag{1.3}$$

where the functions $c_i(\cdot)$ are the components of the vector function $c(\cdot)$. Once an (approximate) solution x_{k+1} of BS(μ_k) is found, the parameter μ_k is updated and attention turns to the next barrier subproblem (see, for instance, [6, 16, 18] for a general survey and [19] for the linear case). Under reasonable conditions [2, 6, 16], it can be shown that the sequence $\{x_k\}$ converges to a stationary point x^* of NLP. A typical stopping criterion for the solution of BS(μ) is

$$\|P_{\mathcal{N}(A)}(\nabla_x \phi(x,\mu))\| \leq \vartheta(\mu), \tag{1.4}$$

where $P_{\mathcal{N}(A)}$ is the orthogonal projection onto the nullspace of A, $\|\cdot\|$ is some norm defined on $\mathbb{R}^n$, and the continuous function $\vartheta : \mathbb{R}_+ \to \mathbb{R}_+$ is a *forcing function*, that is, $\vartheta(\mu) = 0$ if and only if $\mu = 0$.

The most intensive part of the solution procedure is in the approximate solution of successive barrier subproblems BS(μ) (the *inner minimizations*), whose difficulty depends on the chosen starting point. An obvious idea is to start the solution of BS(μ_{k+1}) from x_{k+1}. However, in the primal case, for both linear and nonlinear programming, it has been shown that the unit Newton step for a barrier subproblem is likely not to be accepted as a first step if the minimization process is started from x_{k+1}, even if x_{k+1} is close to a solution of NLP [17]. The determination of better initial points in interior methods has recently been examined by several authors, both for primal [3, 12] and primal-dual barrier methods [1, 22, 23], as well as for exterior penalty methods (see [10], whose results were the inspiration for [5]). In particular, Dussault [5] expands the primal central path about the current iterate. In this paper, we apply a similar analysis in the more general primal-dual framework. This framework offers the advantage of keeping the radius of the "sphere of convergence" of Newton's method away from zero (under nondegeneracy assumptions), whereas this radius is proportional to the barrier parameter in a purely primal scheme, as is shown in [14] for the case of linear programming and in [15] for nonlinear programming.

In this paper, we intend to determine conditions under which, asymptotically, a single Newton step is strictly feasible (in contrast with the purely primal case), and results in a point that satisfies suitable barrier subproblem termination rules, after every reduction of the barrier parameter (see, for instance, [1, 3, 4] for previous work on the subject). This is shown to imply a componentwise Q-superlinear rate of convergence, a stronger result than simply Q-superlinear convergence of the vector of variables and Lagrange multipliers. Furthermore, this rate of convergence may be made arbitrarily close to quadratic. The results we present hold independently of the particular algorithm used for the inner minimization. They may thus be interpreted as giving conditions on the inner minimization stopping criterion to ensure fast convergence, in a manner similar to that studied by [13] for linear complementarity problems.

The motivation for the results presented in this paper is that they cover the general algorithm of [2], as will be discussed below. This algorithm has been implemented as `HSL_VE12` in the Harwell Subroutine Library for the special case of quadratic programming problems. We refer the reader to [2] for further motivation and details,

along with the results of tests performed on a number of large convex and nonconvex quadratic examples.

The paper is organized as follows. Section 2 describes the notation and assumptions used throughout the paper, and section 3 provides useful preliminary results. In section 4, we state the class of algorithms that will be analyzed; in section 5 we describe an extrapolation of the central path that provides a point which turns out to be a very good estimate of the solution of the next subproblem. We subsequently present the local convergence properties of the algorithm when using the aforementioned extrapolation in section 6. In section 7, we review the link between our class of algorithms and the method of [2]; we also briefly discuss the connections with other proposals. We conclude and give some comments in section 8.

2. Notation and assumptions. In this section, we present our notation and outline the required assumptions for the algorithm to converge superlinearly.

2.1. Notation. The following notation will be used throughout the paper. For related positive quantities α and β, we write $\alpha = \mathcal{O}(\beta)$ if there is a constant $\kappa > 0$ such that $\alpha \leq \kappa\beta$ for all β sufficiently small. We write $\alpha = o(\beta)$ if $\alpha/\beta \to 0$ as $\beta \to 0$. We also write $\alpha = \Omega(\beta)$ if $\beta = \mathcal{O}(\alpha)$, and write $\alpha = \Theta(\beta)$ if $\alpha = \mathcal{O}(\beta)$ and $\beta = \mathcal{O}(\alpha)$.

If x is any vector in $\mathbb{R}^n$, the corresponding capital letter X will denote the diagonal matrix $\mathrm{diag}(x)$. The components of x will be denoted by $[x]_1, \ldots, [x]_n$. We shall sometimes define a vector $w \in \mathbb{R}^{n+m+p}$ from $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, and $z \in \mathbb{R}^p$ by $w = (x, y, z)$. For such a vector, we shall use the notation $[w]_x$, $[w]_y$, and $[w]_z$ to refer respectively to its x, y, and z components.

In the remainder of the paper, the statement "μ small enough" is to be understood as "μ is positive and close enough to zero," and the notation $a \searrow b$ as "a decreases monotonically and converges to b." By "global convergence" we shall mean "convergence to a *local* solution, whatever the point from which the process was started." By "local convergence" we mean "convergence to a local solution when the process is started in the vicinity of that solution."

2.1.1. Optimality conditions. The Lagrangian function for NLP is

$$\mathcal{L}(w) = \mathcal{L}(x, y, z) = f(x) + (Ax - b)^T y - c^T(x)z, \tag{2.1}$$

where the Lagrange multipliers $y \in \mathbb{R}^m$ correspond to the equality constraints and $z \in \mathbb{R}^p_+$ to inequalities. We shall conveniently express the optimality conditions and the local analysis developments of sections 5 and 6 in terms of the following family of functions, parameterized by a scalar $\mu \geq 0$:

$$\Psi(w; \mu) \stackrel{\text{def}}{=} \begin{bmatrix} \nabla_x \mathcal{L}(w) \\ Ax - b \\ C(x)z - \mu e \end{bmatrix}. \tag{2.2}$$

Here $\nabla_x \mathcal{L}(w) = \nabla_x f(x) + A^T y - J^T(x)z$, the matrix $J(x)$ denotes the Jacobian matrix of c at x, that is the p by n matrix whose ith row is $(\nabla_x c_i(x))^T$, and e is the vector of all ones. If $w^* \stackrel{\text{def}}{=} (x^*, y^*, z^*)$ is a first-order critical point for NLP, it must satisfy the first-order Karush–Kuhn–Tucker (KKT) conditions, which are that

$$\Psi(w^*; 0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{2.3}$$

and

$$(c(x^*), z^*) \geq 0. \tag{2.4}$$

When solving problem BS(μ), we seek $x(\mu)$ and $y(\mu)$ such that

$$\nabla_x f(x(\mu)) + A^T y(\mu) - \mu J^T(x(\mu)) C^{-1}(x(\mu)) e = 0, \tag{2.5}$$

$$Ax(\mu) = b, \tag{2.6}$$

$$c(x(\mu)) > 0. \tag{2.7}$$

Solving this system corresponds to using a primal approach. Let $x(\mu)$ and $y(\mu)$ be solutions of (2.5)–(2.7). In this case, the set

$$\{x(\mu) \mid \mu > 0\} \tag{2.8}$$

is said to define a (local) primal central path. Under second-order sufficiency assumptions, a linear independence constraint qualification and a strict complementary slackness condition, the primal central path leads to a solution x^* of NLP and $y(\mu)$ converges to a corresponding vector of Lagrange multipliers y^* (see [6]) as μ decreases to zero. Crucially, (2.5)–(2.7) is equivalent to the system

$$\Psi(w(\mu); \mu) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \qquad (c(x(\mu)), z(\mu)) > 0, \tag{2.9}$$

in the sense that if $(x(\mu), y(\mu))$ solves (2.5)–(2.7), the vector $(x(\mu), y(\mu), z(\mu))$ with $z(\mu) = \mu C^{-1}(x(\mu))e$ solves (2.9), while if $(x(\mu), y(\mu), z(\mu))$ solves (2.9), $(x(\mu), y(\mu))$ solves (2.5)–(2.7) and we have $z(\mu) = \mu C^{-1}(x(\mu))e$. Treating $z(\mu)$ as an independent variable when iteratively solving (2.9) is a primal-dual approach and is the one we adopt in this paper. As $\mu \searrow 0$, the solution $w(\mu)$ of (2.9) then converges to a solution w^* of (2.3) under the aforementioned conditions.

In accordance with the primal-dual theory for linear and convex programming, we use here the following terminology. The term *primal variables* refers to the x variables, *Lagrange multipliers* to y, and *dual variables* to z, although the variables z are Lagrange multipliers, too. When solving a primal-dual system, the set

$$\mathcal{C} \stackrel{\text{def}}{=} \{w(\mu) = (x(\mu), y(\mu), z(\mu)) \mid \mu > 0\} \tag{2.10}$$

is said to define a (local) primal-dual central path.

Note that for any $\mu \geq 0$, $\Psi(w; \mu)$ and $\Psi(w; 0)$ satisfy the fundamental relationship

$$\Psi(w; \mu) = \Psi(w; 0) - \begin{bmatrix} 0 \\ 0 \\ \mu e \end{bmatrix}.$$

This implies that the Jacobian matrices $\nabla_w \Psi(w; \mu)$ and $\nabla_w \Psi(w; 0)$ are equal for every w in the domain of interest and satisfy

$$\nabla_w \Psi(w; \mu) = \nabla_w \Psi(w; 0) = \begin{bmatrix} \nabla_{xx} \mathcal{L}(w) & A^T & -J^T(x) \\ A & 0 & 0 \\ ZJ(x) & 0 & C(x) \end{bmatrix}. \tag{2.11}$$

Moreover, $\nabla_\mu \Psi(w; \mu) = [0 \ 0 \ -e^T]^T$.

2.1.2. Norms. We use the symbol $\|\cdot\|$ to represent the Euclidean ℓ_2-norm, unless otherwise specified. We thus have

$$\|X\| = \|x\|_\infty \leq \|x\| \tag{2.12}$$

for any vector x. If S is a symmetric positive definite matrix, the S-norm of x, $\|x\|_S$, is defined as usual by $\|x\|_S^2 \stackrel{\rm def}{=} x^T S x$.

We let the columns of the n by $n-m$ matrix N be an orthonormal basis for the nullspace of A. We denote the smallest and largest eigenvalues of any $n \times n$ symmetric matrix M by $\lambda^{\min}[M]$ and $\lambda^{\max}[M]$. Such a matrix is said to be *second-order sufficient* (with respect to A) if and only if the reduced matrix

$$R[M] = N^T M N$$

is positive definite (see, for instance, [9]).

In the context of our algorithm, we shall choose to measure gradients and related quantities in a seminorm induced by a second-order sufficient iteration-dependent preconditioner M_k, where k is the index of the current iteration.[1] We define the *k-seminorm* of a vector g, $\|g\|_{[k]}$, by

$$\|g\|_{[k]}^2 \stackrel{\rm def}{=} q^T g, \tag{2.13}$$

where q solves the system

$$\begin{bmatrix} M_k & A^T \\ A & 0 \end{bmatrix} \begin{pmatrix} q \\ r \end{pmatrix} = \begin{pmatrix} g \\ 0 \end{pmatrix}.$$

This is actually a norm if g lies in the nullspace of A. In particular,

$$\|g\|_{[k]} = 0 \quad \text{if and only if} \quad \|N^T g\| = 0.$$

A simple calculation (see, for example, [8, section 5.4.1]) reveals that (2.13) may be expressed as

$$\|g\|_{[k]}^2 = g^T N R^{-1}[M_k] N^T g = \|N^T g\|_{R^{-1}[M_k]}^2 = \|R^{-\frac{1}{2}}[M_k] N^T g\|^2. \tag{2.14}$$

The simplest choice $M_k = I$, for which $R[I] = I$, simply measures the size of the projection of g into the nullspace of A. Note that the k-seminorm is invariant for displacements in the range space of A^T, i.e.,

$$\|g + A^T \overline{g}\|_{[k]} = \|g\|_{[k]} \tag{2.15}$$

for any $g \in \mathbb{R}^n$ and any $\overline{g} \in \mathbb{R}^m$.

In addition, because gradients can be interpreted as linear forms on the space of the problem variables, it is natural to measure quantities directly involving these variables, such as the distance between iterates, in a seminorm corresponding to the dual of $\|\cdot\|_{[k]}$. It is easy to verify that such a seminorm is given by

$$\|s\|_k \stackrel{\rm def}{=} \|N^T s\|_{R[M_k]} \tag{2.16}$$

[1] Strictly, this seminorm also depends on A, but we hide this dependence since A is fixed throughout this paper.

and is, in fact, a norm in the nullspace of A. As a consequence, for all $v, s \in \mathbb{R}^n$ such that $As = 0$, i.e., such that $s = NN^T s$, we have that

$$|v^T s| = |v^T N (N^T M_k N)^{-\frac{1}{2}} (N^T M_k N)^{\frac{1}{2}} N^T s| \leq \|v\|_{[k]} \|s\|_k, \tag{2.17}$$

because of the Cauchy–Schwarz inequality. We stress that there is no need for M_k itself to be positive definite, merely that $N^T M_k N$ has to be.

If U is any symmetric matrix, we also define the reduced matrix

$$R[U, M_k] \stackrel{\text{def}}{=} (N^T M_k N)^{-\frac{1}{2}} N^T U N (N^T M_k N)^{-\frac{1}{2}}; \tag{2.18}$$

we denote its smallest and largest eigenvalues by $\lambda^{\min}_{M_k}[U] = \lambda^{\min}[R[U, M_k]]$ and $\lambda^{\max}_{M_k}[U] = \lambda^{\max}[R[U, M_k]]$. We also note that the inertia of $R[U, M_k]$ and $R[U, I] \equiv N^T U N$ are the same. In particular, we have that

$$\lambda^{\min}_{M_k}[U] \geq 0 \quad \text{is equivalent to} \quad \lambda^{\min}_{I}[U] \geq 0. \tag{2.19}$$

We write $\|v\|_\diamond \stackrel{\text{def}}{=} \|N^T v\| = \|NN^T v\|$, the Euclidean norm of the projection of v onto the nullspace of A, and observe that $\|\cdot\|_\diamond$ is a self-dual norm in this nullspace, and that the k-seminorm and $\|\cdot\|_\diamond$ are equivalent if $M_k = I$. Moreover, we have

$$\|g + A^T \overline{g}\|_\diamond = \|g\|_\diamond \tag{2.20}$$

for any $g \in \mathbb{R}^n$ and any $\overline{g} \in \mathbb{R}^m$, which parallels (2.15).

We also notice the equivalence

$$P_{\mathcal{N}(A)}(\nabla_x \phi(x, \mu)) = 0 \Longleftrightarrow N^T \nabla_x \phi(x, \mu) = 0, \tag{2.21}$$

which in turn is equivalent to

$$\|\nabla_x \phi(x, \mu)\|_\diamond = 0 \quad \text{and to} \quad \|\nabla_x \phi(x, \mu)\|_{[k]} = 0 \tag{2.22}$$

for any second-order sufficient matrix M_k. For future reference, we state the expressions of the first and second derivatives of the barrier function $\phi(x, \mu)$ with respect to x:

$$\nabla_x \phi(x, \mu) = \nabla_x f(x) - \sum_{i=1}^{p} \frac{\mu}{c_i(x)} \nabla_x c_i(x), \tag{2.23}$$

$$\nabla_{xx} \phi(x, \mu) = \nabla_{xx} f(x) + \sum_{i=1}^{p} \frac{\mu}{c_i^2(x)} \nabla_x c_i(x) (\nabla_x c_i(x))^T - \sum_{i=1}^{p} \frac{\mu}{c_i(x)} \nabla_{xx} c_i(x). \tag{2.24}$$

2.2. Assumptions. Let $\mathcal{I} = \{x \mid c(x) \geq 0\}$ be the set of points satisfying the inequalities, $\mathcal{E} = \{x \mid Ax = b\}$ be the set of points satisfying the equality constraints, and the intersection $\mathcal{F} \stackrel{\text{def}}{=} \mathcal{I} \cap \mathcal{E}$ be the set of feasible points for NLP. We assume the following.

AS1. There exists x_0 such that $Ax_0 = b$ and $c(x_0) > 0$.

AS2. The functions $f(\cdot)$ and $c_i(\cdot)$ are twice continuously differentiable over an open set containing $\mathcal{F}$.

Furthermore, if w^* is a solution of (2.3)–(2.4), if we let $\mathcal{A} \stackrel{\text{def}}{=} \{i \mid c_i(x^*) = 0\}$ be the set of indices pertaining to the active inequality constraints at x^*, and if $a^{(j)}$ denotes the jth column of A^T, we then assume the following.

AS3. The vectors $\{\nabla_x c_i(x^*)\}_{i\in\mathcal{A}}$ and $\{a^{(j)}\}_{j=1}^m$ form a linearly independent set in $\mathbb{R}^n$.

AS4. The strong second-order sufficiency condition is satisfied at w^*: $v^T\nabla_{xx}\mathcal{L}(w^*)v > 0$ for any vector $v \neq 0, v \in \mathcal{N}(A)$ such that $\nabla_x c_i(x^*)^T v = 0$ for every $i \in \mathcal{A}$.

AS5. Strict complementary slackness holds, that is $[z^*]_i + c_i(x^*) > 0$ for all $i = 1, \ldots, p$.

Note that AS3 implies that the Lagrange multipliers y^* and the dual variables z^* are unique and that the matrix A has full rank, which is not restrictive since it can always be satisfied by preprocessing the linear system $Ax = b$. Under AS3, AS4, and AS5, x^* is an isolated (and thus strict) local solution of NLP. Throughout the paper, the dependence of $\mathcal{A}$ on x^* will remain hidden as only one local solution of NLP is considered.

3. Preliminary results. In this section, we state some results about central paths which will be useful later.

It is shown in [6, 16] that under AS3, AS4, and AS5, $\nabla_w\Psi(w^*;0)$ is nonsingular, and a continuity argument yields that it remains nonsingular in a small neighborhood of w^*. In the following technical lemma, which is a simple extension of that proved in [20] to the case of linear equality constraints, we now verify that the central path is well-defined in the intersection of this neighborhood and $\mathcal{E}$ and show that it has useful continuity properties.

LEMMA 3.1. *Under* AS2–AS5, *let the vector*

$$w(l,r,\zeta) = (x(l,r,\zeta), y(l,r,\zeta), z(l,r,\zeta))$$

be defined implicitly as the solution of the following nonlinear system:

$$\Psi(w;0) = \begin{bmatrix} NN^T l \\ r \\ \zeta \end{bmatrix} \tag{3.1}$$

for given $l \in \mathbb{R}^n, r \in \mathbb{R}^m$, and $\zeta \in \mathbb{R}^p$, with Ψ defined as in (2.2) *and where the columns of N form an orthonormal basis for the nullspace of A. Then there exist constants $\varepsilon > 0$ and $\kappa > 0$ for which the following statements hold.*

(i) *$w(l,r,\zeta)$ is a continuously differentiable function of (l,r,ζ) in the neighborhood*

$$\mathcal{N}(\varepsilon) \stackrel{\text{def}}{=} \{(l,r,\zeta) \mid \|l\|_\diamond + \|r\| + \|\zeta\| \leq \varepsilon\}.$$

(ii) *For $\zeta > 0$ and $(l,r,\zeta) \in \mathcal{N}(\varepsilon)$, we have $[z(l,r,\zeta)]_i > 0$ and $c_i(x(l,r,\zeta)) > 0$ for all $i = 1, \ldots, p$.*

(iii) *Let $(l_1,r_1,\zeta_1), (l_2,r_2,\zeta_2) \in \mathcal{N}(\varepsilon)$ and w_1 and w_2 be the corresponding solutions of* (3.1). *We then have*

$$w_2 - w_1 = (\nabla_w\Psi(w_1;0))^{-1} \begin{bmatrix} NN^T(l_1 - l_2) \\ r_1 - r_2 \\ \zeta_1 - \zeta_2 \end{bmatrix} + \rho, \tag{3.2}$$

where $\rho \in \mathbb{R}^{n+m+p}$ and $\|\rho\| \leq \kappa\left(\|l_1 - l_2\|_\diamond + \|r_1 - r_2\| + \|\zeta_1 - \zeta_2\|\right)^2$, i.e., $\rho = \mathcal{O}(\varepsilon^2)$.

Proof. To prove (i), we note that since $\nabla_w\Psi(w^*;0)$ is nonsingular, $\Psi(w^*;0) = 0$ and $\Psi(\cdot;0)$ is continuously differentiable in a neighborhood of w^*, the implicit function

theorem implies that there exists $\varepsilon > 0$ such that the implicitly defined function $w(l, r, \zeta)$ is continuously differentiable in (l, r, ζ) over the neighborhood $\mathcal{N}(\varepsilon)$.

For (ii), let $(l, r, \zeta) \in \mathcal{N}(\varepsilon)$ with $\zeta > 0$. Taking a smaller ε if necessary, note that $c_i(x(l, r, \zeta)) > 0$ for all $i \notin \mathcal{A}$ and that AS5 yields $[z(l, r, \zeta)]_i > 0$ for all $i \in \mathcal{A}$. On the other hand, since $[z(l, r, \zeta)]_i c_i(x(l, r, \zeta)) = \zeta_i$ for all $i = 1, \ldots, p$, we also obtain $[z(l, r, \zeta)]_i > 0$ for all $i \notin \mathcal{A}$ and $c_i(x(l, r, \zeta)) > 0$ for all $i \in \mathcal{A}$.

Finally, (iii) is shown noting that, by the implicit function theorem, the Jacobian of the function found in (i) is $-(\nabla_w \Psi(w; 0))^{-1}$, from which the result follows, using a first-order Taylor expansion. ∎

We may now apply Lemma 3.1 to obtain a fundamental result relating the optimal w^* to $w(\mu)$. We first observe that, because of Lemma 3.1 (i), when μ is sufficiently small, $w(0, 0, \mu e)$ is unique and is therefore equal to $w(\mu)$, since $w(\mu)$ solves this system by definition. Moreover, because of AS2, $\nabla_w \Psi(\cdot; 0)$ is uniformly nonsingular in a neighborhood of w^*, and the left-hand side of (3.2) is dominated by the first term in its right-hand side for small enough ε. Consequently,

$$\|w_1 - w_2\| = \Theta\left(\|l_1 - l_2\|_\diamond + \|r_1 - r_2\| + \|\zeta_1 - \zeta_2\|\right). \tag{3.3}$$

Substituting $(l_1, r_1, \zeta_1) = (0, 0, \mu e)$ and $(l_2, r_2, \zeta_2) = (0, 0, 0)$ in (3.3) and using the equivalence of $w(0, 0, \mu e)$ and $w(\mu)$, we obtain the important result

$$\|w(\mu) - w^*\| = \Theta(\mu) \tag{3.4}$$

for all sufficiently small μ.

We now show a property of the behavior of the path $x(\mu)$ when it approaches its limit x^*. We know from (2.2) and (2.9) that

$$\Psi(w(\mu); \mu) = \begin{bmatrix} \nabla_x \mathcal{L}(w(\mu)) \\ Ax(\mu) - b \\ C(x(\mu))z(\mu) - \mu e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \tag{3.5}$$

Differentiating system (3.5) with respect to μ and rearranging give

$$\nabla_w \Psi(w(\mu); \mu) \begin{bmatrix} \dot{x}(\mu) \\ \dot{y}(\mu) \\ \dot{z}(\mu) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ e \end{bmatrix} \tag{3.6}$$

for any nonnegative μ. At $\mu = 0$, taking only active constraints into account, the third equation of (3.6) together with (2.11) shows that $\dot{x}(0) \neq 0$ and

$$(\nabla_x c_i(x^*))^T \dot{x}(0) = \frac{1}{[z^*]_i} \quad \text{for all } i \in \mathcal{A}. \tag{3.7}$$

Equation (3.7) means that the trajectory $x(\mu)$ does not skirt the active constraints to reach x^*; that is, its approach is nontangential. Note that this is a consequence of the strict complementarity assumption AS5 (see [21] for details).

Remark 3.1

4. The algorithm. We now state our class of algorithms. Let us first define, for every strictly feasible (x, z),

$$B(x, z) \stackrel{\text{def}}{=} J^T(x) C^{-1}(x) Z J(x), \tag{4.1}$$

the Lagrangian

$$L(x,z) \stackrel{\rm def}{=} f(x) - c^T(x)z, \tag{4.2}$$

and finally

$$V(x,z) \stackrel{\rm def}{=} \nabla_{xx}L(x,z) + B(x,z). \tag{4.3}$$

Note that for any strictly feasible (x,z) and for any $y \in \mathbb{R}^m$

$$\nabla_{xx}L(x,z) = \nabla_{xx}\mathcal{L}(w) = \nabla_{xx}f(x) - \sum_{i=1}^{p}[z]_i\nabla_{xx}c_i(x). \tag{4.4}$$

Note also that $V(x,z)$ is the same as the Hessian of the barrier function (2.24) in the special case where the dual variables $z = \mu C^{-1}(x)e$.

To distinguish the overall algorithm from the inner minimization, that is the approximate solution of the barrier subproblem, we call the former the "outer minimization." Our outer minimization may be formally stated as Algorithm 4.1.

ALGORITHM 4.1 [OUTER MINIMIZATION]

Initialization. An initial barrier parameter $\mu_0 > 0$ and the forcing functions $\epsilon^{\rm C}(\mu)$ and $\epsilon^{\rm D}(\mu)$ are given. Set $k = 0$.

Inner minimization. Approximately minimize the log-barrier function $\phi(x,\mu_k)$. Stop this inner algorithm as soon as an inner iterate (x_{k+1}, z_{k+1}) is found such that

$$Ax_{k+1} = b, \tag{4.5}$$

$$(c(x_{k+1}), z_{k+1}) > 0, \tag{4.6}$$

$$\|C(x_{k+1})z_{k+1} - \mu_k e\| \leq \epsilon^{\rm C}(\mu_k), \quad \text{and} \tag{4.7}$$

$$\|\nabla_x f(x_{k+1}) - J^T(x_{k+1})z_{k+1}\|_{[k+1]} \leq \epsilon^{\rm D}(\mu_k), \tag{4.8}$$

where the norm $\|\cdot\|_{[k+1]}$ is defined with respect to some second-order sufficient preconditioning matrix M_{k+1}. Choose $\mu_{k+1} < \mu_k$, increment k by one, and perform next inner minimization.

A crucial feature of Algorithm 4.1 is that at every stage it generates iterates lying in the constraint manifold $Ax = b$. This allows us to concentrate on the natural curvature of the problem. It also has the important consequence that the Lagrange multipliers y neither appear nor are used anywhere in the algorithm. However, for the needs of the local analysis of section 6, let us define the Lagrange multipliers y_{k+1} by

$$\begin{aligned} A^T y_{k+1} = NN^T &\left[\nabla_x f(x_{k+1}) - J^T(x_{k+1})z_{k+1}\right] \\ &- \left(\nabla_x f(x_{k+1}) - J^T(x_{k+1})z_{k+1}\right) \end{aligned} \tag{4.9}$$

with x_{k+1} and z_{k+1} as given by Algorithm 4.1. Note that the system (4.9) uniquely determines y_{k+1} because the matrix A has full rank. It is easy to check that under our assumptions, as $x_{k+1} \to x^*$ and $z_{k+1} \to z^*$, we have $y_{k+1} \to y^*$. Moreover, the

definition (4.9) implies

$$\begin{aligned}
&\nabla_x f(x_{k+1}) + A^T y_{k+1} - J^T(x_{k+1}) z_{k+1} \\
&= NN^T \left[\nabla_x f(x_{k+1}) - J^T(x_{k+1}) z_{k+1}\right] \\
&= NN^T \left[\nabla_x f(x_{k+1}) + A^T y_{k+1} - J^T(x_{k+1}) z_{k+1}\right], \qquad (4.10)
\end{aligned}$$

or in other words, if we let $w_{k+1} = (x_{k+1}, y_{k+1}, z_{k+1})$, we have

$$\nabla_x \mathcal{L}(w_{k+1}) = NN^T \nabla_x \mathcal{L}(w_{k+1}). \qquad (4.11)$$

Note that conditions (4.7)–(4.8) are relaxations of a part of the optimality system (2.9), and that the stopping condition (4.8) is equivalent to

$$\|\nabla_x \mathcal{L}(w_{k+1})\|_{[k+1]} \leq \epsilon^{\mathrm{D}}(\mu_k),$$

using the identity (2.15).

We do not describe an inner minimization algorithm here (we will return to this question in section 7), but focus on the choice of the preconditioning matrices M_k. Since this preconditioning aims to locally represent the geometry of the log-barrier function, it is natural to assume that M_k is chosen as

$$M_k = W_k + B(x_k, z_k), \qquad (4.12)$$

where the matrix W_k is chosen so that M_k is second-order sufficient, and might, for example, be a suitable approximation of the natural choice $\nabla_{xx} L(x_k, z_k)$. A discussion of the possible practical choices for W_k is out of the scope of this work and we refer the interested reader to [2] for theoretical arguments and results of a practical implementation. Our assumption on M_k is the following.

AS6. There exist $\epsilon_M \in (0,1)$ and $\kappa_{\mathrm{w}} > 0$ such that, for all k, the preconditioner $M_k = W_k + B(x_k, z_k)$ and its component W_k satisfy

$$\lambda^{\min}[N^T M_k N] \geq \epsilon_M \qquad (4.13)$$

and

$$\|N^T W_k N\| \leq \kappa_{\mathrm{w}}. \qquad (4.14)$$

AS6 allows us to analyze the interrelationship of the preconditioners, in that we can deduce an important relation between the norm $\|\cdot\|_\diamond$ and the seminorms (2.13)–(2.14). It is worth emphasizing that this relation does not enforce uniform equivalence between those norms, as it only gives one of the two inequalities required for such an equivalence.

LEMMA 4.1. *Suppose* AS6 *is satisfied and that there exists a constant* $\kappa_J > 0$ *such that, for all* k,

$$\|J(x_k)\| \leq \kappa_J. \qquad (4.15)$$

Then, for any vector $v \in \mathbb{R}^n$ *and for all* k,

$$\|v\|_k \geq \epsilon_M^{1/2} \|v\|_\diamond \qquad (4.16)$$

and

$$\|v\|_{[k]} \leq \epsilon_M^{-1/2} \|v\|_\diamond. \qquad (4.17)$$

Moreover if there exists $\kappa^{(c)} > 0$ *such that*

$$\lim_{\mu \to 0} \frac{\epsilon^C(\mu)}{\mu} \leq \kappa^{(c)}, \tag{4.18}$$

then there exists $\kappa_\diamond > 0$ *such that for all* k

$$\|v\|_{[k]} \geq \kappa_\diamond \min\left[\frac{\min_i c_i(x_k)}{\sqrt{\mu_{k-1}}}, 1\right] \|v\|_\diamond. \tag{4.19}$$

Proof. The following proof is inspired by [2, Lemma 4.2 and Theorem 4.12].

Inequalities (4.16) and (4.17) clearly hold for any v orthogonal to $\mathcal{N}(A)$. Using the identity (2.16), the positive definiteness of $N^T M_k N$ and AS6, we have for all v such that $N^T v \neq 0$,

$$\begin{aligned} \frac{\|v\|_\diamond^2}{\|v\|_k^2} = \frac{\|N^T v\|^2}{\|N^T v\|^2_{R[M_k]}} &= \frac{\|(N^T M_k N)^{-\frac{1}{2}} (N^T M_k N)^{\frac{1}{2}} N^T v\|^2}{\|(N^T M_k N)^{\frac{1}{2}} N^T v\|^2} \\ &\leq \|(N^T M_k N)^{-1}\| \\ &\leq \epsilon_M^{-1}, \end{aligned}$$

which proves (4.16). The proof of (4.17) is similar.

To prove (4.19), first observe that AS6, (2.12), and (4.15) imply that for all k

$$\|N^T M_k N\| \leq \|N^T W_k N\| + \|N^T B(x_k, z_k) N\| \leq \kappa_{\rm w} + \kappa_J^2 \max_i \frac{[z_k]_i}{c_i(x_k)}. \tag{4.20}$$

If r denotes the vector $N^T v$, we have $\|r\| = \|v\|_\diamond$. Consider first the case where there exists $\kappa_\infty > 0$ such that for all $i = 1, \ldots, p$,

$$\limsup_{k \to \infty} \frac{[z_k]_i}{c_i(x_k)} \leq \kappa_\infty < +\infty.$$

Inequality (4.20) then becomes

$$\|N^T M_k N\| \leq \kappa_{\rm w} + \kappa_J^2 \kappa_\infty,$$

and we have from the positive definiteness of $N^T M_k N$

$$\|v\|_{[k]} = \|(N^T M_k N)^{-\frac{1}{2}} r\| \geq (\kappa_{\rm w} + \kappa_J^2 \kappa_\infty)^{-1/2} \|r\|. \tag{4.21}$$

Now consider the other possibility, namely that

$$\limsup_{k \to \infty} \frac{[z_k]_{i_0}}{c_{i_0}(x_k)} = +\infty$$

for some index i_0. Then, using (4.7) and (4.18), we have for all i and for sufficiently large k

$$\frac{[z_k]_i}{c_i(x_k)} \leq \frac{\mu_{k-1}}{c_i(x_k)^2} + \frac{|c_i(x_k)[z_k]_i - \mu_{k-1}|}{c_i(x_k)^2} \leq \frac{\mu_{k-1}}{c_i(x_k)^2} + \frac{\epsilon^{\rm C}(\mu_{k-1})}{c_i(x_k)^2} \leq (1 + 2\kappa^{(c)}) \frac{\mu_{k-1}}{c_i(x_k)^2}.$$

We thus obtain from (4.20) that for large enough k

$$\|N^T M_k N\| \leq \kappa_{\rm w} + \kappa_J^2 \max_i \frac{[z_k]_i}{c_i(x_k)} \leq 2\kappa_J^2 (1 + 2\kappa^{(c)}) \frac{\mu_{k-1}}{\min_i c_i(x_k)^2},$$

so that

$$\|v\|_{[k]} = \|(N^T M_k N)^{-\frac{1}{2}} r\| \geq (2\kappa_J^2(1+2\kappa^{(c)}))^{-\frac{1}{2}} \frac{\min_i c_i(x_k)}{\sqrt{\mu_{k-1}}} \|r\|. \tag{4.22}$$

Putting (4.21) and (4.22) together yields (4.19) with

$$\kappa_\diamond \stackrel{\text{def}}{=} \min\left[(2\kappa_J^2(1+2\kappa^{(c)}))^{-\frac{1}{2}}, (\kappa_W + \kappa_J^2\kappa_\infty)^{-1/2}\right]. \qquad \square$$

5. Choosing the starting point for the inner minimization. As stated in the introduction, a computationally critical part of the algorithm is the choice of the starting point for the inner minimization and we already indicated that choosing x_{k+1} to start the solution of $BS(\mu_{k+1})$ is likely to be inefficient. The purpose of this section is to examine alternative choices from the point of view of improving the local convergence rate. Of course, this rate of convergence depends on the particular choice of the functions $\epsilon^C(\mu)$ and $\epsilon^D(\mu)$. We therefore start by considering appropriate choices for these functions.

5.1. Stopping tolerances. Formally, we shall suppose that the inner iteration, which is the approximate minimization of $\mathrm{BS}(\mu_k)$, starts from the (as yet, undefined) primal-dual point $(x_{k,0}, z_{k,0})$, generates a sequence of iterates $\{(x_{k,j}, z_{k,j})\}_{j\geq 0}$, and terminates at the point $(x_{k,j_k}, z_{k,j_k}) \equiv (x_{k+1}, z_{k+1})$ at which (4.5)–(4.8) are satisfied for some appropriate second-order sufficient matrix M_{k+1}. Let us define the Lagrange multipliers y_{k+1} according to (4.9) and let $w_{k+1} = (x_{k+1}, y_{k+1}, z_{k+1})$. We assume from now on that the tolerances $\epsilon^C(\mu_k)$ and $\epsilon^D(\mu_k)$ asymptotically have the particular form given in Figure 5.1. Observe that (4.7) and (5.1) imply that $c_i(x_{k+1})[z_{k+1}]_i - \mu_k = \mathcal{O}(\mu_k)$ for all $i = 1, \ldots, p$.

Stopping tolerances:
We assume that there exist constants $0 < \underline{\kappa}_\mu^C \leq \overline{\kappa}_\mu^C < 1$ and $0 < \underline{\kappa}_\mu^D \leq \overline{\kappa}_\mu^D$ such that, for sufficiently large values of k,

$$\underline{\kappa}_\mu^C \mu_k \leq \epsilon^C(\mu_k) \leq \overline{\kappa}_\mu^C \mu_k \tag{5.1}$$

and

$$\underline{\kappa}_\mu^D \mu_k^{\gamma_k+1} \leq \epsilon^D(\mu_k) \leq \overline{\kappa}_\mu^D \mu_k^{\gamma_k+1}, \tag{5.2}$$

where

$$0 < \gamma_k < 1. \tag{5.3}$$

FIG. 5.1. *Stopping tolerances for Algorithm* 4.1.

In the context of our local analysis, we now assume that the vector w^*, a solution of (2.3)–(2.4), is a limit point of the sequence $\{w_{k+1}\}$, satisfying (4.5)–(4.8), with y_{k+1} defined by (4.9), as $\mu_k \searrow 0$. More specifically, we assume that there exists an infinite index set $\mathcal{K}$ such that $w_{k+1} \to w^*$ as $k \to \infty$, $k \in \mathcal{K}$. The subsequence of $\{w_{k+1}\}$ indexed by $\mathcal{K}$ is denoted by $\{w_{k+1}\}_\mathcal{K}$ and we write $\{w_{k+1}\}_\mathcal{K} \to w^*$. In what

follows, we consider only $k \in \mathcal{K}$. In addition, AS2 implies that, for all $i = 1, \ldots, p$ and all x sufficiently close to x^*

$$\|\nabla_x f(x)\| \leq \kappa_{\rm g}, \|\nabla_{xx} f(x)\| \leq \kappa_{\rm H}, \|\nabla_x c_i(x)\| \leq \kappa_\gamma \text{ and } \|\nabla_{xx} c_i(x)\| \leq \kappa_\Gamma \tag{5.4}$$

for some $\kappa_{\rm g}, \kappa_{\rm H}, \kappa_\gamma, \kappa_\Gamma > 0$. The third of these bounds and the fact that we restrict our attention to $\mathcal{K}$ imply that (4.15) holds for k sufficiently large, and Lemma 4.1 can thus be applied asymptotically within $\mathcal{K}$.

The next theorem provides bounds on the active and inactive quantities involved in Algorithm 4.1 that result from our choice of stopping tolerances.

THEOREM 5.1. *Assume w^* is a solution of NLP and $\{w_{k+1}\}_{\mathcal{K}} \to w^*$, where $\{w_{k+1}\}$ is a sequence of iterates generated by Algorithm* 4.1 *with y_{k+1} defined by* (4.9). *Under* AS1–AS6 *and* (5.1), *we have that for sufficiently large $k \in \mathcal{K}$,*

(i) *for all $i \in \mathcal{A}$, there exist $\kappa^z \geq \kappa_z > 0$ such that*

$$\frac{1}{\kappa^z}(1 - \overline{\kappa}_\mu^C)\mu_k \leq c_i(x_{k+1}) \leq \frac{1}{\kappa_z}(1 + \overline{\kappa}_\mu^C)\mu_k, \tag{5.5}$$

$$\kappa_z \leq [z_{k+1}]_i \leq \kappa^z; \tag{5.6}$$

(ii) *for all $i \notin \mathcal{A}$, there exist $\kappa^c \geq \kappa_c > 0$ such that*

$$\frac{1}{\kappa^c}(1 - \overline{\kappa}_\mu^C)\mu_k \leq [z_{k+1}]_i \leq \frac{1}{\kappa_c}(1 + \overline{\kappa}_\mu^C)\mu_k, \tag{5.7}$$

$$\kappa_c \leq c_i(x_{k+1}) \leq \kappa^c, \tag{5.8}$$

where $\overline{\kappa}_\mu^C$ is defined in (5.1).

Proof. The strict complementarity assumption AS5 implies that for $i \in \mathcal{A}$, we have for sufficiently large $k \in \mathcal{K}$

$$0 < \tfrac{1}{2}[z^*]_i \leq [z_{k+1}]_i \leq 2[z^*]_i. \tag{5.9}$$

Relation (5.6) then follows with $\kappa_z = \frac{1}{2}\min_{i\in\mathcal{A}}[z^*]_i$ and $\kappa^z = 2\max_{i\in\mathcal{A}}[z^*]_i$.

On the other hand, stopping condition (4.7) yields that for all $i = 1, \ldots, p$,

$$-\epsilon^{\rm C}(\mu_k) \leq c_i(x_{k+1})[z_{k+1}]_i - \mu_k \leq \epsilon^{\rm C}(\mu_k). \tag{5.10}$$

Using the rightmost inequality in (5.1), (5.6), and (5.10) yields (5.5).

Observe that if $i \notin \mathcal{A}$, then for sufficiently large $k \in \mathcal{K}$

$$0 < \tfrac{1}{2}c_i(x^*) \leq c_i(x_{k+1}) \leq 2c_i(x^*). \tag{5.11}$$

Relation (5.8) then follows with $\kappa_c = \frac{1}{2}\min_{i\notin\mathcal{A}} c_i(x^*)$ and $\kappa^z = 2\max_{i\notin\mathcal{A}} c_i(x^*)$. The proof of (5.7) is similar to that of (5.5) using (5.8) and (5.10). ∎

We now show that a termination criterion such as (4.7)–(4.8) coupled with (5.1) and (5.2) guarantees that w_{k+1} lies within a constant factor of $(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k)$ from an exact solution $w(\mu_k)$ of BS(μ_k) and from w^* in the usual, Euclidean norm.

THEOREM 5.2. *Suppose that* AS1–AS6, (5.1), *and* (5.2) *are satisfied, with γ_k as specified in* (5.3). *Assume furthermore that w^* is a solution of NLP and that $\{w_{k+1}\}_{\mathcal{K}} \to w^*$, where $\{w_{k+1}\}$ is a sequence of iterates generated by Algorithm* 4.1

with y_{k+1} defined by (4.9). Then, we have that, for sufficiently large $k \in \mathcal{K}$, there exist constants $\kappa_{dst}, \kappa^*_{dst} > 0$ such that

$$\|w_{k+1} - w(\mu_k)\| \leq \kappa_{dst} \, (\mu_k^{\gamma_k + \frac{1}{2}} + \mu_k) \tag{5.12}$$

and

$$\|w_{k+1} - w^*\| \leq \kappa^*_{dst} \, (\mu_k^{\gamma_k + \frac{1}{2}} + \mu_k). \tag{5.13}$$

Proof. Observe that, under the stated assumptions, we may apply (4.19) which, together with the relations (2.20), (4.8), and (5.2), yields

$$\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond \leq \kappa_\diamond^{-1} \max \left[\frac{\epsilon^{\mathrm{D}}(\mu_k)\sqrt{\mu_k}}{\min_i c_i(x_{k+1})}, \epsilon^{\mathrm{D}}(\mu_k) \right] \tag{5.14}$$

$$\leq \kappa_\diamond^{-1} \overline{\kappa}_\mu^{\mathrm{D}} \max \left[\frac{\mu_k^{\gamma_k + 3/2}}{\min_i c_i(x_{k+1})}, \mu_k^{\gamma_k + 1} \right]. \tag{5.15}$$

First consider the case where the active set $\mathcal{A}$ is nonempty. In view of (5.5) and (5.8), the index i that realizes the minimum in (5.15) certainly asymptotically satisfies (5.5), which implies

$$\min_i c_i(x_{k+1}) = \min_{i \in \mathcal{A}} c_i(x_{k+1}) \geq \frac{1}{\kappa^z}(1 - \overline{\kappa}_\mu^{\mathrm{C}})\mu_k. \tag{5.16}$$

Combining (5.15) and (5.16), we obtain

$$\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond \leq \kappa_{\mathcal{L}} \max \left[\frac{\kappa^z}{1 - \overline{\kappa}_\mu^{\mathrm{C}}} \mu_k^{\gamma_k + \frac{1}{2}}, \mu_k^{\gamma_k + 1} \right] = \kappa_{\mathcal{L}} \frac{\kappa^z}{1 - \overline{\kappa}_\mu^{\mathrm{C}}} \mu_k^{\gamma_k + \frac{1}{2}} \tag{5.17}$$

for sufficiently large $k \in \mathcal{K}$, where we have set $\kappa_{\mathcal{L}} = \kappa_\diamond^{-1} \overline{\kappa}_\mu^{\mathrm{D}}$.

Consider now the case where there are no active constraints. This time, the index i that realizes the minimum in (5.15) satisfies (5.8) and we have $\min_i c_i(x_{k+1}) \geq \kappa_c$. Thus, (5.15) gives that for sufficiently large $k \in \mathcal{K}$

$$\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond \leq \kappa_\diamond^{-1} \overline{\kappa}_\mu^{\mathrm{D}} \max \left[\kappa_c^{-1} \mu_k^{\gamma_k + 3/2}, \mu_k^{\gamma_k + 1} \right] = \kappa_{\mathcal{L}} \mu_k^{\gamma_k + 1}. \tag{5.18}$$

From the definition (4.9) of y_{k+1}, (4.11) together with (4.5) guarantees that we have $w_{k+1} = w(\nabla_x \mathcal{L}(w_{k+1}), 0, C(x_{k+1})z_{k+1})$ in Lemma 3.1. Moreover, (4.7), (5.1), (5.17), and (5.18) guarantee that $\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond + \|C(x_{k+1})z_{k+1}\|$ is smaller than the threshold ε defined in Lemma 3.1 for sufficiently large $k \in \mathcal{K}$. Invoking (3.3) with the parameters $(\nabla_x \mathcal{L}(w_{k+1}), 0, C(x_{k+1})z_{k+1})$ and $(0, 0, \mu_k e)$ thus gives

$$\|w_{k+1} - w(\mu_k)\| = \Theta(\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond + \|C(x_{k+1})z_{k+1} - \mu_k e\|) \tag{5.19}$$

$$= \mathcal{O}(\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond + \epsilon^{\mathrm{C}}(\mu_k)) \tag{5.20}$$

$$= \mathcal{O}(\|\nabla_x \mathcal{L}(w_{k+1})\|_\diamond + \mu_k), \tag{5.21}$$

where we have used (4.7) and (5.1).

When $\mathcal{A}$ is nonempty, we obtain from (5.17) and (5.21) that, for sufficiently large $k \in \mathcal{K}$

$$\|w_{k+1} - w(\mu_k)\| = \mathcal{O}(\mu_k^{\gamma_k + \frac{1}{2}} + \mu_k), \tag{5.22}$$

while if $\mathcal{A}$ is empty, (5.18) and (5.21) yield

$$\|w_{k+1} - w(\mu_k)\| = \mathcal{O}(\mu_k^{\gamma_k+1} + \mu_k) \tag{5.23}$$

$$= \mathcal{O}(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k) \tag{5.24}$$

because of (5.3). Putting (5.22) and (5.24) together proves (5.12).

Using the triangle inequality, (3.4), (5.3), and (5.12), there exists a constant $\kappa^* > 0$ such that

$$\|w_{k+1} - w^*\| \leq \|w_{k+1} - w(\mu_k)\| + \|w(\mu_k) - w^*\| \tag{5.25}$$

$$\leq \kappa_{\rm dst}(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k) + \kappa^*\mu_k, \tag{5.26}$$

$$\leq (\kappa_{\rm dst} + \kappa^*)(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k), \tag{5.27}$$

which proves (5.13) with $\kappa^*_{\rm dst} = \kappa_{\rm dst} + \kappa^*$. ∎

Remark 5.1. In condition (5.14), the minimum is certainly asymptotically attained for an active index, if any. In that case, if AS5 is satisfied, this minimum is of the order of μ_k (and of the order of $\mu_k^{1/2}$ if AS5 fails to be satisfied)—see, for instance, [11, 21]. In our case, in order for the sequence $\{\|\nabla_x\mathcal{L}(w_{k+1})\|_\diamond\}$ to converge to zero, we thus require that $\epsilon^{\rm D}(\mu_k)$ converges to zero faster than $\mu_k^{1/2}$ (as guaranteed by (5.2)), which is usually sufficient in practice to ensure convergence of the outer minimization.

Examining (4.19), we see that in the nondegenerate case, and when there are active constraints, we asymptotically have $\|v\|_{[k]} \geq \kappa^{\rm nd}_\diamond\sqrt{\mu_k}\|v\|_\diamond$ for some constant $\kappa^{\rm nd}_\diamond > 0$. Hence, conditions (4.8) and (5.2) amount to $\|\nabla_x\mathcal{L}(w_{k+1})\|_\diamond = \mathcal{O}(\mu_k^{\gamma_k+\frac{1}{2}})$ which may be weaker than the usual stopping criterion $\|\nabla_x\mathcal{L}(w_{k+1})\|_\diamond = \mathcal{O}(\mu_k)$ whenever $\gamma_k < \frac{1}{2}$. In this case, the right-hand sides of (5.12) and (5.13) are $\mathcal{O}(\mu_k^{\gamma_k+\frac{1}{2}})$, which is weaker than the usual bound $\mathcal{O}(\mu_k)$. If $\gamma_k > \frac{1}{2}$, the stopping criterion is tightened but the right-hand sides of (5.12) and (5.13) are required to take the traditional form $\mathcal{O}(\mu_k)$. Finally, if $\gamma_k = \frac{1}{2}$, the stopping criterion (4.8) and the bounds (5.12) and (5.13) coincide with the traditional ones.

In the degenerate case, (4.19) becomes $\|v\|_{[k]} \geq \kappa^{\rm d}_\diamond\|v\|_\diamond$ for some constant $\kappa^{\rm d}_\diamond > 0$, and shows, together with (4.17), that the norms $\|\cdot\|_{[k]}$ and $\|\cdot\|_\diamond$ are equivalent in the nullspace of A. Using (5.2), condition (4.8) then amounts to $\|\nabla_x\mathcal{L}(w_{k+1})\|_\diamond = \mathcal{O}(\mu_k^{\gamma_k+1})$ and shows that we are more restrictive in this case for any $\gamma_k > 0$.

5.2. Simple choices for the starting point. In the unpreconditioned primal case, it has previously been suggested [3] that the tolerance $\epsilon^{\rm D}(\mu)$ be set to $\mathcal{O}(\mu)$. In that case, inequality (4.8) and the boundedness of μ_{k+1} imply that

$$\|\nabla_x\mathcal{L}(w^{\rm P}_{k+1})\| = \mathcal{O}(\mu_k) = \mathcal{O}\left(\frac{\mu_k}{\mu_{k+1}}\right) \tag{5.28}$$

for the primal choice $w^{\rm P}_{k+1} = (x_{k+1}, y_{k+1}, \mu_{k+1}C^{-1}(x_{k+1})e)$. This observation parallels [5, Lemma 1] and reinforces the results in [17] by suggesting that if the parameter μ_k is reduced too fast (i.e., $\mu_{k+1} \ll \mu_k$), it is unlikely that $w^{\rm P}_{k+1}$ will be an accurate estimate of the solution $w(\mu_{k+1})$ for the forthcoming outer iteration.

By contrast, [3] suggests that, letting $x^{\rm O}_{k+1} = x_{k+1}$ and $z^{\rm O}_{k+1} = \mu_k C^{-1}(x_{k+1})e$ (as opposed to the value $\mu_{k+1}C^{-1}(x_{k+1})e$ which would have been used in a purely primal context without extrapolation), a good initial point for BS(μ_{k+1}) might be $w_{k+1,0} = w^{\rm O}_{k+1} + d^{\rm N}_{k+1} = (x_{k+1,0}, y_{k+1,0}, z_{k+1,0})$, where $d^{\rm N}_{k+1}$ is the full Newton step

taken from $w^{\mathrm{O}}_{k+1} = (x^{\mathrm{O}}_{k+1}, y^{\mathrm{O}}_{k+1}, z^{\mathrm{O}}_{k+1})$ (for some y^{O}_{k+1}). Simply restating a few vital steps from [3], it can be shown that d^{N}_{k+1} is asymptotically feasible and

$$\|\nabla_x \mathcal{L}(w_{k+1,0})\| = \mathcal{O}(\mu_k^2),$$
$$\|C(x_{k+1,0})z_{k+1,0} - \mu_{k+1}e\| = \mathcal{O}(\mu_k^2),$$

which would be accepted by any primal-dual stopping rule for which $\epsilon^{\mathrm{C}}(\mu) = \epsilon^{\mathrm{D}}(\mu) = \mathcal{O}(\mu)$ provided that $\mu_{k+1} = \Omega(\mu_k^2)$. Continuing in this vein, we would subsequently have

$$z_{k+1,0} = \mu_{k+1}C^{-1}(x_{k+1,0})e + \mathcal{O}\left(\frac{\mu_k^2}{\mu_{k+1}}\right),$$

which then provides the bound

$$\left\|\nabla_x f(x_{k+1,0}) - \mu_{k+1}J^T(x_{k+1,0})C^{-1}(x_{k+1,0})e\right\| = \mathcal{O}\left(\frac{\mu_k^2}{\mu_{k+1}}\right).$$

Hence, if the inner minimization corresponding to μ_{k+1} is started with $x_{k+1,0} = x^{\mathrm{O}}_{k+1} + [d^{\mathrm{N}}_{k+1}]_x$ and $z_{k+1,0} = z^{\mathrm{O}}_{k+1} + [d^{\mathrm{N}}_{k+1}]_z$, and assuming a subsequent Newton step is acceptable to the inner minimization method (this can be shown to be the case), the size of $\nabla_x \mathcal{L}(w)$ at the resulting iterate will be $\mathcal{O}(\mu_k^4/\mu_{k+1}^2)$. Consequently, if we wish the resulting iterate to satisfy a primal stopping rule of the form $\epsilon^{\mathrm{D}}(\mu) = \mathcal{O}(\mu)$, this requires that

$$\mu_{k+1} = \Omega(\mu_k^{4/3}), \tag{5.29}$$

which suggests a superlinear rate of convergence in μ is possible.

5.3. An alternative choice based on extrapolating the central path. We now intend to parallel the approach of section 5.2 in the primal-dual case, with the hope of improving the bound (5.28). Assume we update the barrier parameter from μ_k to μ_{k+1}. In order to solve problem BS(μ_{k+1}) efficiently, it is natural to aim to choose a starting point which is as close as possible to a stationary point $w(\mu_{k+1})$ of this problem. We thus wish to (approximately) solve the system

$$\Psi(w; \mu_{k+1}) = 0. \tag{5.30}$$

An attractive possibility is therefore to choose the starting point for the inner minimization as the result of a single Newton iteration for this system. This point, which we denote w^{PD}_{k+1}, is obtained from the solution of the linearized version of (5.30), that is,

$$\nabla_w \Psi(w_{k+1}; \mu_{k+1})(w^{\mathrm{PD}}_{k+1} - w_{k+1}) = -\begin{bmatrix} \nabla_x \mathcal{L}(w_{k+1}) \\ 0 \\ C(x_{k+1})z_{k+1} - \mu_{k+1}e \end{bmatrix}, \tag{5.31}$$

where $\nabla_w \Psi(w; \mu)$ is given by (2.11). If we let

$$d_{k+1} = ([d_{k+1}]_x, [d_{k+1}]_y, [d_{k+1}]_z) = w^{\mathrm{PD}}_{k+1} - w_{k+1}, \tag{5.32}$$

we may eliminate $[d_{k+1}]_z$, use the first identity of (4.4) together with (4.3), the fact that $\nabla_x \mathcal{L}(w_{k+1}) = \nabla_x f(x_{k+1}) + A^T y_{k+1} - J^T(x_{k+1})z_{k+1}$, rearrange and obtain the *reduced* system

$$\begin{bmatrix} V(x_{k+1}, z_{k+1}) & A^T \\ A & 0 \end{bmatrix} \begin{bmatrix} [d_{k+1}]_x \\ [w^{\mathrm{PD}}_{k+1}]_y \end{bmatrix} = -\begin{bmatrix} \nabla_x \phi(x_{k+1}, \mu_{k+1}) \\ 0 \end{bmatrix} \tag{5.33}$$

from which we may recover

$$[d_{k+1}]_z = -z_{k+1} + \mu_{k+1} C^{-1}(x_{k+1})e - C^{-1}(x_{k+1}) Z_{k+1} J(x_{k+1})[d_{k+1}]_x.$$

Note that the right-hand side of (5.33) is independent of z_{k+1}. This system entirely determines $[d_{k+1}]_x$ and $[w^{\rm PD}_{k+1}]_y$ provided that the matrix $V(x_{k+1}, z_{k+1})$ is positive definite on the nullspace of A and A has full rank. This is equivalent to requiring that the matrix $V(x_{k+1}, z_{k+1})$ is second-order sufficient (see section 2.1.2).

Before studying the implications of this step in terms of local convergence rate, we first give an alternative geometrical interpretation of the step d_{k+1}, in the spirit of [5], arising from the implicit function theorem.

From the point w_{k+1} and with the barrier parameter μ_k, assume we wish to find an estimate $w^{\rm EX}_{k+1}$ to an exact solution w_+ of

$$\Psi(w; \mu_{k+1}) = \Psi(w_{k+1}; \mu_k), \tag{5.34}$$

since we know from (4.7) and (4.8) that the right-hand side of (5.34) is "small." Notice that if we compute w_+ as the solution of (5.34), the equality constraints remain satisfied.

Let us consider the system

$$\Psi(w; \mu) = \Psi(w_{k+1}; \mu_k) \tag{5.35}$$

and think of its solution in the variable w as a function of the barrier parameter μ, say $w = \varphi(\mu)$, where the (implicit) function $\varphi(\mu)$ is defined in a small neighborhood of μ_k, and where $w_{k+1} = \varphi(\mu_k)$. As already mentioned in section 3, for small enough values of μ_k, the Jacobian $\nabla_w \Psi(w_{k+1}; \mu_k) = \nabla_w \Psi(w_{k+1}; 0)$ is nonsingular and therefore the implicit function theorem yields that, in a small vicinity of μ_k, the function φ is well-defined and differentiable. In particular, if we let $w'_{k+1} = \varphi'(\mu_k)$, we have

$$w'_{k+1} = -\left(\nabla_w \Psi(w_{k+1}; \mu_k)\right)^{-1} \nabla_\mu \Psi(w_{k+1}; \mu_k) = \left(\nabla_w \Psi(w_{k+1}; \mu_k)\right)^{-1} \begin{bmatrix} 0 \\ 0 \\ e \end{bmatrix}. \tag{5.36}$$

The estimate $w^{\rm EX}_{k+1}$ is computed as the following first-order Taylor expansion of $\varphi(\mu)$ about μ_k, which represents an extrapolation from the parameters $(\mu_k, \Psi(w_{k+1}; \mu_k))$ to $(\mu_{k+1}, \Psi(w_{k+1}; \mu_k))$ and defines the step:

$$w^{\rm EX}_{k+1} = \varphi(\mu_k) + \varphi'(\mu_k)(\mu_{k+1} - \mu_k) = w_{k+1} + w'_{k+1}(\mu_{k+1} - \mu_k). \tag{5.37}$$

Assuming all the functions of interest are three times continuously differentiable, so that φ is twice continuously differentiable, we have by Taylor's theorem that the point $w^{\rm EX}_{k+1}$ is within $\mathcal{O}((\mu_{k+1} - \mu_k)^2)$ of $w_+ = \varphi(\mu_{k+1})$; see [5].

Since the matrix $\nabla_w \Psi(w_{k+1}; \mu_k)$ used in (5.36) is *the same* as the one needed to compute a Newton step from w_{k+1} (see (2.11)), it is now possible to take, from $w^{\rm EX}_{k+1}$, the Newton step we would have taken had we stayed at w_{k+1}, which then defines the step

$$d^{\rm NW}_{k+1} = -\left(\nabla_w \Psi(w_{k+1}; \mu_k)\right)^{-1} \Psi(w_{k+1}; \mu_k),$$

and thereby define our composite extrapolation step

$$w^{\rm PD}_{k+1} \stackrel{\rm def}{=} w^{\rm EX}_{k+1} + d^{\rm NW}_{k+1}$$

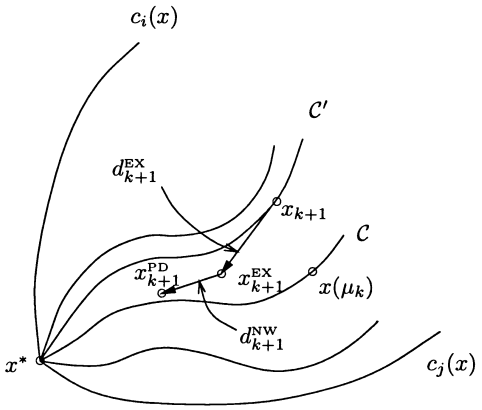


FIG. 5.2. *The above picture simplifies the situation as the x-space only is considered. The infeasible region is the "outer" part of the picture. Constraints i and j are active at the local constrained minimizer x^*. The exact solution of the current barrier subproblem, $x(\mu_k)$, lies exactly on the primal-dual central path $\mathcal{C}$. The second path $\mathcal{C}'$ that is pictured is defined implicitly by $\Psi(w;\mu) = \Psi(w_{k+1};\mu_k)$ as in* (5.35). *The two outermost paths represent the neighborhood of $\mathcal{C}$ defined by the forcing functions $\epsilon^C(\mu)$ and $\epsilon^D(\mu)$. The extrapolated step d^{EX}_{k+1} taken from w_{k+1} leaves the path $\mathcal{C}'$ tangentially and leads to w^{EX}_{k+1}. To that step is added the Newton step d^{NW}_{k+1} that would have been taken from w_{k+1}, as represented on the picture. The step $d^{EX}_{k+1} + d^{NW}_{k+1}$ is the step leading to w^{PD}_{k+1} as given by* (5.31).

$$
\begin{aligned}
&= w_{k+1} + \left(\nabla_w \Psi(w_{k+1};\mu_k)\right)^{-1} \left(\begin{bmatrix} 0 \\ 0 \\ e \end{bmatrix} (\mu_{k+1} - \mu_k) - \Psi(w_{k+1};\mu_k) \right) \\
&= w_{k+1} - \left(\nabla_w \Psi(w_{k+1};\mu_k)\right)^{-1} \begin{bmatrix} \nabla_x \mathcal{L}(w_{k+1}) \\ 0 \\ C(x_{k+1}) z_{k+1} - \mu_{k+1} e \end{bmatrix}, \qquad (5.38)
\end{aligned}
$$

where we have used (2.2), (5.36), (5.37), and the definition of d^{NW}_{k+1}. The step (5.38) amounts to an extrapolation from the parameters $(\mu_k, \Psi(w_{k+1};\mu_k))$ to $(\mu_{k+1}, 0)$, in the spirit of the predictor-corrector approach used in linear programming. Notice that, since the Jacobian matrix $\nabla_w \Psi(w;\mu)$ is independent of μ, the steps (5.31) and (5.38) are identical. This is not true in the purely primal case, as the Jacobian matrix $\nabla_w \Psi(w;\mu)$ is no longer independent of μ. An illustration of the decomposition of the step (5.38) appears in Figure 5.2.

As $\mu \searrow 0$, the trajectory given by (5.35) and represented by $\varphi(\mu)$ obviously gets closer and closer to the primal-dual central path $\mathcal{C}$ represented by $w(\mu)$, until both coincide at w^*. Moreover, in this case, its derivative

$$
\varphi'(\mu) = -\left(\nabla_w \Psi(w_{k+1};\mu)\right)^{-1} \nabla_\mu \Psi(w_{k+1};\mu)
$$

converges to

$$
-\left(\nabla_w \Psi(w^*;0)\right)^{-1} \begin{bmatrix} 0 \\ 0 \\ -e \end{bmatrix}
$$

because of (2.11), which, in view of (3.6), equals $\dot{w}(0)$. The two paths thus coincide up to first order at w^*. This intuitively guarantees that expanding the function φ asymptotically gives an accurate approximation of the primal-dual central path.

6. Local convergence analysis. Having proposed the point $w_{k+1}^{\rm PD}$ as a possible starting point for the inner minimization, we now wish to analyze its properties from the point of view of improving the local convergence rate of the algorithm. We will prove that this particular choice is not only strictly feasible—in contrast with the purely primal case [17]—but also dramatically improves on the bound (5.28). In fact, we shall show that this point asymptotically satisfies the stopping criterion (4.5)–(4.8), which means that the inner minimization algorithm is ultimately not needed. In particular, this means that, asymptotically, only one linear system (5.31) need be solved per update of the barrier parameter μ.

We start by verifying that this system is asymptotically well posed, at least along the converging subsequence.

LEMMA 6.1. *Under the assumptions of Theorem* 5.1*, there exists a closed and bounded neighborhood* $\mathcal{V}$ *of* w^* *such that the matrix* $V(x,z)$ *defined by* (4.3) *is positive definite over the nullspace of* A *for all* $w \in \mathcal{V}$*.*

Proof. The result follows from the application of [16, Theorem 8(iii)] to the reduced Hessian matrix $N^T V(x,z) N$. ∎

This result indicates that the systems (5.31) and (5.33) are asymptotically well posed for $k \in \mathcal{K}$ sufficiently large.

In the next major stage of our analysis, we verify that the stopping conditions for the inner minimization are all satisfied at $w_{k+1}^{\rm PD}$, provided we impose further conditions on the barrier parameter updating rule. Furthermore, we also show that the bound (5.28) can be improved in this context.

THEOREM 6.2. *Under* AS1–AS6*, assume that* w^* *is a solution of NLP, that the sequence* $\{w_{k+1}\}_{\mathcal{K}} \to w^*$*, where* $\{w_{k+1}\}$ *is a sequence of iterates generated by Algorithm* 4.1 *with* y_{k+1} *defined by* (4.9)*, and that the functions* f *and* c_i $(i = 1, \ldots, p)$ *are three times continuously differentiable over an open neighborhood of* x^**. Assume furthermore that* (5.1)–(5.3) *are satisfied, that* $0 < \epsilon_\tau < 1/2$ *is a given constant, that* $0 < \gamma_k \leq (1 - 2\epsilon_\tau)/(1 + 2\epsilon_\tau)$*, and that the barrier parameter updating rule satisfies*

$$\mu_{k+1} = \Omega(\mu_k^{\tau_k}) \qquad 1 + \epsilon_\tau \leq \tau_k \leq \frac{2}{1+\gamma_{k+1}} - \epsilon_\tau. \tag{6.1}$$

Then, we have that, for $k \in \mathcal{K}$ *sufficiently large,*

$$Ax_{k+1}^{PD} = b, \tag{6.2}$$

$$(c(x_{k+1}^{PD}), z_{k+1}^{PD}) > 0, \tag{6.3}$$

$$\|C(x_{k+1}^{PD}) z_{k+1}^{PD} - \mu_{k+1} e\| \leq \epsilon^C(\mu_{k+1}), \tag{6.4}$$

$$\|\nabla_x f(x_{k+1}^{PD}) - J^T(x_{k+1}^{PD}) z_{k+1}^{PD}\|_{[k+2]} \leq \epsilon^D(\mu_{k+1}), \tag{6.5}$$

and

$$\|\Psi(w_{k+1}^{PD}; \mu_{k+1})\| = o(\mu_{k+1}). \tag{6.6}$$

Proof. Observe first that (5.3) and (6.1) imply that

$$\mu_k^2 = o(\mu_{k+1}). \tag{6.7}$$

We start by proving (6.2). From (5.31), the direction d_{k+1} satisfies the equations

$$\nabla_{xx}L(x_{k+1},z_{k+1})[d_{k+1}]_x + A^T[d_{k+1}]_y - J^T(x_{k+1})[d_{k+1}]_z = -\nabla_x\mathcal{L}(w_{k+1}), \tag{6.8}$$

$$A[d_{k+1}]_x = 0, \tag{6.9}$$

and

$$Z_{k+1}J(x_{k+1})[d_{k+1}]_x + C(x_{k+1})[d_{k+1}]_z = \mu_{k+1}e - C(x_{k+1})z_{k+1}, \tag{6.10}$$

where $\nabla_{xx}L(x,z)$ is defined in (4.4). It follows from (6.9) that x^{PD}_{k+1} satisfies the equality constraints, which implies that (6.2) holds for all $k \in \mathcal{K}$. Note that since the right-hand side of (6.8)–(6.10), which is $-((\mu_{k+1}-\mu_k)(0\ 0\ -e^T)^T + \Psi(w_{k+1};\mu_k))$, is $\mathcal{O}(\mu_k)$ because of (4.7), (4.8) and (5.1)–(5.3), and as the Jacobian $\nabla_w\Psi(w_{k+1};\mu_{k+1})$ remains uniformly nonsingular in the vicinity of w^*, we also obtain that

$$d_{k+1} = \mathcal{O}(\mu_k) \tag{6.11}$$

for all sufficiently large $k \in \mathcal{K}$. As a consequence, the sequence $\{w^{\mathrm{PD}}_{k+1}\}_{k\in\mathcal{K}}$ also converges to w^* since μ_k converges to zero.

We next show that $c(x^{\mathrm{PD}}_{k+1}) > 0$, which is part of (6.3). If constraint i is inactive, $\{c_i(x_{k+1})\}_{\mathcal{K}} \to c_i(x^*) > 0$ as $\mu_k \searrow 0$. Taylor's expansion of c_i around x_{k+1}, (5.4), and (6.11) give that

$$c_i(x_{k+1} + [d_{k+1}]_x) = c_i(x_{k+1}) + \mathcal{O}(\mu_k)$$

and thus, asymptotically,

$$0 < \tfrac{1}{2}c_i(x_{k+1}) \le c_i(x^{\mathrm{PD}}_{k+1}) \le 2c_i(x_{k+1}), \tag{6.12}$$

which shows that x^{PD}_{k+1} is strictly feasible with respect to the inactive constraints. Now consider the active constraints, if any. Premultiplying (6.10) by Z^{-1}_{k+1} and rearranging, we obtain that for all $i = 1,\dots,p$,

$$c_i(x_{k+1}) + \nabla_x c_i(x_{k+1})^T[d_{k+1}]_x = \mu_{k+1}[z_{k+1}]_i^{-1} - [z_{k+1}]_i^{-1}c_i(x_{k+1})[[d_{k+1}]_z]_i. \tag{6.13}$$

For all active indices, (5.5), (5.6), and (6.11) show that the last term of the right-hand side of (6.13) is $\mathcal{O}(\mu_k^2)$ so that we obtain

$$c_i(x_{k+1}) + \nabla_x c_i(x_{k+1})^T[d_{k+1}]_x = \mu_{k+1}[z_{k+1}]_i^{-1} + \mathcal{O}(\mu_k^2) \qquad (i \in \mathcal{A}). \tag{6.14}$$

Substituting this equation in the expansion

$$c_i(x_{k+1} + [d_{k+1}]_x) = c_i(x_{k+1}) + \nabla_x c_i(x_{k+1})^T[d_{k+1}]_x + \mathcal{O}(\|[d_{k+1}]_x\|^2), \tag{6.15}$$

where we have used (5.4), and using (6.11) gives, for all $i \in \mathcal{A}$,

$$c_i(x_{k+1} + [d_{k+1}]_x) = \mu_{k+1}[z_{k+1}]_i^{-1} + \mathcal{O}(\mu_k^2). \tag{6.16}$$

If constraint i is active, $\{c_i(x_{k+1})\}_{\mathcal{K}} \to c_i(x^*) = 0$ as $\mu_k \searrow 0$. Using now the bounds (5.6), (6.16) yields

$$\frac{\mu_{k+1}}{\kappa^z} + \mathcal{O}(\mu_k^2) \le c_i(x^{\mathrm{PD}}_{k+1}) \le \frac{\mu_{k+1}}{\kappa_z} + \mathcal{O}(\mu_k^2). \tag{6.17}$$

Combining (6.7) and (6.17), we obtain

$$0 < \frac{1}{2\kappa^z}\mu_{k+1} \leq c_i(x_{k+1}^{\rm PD}) \leq \frac{2}{\kappa_z}\mu_{k+1} \tag{6.18}$$

as soon as μ_k is sufficiently small. Relations (6.12) and (6.18) together show that $x_{k+1}^{\rm PD}$ is asymptotically strictly feasible.

In order to complete our proof of (6.3), we now consider the feasibility of $z_{k+1}^{\rm PD}$. Note that since the direction d_{k+1} is $\mathcal{O}(\mu_k)$ because of (6.11), the same holds for $[d_{k+1}]_z$. We then have that, for every $i \in \mathcal{A}$,

$$[z_{k+1}]_i + [[d_{k+1}]_z]_i = [z_{k+1}]_i + \mathcal{O}(\mu_k), \tag{6.19}$$

which implies, by AS5 and for sufficiently large $k \in \mathcal{K}$, that

$$0 < \tfrac{1}{2}[z_{k+1}]_i \leq [z_{k+1}^{\rm PD}]_i \leq 2[z_{k+1}]_i, \tag{6.20}$$

so that $[z_{k+1}^{\rm PD}]_i$ is asymptotically positive. For the inactive constraints, (6.11) indicates that there exists $\kappa^d > 0$ such that $\|[d_{k+1}]_x\| \leq \kappa^d \mu_k$. From (6.10) and the Cauchy–Schwarz inequality, we know that for all $i = 1, \ldots, p$,

$$[z_{k+1}]_i + [[d_{k+1}]_z]_i = c_i^{-1}(x_{k+1})\left(\mu_{k+1} - [z_{k+1}]_i(\nabla_x c_i(x_{k+1}))^T[d_{k+1}]_x\right) \tag{6.21}$$

$$\geq c_i^{-1}(x_{k+1})\left(\mu_{k+1} - [z_{k+1}]_i\|\nabla_x c_i(x_{k+1})\|\|[d_{k+1}]_x\|\right). \tag{6.22}$$

Using (5.4), (5.7), and (6.22), we have that, for $i \notin \mathcal{A}$ and $k \in \mathcal{K}$ sufficiently large,

$$\begin{aligned}
[z_{k+1}^{\rm PD}]_i &\geq c_i^{-1}(x_{k+1})\left(\mu_{k+1} - [z_{k+1}]_i\kappa_\gamma\kappa^d\mu_k\right) \\
&\geq c_i^{-1}(x_{k+1})\left(\mu_{k+1} - \frac{(1+\overline{\kappa}_\mu^{\rm C})}{\kappa_{\rm c}}\kappa_\gamma\kappa^d\mu_k^2\right) \\
&> 0,
\end{aligned}$$

where the last inequality follows from (6.7). Thus (6.3) holds for sufficiently large $k \in \mathcal{K}$.

We next prove (6.6). In view of (2.11), our differentiability assumptions, (5.6), and (5.7) imply that the partial derivatives with respect to x, y, and z of each of the elements of $\nabla_w\Psi(w;\mu)$ clearly remain bounded in a neighborhood of w^* as μ goes to zero by (5.4). Consequently, applying Taylor's theorem to Ψ, we have

$$\Psi(w_{k+1}^{\rm PD};\mu_{k+1}) = \Psi(w_{k+1};\mu_{k+1}) + \nabla_w\Psi(w_{k+1};\mu_{k+1})d_{k+1} + O(\|d_{k+1}\|^2). \tag{6.23}$$

From the definition (5.31) of $w_{k+1}^{\rm PD}$, the first two terms of (6.23) vanish, and hence we deduce

$$\|\Psi(w_{k+1}^{\rm PD};\mu_{k+1})\| = \mathcal{O}(\mu_k^2) \tag{6.24}$$

from (6.11). We finally deduce (6.6) from this bound and (6.7).

That (6.4) holds now immediately follows from (6.6) and (5.1). We next prove (6.5). Using (4.17), we have that

$$\|\nabla_x\mathcal{L}(w_{k+1}^{\rm PD})\|_{[k+2]} \leq \epsilon_M^{-1/2}\|\nabla_x\mathcal{L}(w_{k+1}^{\rm PD})\|_\diamond \leq \epsilon_M^{-1/2}\|\nabla_x\mathcal{L}(w_{k+1}^{\rm PD})\| = \mathcal{O}(\mu_k^2), \tag{6.25}$$

where the last equation follows from (6.24). Now, using (6.1) and (5.2), we have that

$$\mu_k^2 = \mathcal{O}\left(\mu_{k+1}^{\frac{2}{\tau_k}}\right) = o\left(\mu_{k+1}^{1+\gamma_{k+1}}\right) = o(\epsilon^{\rm D}(\mu_{k+1})),$$

which, with (6.25), implies that (6.5) holds for $k \in \mathcal{K}$ sufficiently large. ∎

An important consequence of this result is that, once w_{k+1}^{PD} has been computed, the inner minimization is ultimately unnecessary, since this "starting point" already satisfies the stopping conditions for this minimization. Thus we choose, in what follows,

$$x_{k+2} \stackrel{\mathrm{def}}{=} x_{k+1}^{\mathrm{PD}}, \quad z_{k+2} \stackrel{\mathrm{def}}{=} z_{k+1}^{\mathrm{PD}}, \quad \text{and} \quad y_{k+2} \text{ according to (4.9).} \tag{6.26}$$

Observe that this makes the complete algorithm asymptotically independent of the procedure chosen for the inner minimization, since this procedure is asymptotically never used.

Observe also that, from (5.12) and (6.11),

$$\|w_{k+1}^{\mathrm{PD}} - w(\mu_k)\| = \mathcal{O}\left(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k\right),$$

and from (3.3) we have $\|w(\mu_k) - w(\mu_{k+1})\| = \mathcal{O}(\mu_k - \mu_{k+1}) = \mathcal{O}(\mu_k)$. Combining these two observations, we obtain

$$\|w_{k+1}^{\mathrm{PD}} - w(\mu_{k+1})\| = \mathcal{O}\left(\mu_k^{\gamma_k+\frac{1}{2}} + \mu_k\right). \tag{6.27}$$

In the primal-dual case, [1] and [15] show that the radius of the sphere of convergence of Newton's method for BS(μ_{k+1}) remains both finite and bounded away from zero as μ decreases to zero. Hence, (6.27) shows that w_{k+1}^{PD} asymptotically lies inside that sphere and thus that Newton's method started from w_{k+1}^{PD} would generate points that converge quadratically to $w(\mu_{k+1})$, if an inner minimization were to be used. Also note that (6.1) indicates that the rate of decrease of the barrier parameter must not be too large.

Theorem 6.2 shows that, as soon as the barrier parameter is sufficiently small, the point (5.31) lies strictly inside the feasible region. In a practical implementation, it might be preferable to decide whether or not the algorithm is in a sufficiently advanced stage to use (5.31) by checking its feasibility in conjunction with a test of the form

$$\|\nabla_x \mathcal{L}(w_{k+1}^{\mathrm{PD}})\|_{[k+2]} \leq \max\left(\eta, \nu \|\nabla_x \mathcal{L}(w_{k+1})\|_{[k+2]}\right),$$

where the parameter $\eta > 0$ might be a small multiple of the machine precision and $0 < \nu < 1$, and to ignore the "improved" starting point if this test is violated.

If we now wish to pursue our rate of convergence analysis, we must be more specific about the rule used to update the barrier parameter. So far, we have assumed that $\mu_{k+1} \leq \mu_k$ and (6.1); from now on, we will assume that

$$\mu_{k+1} = \Theta(\mu_k^{\tau_k}), \tag{6.28}$$

where τ_k remains within the bounds specified in (6.1).

The rate of convergence of $\{\mu_k\}$ implied by the updating rule (6.28) directly depends on the sequence $\{\gamma_k\}$ chosen in (5.2). The rule implies

$$\mu_{k+1} = \Omega\left(\mu_k^{\frac{2}{1+\gamma_{k+1}}-\epsilon_\tau}\right),$$

from which we may retrieve the rule (5.29) if we choose $\gamma_k \geq (2 - 3\epsilon_\tau)/(4 + 3\epsilon_\tau)$ for all k. If we choose ϵ_τ sufficiently small and impose $\lim_{k\to\infty} \gamma_k = 0$, then the rate at

which the barrier parameter approaches zero can be made as close to quadratic as one desires.

It is important to make a distinction between a rate of convergence in μ and a rate of convergence in the variables w of the problem. In view of (3.4), if one is able to compute the *exact* solution of BS(μ) for every μ, $w(\mu)$ converges to w^* exactly as fast as μ decreases to zero. Intuitively, when $\gamma_k \geq \frac{1}{2}$, and because (5.12) and (5.13) are always satisfied, one can reasonably expect the same rate of convergence in the approximate solutions w_{k+1} as in μ_k, and we have observed this in practice. However, it is not immediately clear that this may be made rigorous, since the bound (5.13) is only one-sided. In the next results, we show that even in the case $\gamma_k < \frac{1}{2}$, not only can we show that $\{w_{k+1}\}_{\mathcal{K}}$ converges R-superlinearly to w^*, but we obtain Q-superlinear convergence of the whole sequence $\{w_{k+1}\}$ without restrictions on the sequence of scalars $\{\gamma_k\}$. The following lemma parallels [10, Lemma 5.13].

LEMMA 6.3. *Under* AS1–AS6*, assume that w^* is a solution of NLP, that $\{w_{k+1}\}_{\mathcal{K}} \to w^*$, where $\{w_{k+1}\}$ is a sequence of iterates generated by Algorithm* 4.1 *with y_{k+1} defined by* (4.9)*, and that the functions f and c_i $(i = 1, \ldots, p)$ are three times continuously differentiable over an open neighborhood of x^*. Assume furthermore that the barrier parameter μ_k is updated using* (6.28) *and that it is small enough to ensure that w_{k+1}^{PD} defined by* (5.31) *is strictly feasible. Then we have the estimate*

$$w_{k+2} = w(0) + \mu_{k+1}\dot{w}(0) + o(\mu_{k+1}) \tag{6.29}$$

for all sufficiently large $k \in \mathcal{K}$, where w_{k+2} is defined by (6.26) *and $\dot{w}(0) \neq 0$.*

Proof. Proceeding as in the proof of Theorem 6.2, a second-order Taylor expansion of $\Psi(w;\mu)$ about $(w,\mu) = (w^*, 0)$ and the optimality conditions (2.3) yield

$$\Psi(w_{k+2};\mu_{k+1}) = \nabla_w \Psi(w^*;0)(w_{k+2} - w^*) + \nabla_\mu \Psi(w^*;0)\mu_{k+1} + r \tag{6.30}$$

$$= \nabla_w \Psi(w^*;0)(w_{k+2} - w^*) + \begin{bmatrix} 0 \\ 0 \\ -\mu_{k+1}e \end{bmatrix} + r, \tag{6.31}$$

where

$$\|r\| = \mathcal{O}(\max(\|w_{k+2} - w^*\|^2, \mu_{k+1}^2)). \tag{6.32}$$

We may rewrite (6.31) as

$$\begin{bmatrix} \nabla_x \mathcal{L}(w_{k+2}) \\ 0 \\ C(x_{k+2})z_{k+2} \end{bmatrix} = \nabla_w \Psi(w^*;0)(w_{k+2} - w^*) + r. \tag{6.33}$$

Since $\nabla_x \mathcal{L}(w_{k+2})$ lies in the nullspace of A because of (4.11) and (6.26), we have

$$\|\nabla_x \mathcal{L}(w_{k+2})\| = \|\nabla_x \mathcal{L}(w_{k+2})\|_\diamond = \|\nabla_x \mathcal{L}(w_{k+1}^{\mathrm{PD}})\|_\diamond = o(\mu_{k+1}) \tag{6.34}$$

and

$$C(x_{k+2})z_{k+2} = \mu_{k+1}e + o(\mu_{k+1}), \tag{6.35}$$

using (6.6). Consequently, substituting into (6.33) and using the nonsingularity of $\nabla_w \Psi(w^*;0)$, we obtain

$$w_{k+2} = w(0) + \mu_{k+1}\dot{w}(0) + o(\mu_{k+1}) + (\nabla_w \Psi(w^*;0))^{-1} r, \tag{6.36}$$

where we observed that $w^* = w(0)$ and $(\nabla_w \Psi(w^*;0))^{-1}\,[0\ 0\ e^T]^T = \dot{w}(0) \neq 0$ because of (3.6).

To complete the proof, it remains to show that $\|r\| = o(\mu_{k+1})$. From (6.34) and (6.35), for sufficiently large $k \in \mathcal{K}$, $\|\nabla_x \mathcal{L}(w_{k+2})\|_\diamond + \|C(x_{k+2})z_{k+2}\|$ is smaller than the threshold ε given in Lemma 3.1, and thus, applying (3.3) with the parameters $(\nabla_x \mathcal{L}(w_{k+2}), 0, C(x_{k+2})z_{k+2})$ and $(0,0,0)$ yields

$$\|w_{k+2} - w^*\| = \Theta(\|\nabla_x \mathcal{L}(w_{k+2})\|_\diamond + \|C(x_{k+2})z_{k+2}\|) \tag{6.37}$$

$$= \mathcal{O}(\mu_{k+1}), \tag{6.38}$$

where the last equality is due to (6.34) and (6.35). We thus obtain that $\|w_{k+2} - w^*\|^2 = \mathcal{O}(\mu_{k+1}^2) = o(\mu_{k+1})$, and thus (6.32) clearly implies that $\|r\| = o(\mu_{k+1})$ and proves (6.29). ∎

The result contained in Lemma 6.3 parallels the well-known expansion

$$w(\mu_{k+1}) = w(0) + \mu_{k+1}\dot{w}(0) + o(\mu_{k+1}),$$

which holds for *exact* solutions of BS(μ). It also confirms the suggestion in Figure 5.2 that the trajectory $\mathcal{C}'$ is close to the primal-dual central path. Moreover, it reinforces the observation that the paths $\mathcal{C}$ and $\mathcal{C}'$ coincide up to first order at w^*.

Considering the indentity (6.29) componentwise, we immediately have the following corollary.

COROLLARY 6.4. *Under the assumptions of Lemma* 6.3*, we have*

$$[w_{k+2}]_i = [w^*]_i + \mu_{k+1}[\dot{w}(0)]_i + o(\mu_{k+1}), \quad i = 1, \ldots, n+m+p, \tag{6.39}$$

for all sufficiently large $k \in \mathcal{K}$, with w_{k+2} defined by (6.26).

Consequently, there exists a constant $\kappa^w > 0$ such that, for all $k \in \mathcal{K}$ sufficiently large,

$$|[w_{k+2}]_i - [w^*]_i| \leq \kappa^w \mu_{k+1}, \quad i = 1, \ldots, n+m+p, \tag{6.40}$$

which, unlike (5.13), is independent of the sequence $\{\gamma_k\}$.

So far, we have simply assumed that w^* is a limit point of the sequence $\{w_{k+1}\}_{k\in\mathcal{K}}$. We are now in position to prove that the whole sequence of iterates w_{k+1} converges Q-superlinearly to w^*. Moreover, upon defining

$$\mathcal{J} \stackrel{\text{def}}{=} \{i = 1, \ldots, n+m+p \mid [\dot{w}(0)]_i \neq 0\}, \tag{6.41}$$

this convergence occurs componentwise for $i \in \mathcal{J}$, showing that all errors $|[w_{k+2}]_i - [w^*]_i|$ with $i \in \mathcal{J}$ are of comparable size, and thus that the corresponding variables converge to their limit at a comparable rate. The following result is inspired by [10, Theorem 5.14].

THEOREM 6.5. *Under* AS1–AS6*, assume that w^* is a solution of NLP, that the sequence $\{w_{k+1}\}_{\mathcal{K}} \to w^*$, where $\{w_{k+1}\}$ is a sequence of iterates generated by Algorithm* 4.1 *with y_{k+1} defined by* (4.9)*, and that the functions f and c_i $(i = 1, \ldots, p)$ are three times continuously differentiable over an open set containing $\mathcal{F}$. Assume furthermore that* (5.1)–(5.3) *are satisfied and that the barrier parameter μ_k is updated using* (6.28)*. Assume finally that* (6.26) *is used for all $k \in \mathcal{K}$ large enough to ensure that Theorem* 6.2 *holds. Then the complete sequence $\{w_{k+1}\}$ converges to w^* and*

$$\left|\frac{[w_{k+2}]_i - [w^*]_i}{[w_{k+1}]_i - [w^*]_i}\right| = \Theta(\mu_k^{\tau_k - 1}), \qquad i \in \mathcal{J}, \tag{6.42}$$

for k sufficiently large, which implies that the iterates w_{k+1} converge componentwise Q-superlinearly to w^*, along all those components i for which $[\dot{w}(0)]_i \neq 0$.

Proof. First note that (6.40) and the convergence of μ_{k+1} to zero imply that $\{w_{k+2}\}_{k\in\mathcal{K}}$ also converges to w^*. If we set $\mathcal{K}^+ = \mathcal{K} \cup \{k+1 \mid k \in \mathcal{K}\}$, we thus have that $\{w_{k+1}\}_{k\in\mathcal{K}^+}$ converges to w^*, and we may reapply (6.40) to this subsequence, to conclude that $\{w_{k+2}\}_{k\in\mathcal{K}^+}$ also converges to w^*. Applying this argument inductively, we obtain that the complete sequence $\{w_k\}$ converges to w^* and therefore that $\mathcal{K}$ may be identified with the set of all positive integers. Our assumptions then yield that Theorem 6.2 holds and that (6.26) is used for all k sufficiently large. Moreover, the estimate (6.39) also holds for all k sufficiently large, implying that $|[w_{k+1}]_i - [w^*]_i| = \Theta(\mu_k)$ for all $i \in \mathcal{J}$. This proves that

$$\left|\frac{[w_{k+2}]_i - [w^*]_i}{[w_{k+1}]_i - [w^*]_i}\right| = \Theta\left(\frac{\mu_{k+1}}{\mu_k}\right), \qquad i \in \mathcal{J},$$

which then gives (6.42) because of (6.28). The componentwise Q-superlinear convergence of the iterates to w^* then follows from the convergence of μ_k to zero and the inequality $\tau_k \geq 1 + \epsilon_\tau$. ∎

Theorem 6.5 has the following consequence.

COROLLARY 6.6. *Under the assumptions of Theorem* 6.5, *suppose that*

$$\tau_k = \frac{2}{1+\gamma_{k+1}} - \epsilon_\tau, \tag{6.43}$$

where $0 < \epsilon_\tau < 1/2$ is a given constant, with γ_k satisfying $0 < \gamma_k \leq (1-2\epsilon_\tau)/(1+2\epsilon_\tau)$ and

$$\lim_{k\to\infty} \gamma_k = 0.$$

Then, for any $\sigma \in (1, 2-\epsilon_\tau)$, there exists a constant $q_\sigma > 0$ such that

$$|[w_{k+2}]_i - [w^*]_i| \leq q_\sigma |[w_{k+1}]_i - [w^*]_i|^\sigma, \qquad i \in \mathcal{J},$$

with $\mathcal{J}$ as defined in (6.41).

Proof. First recall that Corollary 6.4 and the fact that $[\dot{w}(0)]_i \neq 0$ for $i \in \mathcal{J}$ implies that $|[w_{k+1}]_i - [w^*]_i| = \Theta(\mu_k)$ for all k sufficiently large. This and (6.28) yield that

$$\frac{|[w_{k+2}]_i - [w^*]_i|}{|[w_{k+1}]_i - [w^*]_i|^\sigma} = \Theta(\mu_k^{\tau_k - \sigma}), \qquad i \in \mathcal{J}.$$

Our assumptions and the fact that, for any $\sigma \in (1, 2-\epsilon_\tau)$,

$$\tau_k = \frac{2}{1+\gamma_{k+1}} - \epsilon_\tau \geq \sigma$$

for k sufficiently large then implies the desired result. ∎

Strictly speaking, our componentwise convergence results hold for all components only if all components of $\dot{w}(0)$ are nonzero. A suitable change of coordinates can transform any nonzero vector $\dot{w}(0)$ into a vector parallel to the vector of all ones without modifying the nature of problem NLP or the convergence of $\{w_k\}$ to w^*. For instance, the $(n+m+p) \times (n+m+p)$ Householder reflection $P = I - 2/\|v\|^2\ vv^T$, where

$v = \dot{w}(0) \pm e \, \|\dot{w}(0)\|/\sqrt{n+m+p}$, is such a transformation. In the transformed space, Theorem 6.5 and Corollary 6.6 therefore hold for *all* components $i = 1, \ldots, n+m+p$. We should, however, be cautious when transforming back for the same rate of convergence might not apply to those components for which $[\dot{w}(0)]_i = 0$. In other words, a fast componentwise Q-rate of convergence in a particular coordinate system does not necessarily imply that the same rate of convergence still applies to all the components in a different coordinate system, although, of course, it does apply normwise.

Interior point methods of the type studied above are thus likely to achieve a rate of convergence that is in practice as fast as that of exterior-penalty methods. In addition, the rate of convergence implied by our theory is governed by ϵ_τ, and Corollary 6.6 shows that this rate can be made as close to quadratic as we wish by choosing ϵ_τ sufficiently close to zero in (6.43). Note that Corollary 6.4 also holds for [10, Lemma 5.13] in connection with exterior penalty methods, and thus that componentwise Q-superlinear convergence also occurs in that case.

Remark 6.1. Most of the qualitative observations made in this paper essentially remain true in the purely primal case. When considering the primal approach, one has to replace (2.2) with

$$\Psi_p(x, y; \mu) = \begin{bmatrix} \nabla_x \mathcal{L}_p(x, y; \mu) \\ Ax - b \end{bmatrix},$$

where $\nabla_x \mathcal{L}_p(x, y; \mu) = \nabla_x f(x) + A^T y - \mu J^T(x) C^{-1}(x) e$, since the left-hand side of (4.7) is always identically zero. The primal case is analyzed both for interior and exterior penalty functions by Dussault [5]. From the quantitative point of view, one obtains two-step superlinear convergence in the primal case as opposed to one-step superlinear convergence in the primal-dual case, using the same sort of Newton-like extrapolation step. In the primal case, the extrapolation step itself is not enough to satisfy the termination tolerances, and one has to perform an additional Newton step. The result obtained in [5] is that the updating rule for the barrier parameter has to satisfy $\mu_{k+1} = \Omega(\mu_k^{4/3})$, thereby limiting the speed of convergence.

7. An application. We now show that the theoretical framework developed above can be applied. In particular we first examine why it applies to the method proposed in [2]. This algorithm consists of an inner iteration imbricated in an outer one. We do not describe the inner minimization here, but refer the interested reader to [2] for a discussion. Suffice it to say that it uses a trust-region algorithm with a primal-dual model of the log-barrier function (1.3). The stopping conditions for this inner iteration are exactly (4.5)–(4.8) augmented by the requirement that

$$\lambda^{\min}_{M_{k+1}} [V(x_{k+1}, z_{k+1})] \geq -\epsilon^{\mathrm{E}}(\mu_k) \tag{7.1}$$

for some forcing function $\epsilon^{\mathrm{E}}(\mu)$. This additional condition is meant to enforce convergence to a second-order critical point of the log-barrier function.

The global convergence of the resulting minimization procedure (inner and outer minimizations together) to weak second-order critical points is guaranteed under standard assumptions [2, Theorem 4.12]. These assumptions are slightly different from those used here: in particular, strict complementarity slackness is not required, and approximate second-order derivatives of the objective function and constraints are allowed, while we concentrate here on the case where they are exact. However, this convergence result depends on three additional conditions, namely that

$$\lim_{k \to \infty} \frac{\epsilon^{\mathrm{D}}(\mu_k)\sqrt{\mu_k}}{\min_i c_i(x_{k+1})} = 0, \tag{7.2}$$

that the tolerance $\epsilon^C(\mu)$ is asymptotically of the form $\mathcal{O}(\mu)$, and that the barrier function is bounded from below on $\mathcal{F}$ for all generated iterates and all small μ. Note that (5.1) implies that $\epsilon^C(\mu_k)$ is asymptotically of the order of μ_k and that (5.2) implies that condition (7.2) is satisfied because of (5.5) and (5.8). As a consequence we see that the global convergence theory for the particular implementation described in [2] is not upset by our choice of stopping tolerances. Moreover, the boundedness of the log-barrier function is guaranteed here, because of AS2, Theorem 5.1, and the fact that our analysis only considers convergent subsequences.

In order to apply our rate of convergence results, we finally have to verify that introducing condition (7.1) in the set of stopping criterion for the inner minimization algorithm does not affect our conclusions. Fortunately, we may deduce from Lemma 6.1 that the matrix $V(x, z)$ is asymptotically second-order sufficient and thus ultimately that condition (7.1) will automatically be satisfied at the iterates generated by Algorithm 4.1 that are sufficiently close to a local solution w^*. The w_{k+1} ($k \in \mathcal{K}$) are such iterates. Consequently, Theorem 6.5 and Corollary 6.6 apply for the algorithm proposed by [2].

It is tempting, although technically difficult, to attempt to apply our results to other primal-dual methods for nonlinear optimization. In particular, the methods of Gay, Overton, and Wright [7] and Byrd, Liu, and Nocedal [1] seem natural candidates. However, a fully unified theory appears to require more work. In particular, besides the fact that these methods handle the full nonlinear program, including nonlinear equality constraints, and allow infeasible iterates with respect to those constraints, they also differ from our framework in further respects. The method of Gay, Overton, and Wright uses a watchdog technique to allow a possible nonmonotone behavior of the sequence of values of a log-barrier based merit function, while our technique does not impose any condition on this sequence. The method of Byrd, Liu, and Nocedal uses slack variables to transform general inequalities into bound constraints. Both methods impose the same accuracy requirement for (4.7) and (4.8) while our approach differentiates between those two components of the optimality conditions (see (5.1)–(5.3)). Moreover, the rules to update the barrier parameter differ in both cases from those considered here.

8. Conclusion. In this paper, we have studied the local convergence properties of primal-dual interior point algorithms for minimizing a general, nonconvex, objective function subject to linear equality constraints and nonconvex inequality constraints, of which the method proposed by Conn et al. [2] is a prime example. Our analysis is inspired by those of [3, 5, 10, 21]. The theoretical results show a convergence rate for barrier methods that is essentially as fast as that previously obtained for exterior penalty methods [10]. These results rely on a suitable extrapolation of the central path from the current iterate w_{k+1} which leads to an asymptotically acceptable w_{k+2}, i.e., a point which immediately satisfies the tolerance requirements corresponding to the updated barrier parameter μ_{k+1}. This is shown to imply a componentwise Q-superlinear convergence rate, and one asymptotically has to solve, in each outer iteration, a single linear system whose coefficient matrix is that of the Newton equations at w_{k+1}. Nevertheless, fast convergence in a particular component depends crucially upon the corresponding component of $\dot{w}(0)$, the tangent to the central path at the solution, being nonzero. It is worth emphasizing that the results presented here hold independently of the exact inner minimization procedure used, provided it ensures that (4.5)–(4.8) are satisfied. The componentwise Q-superlinear convergence of w_k to w^* also holds independently of any particular updating rule for the variable γ_k used

in (5.2), provided it satisfies (5.3). Moreover, if the sequence $\{\gamma_k\}$ converges to zero, then the rate of convergence can be made as close to quadratic as desired by choosing ϵ_τ sufficiently small in (6.43). A consequence is that we may alternatively view the results of the present paper as giving conditions on the stopping criterion of any barrier subproblem solver that ensure componentwise Q-superlinear convergence of the outer iterates. This parallels the results of [13] for linear complementarity problems.

Reasons why one should use extrapolated steps in barrier-type methods are developed in [3, 12, 17], and an analysis similar to that developed in the present paper is developed in [1], where one-step superlinear convergence of an interior point primal-dual trust-region algorithm is exhibited. As superlinear convergence has already been observed in practice during tests on quadratic programs (see [2]), the authors believe that it will be equally worthwhile to experiment with the strategy sketched in this paper on highly nonlinear and high-dimensional optimization problems. It should be mentioned, however, that the extrapolation strategy is only likely to be numerically efficient in conjunction with a method that solves the Newton equations accurately, without suffering from any ill-conditioning that is not already present in problem NLP [5]. Furthermore, higher convergence rates analysis, achievable by taking a further Newton step from the extrapolated point, will be analyzed in a companion paper. In view of the analysis conducted in [21], one may reasonably hope that the results exhibited in the present paper remain essentially true when the linear independence constraint qualification is replaced by the weaker Mangasarian–Fromovitz constraint qualification. One may also hope to obtain interesting, yet similar, results when the strict complementarity condition is dropped. Relaxation of those assumptions and further investigation on the componentwise convergence properties in different coordinate systems are left for future work.

Acknowledgment. The authors are grateful to Jorge Nocedal for his helpful comments on an earlier draft of this paper.

REFERENCES

[1] R. H. Byrd, G. Liu, and J. Nocedal, *On the local behavior of an interior point method for nonlinear programming*, in Numerical Analysis 1997, D. Griffiths and D. Higham, eds., Addison Wesley Longman, Reading, MA, 1997, pp. 37–56.

[2] A. R. Conn, N. I. M. Gould, D. Orban, and P. L. Toint, *A primal-dual trust-region algorithm for non-convex nonlinear programming*, Math. Program. Ser. B, 87 (2000), pp. 215–249.

[3] A. R. Conn, N. I. M. Gould, and P. L. Toint, *A note on using alternative second-order models for the subproblems arising in barrier function methods for minimization*, Numer. Math., 68 (1994), pp. 17–33.

[4] A. R. Conn, N. I. M. Gould, and P. L. Toint, *On the number of inner iterations per outer iteration of a globally convergent algorithm for optimization with general nonlinear inequality constraints and simple bounds*, Comput. Optim. Appl., 7 (1997), pp. 41–70.

[5] J.-P. Dussault, *Numerical stability and efficiency of penalty algorithms*, SIAM J. Numer. Anal., 32 (1995), pp. 296–317.

[6] A. V. Fiacco and G. P. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, J. Wiley and Sons, Chichester, England, 1968. Reprinted as Classics Appl. Math. 4, SIAM, Philadelphia, 1990.

[7] D. M. Gay, M. L. Overton, and M. H. Wright, *A primal-dual interior method for nonconvex nonlinear programming*, in Advances in Nonlinear Programming, Y. Yuan, ed., Kluwer, Dordrecht, the Netherlands, 1998, pp. 31–56.

[8] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization*, Academic Press, London, 1981.

[9] N. I. M. Gould, *On practical conditions for the existence and uniqueness of solutions to the general equality quadratic-programming problem*, Math. Program., 32 (1985), pp. 90–99.

[10] N. I. M. Gould, *On the convergence of a sequential penalty function method for constrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 107–128.

[11] R. Mifflin, *Convergence bounds for nonlinear programming algorithms*, Math. Program., 8 (1975), pp. 251–271.

[12] S. G. Nash and A. Sofer, *Why Extrapolation Helps in Barrier Methods*, Technical report, Operations Research and Engineering Department, George Mason University, Fairfax, VA, 1998.

[13] F. Potra, *Q-Superlinear Convergence of the Iterates in Primal-Dual Interior-Point Methods*, Technical report, University of Maryland–Baltimore County, 1999.

[14] M. C. Villalobos, R. A. Tapia, and Y. Zhang, *The Local Behavior of Newton's Method on Two Equivalent Systems from Linear Programming*, Technical report CRPC-TR98770-S, Department of Computational and Applied Mathematics, Rice University, Houston, TX, 1998.

[15] M. C. Villalobos, R. A. Tapia, and Y. Zhang, *The Sphere of Convergence of Newton's Method on Two Equivalent Systems from Nonlinear Programming*, Technical report TR9913, Department of Computational and Applied Mathematics, Rice University, Houston, TX, 1999.

[16] M. H. Wright, *Interior methods for constrained optimization*, Acta Numer., 1 (1992), pp. 341–407.

[17] M. H. Wright, *Why a pure primal Newton barrier step may be infeasible*, SIAM J. Optim., 5 (1995), pp. 1–12.

[18] M. H. Wright, *The interior-point revolution in constrained optimization*, in High Performance Algorithms and Software in Nonlinear Optimization, R. DeLeone, A. Murli, P. Pardalos, and G. Toraldo, eds., Kluwer, Dordrecht, the Netherlands, 1998, pp. 359–381.

[19] S. J. Wright, *Primal-Dual Interior-Point Methods*, SIAM, Philadelphia, 1997.

[20] S. J. Wright and F. Jarre, *The role of linear objective functions in barrier methods*, Math. Program., 84 (1999), pp. 357–373.

[21] S. J. Wright and D. Orban, *Local Convergence of the Newton/Log-Barrier Method for Degenerate Problems*, Technical report ANL/MCS-P772-0799, Argonne National Laboratory, Argonne, IL, 1999, submitted to Math. Oper. Res.

[22] H. Yabe and H. Yamashita, *Q-superlinear convergence of primal-dual interior point quasi-Newton methods for constrained optimization*, J. Oper. Res. Soc. Japan, 40 (1997), pp. 415–436.

[23] H. Yamashita and H. Yabe, *Superlinear and quadratic convergence of some primal-dual interior point methods for constrained optimization*, Math. Program. Ser. A, 75 (1996), pp. 377–397.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1003–1018

THE SET OF DIVERGENT DESCENT METHODS IN A BANACH SPACE IS σ-POROUS*

SIMEON REICH† AND ALEXANDER J. ZASLAVSKI†

Abstract. Given a Lipschitzian convex function f on a Banach space X, we consider a complete metric space $\mathcal{A}$ of vector fields V on X with the topology of uniform convergence on bounded subsets. With each such vector field we associate two iterative processes. We introduce the class of regular vector fields $V \in \mathcal{A}$ and prove (under two mild assumptions on f) that the complement of the set of regular vector fields is not only of the first category, but also σ-porous. We then show that for a locally uniformly continuous regular vector field V and a coercive function f, the values of f tend to its infimum for both processes.

Key words. Banach space, complete metric space, convex function, descent method, generic property, iterative process, porous set

AMS subject classifications. 49M07, 49M10, 49M45, 90C25

PII. S1052623400370357

1. Introduction.

Assume that $(X, ||\cdot||)$ is a Banach space with norm $||\cdot||$, $(X^*, ||\cdot||_*)$ is its dual space with the norm $||\cdot||_*$, and $f : X \to R^1$ is a convex continuous function which is bounded from below. Recall that for each pair of sets $A, B \subset X^*$,

$$H(A,B) = \max\left\{\sup_{x\in A}\inf_{y\in B}||x-y||_*,\ \sup_{y\in B}\inf_{x\in A}||x-y||_*\right\}$$

is the Hausdorff distance between A and B.

For each $x \in X$, let

$$\partial f(x) = \{l \in X^* :\ f(y) - f(x) \geq l(y-x) \text{ for all } y \in X\}$$

be the subdifferential of f at x. It is well known that the set $\partial f(x)$ is nonempty and bounded (in the norm topology). Set

$$\inf(f) = \inf\{f(x) :\ x \in X\}.$$

Denote by $\mathcal{A}$ the set of all mappings $V : X \to X$ such that V is bounded on every bounded subset of X (i.e., for each $K_0 > 0$ there is $K_1 > 0$ such that $||Vx|| \leq K_1$ if $||x|| \leq K_0$), and for each $x \in X$ and each $l \in \partial f(x)$, $l(Vx) \leq 0$. We denote by $\mathcal{A}_c$ the set of all continuous $V \in \mathcal{A}$, by $\mathcal{A}_u$ the set of all $V \in \mathcal{A}$ which are uniformly continuous on each bounded subset of X, and by $\mathcal{A}_{au}$ the set of all $V \in \mathcal{A}$ which are uniformly continuous on the subsets

$$\{x \in X :\ ||x|| \leq n \text{ and } f(x) \geq \inf(f) + 1/n\}$$

*Received by the editors April 7, 2000; accepted for publication (in revised form) November 29, 2000; published electronically April 26, 2001. The work of the first author was partially supported by the Israel Science Foundation founded by the Israel Academy of Sciences and Humanities, by the Fund for the Promotion of Research at Technion, and by the Technion VPR Fund—E. and M. Mendelson Research Fund.

http://www.siam.org/journals/siopt/11-4/37035.html

†Department of Mathematics, Technion, Israel Institute of Technology, 32000 Haifa, Israel (sreich@tx.technion.ac.il, ajzasl@tx.technion.ac.il).

for each integer $n \geq 1$. Finally, let $\mathcal{A}_{auc} = \mathcal{A}_{au} \cap \mathcal{A}_c$.

Next we endow the set $\mathcal{A}$ with a metric ρ: For each $V_1, V_2 \in \mathcal{A}$ and each integer $i \geq 1$, we first set

$$\rho_i(V_1, V_2) = \sup\{||V_1x - V_2x|| : \ x \in X \ \text{ and } ||x|| \leq i\} \tag{1.1}$$

and then define

$$\rho(V_1, V_2) = \sum_{i=1}^{\infty} 2^{-i}[\rho_i(V_1, V_2)(1 + \rho_i(V_1, V_2))^{-1}]. \tag{1.2}$$

Clearly, $(\mathcal{A}, \rho)$ is a complete metric space. It is also not difficult to see that the collection of the sets

$$E(N, \epsilon) = \{(V_1, V_2) \in \mathcal{A} \times \mathcal{A} : \ ||V_1x - V_2x|| \leq \epsilon, \ x \in X, \ ||x|| \leq N\}, \tag{1.3}$$

where $N, \epsilon > 0$, is a base for the uniformity generated by the metric ρ. Evidently $\mathcal{A}_c$, $\mathcal{A}_u$, $\mathcal{A}_{au}$, and $\mathcal{A}_{auc}$ are closed subsets of the metric space $(\mathcal{A}, \rho)$. In what follows we assign to all these spaces the same metric ρ.

To compute $\inf(f)$, we associate with each vector field $W \in \mathcal{A}$ two gradient-like iterative processes (see (1.5) and (1.7) below).

The study of minimization methods for convex functions is a central topic in optimization theory. See, for example, [1, 2, 4, 9, 10] and references therein. Note, in particular, that the counterexample studied in section 2.2, Chapter VIII of [10] shows that, even for two-dimensional problems, the simplest choice for a descent direction, namely, the normalized steepest descent direction,

$$V(x) = \operatorname{argmin}\left\{\max_{l \in \partial f(x)} < l, d >: ||d|| = 1\right\},$$

may produce sequences whose functional values fail to converge to the infimum of f. This vector field V belongs to $\mathcal{A}$ and the Lipschitzian function f attains its infimum. The steepest descent scheme (Algorithm 1.1.7 in section 1.1, Chapter VIII of [10]) corresponds to either of the two iterative processes we consider below.

In infinite dimensions the problem is even more difficult and less understood. Moreover, positive results usually require special assumptions on the space and the functions. However, as shown in our previous paper [15] (under certain assumptions on the function f), for an arbitrary Banach space X and a generic vector field $V \in \mathcal{A}$, the values of f tend to its infimum for both processes. In that paper, instead of considering a certain convergence property for a method generated by a single vector field V, we investigated it for the whole space $\mathcal{A}$ and showed that this property held for most of the vector fields in $\mathcal{A}$. This approach has also been successfully applied in the theory of dynamical systems [5, 13], approximation theory [6], optimization [8, 11, 14, 16], as well as in optimal control [20, 21, 22].

In the present paper we introduce the class of regular vector fields $V \in \mathcal{A}$. Our first result, Theorem 1, shows (under the two mild assumptions A(i) and A(ii) on f stated below) that the complement of the set of regular vector fields is not only of the first category, but also σ-porous in each of the spaces $\mathcal{A}$, $\mathcal{A}_c$, $\mathcal{A}_u$, $\mathcal{A}_{au}$, and $\mathcal{A}_{auc}$. We then show (Theorem 2) that for any regular vector field $V \in \mathcal{A}_{au}$, if the constructed sequence $\{x_i\}_{i=0}^{\infty} \subset X$ has a bounded subsequence (in the case of the first process) or is bounded (in the case of the second one), then the values of the function f tend

to its infimum for both processes. If, in addition to A(i) and A(ii), f also satisfies the assumption A(iii), then this convergence result is valid for any regular $V \in \mathcal{A}$. Note that if the function f is coercive, then the constructed sequences will always stay bounded. Thus we see, by Theorem 1, that for a coercive f the set of divergent descent methods is σ-porous. Our last result, Theorem 3, shows that in this case we obtain not only convergence, but also stability.

Before we continue we recall the concept of porosity [3, 6, 7, 17, 18, 19].

Let (Y, d) be a complete metric space. We denote by $B(y, r)$ the closed ball of center $y \in Y$ and radius $r > 0$. A subset $E \subset Y$ is called porous in (Y, d) if there exist $\alpha \in (0, 1)$ and $r_0 > 0$ such that for each $r \in (0, r_0]$ and each $y \in Y$ there exists $z \in Y$ for which

$$B(z, \alpha r) \subset B(y, r) \setminus E.$$

A subset of the space Y is called σ-porous in (Y, d) if it is a countable union of porous subsets in (Y, d).

Remark 1.1. It is known that in the above definition of porosity the point y can be assumed to belong to E.

Other notions of porosity have been used in the literature [3, 17]. We use the rather strong notion which appears in [6, 7].

Since porous sets are nowhere dense, all σ-porous sets are of the first category. If Y is a finite-dimensional Euclidean space, then σ-porous sets are of Lebesgue measure 0. In fact, the class of σ-porous sets in such a space is much smaller than the class of sets which have measure 0 and are of the first category. Also, every Banach space contains a set of the first category which is not σ-porous [12, 17, 18]. Moreover, every complete metric space without isolated points contains a closed nowhere dense set which is not σ-porous [19].

To point out the difference between porous and nowhere dense sets note that if $E \subset Y$ is nowhere dense, $y \in Y$, and $r > 0$, then there is a point $z \in Y$ and a number $s > 0$ such that $B(z, s) \subset B(y, r) \setminus E$. If, however, E is also porous, then for small enough r we can choose $s = \alpha r$, where $\alpha \in (0, 1)$ is a constant which depends only on E.

Our results will be established in any Banach space and for those convex functions which satisfy the following two assumptions.

A(i) There exists a bounded (in the norm topology) set $X_0 \subset X$ such that

$$\inf(f) = \inf\{f(x) : \ x \in X\} = \inf\{f(x) : \ x \in X_0\};$$

A(ii) for each $r > 0$ the function f is Lipschitzian on the ball $\{x \in X : \ ||x|| \leq r\}$.

Remark 1.2. We may assume that the set X_0 in A(i) is closed and convex.

Remark 1.3. Clearly, assumption A(i) holds if $\lim_{||x|| \to \infty} f(x) = \infty$.

We will say that a mapping $V \in \mathcal{A}$ is regular if for any natural number n there exists a positive number $\delta(n)$ such that for each $x \in X$ satisfying

$$||x|| \leq n \text{ and } f(x) \geq \inf(f) + 1/n,$$

and each $l \in \partial f(x)$, we have

$$l(Vx) \leq -\delta(n).$$

Denote by $\mathcal{F}$ the set of all regular vector fields $V \in \mathcal{A}$.

It is not difficult to verify the following property of regular vector fields. It means, in particular, that $\mathcal{G} = \mathcal{A} \setminus \mathcal{F}$ is a face of the convex cone $\mathcal{A}$ in the sense that if a nontrivial convex combination of two vector fields in $\mathcal{A}$ belongs to $\mathcal{G}$, then both of them must belong to $\mathcal{G}$.

PROPOSITION 1.1. *Assume that $V_1, V_2 \in \mathcal{A}$, V_1 is regular, $\phi : X \to [0,1]$, and that for each integer $n \geq 1$,*

$$\inf\{\phi(x) : \ x \in X \ \text{and} \ ||x|| \leq n\} > 0.$$

Then the mapping $x \to \phi(x)V_1x + (1 - \phi(x))V_2x$, $x \in X$, also belongs to $\mathcal{F}$.

Our first result shows that in a very strong sense most of the vector fields in $\mathcal{A}$ are regular.

THEOREM 1. *Assume that both A(i) and A(ii) hold. Then $\mathcal{A} \setminus \mathcal{F}$ (respectively, $\mathcal{A}_c \setminus \mathcal{F}$, $\mathcal{A}_{au} \setminus \mathcal{F}$, and $\mathcal{A}_{auc} \setminus \mathcal{F}$) is a σ-porous subset of the space $\mathcal{A}$ (respectively, $\mathcal{A}_c$, $\mathcal{A}_{au}$, and $\mathcal{A}_{auc}$). Moreover, if f attains its infimum, then the set $\mathcal{A}_u \setminus \mathcal{F}$ is also a σ-porous subset of the space $\mathcal{A}_u$.*

Now let $W \in \mathcal{A}$. We associate with W two iterative processes.

For $x \in X$ we denote by $P_W(x)$ the set of all

$$y \in \{x + \alpha Wx : \ \alpha \in [0,1]\}$$

such that

$$f(y) = \inf\{f(x + \beta Wx) : \ \beta \in [0,1]\}. \tag{1.4}$$

Given any initial point $x_0 \in X$, one can construct a sequence $\{x_i\}_{i=0}^{\infty} \subset X$ such that for all $i = 0, 1, \ldots,$

$$x_{i+1} \in P_W(x_i). \tag{1.5}$$

This is our first iterative process.

Next we describe the second iterative process.

Given a sequence $\mathbf{a} = \{a_i\}_{i=0}^{\infty} \subset (0,1]$ such that

$$\lim_{i \to \infty} a_i = 0 \ \text{and} \ \sum_{i=0}^{\infty} a_i = \infty, \tag{1.6}$$

we construct for each initial point $x_0 \in X$ a sequence $\{x_i\}_{i=0}^{\infty} \subset X$ according to the following rule:

$$x_{i+1} = x_i + a_i W(x_i) \ \text{if} \ f(x_i + a_i W(x_i)) < f(x_i), \tag{1.7}$$

$$x_{i+1} = x_i \ \text{otherwise},$$

where $i = 0, 1, \ldots$.

In what follows we will also make use of the following assumption.

A(iii) For each integer $n \geq 1$ there exists $\delta > 0$ such that for each $x_1, x_2 \in X$ satisfying

$$||x_1||, ||x_2|| \leq n, \ f(x_i) \geq \inf(f) + 1/n, \ i = 1, 2, \ \text{and} \ ||x_1 - x_2|| \leq \delta,$$

the following inequality holds:

$$H(\partial f(x_1), \partial f(x_2)) \leq 1/n.$$

This assumption is certainly satisfied if f is differentiable and its derivative is uniformly continuous on those bounded subsets of X over which the infimum of f is larger than $\inf(f)$.

Our next result is a convergence theorem for those iterative processes associated with regular vector fields. It is of interest to note that we obtain convergence when either the regular vector field W or the subdifferential ∂f enjoys a certain uniform continuity property.

THEOREM 2. *Assume that $W \in \mathcal{A}$ is regular, A(i), A(ii) are valid, and that at least one of the following conditions holds:* 1. $W \in \mathcal{A}_{au}$; 2. *A(iii) is valid. Then the following two assertions are true:*

(i) *Let the sequence $\{x_i\}_{i=0}^{\infty} \subset X$ satisfy (1.5) for all $i = 0, 1, \ldots$. If*

$$\liminf_{i\to\infty} ||x_i|| < \infty,$$

then $\lim_{i\to\infty} f(x_i) = \inf(f)$.

(ii) *Let a sequence $\mathbf{a} = \{a_i\}_{i=0}^{\infty} \subset (0, 1]$ satisfy (1.6) and let the sequence $\{x_i\}_{i=0}^{\infty} \subset X$ satisfy (1.7) for all $i = 0, 1, \ldots$. If $\{x_i\}_{i=0}^{\infty}$ is bounded, then*

$$\lim_{i\to\infty} f(x_i) = \inf(f).$$

Finally, we impose an additional coercivity condition on f and establish the following stability theorem. Note that this coercivity condition implies A(i) (Remark 1.3).

THEOREM 3. *Assume that $f(x) \to \infty$ as $||x|| \to \infty$, $V \in \mathcal{A}$ is regular, A(ii) is valid, and that at least one of the following conditions holds:* 1. $V \in \mathcal{A}_{au}$; 2. *A(iii) is valid.*

Let $K, \epsilon > 0$ be given. Then there exist a neighborhood $\mathcal{U}$ of V in $\mathcal{A}$ and a natural number N_0 such that the following two assertions are true:

(i) *For each $W \in \mathcal{U}$ and each sequence $\{x_i\}_{i=0}^{N_0} \subset X$ which satisfies $||x_0|| \leq K$ and (1.5) for all $i = 0, \ldots, N_0 - 1$, the inequality $f(x_{N_0}) \leq \inf(f) + \epsilon$ holds.*

(ii) *For each sequence of numbers $\mathbf{a} = \{a_i\}_{i=0}^{\infty} \subset (0, 1]$ satisfying (1.6), there exists a natural number N such that for each $W \in \mathcal{U}$ and each sequence $\{x_i\}_{i=0}^{N} \subset X$ which satisfies $||x_0|| \leq K$ and (1.7) for all $i = 0, \ldots, N-1$, the inequality $f(x_N) \leq \inf(f) + \epsilon$ holds.*

Comparing the present paper with [15], we note the following significant improvements.

In the present paper we show that the set of divergent descent methods in a Banach space is not only of the first category, but also σ-porous.

We point out a simple property of vector fields (namely, regularity) which yields convergence of the corresponding descent methods.

In contrast to [15], where the rather restrictive assumption A(iii) was used in all our results, in the present paper we establish convergence results for regular vector fields in $\mathcal{A}_{au}$ assuming only the mild assumptions A(i) and A(ii).

Our paper is organized as follows. The proof of Theorem 1 is given in section 3. It is preceded by an auxiliary result in section 2. The proofs of Theorems 2 and 3 are given in section 5. They are preceded by a basic lemma which is presented in section 4.

2. An auxiliary result. Assume that $\mathcal{K}$ is a nonempty closed convex subset of X. We consider the topological subspace $\mathcal{K} \subset X$ with the relative topology. For each function $h : \mathcal{K} \to R^1$ define $\inf(h) := \inf\{h(x) : \ x \in \mathcal{K}\}$.

PROPOSITION 2.1. *Let $g : \mathcal{K} \to R^1$ be a convex, bounded from below, function which is uniformly continuous on bounded subsets of $\mathcal{K}$. Assume that there exists a bounded convex set $\mathcal{K}_0 \subset \mathcal{K}$ such that for each $x \in \mathcal{K}$ there exists $y \in \mathcal{K}_0$ for which $g(y) \le g(x)$.*

Then there exists a continuous mapping $A_g : \mathcal{K} \to \mathcal{K}_0$ which satisfies $g(A_g x) \le g(x)$ for all $x \in \mathcal{K}$ and has the following two properties:

B(i) *For each integer $n \ge 1$, the mapping A_g is uniformly continuous on the set*

$$\{x \in \mathcal{K} : \ ||x|| \le n \text{ and } g(x) \ge \inf(g) + 1/n\};$$

B(ii) *if $g(x) \ge \inf(g) + \epsilon$ for some $\epsilon > 0$ and $x \in \mathcal{K}$, then*

$$g(A_g x) \le g(x) - \epsilon/2.$$

Proof. If there exists $x \in \mathcal{K}$ for which $g(x) = \inf(g)$, then there exists $x^* \in \mathcal{K}_0$ for which $g(x^*) = \inf(g)$ and we can set $A_g(y) = x^*$ for all $y \in \mathcal{K}$. Therefore we may assume that

$$\{x \in \mathcal{K} : \ g(x) = \inf(g)\} = \emptyset.$$

For each integer $i \ge 0$, there exists $y_i \in \mathcal{K}_0$ such that

$$g(y_i) \le (4(i+1))^{-1} + \inf(g). \tag{2.1}$$

Consider now the linear segments which join $y_0, y_1, \dots, y_n, \dots$ (all contained in $\mathcal{K}_0$ by the convexity of $\mathcal{K}_0$), represented as a continuous curve $\gamma : \ [0, \infty) \to \mathcal{K}_0$, and parametrized so that

$$\gamma(t) = y_i + (t - i)(y_{i+1} - y_i) \text{ if } i \le t < i + 1 \ (i = 0, 1, 2, \dots). \tag{2.2}$$

The curve γ is Lipschitzian because the set $\mathcal{K}_0$ is bounded. Define

$$A_g x = \gamma(g(x) - (\inf(g))^{-1}), \ x \in \mathcal{K}. \tag{2.3}$$

It is easy to see that $A_g x \in \mathcal{K}_0$ for all $x \in \mathcal{K}$, the mapping A_g is continuous on $\mathcal{K}$, and that it is uniformly continuous on the subsets

$$\{x \in \mathcal{K} : \ ||x|| \le n \text{ and } g(x) \ge \inf(g) + 1/n\}$$

for each integer $n \ge 1$.

Assume that

$$x \in \mathcal{K}, \ \epsilon > 0, \text{ and } g(x) \ge \inf(g) + \epsilon. \tag{2.4}$$

There is an integer $i \ge 0$ such that

$$g(x) - \inf(g) \in ((i+1)^{-1}, i^{-1}] \tag{2.5}$$

(we assume that $0^{-1} = \infty$). Then

$$(g(x) - \inf(g))^{-1} \in [i, i+1), \tag{2.6}$$

and, by (2.3), (2.2), and (2.6),

$$A_g x = \gamma(g(x) - (\inf(g))^{-1}) = y_i + ((g(x) - \inf(g))^{-1} - i)(y_{i+1} - y_i).$$

It follows from this relation, (2.1), (2.4), (2.5), and the convexity of g that

$$g(A_g x) \le \max\{g(y_i), g(y_{i+1})\} \le \inf(g) + (4(i+1))^{-1}$$

$$\le \inf(g) + 4^{-1}(g(x) - \inf(g)) = g(x) - 3 \cdot 4^{-1}(g(x) - \inf(g))$$

$$\le g(x) - 3 \cdot 4^{-1}\epsilon.$$

This completes the proof of Proposition 2.1. □

Note that in [15] we constructed a mapping A_g satisfying B(ii). Here we constructed a mapping which possesses both properties B(i) and B(ii). This will allow us to establish our results for the spaces $\mathcal{A}_{au}$ and $\mathcal{A}_{auc}$.

3. Proof of Theorem 1. We first note the following simple lemma.

LEMMA 3.1. *Assume that $V_1, V_2 \in \mathcal{A}$, $\phi : X \to [0,1]$, and that*

$$Vx = (1 - \phi(x))V_1 x + \phi(x)V_2 x,\ x \in X.$$

Then $V \in \mathcal{A}$. If $V_1, V_2 \in \mathcal{A}_c$ and ϕ is continuous on X, then $V \in \mathcal{A}_c$. If $V_1, V_2 \in \mathcal{A}_u$ (respectively, $\mathcal{A}_{au}$, $\mathcal{A}_{auc}$) and ϕ is uniformly continuous on bounded subsets of X, then $V \in \mathcal{A}_u$ (respectively, $\mathcal{A}_{au}$, $\mathcal{A}_{auc}$).

For each pair of integers $m, n \ge 1$, denote by Ω_{mn} the set of all $V \in \mathcal{A}$ such that

$$||Vx|| \le m \text{ for all } x \in X \text{ satisfying } ||x|| \le n + 1 \tag{3.1}$$

and

$$\sup\{l(Vx) :\ x \in X,\ ||x|| \le n,\ f(x) \ge \inf(f) + 1/n,\ l \in \partial f(x)\} = 0. \tag{3.2}$$

Clearly,

$$\cup_{m=1}^{\infty} \cup_{n=1}^{\infty} \Omega_{mn} = \mathcal{A} \setminus \mathcal{F}. \tag{3.3}$$

Therefore, to prove Theorem 1 it is sufficient to show that for each pair of integers $m, n \ge 1$, the set Ω_{mn} (respectively, $\Omega_{mn} \cap \mathcal{A}_c$, $\Omega_{mn} \cap \mathcal{A}_{au}$, $\Omega_{mn} \cap \mathcal{A}_{auc}$) is a porous subset of $\mathcal{A}$ (respectively, $\mathcal{A}_c$, $\mathcal{A}_{au}$, $\mathcal{A}_{auc}$) and, if f attains its minimum, then $\Omega_{mn} \cap \mathcal{A}_u$ is a porous subset of $\mathcal{A}_u$.

By assumption A(i), there is a bounded convex set $X_0 \subset X$ with the following property:

C(i) For each $x \in X$ there is $x_0 \in X_0$ such that $f(x_0) \le f(x)$. If f attains its minimum, then X_0 is a singleton.

By Proposition 2.1, there is a continuous mapping $A_f : X \to X$ such that

$$A_f(X) \subset X_0,\ f(A_f x) \le f(x) \text{ for all } x \in X \tag{3.4}$$

and which has the following two properties:

C(ii) If $x \in X$, $\epsilon > 0$, and $f(x) \ge \inf(f) + \epsilon$, then $f(A_f x) \le f(x) - \epsilon/2$;

C(iii) for any natural number n, the mapping A_f is uniformly continuous on the set

$$\{x \in X : \ ||x|| \le n \text{ and } f(x) \ge \inf(f) + 1/n\}.$$

Let $m, n \ge 1$ be integers. In what follows we will use the piecewise linear function $\phi : R^1 \to R^1$ defined by

$$\phi(x) = 1, \ x \in [-n, n], \ \phi(x) = 0, \ |x| \ge n + 1 \tag{3.5}$$

and

$$\phi(-n - 1 + t) = t, \ t \in [0, 1], \ \phi(n + t) = 1 - t, \ t \in [0, 1].$$

By assumption A(ii), there is $c_0 > 1$ such that

$$|f(x) - f(y)| \le c_0||x - y|| \tag{3.6}$$

for all $x, y \in X$ satisfying $||x||, ||y|| \le n + 2$. Choose $\alpha \in (0, 1)$ such that

$$\alpha c_0 2^{n+2} < (2n)^{-1}2^{-1}(1 - \alpha)(m + n + 2 + \sup\{||x|| : \ x \in X_0\})^{-1}. \tag{3.7}$$

Assume that $V \in \Omega_{mn}$ and $r \in (0, 1]$. Let

$$\gamma = 2^{-1}(1 - \alpha)r(m + n + 2 + \sup\{||x|| : \ x \in X_0\})^{-1} \tag{3.8}$$

and define $V_\gamma : \ X \to X$ by

$$V_\gamma x = (1 - \gamma\phi(||x||))Vx + \gamma\phi(||x||)(A_f x - x), \ x \in X. \tag{3.9}$$

By Lemma 3.1, $V_\gamma \in \mathcal{A}$ and, moreover, if $V \in \mathcal{A}_c$ (respectively, $\mathcal{A}_{au}$, $\mathcal{A}_{auc}$), then $V_\gamma \in \mathcal{A}_c$ (respectively, $\mathcal{A}_{au}$, $\mathcal{A}_{auc}$) and, if $V \in \mathcal{A}_u$ and f attains its minimum, then A_f is constant (see C(i)) and $V_\gamma \in \mathcal{A}_u$.

Next we estimate the distance $\rho(V_\gamma, V)$. It follows from (3.9) and the definition of ϕ (see (3.5)) that $V_\gamma x = Vx$ for all $x \in X$ satisfying $||x|| \ge n + 1$ and

$$\rho_i(V_\gamma, V) = \rho_{n+1}(V_\gamma, V) \text{ for all integers } i \ge n + 1.$$

Since $V \in \Omega_{mn}$, the above equality, when combined with (1.2), (1.1), (3.9), (3.5), and (3.4), yields

$$\rho(V_\gamma, V) \le \sum_{i=1}^{\infty} 2^{-i}\rho_i(V, V_\gamma) \le \rho_{n+1}(V, V_\gamma) \tag{3.10}$$

$$= \sup\{||Vx - V_\gamma x|| : \ x \in X, \ ||x|| \le n + 1\}$$

$$\le \sup\{\gamma\phi(||x||)(||Vx|| + ||A_f x - x||) : \ x \in X, \ ||x|| \le n + 1\}$$

$$\le \gamma(m + 1) + \gamma(n + 1) + \gamma \sup\{||x|| : \ x \in X_0\}.$$

Assume that $W \in \mathcal{A}$ with

$$\rho(W, V_\gamma) \le \alpha r. \tag{3.11}$$

By (3.11), (3.10), and (3.8),

$$\rho(W,V) \le \alpha r + \gamma(m+n+2+\sup\{||x||:\ x\in X_0\}) \le 2^{-1}(1+\alpha)r < r. \tag{3.12}$$

Assume now that

$$x\in X,\ ||x||\le n,\ f(x)\ge \inf(f)+1/n,\ \text{and } l\in\partial f(x). \tag{3.13}$$

The inequality (3.6) implies that

$$||l||_* \le c_0.$$

By (3.9), (3.13), the definition of ϕ (see (3.5)), and C(ii),

$$l(V_\gamma x) = l((1-\gamma\phi(||x||))Vx + \gamma\phi(||x||)(A_f x - x)) \tag{3.14}$$

$$\le \gamma\phi(||x||)l(A_f x - x) = \gamma l(A_f x - x) \le \gamma(f(A_f x) - f(x)) \le -\gamma(2n)^{-1}.$$

It follows from (3.13) and (1.1) that

$$||Wx - V_\gamma x|| \le \rho_n(W, V_\gamma). \tag{3.15}$$

By (3.11), (3.15), and the inequality $||l||_* \le c_0$, we have

$$2^{-n}\rho_n(W,V_\gamma)(1+\rho_n(W,V_\gamma))^{-1} \le \rho(W,V_\gamma) \le \alpha r, \tag{3.16}$$

$$\rho_n(W,V_\gamma)(1+\rho_n(W,V_\gamma))^{-1} \le 2^n\alpha r,$$

$$\rho_n(W,V_\gamma)(1-2^n\alpha r) \le 2^n\alpha r,\ ||Wx - V_\gamma x|| \le 2^n\alpha r(1-2^n\alpha r)^{-1},$$

and

$$|l(Wx) - l(V_\gamma x)| \le c_0 2^n\alpha r(1-2^n\alpha r)^{-1}. \tag{3.17}$$

By (3.17), (3.14), (3.8), and (3.7),

$$l(Wx) \le l(V_\gamma x) + c_0 2^n\alpha r(1-2^n\alpha r)^{-1}$$

$$\le -\gamma(2n)^{-1} + c_0 2^n\alpha r(1-2^n\alpha r)^{-1} = c_0 2^n\alpha r(1-2^n\alpha r)^{-1}$$

$$-(2n)^{-1}2^{-1}(1-\alpha)r(m+n+2+\sup\{||x||:\ x\in X_0\})^{-1}$$

$$\le -r[-c_0 2^n\alpha\cdot 2 + (2n)^{-1}2^{-1}(1-\alpha)(m+n+2+\sup\{||x||:\ x\in X_0\})^{-1}]$$

$$\le -2rc_0 2^n\alpha.$$

Thus

$$\{W\in\mathcal{A}:\ \rho(W,V_\gamma)\le\alpha r\}\cap\Omega_{mn} = \emptyset.$$

In view of Remark 1.1 and (3.12), we can conclude that Ω_{mn} is porous in $\mathcal{A}$, $\Omega_{mn}\cap\mathcal{A}_c$ is porous in $\mathcal{A}_c$, $\Omega_{mn}\cap\mathcal{A}_{au}$ is porous in $\mathcal{A}_{au}$, $\Omega_{mn}\cap\mathcal{A}_{auc}$ is porous in $\mathcal{A}_{auc}$, and if f attains its minimum, then $\Omega_{mn}\cap\mathcal{A}_u$ is porous in $\mathcal{A}_u$. This completes the proof of Theorem 1. □

4. Basic lemma. The following result is our key lemma. It improves upon [15, Lemma 3.4] which is concerned only with condition 2 of the present lemma and with regular mappings of a special type.

LEMMA 4.1. *Assume that $V \in \mathcal{A}$ is regular, A(i), A(ii) are valid, and that at least one of the following conditions holds:* 1. $V \in \mathcal{A}_{au}$; 2. *A(iii) is valid.*

Let $\bar{K}$ and $\bar{\epsilon}$ be positive. Then there exist a neighborhood $\mathcal{U}$ of V in $\mathcal{A}$ and positive numbers $\bar{\alpha}$ and γ such that for each $W \in \mathcal{U}$, each $x \in X$ satisfying

$$||x|| \le \bar{K},\ f(x) \ge \inf(f) + \bar{\epsilon}, \tag{4.1}$$

and each $\beta \in (0, \bar{\alpha}]$,

$$f(x) - f(x + \beta W x) \ge \beta\gamma. \tag{4.2}$$

Proof. There exists $K_0 > \bar{K} + 1$ such that

$$||Vx|| \le K_0 \text{ if } x \in X \text{ and } ||x|| \le \bar{K} + 2. \tag{4.3}$$

By assumption A(ii), there exists a constant $L_0 > 4$ such that

$$|f(x_1) - f(x_2)| \le L_0||x_1 - x_2|| \tag{4.4}$$

for all $x_1, x_2 \in X$ satisfying $||x_1||, ||x_2|| \le 2K_0 + 4$. Since V is regular, there exists a positive number $\delta_0 \in (0, 1)$ such that

$$\xi(Vy) \le -\delta_0 \tag{4.5}$$

for each $y \in X$ satisfying $||y|| \le K_0 + 4$, $f(y) \ge \inf(f) + \bar{\epsilon}/4$, and each $\xi \in \partial f(y)$. Choose $\delta_1 \in (0, 1)$ such that

$$4\delta_1(K_0 + L_0) < \delta_0. \tag{4.6}$$

There exists a positive number $\bar{\alpha}$ such that the following conditions hold:

$$8\bar{\alpha}(L_0 + 1)(K_0 + 1) < \min\{1, \bar{\epsilon}\}; \tag{4.7}$$

(a) if $V \in \mathcal{A}_{au}$, then for each $x_1, x_2 \in X$ satisfying

$$||x_1||, ||x_2|| \le \bar{K} + 4,\ \min\{f(x_1), f(x_2)\} \ge \inf(f) + \bar{\epsilon}/4, \tag{4.8}$$

$$\text{and } ||x_1 - x_2|| \le \bar{\alpha}(K_0 + 1),$$

the following inequality is true:

$$||Vx_1 - Vx_2|| \le \delta_1; \tag{4.9}$$

(b) if A(iii) is valid, then for each $x_1, x_2 \in X$ satisfying (4.8) the following inequality is true:

$$H(\partial f(x_1), \partial f(x_2)) < \delta_1. \tag{4.10}$$

Next choose a positive number δ_2 such that

$$8\delta_2(L_0 + 1) < \delta_1\delta_0. \tag{4.11}$$

Now choose a positive number γ such that

$$\gamma < \delta_0/8 \tag{4.12}$$

and define

$$\mathcal{U} = \{W \in \mathcal{A} : \ ||Wx - Vx|| \le \delta_2, \ x \in X, \ \text{and} \ ||x|| \le \bar{K}\}. \tag{4.13}$$

Assume that $W \in \mathcal{U}$, $x \in X$ satisfies (4.1), and $\beta \in (0, \bar{\alpha}]$. We intend to show that (4.2) holds. To this end, we first note that (4.1), (4.3), (4.7), (4.13), and (4.11) yield

$$||x + \beta Vx|| \le \bar{K} + \beta K_0 \le \bar{K} + \bar{\alpha} K_0 \le \bar{K} + 1$$

and

$$||x + \beta Wx|| \le \delta_2 \beta + ||x + \beta Vx|| \le \bar{K} + 1 + \bar{\alpha}\delta_2 \le \bar{K} + 2.$$

By these inequalities and the definition of L_0 (see (4.4)) and (4.13),

$$|f(x + \beta Vx) - f(x + \beta Wx)| \le L_0 \beta ||Wx - Vx|| \le L_0 \beta \delta_2. \tag{4.14}$$

Next we will estimate $f(x) - f(x + \beta Vx)$. There exist $\theta \in [0, \beta]$ and $l \in \partial f(x + \theta Vx)$ such that

$$f(x + \beta Vx) - f(x) = l(Vx)\beta. \tag{4.15}$$

By (4.1), (4.3), and (4.7),

$$||x|| \le \bar{K}, \ ||Vx|| \le K_0, \ ||\theta Vx|| \le \bar{\alpha} K_0, \ \text{and} \ ||x + \theta Vx|| \le \bar{K} + 1. \tag{4.16}$$

It follows from (4.16) and the definition of L_0 (see (4.4)) that

$$||l||_* \le L_0. \tag{4.17}$$

It follows from (4.16), the definition of L_0 (see (4.4)), (4.7), and (4.1) that

$$f(x + \theta Vx) \ge f(x) - L_0 ||\theta Vx|| \tag{4.18}$$

$$\ge f(x) - L_0 \bar{\alpha} K_0 \ge f(x) - 8^{-1}\bar{\epsilon} \ge \inf(f) + \bar{\epsilon}/2.$$

Consider the case where $V \in \mathcal{A}_{au}$. By (4.17), condition (a), (4.16), (4.1), and (4.18),

$$\beta l(Vx) \le \beta l(V(x + \theta Vx)) + \beta ||l||_* (||V(x + \theta Vx) - Vx||) \tag{4.19}$$

$$\le \beta l(V(x + \theta Vx)) + \beta L_0 ||V(x + \theta Vx) - Vx||$$

$$\le \beta l(V(x + \theta Vx)) + \beta L_0 \delta_1.$$

By (4.16), (4.18), and the definition of δ_0 (see (4.5)),

$$l(V(x + \theta Vx)) \le -\delta_0.$$

Combined with (4.19) and (4.6) this inequality implies that

$$\beta l(Vx) \le -\beta\delta_0 + \beta L_0\delta_1 \le -\beta\delta_0/2.$$

By these inequalities and (4.15),

$$f(x+\beta Vx) - f(x) \le -\beta\delta_0/2. \tag{4.20}$$

Assume now that A(iii) is valid. It then follows from condition (b), (4.16), (4.1), and (4.18) that

$$H(\partial f(x), \partial f(x+\theta Vx)) < \delta_1.$$

Therefore, there exists $\bar{l} \in \partial f(x)$ such that $||\bar{l} - l||_* \le \delta_1$. Combined with (4.15) and (4.16) this fact implies that

$$f(x+\beta Vx) - f(x) = \beta l(Vx) \le \beta\bar{l}(Vx) \tag{4.21}$$

$$+ \beta||\bar{l} - l||_*||Vx|| \le \beta\bar{l}(Vx) + \beta\delta_1 K_0.$$

It follows from the definition of δ_0 (see (4.5)) and (4.1) that $\beta\bar{l}(Vx) \le -\beta\delta_0$. Combining this inequality with (4.21) and (4.6), we see that

$$f(x+\beta Vx) - f(x) \le -\beta\delta_0 + \beta\delta_1 K_0 \le -\beta\delta_0/2.$$

Thus in both cases (4.20) is true. It now follows from (4.20), (4.14), (4.11), and (4.12) that

$$f(x+\beta Wx) - f(x) \le f(x+\beta Vx) - f(x) + f(x+\beta Wx) - f(x+\beta Vx)$$

$$\le -\beta\delta_0/2 + L_0\beta\delta_2 \le -\beta\delta_0/4 \le -\gamma\beta.$$

Thus (4.2) holds. Lemma 4.1 is proved. □

5. Proofs of Theorems 2 and 3. Parts of the following proofs are somewhat similar to parts of the proofs of [15, Theorems 1.1 and 1.2]. However, in the present paper we strongly rely on the new lemma, Lemma 4.1.

Proof of Theorem 2. To show that assertion (i) holds, suppose that

$$\{x_i\}_{i=0}^{\infty} \subset X,\ x_{i+1} \in P_W x_i,\ i = 0, 1, \ldots, \text{ and } \liminf_{i\to\infty} ||x_i|| < \infty. \tag{5.1}$$

We will show that

$$\lim_{i\to\infty} f(x_i) = \inf(f). \tag{5.2}$$

Assume the contrary. Then there exists $\epsilon > 0$ such that

$$f(x_i) \ge \inf(f) + \epsilon,\ i = 0, 1, \ldots. \tag{5.3}$$

There exists a number $S > 0$ and a strictly increasing sequence of natural numbers $\{i_k\}_{k=1}^{\infty}$ such that

$$||x_{i_k}|| \le S,\ k = 1, 2, \ldots. \tag{5.4}$$

By Lemma 4.1, there exist numbers $\alpha, \gamma \in (0,1)$ such that for each $x \in X$ satisfying

$$||x|| \leq S,\ f(x) \geq \inf(f) + \epsilon, \tag{5.5}$$

and each $\beta \in (0, \alpha]$,

$$f(x) - f(x + \beta W x) \geq \gamma\beta. \tag{5.6}$$

It follows from (5.1), (1.4), (1.5), the definitions of α and γ, (5.4), and (5.3) that for each integer $k \geq 1$,

$$f(x_{i_k}) - f(x_{i_k+1}) \geq f(x_{i_k}) - f(x_{i_k} + \alpha W x_{i_k}) \geq \gamma\alpha.$$

Since this inequality holds for all integers $k \geq 1$, we conclude that

$$\lim_{n\to\infty} (f(x_0) - f(x_n)) = \infty.$$

This contradicts our assumption that f is bounded from below. Therefore, (5.2) and assertion (i) are indeed true, as claimed.

We turn now to assertion (ii). Let $\mathbf{a} = \{a_i\}_{i=0}^{\infty} \subset (0,1]$ satisfy (1.6) and let a bounded $\{x_i\}_{i=0}^{\infty} \subset X$ satisfy (1.7) for all integers $i \geq 0$. We will show that (5.2) holds. Indeed, assume that (5.2) is not true. Then there exists $\epsilon > 0$ such that (5.3) holds. Since the sequence $\{x_i\}_{1=0}^{\infty}$ is bounded, there exists a number $S > 0$ such that

$$S > ||x_i||,\ i = 0, 1, \ldots. \tag{5.7}$$

By Lemma 4.1, there exist numbers $\alpha, \gamma \in (0,1)$ such that for each $x \in X$ satisfying (5.5) and each $\beta \in (0, \alpha]$, the inequality (5.6) holds. Since $a_i \to 0$ as $i \to \infty$, there exists a natural number i_0 such that

$$a_i < \alpha \text{ for all integers } i \geq i_0. \tag{5.8}$$

Let $i \geq i_0$ be an integer. Then it follows from (5.7), (5.3), the definitions of α and γ, and (5.8) that

$$f(x_i) - f(x_i + a_i W x_i) \geq \gamma a_i,\ x_{i+1} = x_i + a_i W x_i,$$

and

$$f(x_i) - f(x_{i+1}) \geq \gamma a_i.$$

Since $\sum_{i=0}^{\infty} a_i = \infty$, we conclude that

$$\lim_{n\to\infty} (f(x_0) - f(x_n)) = \infty.$$

The contradiction we have reached shows that (5.2), assertion (ii), and Theorem 2 are all true. □

Proof of Theorem 3. Let

$$K_0 > \sup\{f(x) :\ x \in X, ||x|| \leq K + 1\} \tag{5.9}$$

and set

$$E_0 = \{x \in X :\ f(x) \leq K_0 + 1\}. \tag{5.10}$$

Clearly, E_0 is bounded and closed. Choose

$$K_1 > \sup\{||x|| : \ x \in E_0\} + 1 + K. \tag{5.11}$$

By Lemma 4.1, there exist a neighborhood $\mathcal{U}$ of V in $\mathcal{A}$ and numbers $\alpha, \gamma \in (0,1)$ such that for each $W \in \mathcal{U}$, each $x \in X$ satisfying

$$||x|| \le K_1, \ f(x) \ge \inf(f) + \epsilon, \tag{5.12}$$

and each $\beta \in (0, \alpha]$,

$$f(x) - f(x + \beta W x) \ge \gamma\beta. \tag{5.13}$$

Now choose a natural number N_0 which satisfies

$$N_0 > (\alpha\gamma)^{-1}(K_0 + 4 + |\inf(f)|). \tag{5.14}$$

First we will show that assertion (i) is true. Assume that $W \in \mathcal{U}$, $\{x_i\}_{i=0}^{N_0} \subset X$,

$$||x_0|| \le K, \text{ and } x_{i+1} \in P_W x_i, \ i = 0, \dots, N_0 - 1. \tag{5.15}$$

Our aim is to show that

$$f(x_{N_0}) \le \inf(f) + \epsilon. \tag{5.16}$$

Assume that (5.16) is not true. Then

$$f(x_i) > \inf(f) + \epsilon, \ i = 0, \dots, N_0. \tag{5.17}$$

By (5.15) and (5.9)–(5.11) we also have

$$||x_i|| \le K_1, \ i = 0, \dots, N_0. \tag{5.18}$$

Let $i \in \{0, \dots, N_0 - 1\}$. It follows from (5.18), (5.17), and the definitions of $\mathcal{U}$, α, and γ (see (5.12) and (5.13)) that

$$f(x_i) - f(x_{i+1}) \ge f(x_i) - f(x_i + \alpha W x_i) \ge \gamma\alpha.$$

Summing up from $i = 0$ to $N_0 - 1$ we conclude that

$$f(x_0) - f(x_{N_0}) \ge N_0\gamma\alpha.$$

It follows from this inequality, (5.9), (5.14), and (5.15) that

$$\inf(f) \le f(x_{N_0}) \le f(x_0) - N_0\gamma\alpha \le K_0 - N_0\gamma\alpha \le -4 - |\inf(f)|.$$

Since we have reached a contradiction, we see that (5.16) must be true and assertion (i) is proved.

Now we will show that assertion (ii) is also valid. To this end, let a sequence $\mathbf{a} = \{a_i\}_{i=0}^{\infty} \subset (0,1]$ satisfy

$$\lim_{i\to\infty} a_i = 0 \text{ and } \sum_{i=0}^{\infty} a_i = \infty. \tag{5.19}$$

Clearly, there exists a natural number N_1 such that

$$a_i \le \alpha \text{ for all } i \ge N_1. \tag{5.20}$$

Choose a natural number $N > N_1 + 4$ such that

$$\gamma \sum_{i=N_1}^{N-1} a_i > K_0 + 4 + |\inf(f)|. \tag{5.21}$$

Now assume that $W \in \mathcal{U}$, $\{x_i\}_{i=0}^{N} \subset X$, $||x_0|| \le K$, and that (1.7) holds for all $i = 0, \dots, N-1$. We will show that

$$f(x_N) \le \inf(f) + \epsilon. \tag{5.22}$$

Assume the contrary. Then

$$f(x_i) > \inf(f) + \epsilon,\ i = 0, \dots, N. \tag{5.23}$$

Since $||x_0|| \le K$, we see by (1.7) and (5.9)–(5.11) that

$$||x_i|| \le K_1,\ i = 0, \dots, N. \tag{5.24}$$

Let $i \in \{N_1, \dots, N-1\}$. It follows from (5.24), (5.23), (5.20), and the definitions of α and γ (see (5.12) and (5.13)) that

$$f(x_i) - f(x_i + a_i W x_i) \ge \gamma a_i.$$

This implies that

$$f(x_{N_1}) - f(x_N) \ge \gamma \sum_{i=N_1}^{N-1} a_i.$$

By this inequality, (1.7), the inequality $||x_0|| \le K$, (5.9), and (5.21), we obtain

$$\inf(f) \le f(x_N) \le f(x_{N_1}) - \gamma \sum_{i=N_1}^{N-1} a_i$$

$$\le K_0 - \gamma \sum_{i=N_1}^{N-1} a_i < -4 - |\inf(f)|.$$

The contradiction we have reached proves (5.22) and assertion (ii). This completes the proof of Theorem 3. □

Acknowledgments. Both authors are grateful to two anonymous referees for many helpful comments and to J. Revalski for useful discussions.

REFERENCES

[1] Y.I Alber, A.N. Iusem, and M.V. Solodov, *Minimization of nonsmooth convex functionals in Banach spaces*, J. Convex Anal., 4 (1997), pp. 235–255.

[2] A. Ben-Tal and M. Zibulevsky, *Penalty/barrier multiplier methods for convex programming problems*, SIAM J. Optim., 7 (1997), pp. 347–366.

[3] Y. Benyamini and J. Lindenstrauss, *Geometric Nonlinear Functional Analysis*, AMS, Providence, RI, 2000.
[4] R. Correa and C. Lemaréchal, *Convergence of some algorithms for convex minimization*, Math. Programming, 62 (1993), pp. 261–275.
[5] F.S. De Blasi and J. Myjak, *Generic flows generated by continuous vector fields in Banach spaces*, Adv. in Math., 50 (1983), pp. 266–280.
[6] F.S. De Blasi and J. Myjak, *On a generalized best approximation problem*, J. Approx. Theory, 94 (1998), pp. 54–72.
[7] R. Deville and J. Revalski, *Porosity of ill-posed problems*, Proc. Amer. Math. Soc., 128 (2000), pp. 1117–1124.
[8] A.L. Dontchev and T. Zolezzi, *Well-Posed Optimization Problems*, Lecture Notes in Math. 1543, Springer, Berlin, 1993.
[9] M.S. Gowda and M. Teboulle, *A comparison of constraint qualifications in infinite-dimensional convex programming*, SIAM J. Control Optim., 28 (1990), pp. 925–935.
[10] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms*, Springer, Berlin, 1993.
[11] A.D. Ioffe and A.J. Zaslavski, *Variational principles and well-posedness in optimization and calculus of variations*, SIAM J. Control Optim., 38 (2000), pp. 566–581.
[12] D. Preiss and L. Zajicek, *Fréchet differentiation of convex functions in a Banach space with a separable dual*, Proc. Amer. Math. Soc., 91 (1984), pp. 202–204.
[13] S. Reich and A.J. Zaslavski, *Convergence of generic infinite products of nonexpansive and uniformly continuous operators*, Nonlinear Anal., 36 (1999), pp. 1049–1065.
[14] S. Reich and A.J. Zaslavski, *On the minimization of convex functionals*, in Calculus of Variations and Differential Equations, Res. Notes Math. Ser. 410, CRC Press, Boca Raton, FL, 1999, pp. 200–209.
[15] S. Reich and A.J. Zaslavski, *Generic convergence of descent methods in Banach spaces*, Math. Oper. Res., 25 (2000), pp. 231–242.
[16] J. Revalski, *Generic properties concerning well-posed optimization problems*, C. R. Acad. Bulg. Sci., 38 (1985), pp. 1431–1434.
[17] L. Zajicek, *Porosity and σ-porosity*, Real Anal. Exchange, 13 (1987), pp. 314–350.
[18] L. Zajicek, *Products of non-σ-porous sets and Foran systems*, Atti Sem. Mat. Fis. Univ. Modena, 44 (1996), pp. 497–505.
[19] L. Zajicek, *Small non-σ-porous sets in topologically complete metric spaces*, Colloq. Math., 77 (1998), pp. 293–304.
[20] A.J. Zaslavski, *Optimal programs on infinite horizon* 1 *and* 2, SIAM J. Control Optim., 33 (1995), pp. 1643–1686.
[21] A.J. Zaslavski, *Dynamic properties of optimal solutions of variational problems*, Nonlinear Anal., 27 (1996), pp. 895–932.
[22] A.J. Zaslavski, *Existence of solutions of optimal control problems for a generic integrand without convexity assumptions*, Nonlinear Anal., 43 (2001), pp. 339–361.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1019–1036

FATCOP: A FAULT TOLERANT CONDOR-PVM MIXED INTEGER PROGRAMMING SOLVER*

QUN CHEN† AND MICHAEL C. FERRIS‡

Abstract. We describe FATCOP, a new parallel mixed integer program solver written in PVM. The implementation uses the Condor resource management system to provide a virtual machine composed of otherwise idle computers. The solver differs from previous parallel branch-and-bound codes by implementing a general purpose parallel mixed integer programming algorithm in an opportunistic multiple processor environment, as opposed to a conventional dedicated environment. It shows how to make effective use of resources as they become available while ensuring the program tolerates resource retreat. The solver performs well on test problems arising from real applications and is particularly useful for solving long running hard mixed integer programming problems.

Key words. mixed integer programming, Condor, opportunistic environment, branch-and-bound, fault tolerance

AMS subject classifications. 90C11, 65K05

PII. S1052623499353911

1. Introduction.

Mixed integer programming (MIP) problems are difficult and commonplace. For many of these hard problems, only small instances can be solved in a reasonable amount of time on sequential computers, resulting in mixed integer programming being a frequently cited application of parallel computing. Most available general-purpose large-scale MIP codes use branch-and-bound to search for an optimal integer solution by solving a sequence of related linear programming (LP) relaxations that allow possible fractional values. This paper discusses a new parallel mixed integer program solver, written in PVM, that runs in the opportunistic computing environment provided by the Condor resource management system.

Parallel branch-and-bound algorithms for MIP have attracted many researchers (see [11, 15, 24] and references therein). Most parallel branch-and-bound programs were developed for large centralized mainframes or supercomputers that are typically very expensive. Users of these facilities usually have only a certain amount of time allotted to them and have to wait their turn to run their jobs. Due to the decreasing cost of lower-end workstations, large heterogeneous clusters of workstations connected through fast local networks are becoming common in workplaces such as universities and research institutions. In this paper we shall refer to the former resources as dedicated resources and the latter as distributed ownership resources. The principal goal of the research outlined in this paper is to exploit distributed ownership resources to solve extremely difficult mathematical programming problems. We believe that environments of this type are important for the future of certain numerical computations, including computations for discrete optimization problems. In fact, during the time that this paper was under review, several other authors had adapted

*Received by the editors March 17, 1999; accepted for publication (in revised form) December 6, 2000; published electronically May 10, 2001. This material is based on research supported in part by National Science Foundation grants CDA-9726385, CCR-9619765, and CCR-9972372 and by Air Force Office of Scientific Research grant F49620-98-1-0417.

http://www.siam.org/journals/siopt/11-4/35391.html

†Portland Development Center, Oracle Corporation, 1211 SW 5th Avenue, Portland, OR 97204 (qun.chen@oracle.com).

‡Computer Sciences Department, University of Wisconsin, Madison, WI 53706 (ferris@cs.wisc.edu).

the scheme first outlined in this work for other branch-and-bound applications, most notably in [1, 16, 27].

A parallel virtual machine (PVM) is a programming environment that allows a heterogeneous network of computers to appear as a single concurrent computational resource [14]. It provides a unified framework within which parallel programs for a heterogeneous collection of machines can be developed in an efficient manner. However, PVM is not sufficient to develop an efficient parallel branch-and-bound program in a distributed ownership environment. The machines in such an environment are usually dedicated to the exclusive use of individuals. The application programming interface defined by PVM requires that users explicitly select machines on which to run their programs. Therefore, they must have permission to access the selected machines and cannot be expected to know the load on the machines in advance. Furthermore, when a machine is claimed by a PVM program, the required resources in the machine will be "occupied" during the life cycle of the program. This is not a desirable situation when the machine is owned by a person different from the user of the MIP solver.

Condor [12, 20] is a distributed resource management system that can help to overcome these problems. Condor manages large heterogeneous clusters of machines in an attempt to use the idle cycles of some users' machines to satisfy the needs of others who have computing intensive jobs. It was first developed for long running sequential batch jobs. The current version of Condor provides a framework (Condor-PVM) to run parallel programs written in PVM in a distributed ownership environment. In such programs, Condor is used to dynamically construct a PVM out of nondedicated desktop machines on the network. Condor allows users' programs to run on any machine in the pool of machines managed by Condor, regardless of whether or not the user submitting the job has an account there, and guarantees that heavily loaded machines will not be selected for an application. To protect ownership rights, whenever a machine's owner returns, Condor immediately interrupts any job that is running on that machine, migrating the job to another idle machine. Since owners and many other Condor users compete for resources managed by Condor we refer to such resources as Condor's *opportunistic resources* and the Condor-PVM parallel programming environment as the Condor-PVM *opportunistic environment*.

FATCOP represents a first attempt to develop a general purpose parallel solver for mixed integer programs in Condor's opportunistic environment. It is hoped that many of the lessons learned in developing FATCOP can be incorporated into more general branch-and-bound codes for other applications. FATCOP is implemented on top of both SOPLEX, a public available simplex object-oriented linear programming solver [28], and the CPLEX LP solver [9]. FATCOP is written in the C++ programming language with calls to the PVM library. It is designed to make best use of participating resources managed by Condor while handling resource retreat carefully in order to ensure the eventual and correct completion of a FATCOP job. Key features of FATCOP include

- parallel implementation under Condor-PVM framework;
- greedy utilization of Condor's opportunistic resources;
- powerful MIP techniques including strong branching, pseudocost estimation searching, preprocessing, and cutting plane generation;
- the ability to process both MPS [21] and GAMS [7] models;
- the use of both CPLEX and SOPLEX as its LP solver.

The remainder of this paper is organized as follows. Section 2 is a review of the standard MIP algorithm components of FATCOP that are implemented to ensure

that the branch-and-bound algorithm generates reasonable search trees. Section 3 introduces a Condor-PVM parallel programming framework and the parallel implementation of FATCOP. In section 4, we present some numerical results that exhibit important features of FATCOP. A brief summary and future directions are given in section 5.

2. Components of the sequential program. An MIP can be stated mathematically as follows:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax \le b, \\ & l \le x \le u, \\ & x_j \in Z \qquad \forall j \in I. \end{array}$$

Here Z denotes the integers, A is an $m \times n$ matrix, and I is a subset of the indices identifying the integer variables.

Integer programming textbooks such as [22] describe the fundamental branch-and-bound algorithm for the above MIP problem. Basically, the method explores a binary tree of subproblems. Branching refers to the process of creating refinements of the current relaxation, while bounding of the LP solution is used to eliminate exploration of parts of the tree. The remainder of this section describes refinements to this basic framework.

2.1. Preprocessing. Preprocessing refers to a set of reformulations performed on a problem instance. In LP this typically leads to problem size reductions. FATCOP identifies infeasibilities and redundancies, tightens bounds on variables, and improves the coefficients of constraints [25]. At the root node, FATCOP analyzes every row of the constraint matrix. If, after processing, some variables are fixed or some bounds are improved, the process is repeated until no further model reduction occurs.

In contrast to LP, preprocessing may reduce the *integrality gap*, i.e., the difference between the optimal solution value and its LP relaxation as well as the size of an MIP problem. For example, for the model *p0548* from MIPLIB [5], an electronically available library of both pure and mixed integer programs arising from real applications, the FATCOP preprocessor can remove only 12 rows and 16 columns and modify 176 coefficients from the original model that has 176 rows, 548 columns, and 1711 nonzero coefficients, but pushes the optimal value of the initial LP relaxation from 315.29 up to 3125.92.

2.2. Cutting planes and reduced cost fixing. It is well known that cutting planes can strengthen MIP formulations [4]. FATCOP generates knapsack cuts at each subproblem as described in [18]. There are about 10 models in MIPLIB for which knapsack cuts are useful. We again take *p0548* as an example; the FATCOP code can solve the model in 350 nodes with knapsack cuts applied at each node. However, it is not able to solve the problem to optimality in 100,000 nodes without knapsack cuts.

FATCOP also incorporates a standard reduced cost fixing procedure [11] that fixes integer variables to their upper or lower bounds by comparing their reduced costs to the gap between a linear programming solution value and the current problem best upper bound.

2.3. Variable and node selection. Several reasonable criteria exist for selecting branching variables. FATCOP currently provides four variable selection options:

pseudocost [19], strong branching [6], and maximum and minimum integer infeasibility [2]. Since the pseudocost method is widely used and known to be efficient, we set it as the default branching strategy. FATCOP can also accept user defined priorities on integer variables.

FATCOP provides five options for selecting a node from those remaining: depth-first, best-bound, best-estimation [2], a mixed strategy of depth-first and best-bound [11] (mixed strategy 1), and a mixed strategy of best-estimation and best-bound (mixed strategy 2).

Mixed strategy 1 expands the subproblems in the best-first order, but with an initial depth-first phase. FATCOP keeps track of the number of node evaluations over which the best integer solution has not been updated. It then switches searching strategy from depth-first to best-first after this number exceeds a prespecified fixed number. Mixed strategy 2 is similar to mixed strategy 1, but starts the algorithm with best-estimation search first. Since best-estimation often finds better solutions than depth-first does, mixed strategy 2 is set as the default searching strategy for FATCOP.

3. Condor-PVM parallel implementation of FATCOP. In this section we first give a brief overview of Condor, PVM, and the Condor-PVM parallel programming environment. Then we discuss the parallel scheme we selected for FATCOP and the differences between normal PVM and Condor-PVM programming. At the end of the section, we present a detailed implementation of FATCOP.

3.1. The Condor-PVM parallel programming environment. Heterogeneous clusters of workstations are becoming an important source of computing resources. Two approaches have been proposed to make effective use of such resources. One approach provides efficient resource management by allowing users to run their jobs on idle machines that belong to somebody else. Condor, developed at the University of Wisconsin-Madison, is one such system. It monitors the activity on all participating machines, placing idle machines in the Condor pool. Machines are then allocated from the pool when users send job requests to Condor. Machines enter the pool when they become idle, and leave when they get busy, e.g., the machine owner returns. When an executing machine becomes busy, the job running on this machine is initially suspended in case the executing machine becomes idle again within a short timeout period. If the executing machine remains busy, then the job is migrated to another idle workstation in the pool or returned to the job queue. For a job to be restarted after migration to another machine, a checkpoint file is generated that allows the exact state of the process to be re-created. This design feature ensures the eventual completion of a job. There are various priority orderings used by Condor for determining which jobs and machines are matched at any given instance. Based on these orderings, running jobs may sometimes be preempted to allow higher priority jobs to run. Condor is freely available and has been used in a wide range of production environments for more than ten years.

Another approach to exploit the power of a workstation cluster is from the perspective of parallel programming. Research in this area has developed message passing environments allowing people to solve a single problem in parallel using multiple resources. One of the most widely used message passing environments is PVM, which was developed at the Oak Ridge National Laboratory. PVM's design centers around the idea of a *virtual machine*, a very general notion that can encompass a nearly arbitrary collection of computing resources, from desktop workstations to multiprocessors to massively parallel homogeneous supercomputers. The goal of PVM is to

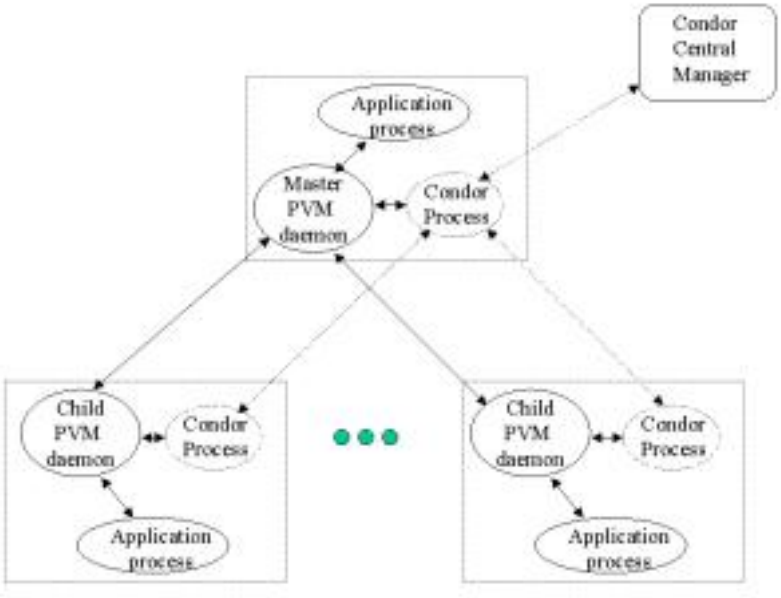


FIG. 3.1. *Architecture of Condor-PVM.*

make programming straightforward for a heterogeneous collection of machines. PVM provides process control and resource management functions that allow spawning and termination of arbitrary processes and the addition and deletion of hosts at run time. The PVM system is composed of two parts. The first part is a daemon that resides on all the computers comprising the virtual machine. The second part of the system is the PVM library. It contains user-callable routines for message passing, process spawning, virtual machine modification, and task coordination. PVM transparently handles all message routing, data conversion, and task scheduling across a network of incompatible computer architectures. A similar environment is the message-passing interface (MPI) [17]. Both systems center around a message-passing model, providing point-to-point as well as collective communication between distributed processes.

The development of resource management systems and message-passing environments have been independent of each other for many years. Researchers at the University of Wisconsin have recently developed a parallel programming framework that interfaces Condor and PVM [23]. The reason to select PVM instead of MPI is that the implementation of MPI has no concept of process control, and hence cannot handle resource addition and retreat in an opportunistic environment. Figure 3.1 shows the architecture of Condor-PVM. There are three processes on each machine running a Condor-PVM application: the PVM daemon, the Condor process, and the user application process. The Condor-PVM framework still relies on the PVM primitives for application communication, but provides resource management in the opportunistic environment through Condor. Each PVM daemon has a Condor process associated with it, acting as the *resource manager*. The Condor process interacts with the PVM daemon to start tasks, send signals to suspend, resume and kill tasks, and receive process completion information. The Condor process running on the master machine

is special. It communicates with Condor processes running on the other machines, keeps information about the status of the machines, and forwards resource requests to the Condor central manager. This Condor process is called the *global resource manager*. When a Condor-PVM application asks for a host (we will use host and machine interchangeably in what follows), the global resource manager communicates with the Condor central manager to schedule a new machine. After Condor grants a machine to the application, it starts a Condor process (resource manager) and a PVM daemon on the new machine. If a machine needs to leave the pool, the resource manager will send signals to the PVM daemon to suspend tasks. The master user application is notified of that via normal PVM notification mechanisms.

Compared with a conventional dedicated environment, the Condor-PVM opportunistic environment has the following characteristics:

1. There is usually a large amount of heterogeneous resources available for an application, but in each time instance, the amount of available resources is random, dependent on the status of machines managed by Condor. Owners and other Condor users compete for the resources.
2. Resources used by an application may disappear during the life cycle of the application.
3. The execution order of components in a particular application is highly nondeterministic, leading to different solution and execution times.

Therefore a good Condor-PVM application should be tolerant to loss of resources (host suspension and deletion) and dynamically adaptive to the current status of the Condor pool in order to make effective use of opportunistic resources.

3.2. The parallel scheme for FATCOP. FATCOP introduces parallelism when building the branch-and-bound tree. It simultaneously performs bounding operations on several subproblems. This approach may affect the order of subproblems generated during the expansion of the branch-and-bound tree. Hence more or less subproblems could be evaluated by the parallel program than with its sequential version. Such phenomena are known as *search anomalies* and examples are given in section 4.

FATCOP was designed in the master-worker paradigm. One host, called the master, manages the work pool and sends subproblems out to other hosts, called workers, that solve LPs and send the results back to the master. When using a large number of workers, this centralized parallel scheme can become a bottleneck in processing the returned information, thus keeping workers idle for large amounts of time. However, this scheme can handle different kinds of resource failure well in Condor's opportunistic environment and thus achieve the best degree of fault tolerance. The basic idea is that the master keeps track of which subproblem has been sent to each worker and does not actually remove the subproblem from the work pool. All the subproblems that are sent out are marked as "in progress by worker i." If the master is then informed that a worker has disappeared, it simply unmarks the subproblems assigned to that worker.

The remaining design issue is how to use the opportunistic resources provided by Condor to adapt to changes in the number of available resources. The changes include newly available machines, machine suspension and resumption, and machine failure. In a conventional dedicated environment, a parallel application usually is developed for running with a fixed number of processors and the solution process will not be started until the required number of processors is obtained and initialized. In Condor's opportunistic environment, doing so may cause a serious delay. In fact the time to

obtain the required number of new hosts from the Condor pool can be unbounded. Therefore we implement FATCOP in such a way that the solution process starts as soon as it obtains a single host. The solver then attempts to acquire new hosts as often as possible. At the beginning of the program, FATCOP places a number of requests for new hosts from Condor. Whenever it gets a host, it then allocates work to this host and immediately requests a new host. Thus, in each period between when Condor assigns a machine to FATCOP and when the new host request is received by Condor, there is at least one "new host" request from FATCOP waiting to be processed by Condor. This greedy implementation makes it possible for a FATCOP job to collect a significant amount of hosts during its life cycle.

3.3. Differences between PVM and Condor-PVM programming. PVM and Condor-PVM are binary compatible with each other. However, there exist some run time differences between PVM and Condor-PVM. The most important difference is the concept of machine class. In a regular PVM application, the configuration of hosts that PVM combines into a virtual machine usually is defined in a file in which host names have to be explicitly given. Under the Condor-PVM framework, Condor selects the machines on which a job will run, so the dependency on host names must be removed from an application. Instead, the applications must use class names. Machines of different architecture attributes belong to different machine classes. Machine classes are numbered 0, 1, etc., and hosts are specified through machine classes. A machine class is specified in the submit-description file submitted to Condor that specifies the program name, input file name, requirement on machines' architecture, operating system and memory, etc.

Another difference is that Condor-PVM has "host suspend" and "host resume" notifications in addition to the "host add," "host deletion," and "task exit" notifications of PVM. When Condor detects activity of a workstation owner, it suspends all Condor processes running there rather than killing them immediately. If the owner remains for less than a prespecified cutoff time, the suspended processes will resume. To help an application to deal with this situation, Condor-PVM makes some extensions to PVM's notification mechanism.

The last difference is that adding a host is nonblocking in Condor-PVM. When a Condor-PVM application requests that a new host be added to the virtual machine, the request is sent to Condor. Condor then attempts to schedule one from the pool of idle machines. This process can take a significant amount of time if, for example, there are no machines available in Condor's pool. Therefore, Condor-PVM handles requests for new hosts asynchronously. The application can start other work immediately after it sends out a request for a new host. It then uses the PVM notification mechanism to detect when the "host add" request was satisfied. This feature allows our greedy host request scheme to work well in practice.

Documentation and examples about these differences can be found online at http://www.cs.wisc.edu/condor/. FATCOP was first developed as a PVM application, and modified to exploit Condor-PVM.

3.4. The parallel implementation of FATCOP. FATCOP consists of two separate programs: the master program and the worker program. The master program runs on the machine from which the job was submitted to Condor. This machine is supposed to be stable for the life of the run, so it is generally the machine owned by the user. The design of FATCOP makes the program tolerant to any type of failure for workers, but if the machine running the master program crashes due to either system reboot or power outage, the program will be terminated. To make FATCOP tolerant

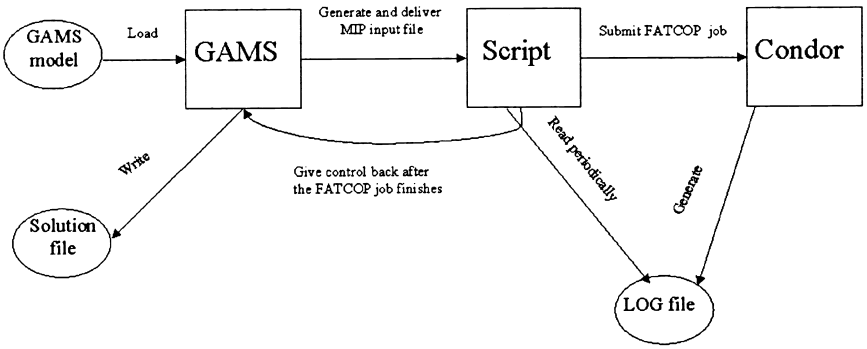


FIG. 3.2. *Interactions among Condor, FATCOP, and GAMS.*

even of these failures, the master program writes information about subproblems in the work pool periodically to a log file on the disk. Each time a FATCOP job is started by Condor, it reads in the MIP problem as well as the log file that stores subproblem information. If the log file does not exist, the job starts from the root of the search tree. Otherwise, it is warm started from some point in the search process. The work pool maintained by the master program has copies for all the subproblems that were sent to the workers, so the master program is able to write complete information about the branch-and-bound process to the log file.

The worker program runs on the machines selected by Condor. The number of running worker programs changes over time during the execution of a FATCOP job.

3.4.1. The master program. FATCOP can take both MPS format and general algebraic modeling system (GAMS) models as input. The interactions among Condor, FATCOP, and GAMS are as follows. A user starts to solve a GAMS model in the usual way from the command line. After GAMS reads in the model, it generates an input file containing a description of the MIP model to be solved. Control is then passed to a PERL script. The script generates a Condor job description file and submits the job to Condor. After submitting the job, the script reads a log file periodically until the submitted job is finished. The log file is generated by Condor and records the status of the finished and executing jobs. After completion, control is returned to GAMS, which then reports the solution to the user. This process is depicted in Figure 3.2. The process is similar for MPS file input.

The MIP model is stored globally as LP and integrality constraints. The master program first solves the LP relaxation. If it is infeasible or the solution satisfies the integrality constraints, the master program stops. Otherwise, it starts a sequential MIP solve process until there are N solved LP subproblems in the work pool. N is a predefined number that has a default value and can be modified by users. This process is based on the observation that using parallelism as soon as a few subproblems become available may not be a good policy, since doing so may expand more nodes compared to a sequential algorithm. Associated with each subproblem in the work pool is the LP relaxation solution and value, modified bound information for the integer variables, pseudocosts used for searching, and an optimal basis that is used for warm starting the simplex method. The subproblems in the work pool are multi-indexed by bound, best-estimation, and the order in which they entered the pool. The indices correspond to different searching rules: best-first, best-estimation, and depth-first.

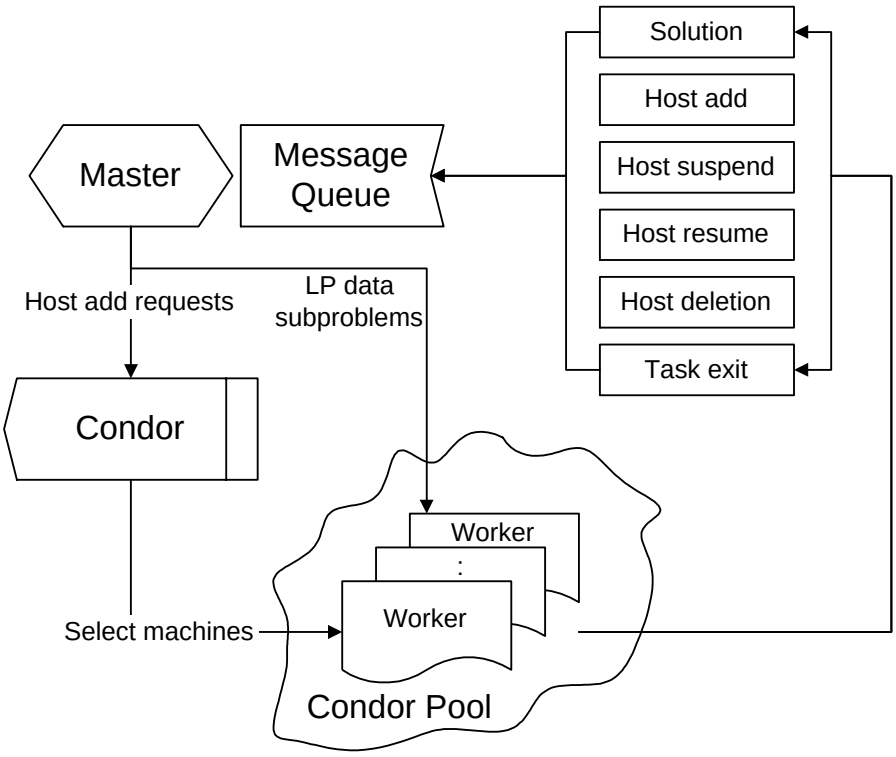


FIG. 3.3. *Message passing inside FATCOP.*

Following the initial subproblem generation stage, the master program sends out a number of requests for new hosts. It then sits in a loop that repeatedly does message receiving. The master accepts several types of messages from workers. The messages passing within FATCOP are depicted in Figure 3.3 and are explained further below. After all workers have sent solutions back and the work pool becomes empty, the master program kills all workers and exits itself.

Host add message: After the master is notified of getting a new host, it spawns a child process on that host and sends an LP copy as well as a subproblem to the new child process. The subproblem is marked in the work pool, but not actually removed from it. Thus the master is capable of recovering from several types of failures. For example, the spawn may fail. Recall that Condor takes the responsibility to find an idle machine and starts a PVM daemon on it. During the time between when the PVM daemon was started and the message is received by the master program, the owner of the selected machine can reclaim it. If a "host add" message was queued waiting for the master program to process other messages, a failure for spawn becomes more likely.

The master program then sends out another request for a new host if the number of remaining subproblems is at least twice as many as the number of workers. The reason for not always asking for a new host is that the overhead associated with spawning processes and initializing new workers is significant. Spawning a new process is not handled asynchronously by Condor-PVM. While a spawn request is processed, the master is blocked. The time to spawn a new process usually takes several seconds.

Therefore if the number of subproblems in the work pool drops to a point close to the number of workers, the master will not ask for more hosts. This implementation guarantees that only the top 50% "promising" subproblems considered by the program can be selected for evaluation. Furthermore, when the branch-and-bound algorithm eventually converges, this implementation prevents the program from asking for excess hosts. However, the program must be careful to ensure that when the ratio of the number of remaining subproblems to the number of hosts becomes bigger than 2, the master restarts requesting hosts.

Solution message: If a received message contains a solution returned by a worker, the master will permanently remove the corresponding subproblem from the work pool that was marked before. It then updates the work pool using the received LP solutions. After that, the master selects one subproblem from the work pool and sends it to the worker that sent the solution message. The subproblem is marked and stays in the work pool for failure recovery. Some worker idle time is generated here, but the above policy typically sends subproblems to workers that exploit the previously generated solution.

Host suspend message: This type of message informs the master that a particular machine has been reclaimed by its owner. If the owner leaves within 10 minutes, the Condor processes running on this machine will resume. We have two choices to deal with this situation. The master program can choose to wait for the solutions from this host or send the subproblem currently being computed in this host to another worker. Choosing to wait may save the overhead involved in solving the subproblem. However, the waiting time can be as long as 10 minutes. If the execution time of a FATCOP job is not significantly longer than 10 minutes, waiting for a suspended worker may cause a serious delay for the program. Furthermore, the subproblems selected from the work pool are usually considered "promising." They should be exploited as soon as possible. Therefore, if a "host suspend" message is received, we choose to recover the corresponding subproblems in the work pool right away. This problem then has a chance to be quickly sent to another worker. If the suspended worker resumes later, the master program has to reject the solutions sent by it so that each subproblem is considered exactly once.

Host resume message: After a host resumes, the master sends a new subproblem to it. Note that the master should reject the first solution message from that worker. The resumed worker picks up in the middle of the LP solve process that was frozen when the host was suspended. After the worker finishes solving the LPs, it sends the solutions back to the master. Since the associated subproblem had been recovered when the host was suspended, these solutions are redundant and hence should be ignored by the master.

Host delete/task exit message: If the master is informed that a host is removed from the PVM or a process running on a host is killed, it recovers the corresponding subproblem from the work pool and makes it available to other workers.

3.4.2. Worker program. The worker program first receives an LP model from the master, then sits in an infinite loop to receive messages from the master. The messages from the master consist of the modified bound information about the subproblem P, the optimal basis to speed up the bounding operation, and the branching variable that is used to define the "up" and "down" children $P+$ and $P-$. The worker performs two bounding operations on $P+$ and $P-$ and sends the results back to the master. The worker program is not responsible for exiting its PVM daemon. It will

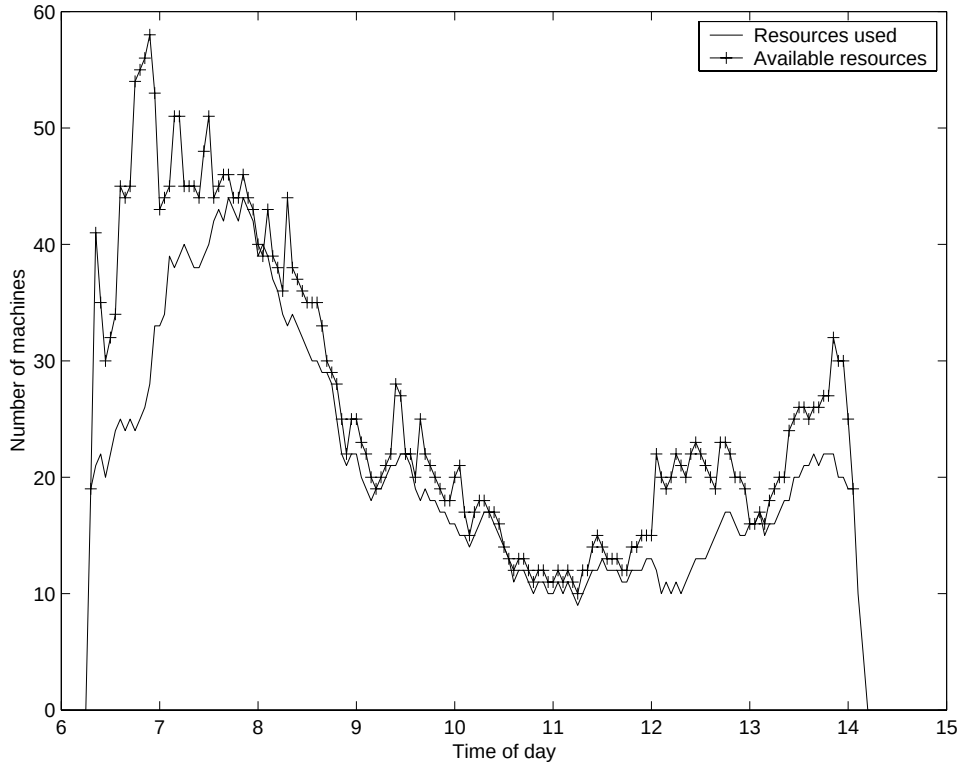


FIG. 4.1. *Resource utilization for one run of FATCOP.*

be killed by the master after the stopping criteria are met.

4. Computational experience. A major design goal of FATCOP is fault tolerance, that is, solving MIP problems correctly using opportunistic resources. Another design goal is to make FATCOP adaptive to changes in available resources provided by Condor in order to achieve maximum possible parallelism. Therefore the principal measures we use when evaluating FATCOP are *correctness* of solutions, and *adaptability* to changes in resources. *Execution time* is another important performance measure, but it is affected by many random factors and heavily dependent on the availability of Condor's resources. For example, a FATCOP job that can be finished in 1 hour at night may take 2 hours to finish during the day because of the high competition for the resources. We first show how FATCOP uses as many resources as it is able to capture, then show how reliable it is to failures in its environment, and conclude this section with numerical results on a variety of test problems from the literature.

4.1. Resource utilization. In Wisconsin's Condor pool there are more than 100 machines in our desired architecture class. Such a large amount of resources makes it possible to solve MIP problems with fairly large search trees. However, the available resources provided by Condor change as the status of participating machines changes. Figure 4.1 demonstrates how FATCOP is able to adapt to Condor's dynamic environment. We submitted a FATCOP job in the early morning. Each time a machine was added or suspended, the program asked Condor for the number of idle

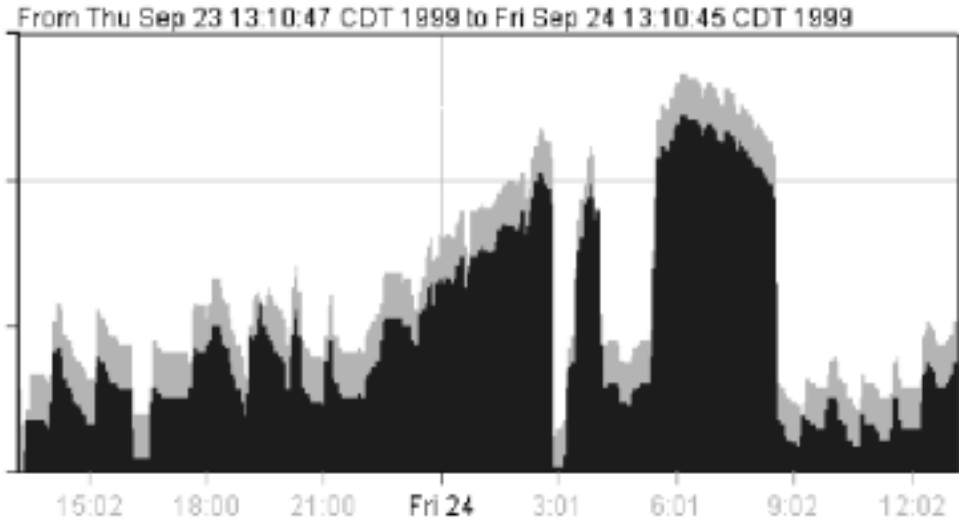


FIG. 4.2. *Daily log for a FATCOP job.*

machines in our desired machine class. We plot the number of machines used by the FATCOP job and the number of machines available to the job in Figure 4.1. In the figure, time goes along the horizontal axis, and the number of machines is on the vertical axis. The solid line is the number of working machines and the dotted line is the number of available machines that includes idle machines and working machines used by our FATCOP job. At the start, there were some idle machines in the Condor pool. The job quickly harnessed about 20 machines and eventually collected more than 40 machines with a speed of roughly one new resource every minute. At 8 A.M. it became difficult to acquire new machines and machines were steadily lost during the next four hours. There were some newly available resources at 8:30 and 10:00 (see the peaks of the dotted lines), but they became unavailable again quickly, either reclaimed by owners or scheduled to other Condor users with higher priority. At noon, another group of machines became available and stayed idle for a relatively long time. The FATCOP job acquired some of these additional machines during that time. In general, the number of idle machines in the Condor pool had been kept at a very low level during the life cycle of the FATCOP job except during the start-up phase. When the number of idle machines stayed high for some time, FATCOP was able to quickly increase the size of its virtual machine. We believe these observations exhibit that FATCOP can utilize opportunistic resources very well.

We show a FATCOP daily log in Figure 4.2. The darkly shaded area in the foreground is the number of machines used and the lightly shaded area is the number of outstanding resource requests to Condor from this FATCOP job. During the entire day, the number of outstanding requests was always about 10, so Condor would consider assigning machines to the job whenever there were idle machines in Condor's pool. At night, this job was able to use up to 85 machines (the horizontal line represents 75 machines, and the vertical line in the figure represents midnight). Note that the Computer Sciences Department at the University of Wisconsin reboots all instructional machines at 3 A.M. every day. This job lost almost all its machines at that time, but it quickly got back the machines after the reboot.

TABLE 4.1
Average number of machine and suspensions for 4 FATCOP runs.

Run	Starting time	Duration	E_{avg}	Number of suspensions
1	07:50	13.5 hrs	32	145
2	12:01	14.9 hrs	29	181
3	16:55	11.1 hrs	40	140
4	21:00	10.1 hrs	49	118

To gain more insight into utilization of opportunistic resources by FATCOP, we define the average number of machines used by a FATCOP job E_{avg} as

$$E_{avg} = \frac{\sum_{k=1}^{E_{max}} k\tau_k}{T},$$

where τ_k is the total time when the FATCOP job has k workers, T is the total execution time for the job, and E_{max} is the number of available machines in the desired class. We ran 4 replications of a MIP problem. The starting time of these runs was distributed over a day. In Table 4.1 we record the average number of machines the FATCOP job was able to use and the number of machines suspended during each run. The first value shows how much parallelism the FATCOP job can achieve and the second value indicates how much additional work had to be done. In general the number of machines used by FATCOP is quite satisfactory. At run 4, this value is as high as 49, implying that on average FATCOP used close to 50% of the total machines in our desired class. However, the values vary greatly due to the different status of the Condor pool during different runs. In working hours it is hard to acquire machines because many of them are in use by owners. After working hours and on the weekend, only other Condor users are our major competitors. As expected FATCOP lost machines frequently during the daytime. However, during the runs at night FATCOP also lost many machines. It is not surprising to see this, because the more machines FATCOP was using, the more likely it would lose some of them to other Condor users.

4.2. Fault tolerance. FATCOP has been tested on the problems from MIPLIB3.0. There are 59 problems in the test set with different size and difficulty. The FATCOP sequential and parallel solvers solved 41 and 44 problems, respectively, with default options, accounting for 70% and 75% of the total test problems. Our computational results show that problems that can be solved in minutes by the FATCOP sequential solver may take longer to solve via the parallel solver. In solving these problems, FATCOP spent a large portion of the total solution time on spawning and initializing workers. It suggests that it is only beneficial to use FATCOP to solve MIPs with large search trees and/or complex LP bounding operations. In this section we report the computational results for the problems that cannot be solved by the FATCOP sequential solver in an hour, but are solvable by the parallel solver. The size of these problems are shown in Table 4.2.

FATCOP was configured to use default branching and node selection strategies, i.e., pseudocost branching and the best-estimation-based mixed searching strategy (mixed strategy 2). We let FATCOP switch to best-bound search after the best integer solution had remained unchanged for 10,000 node evaluations. MIPLIB files do not provide branching priorities, so priority branching is irrelevant to problems from MIPLIB. CPLEX was used as the primary LP solver. Due to licensing limitations,

TABLE 4.2
*Summary of test problems from MIPLIB*3.0.

Name	#rows	#columns	#nonzeros	#integers
10TEAMS	230	2025	12150	1800
AIR04	823	8904	72965	8904
AIR05	426	7195	52121	7195
DANOINT	664	521	3232	56
FIBER	363	1298	2944	1254
L1521AV	97	1989	9922	1989
MODGLOB	291	422	968	98
PK1	45	86	915	55
PP08ACUTS	246	240	839	64
QIU	1192	840	3432	48
ROUT	291	556	2431	315
VPM2	234	378	917	168

TABLE 4.3
Results obtained by the FATCOP sequential solver.

Name	Solution gap(%)	Proven optimal?	Execution time	Nodes
10TEAMS	0	yes	23.2 hrs	163,130
AIR04	0	yes	9.6 hrs	4,606
AIR05	0	yes	29.3 hrs	23,512
DANOINT	0	no	48.0 hrs	640,300
FIBER	0	yes	4.7 hrs	172,788
L1521AV	0	yes	1.9 hrs	17,846
MODGLOB	0	no	48.0 hrs	12,459,812
PK1	0	yes	1.6 hrs	475,976
PP08ACUTS	0	yes	6.4 hrs	3,469,870
QIU	0	yes	1.2 hrs	16,448
ROUT	3	no	48.0 hrs	2,582,441
VPM2	0	yes	2.4 hrs	926,740

when the maximum number of CPLEX copies was reached, SOPLEX was called to perform bounding operations in the workers.

We first tried to solve the problems in Table 4.2 using the FATCOP sequential solver on a SUN Sparc SOLARIS2.6 machine. Each run was limited to 48 hours. We present the results in Table 4.3. The first column in the table shows the relative difference between the best solution found by the FATCOP sequential solver and the known optimal solution. If the optimal solution is found, column 2 shows whether or not the solution is a proven optimal solution. Execution time in column 3 is clock elapsed time. Tree size at the time when the program was terminated is given in column 4.

The test problems were then solved by the FATCOP parallel Condor-PVM solver. The number of problems N generated in the initial stage was set to 20. At the beginning the master sends 10 requests for new hosts to Condor. FATCOP implements an asynchronous algorithm; hence communication may occur at any time and is unpredictable. Furthermore, the number of workers in the life cycle of a FATCOP job keeps changing so that the branch-and-bound process may not follow the same path for different executions. Our experiments show that the search trees were almost never expanded in the same order for a given problem. This feature often leads FATCOP to different execution times. We ran 3 replications for each problem. For all runs, FATCOP found provable optimal solutions for the test problems. We report the average execution time, the number of evaluated nodes in the search tree, resource utilization,

TABLE 4.4
Average results obtained by the FATCOP parallel Condor-PVM solver for 3 replications (all instances were solved to optimality).

Name	Average execution time	Average tree size	E_{avg}	Average # of suspensions	Speedup
10TEAMS	1.9 hrs	201,553	20	256	12.2
AIR04	56.8 mins	5,464	11	94	10.1
AIR05	2.0 hrs	16,842	41	55	14.7
DANOINT	24.2 hrs	3,451,676	22	239	-
FIBER	1.1 hrs	174,489	42	25	4.3
L1521AV	24.6 mins	12,266	35	32	4.6
MODGLOB	44.8 hrs	704,056,478	18	512	-
PK1	14.5 mins	8554.886	28	12	6.6
PP08ACUTS	2.8 hrs	5,001,600	19	122	2.3
QIU	22.2 mins	16,376	27	9	3.2
ROUT	12.3 hrs	22,851,706	29	295	-
VPM2	46.7 mins	1,014,943	15	59	3.1

and resource losses in Table 4.4. During all runs, FATCOP lost some workers, but the program returned correct solutions. Therefore FATCOP was tolerant to the resource retreats in our experiments.

The FATCOP parallel solver found provable optimal solutions for all the test problems. However, the sequential solver failed to prove optimality on *danoint*, *modglob*, and *rout*. For the problems solved by both, the parallel solver achieved reasonable speedup over the sequential solver. Run times for these problems were reduced by factors between 2.3 and 14.7.

We observed from Table 4.4 that many test problems exhibit strong search anomalies. *pp08aCUTS* and *pk1* have much larger search trees when solved by the parallel solver. On the other hand, *10teams*, *air05*, and *l1521av* have smaller search trees. While such search anomalies are well known for parallel branch-and-bound, the highly nondeterministic nature of the Condor opportunistic environment can lead to even more varying search patterns.

A remarkable example in this test set is *modglob*. The FATCOP sequential solver could not find a provable optimal solution for this problem in 48 hours, while it was solved to optimality by the parallel solver in 44.8 hours. It ran over two Computer Sciences Department daily reboot periods, used 18 machines on average, and had 512 machines suspended during the run. To test fault tolerance of the master, we let FATCOP write the work pool information to disk every 100,000 node evaluations. We interrupted the job once (to simulate a master failure) and resubmitted the problem to Condor. FATCOP then started from where the work pool information was last recorded. This indicates that FATCOP is tolerant to both worker and master failures.

4.3. Application test problems. FATCOP was used to solve two classes of problems arising from marketing and electronic data delivery. One class of problems, *VOD*, consists of applications to video-on-demand system design [10, 13]. The other class of problems, *PROD*, consists of applications to product design [26]. Two problems from each application were formulated as GAMS models. The size of the problem instances and results found by FATCOP are reported in Table 4.5. Execution time is clock elapsed time and does not include GAMS compilation and solution report time. User-defined priorities are provided in *VOD2*. It turns out that this information is critical to solve this problem in a reasonable amount of time [13]. For PROD problems, good integer feasible solutions were found using a genetic algorithm (GA) [3]

TABLE 4.5
Results for VOD and PROD problems.

Name	#rows	#columns	#nonzeros	#integers	time	E_{avg}
VOD1	107	306	1207	303	7.5 mins	11
VOD2	715	1513	7316	1510	5.2 mins	18
PROD1	208	251	5350	149	1.2 hrs	25
PROD2	211	301	10501	200	10.8 hrs	35

TABLE 4.6
Comparison with GA and without GA for PROD1.

Relative gap %	Nodes with GA	Nodes without GA
20	1,178	864,448
15	2,230	> 1,000,000
10	6,506	> 1,000,000
5	37,224	> 1,000,000
0	137,866	> 1,000,000

first, and these solutions were delivered to FATCOP as incumbent values. Provable optimal solutions were found for all the problem instances in Table 4.5.

In practice many problem-specific heuristics are effective for quickly finding near-optimal solutions. Marrying the branch-and-bound algorithm with such heuristics can help both heuristic procedures and a branch-and-bound algorithm. For example, heuristics may identify good integer feasible solutions for the early stage of the branch-and-bound process, decreasing overall solution time. On the other hand, the quality of solutions found by heuristic procedures may be measured by the (lower-bounding) branch-and-bound algorithm. FATCOP can use problem-specific knowledge to increase its performance. Based on interfaces defined by FATCOP, users can write their own programs to round an integer infeasible solution, improve an integer feasible solution, and perform operations such as identifying good solutions or adding problem-specific cutting planes at the root node. These user-defined programs are dynamically linked to the solver at run time and can be invoked by turning on appropriate solver options. For product design problems, good solutions found by the GA made the problems solvable. We performed a set of experiments on *PROD1* by turning the GA program on and off at the root node. We limited the number of node evaluations to 1,000,000. The computational results are given in Table 4.6. Without the GA, FATCOP cannot reduce the optimality gap below 15% in the given number of node evaluations. However, with good solution found by the GA at the root node, FATCOP is able to prove optimality for this problem in around 1.2 hours.

4.4. Set partitioning. An encouraging side benefit to the opportunistic environment is the fact that the parallel machine becomes much more powerful as more and newer machines are added to the pool. This is in sharp contrast to the case where a standard supercomputer is obtained, unless a new machine is purchased every six months. Thus, the computational results that we generated six months ago become much easier to obtain today. As an example of this type of progress, we cite two new set partitioning problems arising in transportation scheduling that were solved using the FATCOP solver after the original paper was written.

The first problem, t0415, was started at 18:37 and finished proving optimality two days later (around 31 hours later) at 01:11. The problem has 1518 constraints and 7254 binary variables. During this time, a maximum of 114 workers was used, and a total of 1795 hours of worker CPU time. Note that many of the machines used in this

run were claimed during the work day by their respective owners, but regardless of this, each of the 114 workers was processing linear programs for around 16 hours of the total run, processing 531,707 nodes in the tree in total. FATCOP used its default strategy in this run. Many of the processors used in this run were not installed when this paper was first released, but the same code was able to exploit them as soon as they became part of the Condor pool.

The second problem, t0416, seems at first inspection very similar. It has 1771 constraints and 9345 binary variables. However, this problem proved substantially more difficult to solve, being started on September 16 at 19:49 and finishing on September 26 at 22:48. In solving this problem, three checkpoints were utilized, each time resetting FATCOP's options to switch between best-bound branching and depth-first search after each solve. Thus, the actual time the FATCOP program was running was around 97 hours instead of the possible 242 hours between its start and finish times. This was to control the size of the unexplored tree more carefully than our default strategy allows. The maximum number of workers used was 118, the average number of workers throughout the run was 78, with a total of 5409 hours of worker CPU time (out of a total worker uptime of 7544 hours). In this run, 1,081,271 nodes in the tree were processed in total. Clearly, the second run is less impressive in terms of efficiency and the need for user intervention. However, it should also be noted that we have been unable to solve either test problem using the currently available commercial solvers for mixed integer linear programs.

5. Summary and future work. In this paper, we provide a parallel branch-and-bound implementation for MIPs using distributed privately owned workstations. The solver, FATCOP, is designed in the master-worker paradigm to deal with different types of failures in an opportunistic environment with the help of Condor, a resource management system. To harness the available computing power as much as possible, FATCOP uses a greedy strategy to acquire machines. FATCOP is built upon Condor-PVM and SOPLEX, which are freely available for download from http://www.cs.wisc.edu/condor/ and http://www.zib.de/Optimization/Software/Soplex/.

FATCOP has successfully solved real-life MIP problems such as in applications to video-on-demand system and product design. It was also tested on a set of standard test problems from MIPLIB. Our computational results show that the solver works correctly in the opportunistic environment and is able to utilize opportunistic resources efficiently. A reasonable speedup was achieved on long running MIPs over its sequential counterpart.

Our future work includes strengthening parallel branch-and-bound procedures with more cutting planes such as flow cover cuts and disjunctive cuts and investigating how much a worker processor should do before returning results to the master. Some of these issues have already been addressed in subsequent work [8].

Acknowledgments. The authors are grateful to Miron Livny and Michael Yoder for advice and assistance in using Condor and the Condor-PVM environment. The authors also wish to thank Jeff Linderoth for his very insightful comments.

REFERENCES

[1] K. Anstreicher, N. Brixius, J.-P. Goux, and J. Linderoth, *Solving Large Quadratic Assignment Problems on Computational Grids*, Technical report, Argonne National Laboratory, Argonne, IL, 2000.

[2] M. Avriel and B. Golany, *Mathematical Programming for Industrial Engineers*, Marcel Dekker, New York, 1996.
[3] P. V. Balakrishnan and V. S. Jacob, *Genetic algorithms for product design*, Management Sci., 42 (1996), pp. 1105–1117.
[4] M. Benichou and J. M. Gauthier, *Experiments in mixed-integer linear programming*, Management Sci., 20 (1974), pp. 736–773.
[5] R. E. Bixby, S. Ceria, C. M. McZeal, and M. W. P. Savelsbergh, *MIPLIB* 3.0, available online at http://www.caam.rice.edu/~bixby/miplib/miplib.html.
[6] R. E. Bixby, W. Cook, A. Cox, and E. K. Lee, *Parallel Mixed Integer Programming*, Technical report CRPC-TR95554, Center for Research on Parallel Computation, Rice University, Houston, TX, 1995. Available online at ftp://softlib.rice.edu/pub/CRPC-TRs/reports/CRPC-TR95554.pdf.
[7] A. Brooke, D. Kendrick, and A. Meeraus, *GAMS: A User's Guide*, The Scientific Press, South San Francisco, CA, 1988.
[8] Q. Chen, M. C. Ferris, and J. T. Linderoth, *FATCOP* 2.0: *Advanced features in an opportunistic mixed integer programming solver*, Ann. Oper. Res., to appear.
[9] *CPLEX Optimizer*, available online at http://www.cplex.com.
[10] D. L. Eager, M. C. Ferris, and M. K. Vernon, *Optimized regional caching for on-demand data delivery*, in Multimedia Computing and Networking 1999, Proc. SPIE 3654, SPIE—The International Society for Optical Engineering, Bellingham, WA, 1999, pp. 301–316.
[11] J. Eckstein, *Parallel branch-and-bound algorithms for general mixed integer programming on the CM-5*, SIAM J. Optim., 4 (1994), pp. 794–814.
[12] D. H. Epema and M. Livny, *A worldwide flock of condors: Load sharing among workstation clusters*, Future Generation Comput. Systems, 12 (1996), pp. 53–65.
[13] M. C. Ferris and R. R. Meyer, *Models and solution for on-demand data delivery problems*, in Approximation and Complexity in Numerical Optimization: Continuous and Discrete Problems, Kluwer Academic Publishers, Dordrecht, the Netherlands, 1999, pp. 175–188.
[14] A. Geist, A. Beguelin, J. J. Dongarra, W. Jiang, R. Manchek, and V. S. Sunderam, *PVM: Parallel Virtual Machine—A User's Guide and Tutorial for Networked Parallel Computing*, The MIT Press, Cambridge, MA, 1994.
[15] B. Gendron and T. G. Crainic, *Parallel branch-and-bound algorithms: Survey and synthesis*, Oper. Res., 42 (1994), pp. 1042–1060.
[16] J.-P. Goux, S. Kulkarni, J. Linderoth, and M. E. Yoder, *An enabling framework for master-worker applications on the computational grid*, in Proceedings of the Ninth IEEE International Symposium on High Performance Distributed Computing, Pittsburgh, PA, 2000, pp. 43–50.
[17] W. Gropp, E. Lusk, and A. Skjellum, *Using MPI: Portable Parallel Programming with the Message-Passing Interface*, The MIT Press, Cambridge, MA, 1994.
[18] Z. Gu, G. L. Nemhauser, and M. W. P. Savelsbergh, *Lifted cover inequalities for* 0-1 *integer programs: Computation*, INFORMS J. Comput., 10 (1998), pp. 427–437.
[19] J. Linderoth and M. W. P. Savelsbergh, *A computational study of search strategies for mixed integer programming*, INFORMS J. Comput., 11 (1999), pp. 173–187.
[20] M. J. Litzkow and M. Livny, *Condor—A hunter of idle workstations*, in Proceedings of the 8th International Conference on Distributed Computing Systems, Washington, DC, IEEE Computer Society Press, Los Alamitos, CA, 1988, pp. 108–111.
[21] J. L. Nazareth, *Computer Solution of Linear Programs*, Oxford University Press, Oxford, New York, 1987.
[22] G. L. Nemhauser and L. Wolsey, *Integer and Combinatorial Optimization*, Wiley-Interscience, New York, 1989.
[23] J. Pruyne and M. Livny, *Providing resource management services to parallel applications*, in Proceedings of the Second Workshop on Environments and Tools for Parallel Scientific Computing, J. J. Dongarra and B. Tourancheau, eds., Proceedings in Applied Mathematics 74, SIAM, Philadelphia, 1994, pp. 152–161.
[24] E. A. Pruul and G. L. Nemhauser, *Branch-and-bound and parallel computation: A historical note*, Oper. Res. Lett., 7 (1988), pp. 65–69.
[25] M. W. P. Savelsbergh, *Preprocessing and probing for mixed integer programming problems*, ORSA J. Comput., 6 (1994), pp. 445–454.
[26] L. Shi, S. Ólafsson, and Q. Chen, *An optimization framework for product design*, Management Sci., submitted.
[27] S. J. Wright, *Solving optimization problems on computational grids*, Optima, to appear.
[28] R. Wunderling, *SOPLEX Library Documentation*, available online at http://www.zib.de/Optimization/Software/Soplex.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1037–1053

LIMITING AVERAGE CRITERIA FOR NONSTATIONARY MARKOV DECISION PROCESSES*

XIANPING GUO† AND PENG SHI‡

Abstract. This paper deals with the so-called limiting average criteria for nonstationary Markov decision processes with (possibly unbounded) rewards and Borel state space. A new set of conditions is provided, under which the existence of both a solution to the optimality equations and the limiting average $\epsilon(\geq 0)$-optimal Markov policies is derived. Also, a rolling horizon algorithm for computing limiting average $\epsilon(> 0)$-optimal Markov policies is developed. Furthermore, the results in this paper are illustrated by several examples such as the water regulation problem.

Key words. nonstationary Markov decision processes, limiting average criteria, optimality equations, limiting average ϵ-optimal policies, rolling horizon algorithm

AMS subject classification. 90C40

PII. S1052623499355235

1. Introduction. Infinite horizon Markov decision processes (MDPs) have been extensively studied since the 1950s. One of the most commonly considered versions is the so-called "limiting average reward" model. In this model, the criteria most often used are *average expected criterion* $\bar{V}$(see (2.4) below), *expected average criterion* $\bar{U}$(see (2.5) below), and *sample path average criterion* V_s (see (2.6) below). The three criteria are different for many nontrivial problems and have been studied by many authors. First, we briefly describe the main results about these criteria for the *stationary* MDPs, that is, the rewards and the transition probabilities are independent of time. For the average expected criterion $\bar{V}$, it is well known that if the state and action spaces are finite, then there exists an optimal stationary policy (see Dynkin and Yushkevich [6]). But if the action space is compact or the state space is countable, there may not exist an optimal policy (see Dynkin and Yushkevich [6, p. 178] and Ross [22, p. 90]). For the existence of optimal stationary or ϵ-optimal policies in MDPs with general state and possibly unbounded rewards, many sufficient conditions have been investigated such as ergodic conditions (see Hernandez-Lerma [12, p. 56] and Kurano [18, 17]), the mirror conditions (see Dynkin and Yushkevich [6, p. 187]), the recurrence and Lyapunov conditions (see Arapostathis et al. [2]), and the vanishing discounted factor conditions (see Hernandez-Lerma and Lasserre [13, pp. 83–88], Puterman [20, pp. 415–416], and Arapostathis et al. [2] and the references therein). The algorithms for computing optimal policies such as value iteration, policy iteration, and linear programming are also provided (see [6, pp. 173–178], [20, pp. 452; 462–468; 472–476]). For the expected average criterion $\bar{U}$, the results in Bierth [4] and Blackwell [5] together with the example in [6, p. 178] have shown that

*Received by the editors April 26, 1999; accepted for publication (in revised form) November 21, 2000; published electronically May 10, 2001. This work was partially supported by the Natural Science Foundation of China grant 19901038, the Natural Science Foundation of Guangdong Province, the Foundation of Hong Kong and Zhongshan University Advanced Research Center, and the Centre for Industrial and Applicable Mathematics, the University of South Australia.
http://www.siam.org/journals/siopt/11-4/35523.html

†Department of Mathematics, Zhongshan University, Guangzhou 510275, People's Republic of China (mcsgxp@zsu.edu.cn).

‡Land Operations Division, Defence Science and Technology Organisation, PO Box 1500, Salisbury 5108 SA, Australia (peng.shi@dsto.defence.gov.au). This author's work was performed while he was at the University of South Australia.

optimal stationary policies exist for the case of finite state and action spaces, but may not exist for the case of a finite state model with arbitrary action sets. However, the existence of ϵ-optimal Markov policies is established by Bierth [4] for the finite state model. Feinberg and Park [7] discussed the existence of persistently nearly optimal policies under some conditions such as finite state space. Filar, Krass, and Ross [8] use the results about this criterion to discuss the percentile performance criterion for limiting average MDPs with finite state and action spaces. For the sample path average criterion V_s, Ross and Varadarajian [23] established that there exist ϵ optimal stationary policies for communicating MDPs with finite state and action spaces and gave a parametric linear programming algorithm for constructing ϵ optimal stationary policies. Moreover, they [24] extended the results in [23] to the case of MDPs without the communicating properties. For the case of denumerable state and possibly unbounded rewards, Cavazos-Cadena and Fernández-Gaucherand [21] showed that, under the Lyapunov and continuity-compactness conditions, stationary policies obtained from the average reward optimality equation are not only average expected optimal, but also sample path average optimal. As is well known, most research on the three criteria has focused on the case of stationary MDPs. However, in reality, the rewards and the transition probabilities may be changed with time. Hence, it is more natural to consider the case of both rewards and transition probabilities being time dependent (i.e., nonstationary case). For the nonstationary MDPs, a great amount of work has been done in the past. Now we summarize the main results for the average expected criterion in nonstationary MDPs. For the case of finite state and action spaces, under the ergodic condition, Hopp, Bean, and Smith [15] showed that an accumulation point of a sequence of finite horizon optimal policies is average optimal. Also, Alden and Smith [1] provided an error bound in average expected cost between a rolling horizon policy and an average expected optimal policy. Bean, Smith, and Lasserre [3] extended the results for the above finite state model to a denumerable state case under weak ergodicity. Park, Bean, and Smith [19] proved that, under an ergodic condition, the optimal finite horizon average values converge to the infinite horizon optimal average expected value in the denumerable state case. By using optimality equations, Hou and Guo [16, 9] proved the existence of optimal Markov policies under an ergodic condition. Recently, Guo [10, 11] discussed the properties of optimal policies and the average variance criterion. It should be noted that all the rewards in [15, 1, 3, 19, 16, 9, 10, 11] are assumed to be uniformly bounded. To the best of the authors' knowledge, the problems of the expected average and sample path average criteria in nonstationary MDPs with possible unbounded rewards have not yet been fully investigated.

This paper will deal with the above criteria for nonstationary MDPs with possibly unbounded rewards and Borel state space. Our concern is the following question: To what extent can the classical results about the above three criteria for stationary MDPs be generalized into the nonstationary situation? In the spirit of [6, 12, 18, 17] for general state stationary MDPs and of [1, 3, 19] for denumerable state nonstationary MDPs, we propose a new set of conditions under which we not only establish the existence of a solution to the optimality equations (OEs) for nonstationary MDPs but also prove that the Markov policies obtained from the OEs are ϵ-optimal with respect to all three criteria (i.e., limiting average ϵ-optimal) by using the martingale theory. Also, a rolling horizon algorithm for computing limiting average $\epsilon(> 0)$-optimal Markov policies is developed. Moreover, two examples such as the water regulation problem are given to illustrate our results and to show the potential of the

proposed techniques in this paper.

In section 2 notation and definitions are introduced and the formal description of the underlying problem is given. The OEs are built up in section 3 for nonstationary MDPs with average optimality, and the existence of solutions to the OEs is also verified. The existence of limiting average ϵ-optimal Markov policies is proved in section 4. A rolling horizon algorithm is developed in section 5. An applied example is given in Section 6.

2. Model, notation, and definitions. The model considered in this paper is a six-element tuple $\{S, A, (A_n(i)|i \in S,\ n \geq 0), (P_n), (r_n), (\bar{V}, \bar{U}, \bar{V}_s)\}$ consisting of

(a) a standard Borel space S, the state space, with Borel σ-algebra $\mathcal{B}(S)$;

(b) a Borel space A, the action space, with Borel σ-algebra $\mathcal{B}(A)$;

(c) a family $\{A_n(i)|i \in S, n \geq 0\}$ of nonempty measurable subsets of A, where $A_n(i)$ denotes the set of feasible actions when the system is in state $i \in S$ at stage n, with the property that for each $n \geq 0$ the set

$$K_n := \{(i, a)|i \in S, a \in A_n(i)\}$$

of feasible state-action pairs is a measurable subset of $S \times A$ and contains the graph of a measurable function from S to A;

(d) the stochastic kernel P_n on S given K_n, that is, the transition probability of the system from stage n to stage $n + 1$;

(e) the measurable function $r_n(n \geq 0) : K_n \to R$, that is, the n stage reward function;

(f) the limiting average reward criteria $(\bar{V}, \bar{U}, \bar{V}_s)$ (see (2.4), (2.5), and (2.6) below).

For each $n = 0, 1, \ldots$, we define the space H_n of admissible histories up to time n as $H_0 := S$ and $H_n := K_0 \times K_1 \times \cdots \times K_{n-1} \times S \ \ \forall n \geq 1$. A generic element $h_n \in H_n$, which is an admissible n-history, is a vector of the form $h_n = (i_0, a_0, \ldots, i_{n-1}, a_{n-1}, i_n)$ with $(i_t, a_t) \in K_t \ \forall t = 0, 1, \ldots, n-1$, and $i_n \in S$. Of course, for each $n \geq 0$, H_n is a subspace of $(S \times A)^n \times S$.

A randomized policy π is a sequence $(\pi_0, \ldots, \pi_n, \ldots)$, where stochastic kernel π_n on the action space A given H_n satisfies

$$\pi_n(A_n(i_n)|h_n) = 1 \ \ \forall h_n \in H_n, \quad n = 0, 1, \ldots. \tag{2.1}$$

The set of all randomized policies is denoted by Π. A randomized policy $\pi := (\pi_0, \ldots, \pi_n, \ldots) \in \Pi$ is called a randomized Markov policy if $\pi_n(.|h_n) = \pi_n(.|i_n)\ \forall h_n \in H_n$ and $n \geq 0$. The set of all randomized Markov policies is denoted by Π_m. A randomized Markov policy $\pi = (\pi_0, \ldots, \pi_n, \ldots)$ is called a Markov policy if for each $n \geq 0$ there exists an $f_n \in F_n$ such that $\pi_n(\cdot|i)$ is concentrated at $f_n(i)$ for all $i \in S$, where $F_n(n \geq 0)$ denotes the nonempty set of all measurable functions $f_n : S \to A$ satisfying that $f_n(i) \in A_n(i)\ \forall i \in S$. Clearly, a Markov policy π can be uniquely determined by the sequence $\{f_n\}$ and then written as $\pi := \{f_n\}$. The set of all Markov policies is denoted by Π_m^d. Obviously, $\Pi_m^d \subset \Pi_m \subset \Pi$.

For any $\pi \in \Pi$ and $i \in S$, by the Theorem of Ionescu–Tulcea (see [13, pp. 179 and 16]), there exists a unique probability measure P_π^i on $((S \times A)^\infty, (\mathcal{B}(S) \times \mathcal{B}(A))^\infty)$ such that $P_\pi^i(H_\infty) = 1$, and, $\forall\ B \in \mathcal{B}(S)$, $i \in S$ and $n = 0, 1, \ldots,$

$$P_\pi^i(X_0 = i) = 1, \tag{2.2}$$

$$P_\pi^i(X_{n+1} \in B|X_0, \Delta_0, \ldots, X_n, \Delta_n) = P_n(B|X_n, \Delta_n), \tag{2.3}$$

where X_n and Δ_n are state and action variables at stage n, respectively. The expectation operator with respect to P_π^i is denoted by E_π^i. Since $P_\pi^i(H_\infty) = 1$, the average expected criterion $\bar{V}$, expected average criterion $\bar{U}$, and average sample path criterion V_s are well defined, respectively, as follows:

$$\bar{V}(\pi, i) := \limsup_{N\to\infty} \frac{\sum_{n=0}^{N-1} E_\pi^i r_n(X_n, \Delta_n)}{N}, \tag{2.4}$$

$$\bar{U}(\pi, i) := E_\pi^i \left[\limsup_{N\to\infty} \frac{\sum_{n=0}^{N-1} r_n(X_n, \Delta_n)}{N} \right], \tag{2.5}$$

$$V_s(\pi, i) := \limsup_{N\to\infty} \frac{\sum_{n=0}^{N-1} r_n(X_n, \Delta_n)}{N}. \tag{2.6}$$

For any $i \in S$, let $\bar{V}^*(i) := \sup_{\pi\in\Pi} \bar{V}(\pi, i)$, $\bar{U}^*(i) := \sup_{\pi\in\Pi} \bar{U}(\pi, i)$.

DEFINITION 2.1. *For any $\epsilon \geq 0$, a policy $\pi^* \in \Pi$ is called $\bar{V}$-ϵ-optimal if $\bar{V}(\pi^*, i) \geq \bar{V}^*(i) - \epsilon \quad \forall i \in S$. $\bar{V}$-0-optimal policies are called $\bar{V}$-optimal policies. Similarly, we can define $\bar{U}$-ϵ-optimal policies and $\bar{U}$-optimal policies.*

DEFINITION 2.2. *For any $\epsilon \geq 0$, a policy $\pi^* \in \Pi$ is called V_s-ϵ-optimal if there exists a constant ρ such that for all $i \in S$ and $\pi \in \Pi$,*

$$V_s(\pi^*, i) \geq \rho - \epsilon, \quad a.e.\text{-}P_{\pi^*}^i, \quad \text{and} \tag{2.7}$$

$$V_s(\pi, i) \leq \rho, \quad a.e.\text{-}P_\pi^i. \tag{2.8}$$

A V_s-0-optimal policy is called V_s-optimal.

DEFINITION 2.3. *For any $\epsilon \geq 0$, a policy $\pi^* \in \Pi$ is called limiting average ϵ-optimal if π^* is $w - \epsilon$-optimal for each $w \in \{\bar{V}, \bar{U}, V_s\}$. A limiting average 0-optimal policy is called limiting average optimal.*

Obviously, limiting average optimality is stronger than each one of the average expected optimality, expected average optimality, and sample path average optimality.

3. Optimality equations. In this section, we shall establish the OEs for nonstationary MDPs with limiting average criteria and provide some conditions to guarantee the existence of a solution to the OEs.

Let $\mathcal{P}(S)$ denote the set of all probability measures on $\mathcal{B}(S)$. We recall that a function u on S is called universally measurable if for any $P \in \mathcal{P}(S)$, there exist a measurable function v on S and a measurable set $N \in \mathcal{B}(S)$ such that $P(N) = 1$ and $u(x) = v(x) \ \ \forall x \in N$. The set of all universally measurable functions on S is denoted by $M(S)$.

DEFINITION 3.1. *If there exist a real number sequence $\{g_n\}$ and a function sequence $\{u_n\} \subset M(S)$ such that*

$$g_n + u_n(i) = \sup_{a\in A_n(i)} \left\{ r_n(i, a) + \int_S P_n(dy|i, a) u_{n+1}(y) \right\} \ \ \forall i \in S \text{ and } n \geq 0, \tag{3.1}$$

then we call the functional equations (3.2) *the OEs for our model and both sequences $\{g_n\}$ and $\{u_n\}$ a solution to the OEs* (3.2).

Obviously, if the sequences $\{g_n\}$ and $\{u_n\}$ are a solution to the OEs (3.2), then, for any constant c, the sequences $\{g_n\}$ and $\{u_n + c\}$ are also a solution to the OEs (3.2). Hence, in general, the solutions to the OEs (3.2) are not unique.

In order to derive the existence of a solution to the OEs (3.2), we need the following conditions.

ASSUMPTION 3.1. *For each $n \geq 0$, there exists a measure δ_n on $\mathcal{B}(S)$ such that for all $i \in S$ and $a \in A(i)$,*

$$(3.2)\quad \text{(i)}\quad P_n(B|i,a) \geq (\text{or } \leq)\delta_n(B) \quad \forall B \in \mathcal{B}(S);$$

$$(3.3)\quad \text{(ii)}\quad r_n(i) + \sum_{t=0}^{\infty} \left|(1-\delta_n(S))\cdots(1-\delta_{n+t}(S))\right| \parallel r_{n+t+1} \parallel := R_n(i) < \infty,$$

where $r_n(i) := \sup_{a\in A_n(i)} |r_n(i,a)|$ and $\|r_n\| := \sup_{i\in S a\in A_n(i)} |r_n(i,a)|$.

REMARK 3.1. *The motivation to introduce Assumption* 3.1(i) *stems from the ergodic conditions in* [17, 18, 19, 3, 1] *and in* [12, p. 56] *and the minorant condition in* [6, p. 186]. *Assumption* 3.1(ii) *is for the purpose of convergence. Obviously, Assumption* 3.1 *is the extension of the above ergodic and minorant conditions.*

Since Assumption 3.1(ii) is based on Assumption 3.1(i), we will give some sufficient conditions for Assumption 3.1(i). It should be noted that if the state space is denumerable, then the measure δ_n may be defined as $\delta_n(j) := \sup_{i\in S} \sup_{a\in A_n(i)} P_n(j|i,a)$ or $\delta_n(j) := \inf_{i\in S} \inf_{a\in A_n(i)} P_n(j|i,a)$ $\forall$ $j \in S$ and $n \geq 0$. In general, we have the following result.

COROLLARY 3.2. *If any one of the following conditions holds, then Assumption* 3.1(i) *holds:*

(i) *for each $n \geq 0$, the transition law $P_n(dy|i,a)$ has a density $q_n(y|i,a)$ with respect to a sigma-finite measure μ_n on S, and $q_n(y|i,a) \geq q_n^*(y)$ for all $a \in A_n(i)$ and $i \in S$, where q_n^* is a nonnegative measurable function with $\int_S q_n^*(y)\mu_n(dy) > 0$;*

(ii) *for each $n \geq 0$, the transition law $P_n(dy|i,a)$ has a density $q_n(y|i,a)$ with respect to a sigma-finite measure μ_n on S, and there exists a nonnegative measurable function q_n^* on S such that $\int_S q_n^*(y)\mu_n(dy) < \infty$ and*

$$q_n(y|i,a) \leq q_n^*(y) \quad \forall\ y \in S, a \in A_n(i),\ \text{and } i \in S;$$

(iii) *for each $n \geq 0$, there exist a state $x_n^* \in S$ and a positive number α_n such that*

$$P_n(\{x_n^*\})|i,a) \geq \alpha_n \quad \forall\ a \in A_n(i) \text{ and } i \in S.$$

Proof. Under condition (i) (or (ii)), let $\delta_n(B) := \int_B q_n^*(y)\mu_n(dy)$ $\forall B \in \mathcal{B}(S)$. Then we can show that Assumption 3.1(i) holds. For all $B \in \mathcal{B}(S)$, let $\delta_n(B) := \alpha_n$ if $x_n^* \in B$ and $\delta_n(B) := 0$ if $x_n^* \notin B$; then Assumption 3.1(i) is satisfied under condition (iii). □

Now, we present our results on the solution to the OEs.

THEOREM 3.3. *If Assumption* 3.1 *holds, then*

(i) *there exist a real number sequence $\{g_n\}$ and a function sequence $\{u_n\} \subset M(S)$ satisfying* (3.2);

(ii)

$$(3.4)\quad g_n = \int_S u_{n+1}(y)\delta_n(dy), \quad |u_n(i)| \leq |R_n(i)|, \quad \text{and} \quad u_n(i) \leq R_n(i) \quad \forall i \in S \text{ and } n \geq 0;$$

(iii) *if $\lim_{k\to\infty} |(1-\delta_n(S))\cdots(1-\delta_{n+k-1}(S))|\|R_{n+k}\| = 0$ $\forall$ $n \geq 0$, then the solution to* (3.2) *satisfying* (3.4) *is unique.*

Proof. For all $i \in S$, $n \geq 0$, and $k \geq 1$, let

$$
\begin{aligned}
u^1(i,n) &:= \sup_{a\in A_n(i)} r_n(i,a),\\
u^2(i,n) &:= \sup_{a\in A_n(i)} \left[r_n(i,a) + \int_S (P_n(dy|i,a) - \delta_n(dy))u^1(y,n+1)\right],\\
&\vdots\ \vdots\ \vdots\\
u^{k+1}(i,n) &:= \sup_{a\in A_n(i)} \left[r_n(i,a) + \int_S (P_n(dy|i,a) - \delta_n(dy))u^k(y,n+1)\right]. \qquad (3.5)
\end{aligned}
$$

By Theorems 18.4 and 13.2 in [14], we have, for any fixed $n \geq 0$ and $k \geq 1$

$$u^k(\cdot,n) \in M(S). \qquad (3.6)$$

From (3.5) and by induction, we can obtain, for any $k \geq 2$,

$$|u^k(i,n) - u^{k-1}(i,n)| \leq |(1-\delta_n(S))\cdots(1-\delta_{n+k-2}(S))| \parallel r_{n+k-1} \parallel . \qquad (3.7)$$

For any $m > l > N_0 \geq 1$, by induction and (3.7), one has

$$
\begin{aligned}
|u^m(i,n) - u^l(i,n)| &\leq \sum_{k=l+1}^{m} |u^k(i,n) - u^{k-1}(i,n)|\\
&\leq \sum_{k=l+1}^{m} |(1-\delta_n(S))\cdots(1-\delta_{n+k-2}(S))| \parallel r_{n+k-1} \parallel\\
&\leq \sum_{k=N_0}^{\infty} |(1-\delta_n(S))\cdots(1-\delta_{n+k-1}(S))| \parallel r_{n+k} \parallel . \qquad (3.8)
\end{aligned}
$$

By condition (3.3), for any $\epsilon > 0$, we can find $N_0 \geq 1$ such that

$$\sum_{k=N_0}^{\infty} |(1-\delta_n(S))\cdots(1-\delta_{n+k-1}(S))| \parallel r_{n+k} \parallel < \epsilon. \qquad (3.9)$$

By combining (3.8) with (3.9), we obtain $|u^m(i,n) - u^l(i,n)| < \epsilon \ \forall\ m > l > N_0$. So $\{u^k(i,n)\}$ is a Cauchy sequence. Hence, there exists an $u^*(i,n)$ such that

$$u^k(i,n) \to u^*(i,n), \quad \text{as } k \to \infty. \qquad (3.10)$$

From (3.6), we have $u^*(\cdot,n) \in M(S)$ for all $n \geq 0$. By (3.5) and induction, for any $k \geq 2$, one has

$$u^k(i,n) \leq r_n(i) + \sum_{t=0}^{k-2} |(1-\delta_n(S))\cdots(1-\delta_{n+t}(S))| \parallel r_{n+t+1} \parallel, \qquad (3.11)$$

which yields

$$u^*(i,n) \leq r_n(i) + \sum_{t=0}^{\infty} |(1-\delta_n(S))\cdots(1-\delta_{n+t}(S))| \parallel r_{n+t+1} \parallel = R_n(i). \qquad (3.12)$$

From (3.8) and (3.10), for any $l \geq 1$, it follows that

$$|u^*(i,n) - u^l(i,n)| \leq \sum_{k=l}^{\infty} |(1-\delta_n(S))\cdots(1-\delta_{n+k-1}(S))| \parallel r_{n+k} \parallel . \qquad (3.13)$$

Hence, as $l \to +\infty$, one obtains

$$\sup_{i\in S} |u^*(i,n) - u^l(i,n)| \to 0 \ \forall n \geq 0. \tag{3.14}$$

From (3.10), together with (3.5) and (3.14), for any $l \geq 0$, we have

$$\begin{aligned}
&\left| \sup_{a\in A_n(i)} \left[r_n(i,a) + \int_S (P_n(dy|i,a) - \delta_n(dy)) u^*(y,n+1) \right] - u^*(i,n) \right| \\
&\leq \left| \sup_{a\in A_n(i)} \left[r_n(i,a) + \int_S (P_n(dy|i,a) - \delta_n(dy)) u^*(y,n+1) \right] - u^*(i,n) \right. \\
&\qquad \left. + u^{l+1}(i,n) - u^{l+1}(i,n) \right| \\
&\leq |u^{l+1}(i,n) - u^*(i,n)| + (1 - \delta_n(S)) \sup_{j\in S} |u^*(j,n+1) - u^l(j,n+1)|. \qquad (3.15)
\end{aligned}$$

Also, from (3.10), (3.14), and (3.15), let $l \to +\infty$; it can be shown that

$$u^*(i,n) = \sup_{a\in A_n(i)} \left[r_n(i,a) + \int_S (P_n(dy|i,a) - \delta_n(dy)) u^*(y,n+1) \right]. \tag{3.16}$$

Let $g_n := \int_S u^*(y,n+1)\delta_n(dy)$ and $u_n(i) := u^*(i,n)$ for all $i \in S$ and $n \geq 0$. Equation (3.16) implies that (3.2) holds. This completes the proof of part (i).

From (3.12), we also have, for all $n \geq 0$ and $i \in S$,

$$u_n(i) = u^*(i,n) \leq R_n(i). \tag{3.17}$$

Similarly, we can derive $|u_n(i) \leq |R_n(i)| \ \ \forall i \in S$ and $n \geq 0$. Hence part (ii) is valid.

Now let us prove part (iii). Let $\{g'_n\}$ and $\{u'_n\}$ be a solution to (3.2) and satisfy the conditions in (iii). By induction we have, for all $i \in S$ and $k \geq 1$,

$$\begin{aligned}
|u_n(i) - u'_n(i)| &\leq |(1 - \delta_n(S)) \cdots (1 - \delta_{n+k-1}(S))| \parallel u_{n+k} - u'_{n+k} \parallel \\
&\leq 2|(1 - \delta_n(S)) \cdots (1 - \delta_{n+k-1}(S))| \parallel R_{n+k} \parallel \to 0, \ \textit{as } k \to \infty,
\end{aligned}$$

which ends the proof. □

REMARK 3.2. *Suppose that $\delta_n(S) \neq 1 \ \forall n \geq 0$. For any fixed $n \geq 0$, considering n as the starting time, let $\beta_k^n := (1-\delta_n(S)) \cdots (1-\delta_{n+k}(S)) \ \forall k \geq 1$ and $\beta_0^n := 1$. Let $r_k^n(i,a) := \beta_k^n r_{n+k}(i,a) \ \forall k \geq 0$. Then $r_k^n(i,a)$ denotes a new discounted reward with respect to the time-dependent discounted factor β_k^n at time $n+k$. For any $k \geq 0$, let $\bar{P}_k^n(\cdot|i,a) := (P_{n+k}(\cdot|i,a) - \delta_{n+k}(\cdot))/(1-\delta_{n+k}(S))$ denote a new transition probability from stage $n+k$ to stage $n+k+1$. Then from* (3.5), (3.10), *and* (3.17), *we can see that for each $n \geq 0$ the meaning of u_n in the OEs is the* optimal expected total reward value *with respect to the new discounted rewards and the new transition probabilities from time period (n, ∞), and that the meaning of g_n in the OEs is the* expected value *of u_{n+1} corresponding to measure δ_n.*

For the existence of the maximum points of the right-hand side of (3.2), as usual we need the continuity-compactness conditions.

ASSUMPTION 3.2. (i) *For each $n \geq 0$, K_n is a closed subset of $S \times A$, and A is compact.*

(ii) *For each* $n \geq 0$, *the reward function* r_n *is upper semicontinuous on* K_n.

(iii) *The function* $v_n(i,a) := \int_S P_n(dy|i,a)v(y)$ *on* K_n *satisfies that* v_n *is upper semicontinuous on* K_n *for every* $n \geq 0$ *and every bounded super semicontinuous function* v *on* S.

THEOREM 3.4. *If Assumptions* 3.1 *and* 3.2 *hold, then we have the following.*

(i) *There exist a number sequence* $\{g_n\}$ *and a upper semicontinuous function sequence* $\{u_n\}$ *satisfying* (3.2), $|u_n(i)| \leq |R_n(i)|, u_n(i) \leq R_n(i) \ \ \forall i \in S$ *and* $n \geq 0$.

(ii) *There exists a Markov policy* $\pi^* = \{f_n^*\} \in \Pi_m^d$ *such that for all* $i \in S$ *and* $n \geq 0$,

$$r_n(i, f_n^*(i)) + \int_S P_n(dy|i, f_n^*(i))u_{n+1}(y) = g_n + u_n(i). \tag{3.18}$$

Proof. By Lemmas 17.11, 5.5, and 5.10 in [14], for any fixed $k, n \geq 0$, we have that $u^k(\cdot, n)$ in the proof of Theorem 3.3 is an upper semicontinuous function on S. Then, similar to the proof of Theorem 3.3, we can prove that part (i) is true. Part (ii) follows from part (i) and Theorem 17.9 in [14]. □

4. Limiting average optimality. In this section we will prove the existence of limiting average ϵ-optimal Markov policies from the OEs (3.2). The approach employed here is rather different from those used in [15, 3, 19, 1]. The martingale theory is adopted to develop our main results.

THEOREM 4.1. *If* $\{g_n\}$ *and* $\{u_n\}$ *are a solution to the OEs* (3.2) *such that* $\lim_{N\to\infty} \frac{E_\pi^i u_N(X_N)}{N} = 0 \ \forall i \in S$ *and* $\pi \in \Pi$, *then*

(i) *for all* $i \in S$,

$$\bar{V}^*(i) \leq \limsup_{N\to\infty} \frac{g_0 + g_1 + \cdots + g_{N-1}}{N}; \tag{4.1}$$

(ii) *for any* $\epsilon \geq 0$, *if a Markov policy* $\pi^* = \{f_n^*\}$ *satisfies*

$$r_n(i, f_n^*(i)) + \int_S P_n(dy|i, f_n^*(i))u_{n+1}(y) \geq g_n + u_n(i) - \epsilon \ \ \forall \ i \in S \text{ and } n \geq 0, \tag{4.2}$$

then $\bar{V}(\pi^*, i) \geq \bar{V}^*(i) - \epsilon \ \ \forall i \in S$;

(iii) *if for all* $i \in S$ *and* $n \geq 0$, *a Markov policy* $\pi^* = \{f_n^*\}$ *satisfies*

$$r_n(i, f_n^*(i)) + \int_S P_n(dy|i, f_n^*(i))u_{n+1}(y) = g_n + u_n(i), \tag{4.3}$$

then $\bar{V}(\pi^*, i) = \bar{V}^*(i) \ \forall i \in S$;

(iv) *if* S *is denumerable, and* $A_n(i)$ *is finite for each* $n \geq 0$ *and* $i \in S$, *then there exists a* $\bar{V}$*-optimal Markov policy.*

Proof. (i) For any fixed $i \in S$, $\pi \in \Pi$, and $n \geq 0$, since $\lim_{N\to\infty} \frac{E_\pi^i u_N(X_N)}{N} = 0$, we have $|E_\pi^i u_n(X_n)| < \infty \ \forall n \geq 0$. From (2.3) and (3.2), we obtain

$$\begin{aligned} E_\pi^i[u_{n+1}(X_{n+1})|X_0, \Delta_0, \ldots, X_n, \Delta_n] &= \int_S u_{n+1}(y)P_n(dy|X_n, \Delta_n) \\ &= r_n(X_n, \Delta_n) + \int_S u_{n+1}(y)P_n(dy|X_n, \Delta_n) - r_n(X_n, \Delta_n) \\ &\leq g_n + u_n(X_n) - r_n(X_n, \Delta_n). \end{aligned} \tag{4.4}$$

By taking expectation operator E_π^i on both sides of (4.4), we have

$$E_\pi^i u_{n+1}(X_{n+1}) \le g_n + E_\pi^i u_n(X_n) - E_\pi^i r_n(X_n, \Delta_n). \tag{4.5}$$

By induction and (4.5), for every $N \ge 1$, it gives

$$E_\pi^i u_{N+1}(X_{N+1}) - u_0(i) + \sum_{n=0}^{N} E_\pi^i r_n(X_n, \Delta_n) \le \sum_{n=0}^{N} g_n. \tag{4.6}$$

Since $\lim_{N\to\infty} \frac{E_\pi^i u_{N+1}(X_{N+1})}{N+1} = 0$, from (4.6) and (2.4), for all $i \in S$ and $\pi \in \Pi$, one has

$$\bar{V}(\pi, i) \le \limsup_{N\to\infty} \frac{g_0 + g_1 + \cdots + g_N}{N+1}. \tag{4.7}$$

By (4.7) and the arbitrariness of $\pi \in \Pi$ and $i \in S$, we can complete the proof of part (i).

(ii) Similar to the proof of (4.5), under the condition of part (ii), for all $n \ge 0$, we have

$$E_{\pi^*}^i u_{n+1}(X_{n+1}) \ge g_n + E_{\pi^*}^i u_n(X_n) - \epsilon - E_{\pi^*}^i r_n(X_n, \Delta_n). \tag{4.8}$$

Along the same lines as the proof of (4.7), for all $i \in S$, we can obtain

$$\bar{V}(\pi^*, i) \ge \limsup_{N\to\infty} \frac{g_0 + g_1 + \cdots + g_N}{N+1} - \epsilon. \tag{4.9}$$

Combining part (i) and (4.9), part (ii) has been proved.

Part (iii) follows from parts (i) and (ii). Part (iv) comes from (iii). □

COROLLARY 4.2. *If Assumption* 3.1 *holds, and, for all* $\pi \in \Pi$ *and* $i \in S$, $\lim_{n\to\infty} \frac{E_\pi^i |R_n|}{n} = 0$, *then all the conclusions of Theorem* 4.1 *hold.*

Proof. The proof can be worked out by Theorem 3.3, together with Theorem 4.1. □

REMARK 4.1. *In* [3] *and* [19], *on the average expected criterion, the following additional assumption is needed to obtain conclusion* (iv) *in Theorem* 4.1: *From each state* i, *at stage* n, *under action* $a \in A_n(i)$, *there exists a finite set* $\{j | P_n(j|i,a) > 0\}$. *That is, only a finite set of states is reachable in one step transition from any state, under any action. Furthermore, the reward functions* r_n, $n \ge 0$ *are required to be uniformly bounded in* n, *i.e.,* $\sup_{n\ge 0} \|r_n\| < \infty$.

THEOREM 4.3. *If* $\{g_n\}$ *and* $\{u_n\}$ *are a solution to the OEs* (3.2) *and satisfy*

(i) $\lim_{N\to\infty} \frac{u_N(X_N)}{N} = 0$, *a.e.-*$P_\pi^i$ $\forall\, i \in S$ *and* $\pi \in \Pi$;

(ii) $E_\pi^i u_n^2(X_n)$ *and* $\sum_{n=1}^{\infty} \frac{E_\pi^i [Y_n^2 | X_0, \Delta_0, \ldots, X_{n-1}, \Delta_{n-1}]}{n^2}$ *are finite* $\forall\, n \ge 0, \pi \in \Pi$, *and* $i \in S$, *where* $Y_n := u_n(X_n) - \int_S u_n(X_n) P_{n-1}(dX_n | X_{n-1}, \Delta_{n-1})$ $\forall n \ge 1$,

then, for all $i \in S$ *and* $\pi \in \Pi$, *we have*

$$\text{(a)} \quad \bar{U}^*(i) \le \limsup_{N\to\infty} \frac{g_0 + g_1 + \cdots + g_{N-1}}{N} \quad \forall i \in S; \tag{4.10}$$

$$\text{(b)} \quad V_s(\pi, i) \le \limsup_{N\to\infty} \frac{g_0 + g_1 + \cdots + g_{N-1}}{N}, \ \text{a.e.-} P_\pi^i. \tag{4.11}$$

Proof. For any $i \in S, a \in A_n(i)$, and $n \geq 0$, let

$$Z_n(i,a) := r_n(i,a) + \int_S u_{n+1}(y) P_n(dy|i,a) - g_n - u_n(i).$$

Since $\{g_n\}$ and $\{u_n\}$ are a solution to the OEs (3.2), we have

$$Z_n(i,a) \leq 0 \quad \forall i \in S, a \in A_n(i), \text{ and } n \geq 0. \tag{4.12}$$

For all $h = (i_0, a_0, \ldots, i_n, a_n, \ldots) \in (S \times A)^\infty$ and $n \geq 0$, let

$$M_n(h) := \begin{cases} r_n(i_n,a_n) + u_{n+1}(i_{n+1}) - g_n - u_n(i_n) - Z_n(i_n,a_n) & \text{if } h_n \in H_n; \\ 0 & \text{if } h_n \notin H_n. \end{cases}$$

From (2.1), we can obtain that $M_n = Y_{n+1}$, a.e.-P_π^i $\forall$ $i \in S, \pi \in \Pi$, and $n \geq 0$. For any $i \in S, \pi \in \Pi$ and $n \geq 0$, by (2.3), we have

$$\begin{aligned} &E_\pi^i[M_n|X_0, \Delta_0, \ldots, X_n, \Delta_n] = E_\pi^i[Y_{n+1}|X_0, \Delta_0, \ldots, X_n, \Delta_n] \\ &= E_\pi^i[u_{n+1}(X_{n+1})|X_0, \Delta_0, \ldots, X_n, \Delta_n] - \int_S u_{n+1}(X_{n+1}) P_n(dX_{n+1}|X_n, \Delta_n) = 0. \end{aligned}$$

Hence, we obtain that $\{\sum_{n=0}^{N-1} M_n, \sigma(X_0, \Delta_0, \ldots, X_{N-1}, \Delta_{N-1})\}$ is a martingale. By $E_\pi^i u_n^2(X_n) < \infty$ $\forall n \geq 0$, we can also derive that $\{\sum_{n=0}^{N-1} M_n\}$ is square-integrable. Obviously, $\{N, \sigma(X_0, \Delta_0, \ldots, X_{N-1}, \Delta_{N-1})\}$ is predictable increasing. By Theorem 7.5.4 in [25], we have, for any $i \in S$ and $\pi \in \Pi$,

$$\lim_{N\to\infty} \frac{\sum_{n=0}^{N-1} M_n}{N} = 0, \text{ a.e.-} P_\pi^i, \tag{4.13}$$

which gives us

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \Big[r_n(X_n, \Delta_n) + u_{n+1}(X_{n+1}) - u_n(X_n) - g_n - Z_n(X_n, \Delta_n) \Big] = 0, \quad \text{a.e.-} P_\pi^i. \tag{4.14}$$

Since $-Z_n(X_n, \Delta_n) \geq 0$, a.e.-$P_\pi^i$, and $\lim_{N\to\infty} \frac{1}{N} u_N(X_N) = 0$, a.e.-$P_\pi^i$ $\forall$ $i \in S, \pi \in \Pi$, and $n \geq 0$, from (4.14) we have

$$\begin{aligned} 0 &\geq \limsup_{N\to\infty} \left[\frac{1}{N} \sum_{n=0}^{N-1} r_n(X_n, \Delta_n) - \frac{1}{N} \sum_{n=0}^{N-1} g_n + \frac{1}{N} u_N(X_N) \right] \\ &\geq \limsup_{N\to\infty} \left[\frac{1}{N} \sum_{n=0}^{N-1} r_n(X_n, \Delta_n) \right] + \liminf_{N\to\infty} \left[-\frac{1}{N} \sum_{n=0}^{N-1} g_n \right], \quad \text{a.e.-} P_\pi^i, \end{aligned}$$

which means

$$\limsup_{N\to\infty} \left[\frac{1}{N} \sum_{n=0}^{N-1} r_n(X_n, \Delta_n) \right] \leq \limsup_{N\to\infty} \left[\frac{1}{N} \sum_{n=0}^{N-1} g_n \right], \quad \text{a.e.-} P_\pi^i. \tag{4.15}$$

Therefore, part (b) is valid.

To prove part (a), we take the expectation operator on both sides of (4.15); then

$$\bar{U}(\pi,i) \le \limsup_{N\to\infty} \frac{g_0+g_1+\cdots+g_{N-1}}{N}. \tag{4.16}$$

Note that π and i are arbitrary; the desired result can be obtained from (4.16). □

THEOREM 4.4. *If $\{g_n\}$ and $\{u_n\}$ are a solution to the OEs* (3.2) *and satisfy the following:*

(i) $\lim_{N\to\infty} \frac{u_N}{N} = 0$, *a.e.-$P_\pi^i \forall i \in S$ and $\pi \in \Pi$, and $\{\frac{u_N}{N}\}$ is bounded;*

(ii) $\sum_{N=1}^{\infty} \frac{E_\pi^i[Y_N^2 | X_0,\Delta_0,\ldots,X_{N-1},\Delta_{N-1}]}{N^2} < \infty \ \ \forall i \in S$ *and* $\pi \in \Pi$,

then we have that

(i) *for any $\epsilon \ge 0$, if a Markov policy $\pi^* = \{f_n^*\}$ satisfies*

$$r_n(i, f_n^*(i)) + \int_S P_n(dy|i, f_n^*(i)) u_{n+1}(y) \ge g_n + u_n(i) - \epsilon \ \ \forall i \in \ \text{and } n \ge 0, \tag{4.17}$$

then π^* is ϵ-limiting average optimal;

(ii) *if for all $i \in S$ and $n \ge 0$, a Markov policy $\pi^* = \{f_n^*\}$ satisfies*

$$r_n(i, f_n^*(i)) + \int_S P_n(dy|i, f_n^*(i)) u_{n+1}(y) = g_n + u_n(i), \tag{4.18}$$

then

(a) $V_s(\pi^*, i) \overset{a.e.\text{-}P_{\pi^*}^i}{=} \bar{V}(\pi^*, i) = \bar{U}(\pi^*, i) = \limsup_{N\to\infty} \frac{g_0+g_1+\cdots+g_{N-1}}{N} \ \ \forall i \in S$, *and*

(b) *the policy $\pi^* = \{f_n^*\}$ is limiting average optimal.*

Proof. By replacing π in the proof of Theorem 4.3 with π^* here, for all $n \ge 0$, we have

$$0 \le -Z_n(X_n, \Delta_N) \le \epsilon, \ \text{a.e.-}P_{\pi^*}. \tag{4.19}$$

Hence, from (4.14) and (4.19), we have

$$\begin{aligned} 0 &= \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left[r_n(X_n,\Delta_n) - g_n - Z_n(X_n,\Delta_n) \right] \\ &\le \liminf_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left[r_n(X_n,\Delta_n) - \frac{1}{N}\sum_{n=0}^{N-1} g_n + \epsilon \right] \\ &\le \limsup_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left[r_n(X_n,\Delta_n) \right] + \liminf_{N\to\infty} \left[-\frac{1}{N}\sum_{n=0}^{N-1} g_n \right] + \epsilon, \end{aligned}$$

which yields

$$\limsup_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} \left[r_n(X_n,\Delta_n) \right] \ge \limsup_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} g_n - \epsilon, \ \ \text{a.e.-}P_\pi^*. \tag{4.20}$$

Taking expectation operator $E_{\pi^*}^i$ on both sides of (4.20), one has

$$\bar{U}(\pi^*, i) \ge \limsup_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} g_n - \epsilon. \tag{4.21}$$

Under condition (i), by the controlled convergence theorem, from $\lim_{N\to\infty}\frac{u_N}{N}=0$, we can obtain that $\lim_{N\to\infty}\frac{E^i_{\pi^*}u_N}{N}=0$. By Theorems 4.1 and 4.3, together with (4.20) and (4.21), we have that part (i) is valid.

We now prove part (ii). Since $Z_n(X_n,\Delta_n)=0$, a.e.-$P^i_{\pi^*}$ $\forall i\in S$, from (4.14), we have

$$\lim_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}\Big[r_n(X_n,\Delta_n)-g_n\Big]=0,\quad \text{a.e.-}P^i_{\pi^*}. \tag{4.22}$$

Hence,

$$\begin{aligned}&\Big|\limsup_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}r_n(X_n,\Delta_n)-\limsup_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}g_n\Big|\\&\le\limsup_{N\to\infty}\Big|\frac{1}{N}\sum_{n=0}^{N-1}[r_n(X_n,\Delta_n)-g_n]\Big|\\&=\lim_{N\to\infty}\Big|\frac{1}{N}\sum_{n=0}^{N-1}[r_n(X_n,\Delta_n)-g_n]\Big|=0,\quad \text{a.e.-}P^i_{\pi^*}.\end{aligned} \tag{4.23}$$

Therefore,

$$\limsup_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}r_n(X_n,\Delta_n)=\limsup_{N\to\infty}\frac{1}{N}\sum_{n=0}^{N-1}g_n,\ \text{a.e.-}P^i_{\pi^*}\ \forall\ i\in S. \tag{4.24}$$

By the result of part (i) and Theorems 4.1 and 4.3, we complete the proof of part (ii). □

THEOREM 4.5. *If either one of the following conditions holds:*

(i) *Assumption* 3.1 *holds and S is denumerable;*

(ii) *Assumption* 3.1 *holds and there exists a probability measure sequence $\{\mu_n\}\subset\mathcal{P}(S)$, such that $P_n(\cdot|i,a)$ is absolutely continuous with respect to μ_n $\forall$ $i\in S$, $a\in A_n(i)$, and $n\ge 0$,*

then, for any $\epsilon>0$, there exists a Markov policy $\pi^\epsilon=\{f^\epsilon_n\}$ such that for all $i,k\in S$ and $n\ge 0$

$$r_n(k,f^\epsilon_n(k))+\int_S P_n(dy|k,f^\epsilon_n(k))u_{n+1}(y)\ge g_n+u_n(k)-\epsilon,\ \text{a.e.-}P^i_{\pi^\epsilon}. \tag{4.25}$$

Proof. (i) Under the conditions of (i), by Theorem 3.3, let $\{g_n\}$ and $\{u_n\}$ be a solution to (3.2) and satisfy $u_n(i)\le R_n(i)$ $\forall n\ge 0$. Hence $u_n(i)+g_n<\infty\forall i\in S$ and $n\ge 0$. Then, for any fixed $\epsilon>0$, $i\in S$, and $n\ge 0$, there exists an $f^\epsilon_n(i)\in A_n(i)$ such that

$$r_n(i,f^\epsilon_n(i))+\int_S P_n(dy|i,f^\epsilon_n(i))u_{n+1}(y)\ge u_n(i)+g_n-\epsilon. \tag{4.26}$$

Let $\pi^\epsilon=\{f^\epsilon_n\}$. Since S is denumerable, $\pi^\epsilon\in\Pi^d_m$, the proof is complete.

(ii) Let

$$\mu(\cdot)=\sum_{n=0}^{\infty}\frac{1}{2^{n+1}}\mu_n(\cdot). \tag{4.27}$$

Then $\mu \in \mathcal{P}(S)$, and $P_n(\cdot|i,a)$ is absolutely continuous with respect to $\mu \forall i \in S, a \in A_n(i)$, and $n \geq 0$. By Theorem 3.3, there exist a number sequence $\{g_n\}$ and a universally measurable function sequence $\{u_n\}$ satisfying (3.2), and $u_n(i) \leq R_n(i)\ \forall i \in S$ and $n \geq 0$. Hence, there exist a measurable function sequence $\{V_n\}$ on S and a measurable subset sequence $\{B_n\}$ such that

$$u_n(x) = V_n(x) \quad \forall x \in B_n^c := S - B_n, \quad \mu(B_n) = 0 \tag{4.28}$$

for all $n \geq 0$. By the absolute continuity of $P_n(\cdot|i,a)$ corresponding to μ, we have

$$P_n(B_{n+1}|i,a) = 0 \quad \forall i \in S,\ a \in A_n(i),\ \text{and } n \geq 0. \tag{4.29}$$

Then we have for all $i \in S, a \in A_n(i)$, and $n \geq 0$,

$$\int_S u_{n+1}(y) P_n(dy|i,a) = \int_S V_{n+1}(y) P_n(dy|i,a). \tag{4.30}$$

Hence, for any $i \in B_n^c$ and $n \geq 0$,

$$g_n + V_n(i) = \sup_{a \in A_n(i)} \{r_n(i,a) + \int_S V_{n+1}(y) P_n(dy|i,a)\}. \tag{4.31}$$

For any $\epsilon > 0$ and $n \geq 0$, let

$$K_n(\epsilon) = \Bigg\{ (i,a) | i \in B_n^c,\ a \in A_n(i),\ \text{and}$$

$$r_n(i,a) + \int_S P_n(dy|i,a) V_{n+1}(y) \geq g_n + V_n(i) - \epsilon \Bigg\} \cup (B_n \times A). \tag{4.32}$$

Obviously, $K_n(\epsilon) \in \mathcal{B}(S) \times \mathcal{B}(A)$. For any $i \in S$ and $n \geq 0$, because of $g_n + V_{n+1}(i) < \infty$, we have $K_n(\epsilon)(i) := \{a \in A_n(i) | (i,a) \in K_n(\epsilon)\} \neq \emptyset$. Therefore, by Lemma 12.12 in [14], there exist an $f_n^\epsilon \in F_n$ and a measurable subset $\tilde{B}_n \subset S$ such that

$$f_n^\epsilon(i) \in K_n(\epsilon)(i) \quad \forall i \in \tilde{B}_n^c \ \text{ and } \ \mu(\tilde{B}_n) = 0.$$

Let $\pi^\epsilon = \{f_n^\epsilon\}$ and $B := \cup_{n=0}^\infty (B_n \cup \tilde{B}_n)$; then we have $B \in \mathcal{B}(S)$ and $\mu(B) = 0$. Hence $P_n(B|i,a) = 0$ for all $i \in S$, $a \in A_n(i)$, and $n \geq 0$. Therefore, for $k \in B^c$ and $n \geq 0$, we have

$$r_n(k, f_n^\epsilon(k)) + \int_S P_n(dy|i, f_n^\epsilon(k)) u_{n+1}(y) \geq g_n + u_n(k) - \epsilon, \tag{4.33}$$

and consequently, for all $i, k \in S$ and $n \geq 0$

$$r_n(k, f_n^\epsilon(k)) + \int_S P_n(dy|i, f_n^\epsilon(k)) u_{n+1}(y) \geq g_n + u_n(k) - \epsilon, \quad \text{a.e.-} P_{\pi^\epsilon}^i, \tag{4.34}$$

which completes the proof. □

To ensure the existence of limiting average ϵ-optimal Markov policies, from the proofs of Theorems 4.4 and 4.5 we introduce the following conditions.

ASSUMPTION 4.1. $\sum_{N=1}^\infty \frac{\|R_N\|^2}{N^2} < \infty$.

ASSUMPTION 4.2. *Either one of the following conditions is true:*

(i) *S is denumerable;*

(ii) *there exists a probability measure sequence $\{\mu_n\} \subset \mathcal{P}(S)$ such that $P_n(\cdot|i,a)$ is absolutely continuous with respect to μ_n $\forall$ $i \in S$, $a \in A_n(i)$, and $n \geq 0$.*

Now we provide the main results of this paper.

THEOREM 4.6. (i) *If Assumptions* 3.1, 3.2, *and* 4.1 *hold, then there exists a limiting average optimal Markov policy.*

(ii) *If Assumptions* 3.1, 4.1, *and* 4.2 *hold, then for any $\epsilon > 0$, there exists a limiting average ϵ-optimal Markov policy.*

Proof. Under Assumption 4.1, we are ready to show that $\lim_{N\to\infty} \frac{\|R_N\|}{N} = 0$. By Theorems 3.3, 3.4, and 4.4, we can complete the proof of part (i). By Theorems 3.3, 4.4, and 4.5, part (ii) can be carried out. □

COROLLARY 4.7. *If the following conditions hold:*

(i) *S is denumerable, every $A_n(i)(i \in S,\ n \geq 0)$ is finite;*

(ii) $\beta := \sup_n(1 - \sum_{j\in S}(\inf_{i\in S}\inf_{a\in A_n(i)})P_n(j|i,a)) < 1$*;*

(iii) $\sup_n \|r_n\| < \infty$,

then there exists a limiting average optimal Markov policy.

Proof. By Theorem 4.6, this corollary can be figured out. □

Now we give an example for which Assumptions 3.1, 3.2, and 4.1 hold here, whereas Assumption 4 in [3, 19] fails to hold.

EXAMPLE 4.1. *Let $S := \{1, 2, \ldots, i, \ldots\}$ and $A_n(i)$ be nonempty finite sets for all $n \geq 0$ and $i \in S$. Suppose that $\sup_{n\geq 0} \|r_n\| < \infty$ and $\delta \in (0,1)$. The transition law P_n is partially defined by*

$$P_n(i|1,a) := \frac{1-\delta}{2^{i-1}}, \qquad n \geq 0,\ a \in A_n(1),\ \text{and } i \geq 2, \tag{4.35}$$

$$P_n(1|i,a) := \delta, \qquad i \in S,\ a \in A_n(i),\ \text{and } n \geq 0. \tag{4.36}$$

From (4.35)*, for each $n \geq 0$ and $a \in A_n(1)$, we have that the set*

$$\{j|P_n(j|1,a) > 0\} = S \tag{4.37}$$

is infinite. This implies that Assumption 4 *in both* [3] *and* [19] *fails to hold.*

Take $\delta_n(j) = \inf_{i\in S}\inf_{a\in A_n(i)} P_n(j|i,a)$ $\forall j \in S$ and $n \geq 0$. From (4.36)*, we have*

$$\begin{aligned}\beta &= \sup_{n\geq 0}\left(1 - \sum_{j\in S}\left(\inf_{i\in S}\inf_{a\in A_n(i)} P_n(j|i,a)\right)\right)\\ &\leq \sup_{n\geq 0}\left(1 - \inf_{i\in S}\inf_{a\in A_n(i)} P_n(1|i,a)\right)\\ &= \sup_{n\geq 0}(1-\delta) < 1. \end{aligned} \tag{4.38}$$

From $\beta < 1$ and $\sup_{n\geq 0}\|r_n\| < \infty$, we can easily verify that all assumptions made in this paper are satisfied for this example.

5. A rolling horizon algorithm. In this section, a rolling horizon algorithm for computing limiting average ϵ-optimal Markov policies is proposed.

In order to guarantee the feasibility of this algorithm, we need the following conditions.

ASSUMPTION 5.1. (i) *S is denumerable, and, for each $n \geq 0$ the set $\{j|\delta_n(j) > 0\} \subset S$ is finite;*

(ii) $\beta := \sup_{n\geq 0} |(1-\delta_n(S))| < 1$, *and* $K := \sup_{n\geq 0} \|r_n\| < \infty$;

(iii) *every* $A_n(i)$ *is finite, and from each state* $i \in S$, *at stage* n, *under action* $a \in A_n(i)$, *there exists a finite set* $\{j|P_n(j|i,a) > 0\}$; *that is, only a finite set of states is reachable in one step transition from any state under any action.*

Now, we are ready to develop the algorithm.

Under Assumptions 3.1 and 5.1, for any $\epsilon > 0$, we can choose a positive integer N_0 such that $\beta^{N_0}\frac{K}{1-\beta} < \frac{\epsilon}{2}$. By (3.13) and (3.17), we have

$$\begin{aligned} |u_n(i) - u^l(i,n)| &\leq \sum_{k=l}^{\infty}(1-\delta_n(S))\cdots(1-\delta_{n+k-1}(S))||r_{n+k}|| \\ &\leq K\sum_{k=l}^{\infty}\beta^l \leq \beta^{N_0}\frac{K}{1-\beta} < \frac{\epsilon}{2} \quad \forall l \geq N_0,\ i \in S, \text{ and } n \geq 0. \end{aligned} \tag{5.1}$$

For any $n \geq 0$ and $i \in S$, let

$$\begin{aligned} V_{n+N_0+1}(i) &:= \max_{a\in A_{n+N_0+1}(i)} r_{n+N_0+1}(i,a), \\ V_{n+N_0}(i) &:= \max_{a\in A_{n+N_0}(i)}\left[r_{n+N_0}(i,a) + \sum_{j\in S}(P_{n+N_0}(j|i,a) - \delta_{n+N_0}(j))V_{n+N_0+1}(j)\right], \\ &\vdots\ \vdots\ \vdots \\ V_n(i) &:= \max_{a\in A_n(i)}\left[r_n(i,a) + \sum_{j\in S}(P_n(j|i,a) - \delta_n(j))V_{n+1}(j)\right]. \end{aligned} \tag{5.2}$$

From (3.5) we know that $V_n(i) = u^{N_0+1}(i,n)$ and $V_{n+1}(i) = u^{N_0}(i,n+1)\ \forall\ i \in S$ and $n \geq 0$. For any $i \in S$ and $n \geq 0$, we may choose $f_n^*(i) \in A_n(i)$ such that

$$V_n(i) = r_n(i, f_n^*(i)) + \sum_{j\in S}(P_n(j|i, f_n^*(i)) - \delta_n(j))V_{n+1}(j). \tag{5.3}$$

From (5.1) and (5.3), for all $i \in S$ and $n \geq 0$, one has

$$r_n(i, f_n^*(i)) + \sum_{j\in S} P_n(j|i, f_n^*(i))u_{n+1}(j) \geq u_n(i) + g_n - \epsilon. \tag{5.4}$$

Let $\pi^* = \{f_n^*\}$; by Theorem 4.4, π^* is limiting average ϵ-optimal.

We may summarize the above discussion in the following result.

THEOREM 5.1. *Under Assumptions* 3.1 *and* 5.1, *for any fixed* $\epsilon > 0, n \geq 0$, *and* $i \in S$, *the action* $f_n^*(i)$ *can be calculated in finite steps and the corresponding Markov policy* $\pi^* := \{f_n^*\}$ *is limiting average* ϵ*-optimal.*

Finally, we can provide a rolling horizon algorithm (RHA) to find a limiting average $\epsilon(>0)$-optimal Markov policy $\pi^* = \{f_n^*\}$ as follows.

Step 1. For $\epsilon > 0$, choose a positive integer N_0 such that $\beta^{N_0}\frac{K}{1-\beta} < \frac{\epsilon}{2}$;

Step 2. For a given $n \geq 0$ and $i \in S$, by (5.2), calculate $V_n(i)$;

Step 3. Select $f_n^*(i) \in A_n(i)$ satisfying (5.3).

REMARK 5.1. *The above algorithm is similar to the one given by Alden and Smith in* [1]. *However, the algorithm in* [1] *is restricted to the case of finite state space. In addition, the proof of convergence of our algorithm is simpler and easier to follow.*

6. Application example. In this section we apply our results from previous sections to the water regulation problem.

EXAMPLE 6.1. *Water is stored in a reservoir with the finite volume M and later expended for irrigation in periods of drought. Suppose that s_n is the yearly quantity of water available to replenish the reservoir. The value of s_n depends on the quantity of rainfall, the character of flooding, the thawing of glaciers and so forth, and it is natural to consider it as a random variable with distribution μ_n. Suppose that the random influx of water s_n is satisfied, that is,* $\inf_{n\geq 0}\mu_n([M,\infty)) > 0$. *At the beginning of the period $(n, n+1)$, having a stock i_n of water, we plan the quantity a_n of water to be used during that period for irrigation. The control by expenditures of water seeks to achieve the largest possible reward. We may suppose that the average harvest over the period n is a function $h_n(i,a)$ of the amount i of the stock of water and of the amount a of water released for irrigation. Then we obtain a model for nonstationary MDPs as follows: The state space S here is the segment $[0,M]$; the same segment serves as the action space A; for each state $i \in S$ the set of feasible actions $A_n(i)$ consists of the points $a \in [0,i]$; the reward function $r_n(i,a)$ is equal to $h_n(i,a)$; and the transition probability P_n is defined as $P_n(B|i,a) = \mu_n(\{s \in S : \min(i-a+s, M) \in B\})$ $\forall B \in \mathcal{B}(S)$ and satisfies that for all $i \in S$ and $a \in A_n(i)$ and $v \in M(S)$,*

$$P_n(\{M\}|i,a) = \mu_n(\{s \in S : s \geq M+a-i\}) \geq \mu_n([M,\infty)); \tag{6.1}$$

$$\begin{aligned} v_n(i,a) &:= \int_S v(y)P_n(dy|i,a) = \int_S v(h(i,a,s))\mu_n(ds) \\ &= \int_0^{M-i+a} v(i-a+s)\mu_n(ds) + v(M)\mu_n((M-i+a,\infty)), \end{aligned} \tag{6.2}$$

where $h(i,a,s) := \min\{i-a+s, M\}$ $\forall i \in S, a \in [0,i]$, and $s \geq 0$. Suppose that $h_n(i,a)$ is bounded upper semicontinuous on $K := \{(i,a) : i \in S, a \in A(i)\}$, and $\sup_{n\geq 0}\|r_n\| < \infty$. Since $h(i,a,s)$ is continuous in i and a, by [6, *pp.* 51–53], *from* (6.1) *and* (6.2) *we know that Assumption* 3.2 *holds for this model. Noticing that* $\inf_{n\geq 0}\mu_n([M,\infty)) > 0$ *and* $\sup_{n\geq 0}\|r_n\| < \infty$, *by Assumption* 3.1 *and* (6.1) *we can derive that Assumptions* 4.1 *and* 4.2 *are satisfied. Hence, by Theorem* 4.6, *it follows that* a limiting average optimal Markov policy exists for this water regulation problem.

Acknowledgments. The authors wish to thank Professor Jerzy Filar for a number of discussions and his helpful suggestions. The first author is grateful for the hospitality during his visit to the Centre for Industrial and Applicable Mathematics, the University of South Australia. The authors are also indebted to the associate editor, Professor Andrzej Ruszczyński, and the anonymous referees for many valuable comments and suggestions that have improved the presentation.

REFERENCES

[1] M. ALDEN AND R. L. SMITH, *Rolling horizon procedures in nonhomogeneous Markov decision processes*, Oper. Res., 40 (1992), pp. 183–194.

[2] A. ARAPOSTATHIS, V. S. BORKER, E. FERNÁNDEZ-GAUCHERAND, M. K. GHOSH, AND S. I. MARCUS, *Discrete-time controlled Markov processes with average cost criterion: A survey*, SIAM J. Control Optim., 31 (1993), pp. 282–344.

[3] J. BEAN, R. SMITH, AND J. LASSERRE, *Denumerable state nonhomogeneous Markov decision processes*, J. Math. Anal. Appl., 153 (1990), pp. 64–77.

[4] K. J. BIERTH, *An expected average reward criterion*, Stochastic Process Appl., 26 (1987), pp. 123–140.

[5] D. BLACKWELL, *Discrete dynamic programming*, Ann. Math. Statist., 33 (1962), pp. 719–726.

[6] E. B. Dynkin and A. A. Yushkevich, *Controlled Markov Processes*, Springer-Verlag, New York, 1979.
[7] E. Feinberg and H. Park, *Finite state Markov decision model with average reward criteria*, Stochastic Process. Appl., 49 (1994), pp. 159–177.
[8] J. A. Filar, D. Krass, and K. W. Ross, *Percentile performance criteria for limiting average Markov decision processes*, IEEE Trans. Automat. Control, 40 (1995), pp. 2–10.
[9] X. P. Guo, *Nonstationary MDP average model with incomplete information*, J. Math. Statist. Appl. Prob., 10 (1995), pp. 15–23.
[10] X. P. Guo, *The uniqueness of optimal policies for general Markov decision processes*, Chinese J. Appl. Probab. Statist., 14 (1998), pp. 258–265 (in Chinese).
[11] X. P. Guo, *Nonstationary denumerable state Markov decision processes—with average variance criterion*, Math. Methods Oper. Res., 49 (1999), pp. 87–96.
[12] O. Hernandez-Lerma, *Adaptive Markov Controlled Processes*, Springer-Verlag, New York, 1989.
[13] O. Hernandez-Lerma and J. B. Lasserre, *Discrete-Time Markov Controlled Processes*, Springer-Verlag, New York, 1996.
[14] K. Hinderer, *Foundations of Nonstationary Dynamic Programming with Discrete Time Parameter*, Springer-Verlag, New York, 1970.
[15] W. Hopp, J. Bean, and R. Smith, A new optimality criterion for nonhomogeneous Markov decision processes, Oper. Res., 35 (1987), pp. 875–883.
[16] Z. T. Hou and X. P. Guo, *Markov Decision Processes*, Science and Technology Press, Hunan, China, 1998.
[17] M. Kurano, *Markov decision processes with a Borel measurable cost function—the average case*, Math. Oper. Res., 11(1986), pp. 309–320.
[18] M. Kurano, *Markov decision processes with a minimum-variance criterion*, J. Math. Anal. Appl., 123 (1987), pp. 572–583.
[19] Y. Park, J. Bean, and R. Smith, *Optimal average value convergence in nonhomogeneous Markov decision processes*, J. Math. Anal. Appl., 179 (1993), pp. 525–536.
[20] M. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley, New York, 1994.
[21] R. Cavazos-Cadena and E. Fernández-Gaucherand, *Denumerable controlled Markov chains with average reward criterion: Sample path optimality*, ZOR—Math. Methods Oper. Res., 41 (1995), pp. 89–108.
[22] S. M. Ross, *Introduction to Stochastic Dynamic Programming*. Academic Press, New York, London, 1983.
[23] K. W. Ross and R. Varadarajian, *Markov decision processes with sample path constraints: The communicating case*, Oper. Res., 37 (1989), pp. 780–790.
[24] K. W. Ross and R. Varadarajian, *Markov decision processes with sample path constraints: A decomposition approach*, Math. Oper. Res., 16 (1991), pp. 195–207.
[25] A. N. Shiryayev, *Probability*, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1984.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1054–1064

ON THE GLOBAL CONVERGENCE OF THE BFGS METHOD FOR NONCONVEX UNCONSTRAINED OPTIMIZATION PROBLEMS*

DONG-HUI LI† AND MASAO FUKUSHIMA‡

Abstract. This paper is concerned with the open problem of whether the BFGS method with inexact line search converges globally when applied to nonconvex unconstrained optimization problems. We propose a cautious BFGS update and prove that the method with either a Wolfe-type or an Armijo-type line search converges globally if the function to be minimized has Lipschitz continuous gradients.

Key words. unconstrained optimization, BFGS method, global convergence

AMS subject classifications. 90C30, 65K05

PII. S1052623499354242

1. Introduction.

The BFGS method is a well-known quasi-Newton method for solving unconstrained optimization problems [5, 7]. Because of favorable numerical experience and fast theoretical convergence, it has become a method of choice for engineers and mathematicians who are interested in solving optimization problems.

Local convergence of the BFGS method has been well established [3, 4]. The study on global convergence of the BFGS method has also made good progress. In particular, for convex minimization problems, it has been shown that the iterates generated by the BFGS method are globally convergent if the exact line search or some special inexact line search is used [1, 2, 6, 9, 16, 17, 18]. On the other hand, little is known concerning global convergence of the BFGS method for nonconvex minimization problems. Indeed, so far, no one has proved global convergence of the BFGS method for nonconvex minimization problems or has given a counter example that shows nonconvergence of the BFGS method. Whether the BFGS method converges globally for a nonconvex function remains unanswered. This open problem has been mentioned many times and is currently regarded as one of the most fundamental open problems in the theory of quasi-Newton methods [8, 15].

Recently, the authors [12] proposed a modified BFGS method and established its global convergence for nonconvex unconstrained optimization problems. The authors [11] also proposed a globally convergent Gauss–Newton-based BFGS method for symmetric nonlinear equations that contain unconstrained optimization problems as a special case. The results obtained in [11] and [12] positively support the open problem. However, the original question still remains unanswered.

The purpose of this paper is to study this problem further. We introduce a *cautious update* in the BFGS method and prove that the method with a Wolfe-type or an Armijo-type line search converges globally if the function to be minimized has Lipschitz continuous gradients. Moreover, under appropriate conditions, we show that the cautious update eventually reduces to the ordinary update.

*Received by the editors April 11, 1999; accepted for publication (in revised form) November 6, 2000; published electronically May 10, 2001.
http://www.siam.org/journals/siopt/11-4/35424.html

†Department of Applied Mathematics, Hunan University, Changsha, China 410082 (dhli@mail.hunu.edu.cn). This work was done while this author was visiting Kyoto University.

‡Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan (fuku@amp.i.kyoto-u.ac.jp).

In the next section, we present the BFGS method with a cautious update. In section 3, we prove global convergence and, under additional assumptions, superlinear convergence of the algorithm. In section 4, we report some numerical results with the algorithm.

We introduce some notation: For a real-valued function $f : R^n \to R$, $g(x)$ and $G(x)$ denote the gradient and Hessian matrix of f at x, respectively. For simplicity, $g(x_k)$ and $G(x_k)$ are often denoted by g_k and G_k, respectively. For a vector $x \in R^n$, $\|x\|$ denotes its Euclidean norm.

2. Algorithm. Let $f : R^n \to R$ be continuously differentiable. Consider the following unconstrained optimization problem:

$$\min f(x), \quad x \in R^n. \tag{2.1}$$

The ordinary BFGS method for (2.1) generates a sequence $\{x_k\}$ by the iterative scheme:

$$x_{k+1} = x_k + \lambda_k p_k, \qquad k = 0, 1, 2, \ldots,$$

where p_k is the BFGS direction obtained by solving the linear equation

$$B_k p + g_k = 0. \tag{2.2}$$

The matrix B_k is updated by the BFGS formula

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{y_k^T s_k}, \tag{2.3}$$

where $s_k = x_{k+1} - x_k$ and $y_k = g_{k+1} - g_k$. A good property of formula (2.3) is that B_{k+1} inherits the positive definiteness of B_k as long as $y_k^T s_k > 0$. The condition $y_k^T s_k > 0$ is guaranteed to hold if the stepsize λ_k is determined by the exact line search

$$f(x_k + \lambda_k p_k) = \min_{\lambda > 0} f(x_k + \lambda p_k) \tag{2.4}$$

or the Wolfe-type inexact line search

$$\begin{cases} f(x_k + \lambda_k p_k) \le f(x_k) + \sigma_1 \lambda_k g(x_k)^T p_k, \\ g(x_k + \lambda_k p_k)^T p_k \ge \sigma_2 g(x_k)^T p_k, \end{cases} \tag{2.5}$$

where σ_1 and σ_2 are positive constants satisfying $\sigma_1 < \sigma_2 < 1$. In addition, if $\lambda_k = 1$ satisfies (2.5), we take $\lambda_k = 1$. Global convergence of the BFGS method with the line search (2.4) or (2.5) for convex minimization problems has been studied in [1, 2, 6, 9, 16, 17, 18].

Another important inexact line search is the Armijo-type line search that finds a λ_k that is the largest value in the set $\{\rho^i | i = 0, 1, \ldots\}$ such that the inequality

$$f(x_k + \lambda_k p_k) \le f(x_k) + \sigma \lambda_k g(x_k)^T p_k \tag{2.6}$$

is satisfied, where σ and ρ are constants such that $\sigma, \rho \in (0, 1)$. The Armijo-type line search does not ensure the condition $y_k^T s_k > 0$ and hence B_{k+1} is not necessarily positive definite even if B_k is positive definite. In order to ensure the positive definiteness

of B_{k+1}, the condition $y_k^T s_k > 0$ is sometimes used to decide whether or not B_k is updated . More specifically, B_{k+1} is determined by

$$B_{k+1} = \begin{cases} B_k - \dfrac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \dfrac{y_k y_k^T}{y_k^T s_k} & \text{if } y_k^T s_k > 0, \\ B_k & \text{otherwise.} \end{cases} \tag{2.7}$$

Computationally, the condition $y_k^T s_k > 0$ is often replaced by the condition $y_k^T s_k > \eta$, where $\eta > 0$ is a small constant. In this paper, we propose a cautious update rule similar to the above and establish a global convergence theorem for nonconvex problems. For the sake of motivation, we state a lemma due to Powell [17].

LEMMA 2.1 (Powell [17]). *If the BFGS method with the line search* (2.5) *is applied to a continuously differentiable function f that is bounded below, and if there exists a constant $M > 0$ such that the inequality*

$$\frac{\|y_k\|^2}{y_k^T s_k} \le M \tag{2.8}$$

holds for all k, then

$$\liminf_{k\to\infty} \|g(x_k)\| = 0. \tag{2.9}$$

Notice that if f is twice continuously differentiable and convex, then (2.8) always holds whenever $\{x_k\}$ is bounded. Therefore, global convergence of the BFGS method follows immediately from Lemma 2.1. However, in the case where f is nonconvex, it seems difficult to guarantee (2.8). This is probably the main reason why global convergence of the BFGS method has yet to be proved. In [12], the authors proposed a modified BFGS method by using $\tilde{y}_k = C\|g_k\|s_k + (g_{k+1} - g_k)$ with a constant $C > 0$ instead of y_k in the update formula (2.3). Global convergence of the modified BFGS method in [12] is proved without the convexity assumption on f by means of Lemma 2.1 with a contradictory assumption that $\{\|g_k\|\}$ are bounded away from zero. We further study global convergence of the BFGS method for (2.1). Instead of modifying the method, we introduce a cautious update rule in the ordinary BFGS method. To be precise, we determine B_{k+1} by

$$B_{k+1} = \begin{cases} B_k - \dfrac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \dfrac{y_k y_k^T}{y_k^T s_k} & \text{if } \dfrac{y_k^T s_k}{\|s_k\|^2} \ge \epsilon \|g_k\|^\alpha, \\ B_k & \text{otherwise,} \end{cases} \tag{2.10}$$

where ϵ and α are positive constants.

Now, we state the BFGS method with the cautious update.

ALGORITHM 1.

Step 0 Choose an initial point $x_0 \in R^n$ and an initial symmetric and positive definite matrix $B_0 \in R^{n\times n}$. Choose constants $0 < \sigma_1 < \sigma_2 < 1$, $\alpha > 0$, and $\epsilon > 0$. Let $k := 0$.

Step 1 Solve the linear equation (2.2) to get p_k.

Step 2 Determine a stepsize $\lambda_k > 0$ by (2.5) or (2.6).

Step 3 Let the next iterate be $x_{k+1} := x_k + \lambda_k p_k$.

Step 4 Determine B_{k+1} by (2.10).

Step 5 Let $k := k + 1$ and go to Step 1.

Remark. It is not difficult to see from (2.10) that the matrix B_k generated by Algorithm 1 is symmetric and positive definite for all k, which in turn implies that

$\{f(x_k)\}$ is a decreasing sequence whichever line search (2.5) or (2.6) is used. Moreover, we have from (2.5) or (2.6)

$$-\sum_{k=0}^{\infty} g_k^T s_k < \infty \tag{2.11}$$

if f is bounded below. In particular, we have

$$-\lim_{k\to\infty} \lambda_k g_k^T p_k = -\lim_{k\to\infty} g_k^T s_k = 0. \tag{2.12}$$

3. Global convergence. In this section, we prove global convergence of Algorithm 1 under the following assumption, which we assume throughout this section.

Assumption A. The level set

$$\Omega = \{x \in R^n \mid f(x) \le f(x_0)\}$$

is bounded, the function f is continuously differentiable on Ω, and there exists a constant $L > 0$ such that

$$\|g(x) - g(y)\| \le L\|x - y\| \qquad \forall x, y \in \Omega. \tag{3.1}$$

Since $\{f(x_k)\}$ is a decreasing sequence, it is clear that the sequence $\{x_k\}$ generated by Algorithm 1 is contained in Ω.

For the sake of convenience, we define the index set

$$\bar{K} = \left\{ i \;\middle|\; \frac{y_i^T s_i}{\|s_i\|^2} \ge \epsilon \|g_i\|^\alpha \right\}. \tag{3.2}$$

By means of $\bar{K}$, we may rewrite (2.10) as

$$B_{k+1} = \begin{cases} B_k - \dfrac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \dfrac{y_k y_k^T}{y_k^T s_k} & \text{if } k \in \bar{K}, \\ B_k & \text{otherwise.} \end{cases} \tag{3.3}$$

Now we proceed to establishing global convergence of Algorithm 1. First, we show the following convergence theorem.

THEOREM 3.1. *Let $\{x_k\}$ be generated by Algorithm* 1. *If there are positive constants $\beta_1, \beta_2, \beta_3 > 0$ such that the relations*

$$\|B_k s_k\| \le \beta_1 \|s_k\|, \qquad \beta_2 \|s_k\|^2 \le s_k^T B_k s_k \le \beta_3 \|s_k\|^2 \tag{3.4}$$

hold for infinitely many k, then we have

$$\liminf_{k\to\infty} \|g(x_k)\| = 0. \tag{3.5}$$

Proof. Since $s_k = \lambda_k p_k$, it is clear that (3.4) holds true if s_k is replaced by p_k. Let $\mathcal{K}$ be the set of indices k such that (3.4) hold. It is not difficult to deduce from (2.2) and (3.4) that for each $k \in \mathcal{K}$

$$\beta_2 \|p_k\| \le \|g(x_k)\| \le \beta_1 \|p_k\|. \tag{3.6}$$

Consider the case where the Armijo-type line search (2.6) is used with the backtracking parameter ρ. If $\lambda_k \ne 1$, then we have

$$f(x_k + \rho^{-1}\lambda_k p_k) - f(x_k) > \sigma \rho^{-1} \lambda_k g(x_k)^T p_k. \tag{3.7}$$

By the mean-value theorem, there is a $\theta_k \in (0,1)$ such that

$$\begin{aligned} & f(x_k + \rho^{-1}\lambda_k p_k) - f(x_k) \\ & = \rho^{-1}\lambda_k g(x_k + \theta_k \rho^{-1}\lambda_k p_k)^T p_k \\ & = \rho^{-1}\lambda_k g(x_k)^T p_k + \rho^{-1}\lambda_k (g(x_k + \theta_k \rho^{-1}\lambda_k p_k) - g(x_k))^T p_k \\ & \le \rho^{-1}\lambda_k g(x_k)^T p_k + L\rho^{-2}\lambda_k^2 \|p_k\|^2, \end{aligned} \tag{3.8}$$

where $L > 0$ is the Lipschitz constant of g. Substituting (3.8) into (3.7), we get for any $k \in \mathcal{K}$

$$\lambda_k \ge \frac{-(1-\sigma)\rho g(x_k)^T p_k}{L\|p_k\|^2} = \frac{(1-\sigma)\rho p_k^T B_k p_k}{L\|p_k\|^2} \ge (1-\sigma)\beta_2 L^{-1}\rho.$$

This means that for each $k \in \mathcal{K}$, we have

$$\lambda_k \ge \min\{1, (1-\sigma)\beta_2 L^{-1}\rho\} > 0. \tag{3.9}$$

Consider the case where the Wolfe-type line search (2.5) is used. It follows from the second inequality of (2.5) and the Lipschitz continuity of g that

$$L\lambda_k \|p_k\|^2 \ge (g(x_k + \lambda_k p_k) - g(x_k))^T p_k \ge -(1-\sigma_2) g(x_k)^T p_k.$$

This implies

$$\lambda_k \ge \frac{-(1-\sigma_2) g(x_k)^T p_k}{L\|p_k\|^2} = \frac{(1-\sigma_2) p_k^T B_k p_k}{L\|p_k\|^2} \ge (1-\sigma_2)\beta_2 L^{-1}. \tag{3.10}$$

The inequalities (3.10) together with (3.9) show that $\{\lambda_k\}_{k\in\mathcal{K}}$ is bounded away from zero whenever the line search (2.5) or (2.6) is used. It then follows from (2.2) and (2.12) that $p_k^T B_k p_k = -g(x_k)^T p_k \to 0$ as $k \to \infty$ with $k \in \mathcal{K}$. This together with (3.4) and (3.6) implies (3.5). □

Theorem 3.1 indicates that to prove global convergence of Algorithm 1, it suffices to show that there are positive constants $\beta_1, \beta_2, \beta_3$ such that (3.4) holds for infinitely many k. To this end, we quote the following useful result [1, Theorem 2.1].

LEMMA 3.2. *Let B_k be updated by the BFGS formula* (2.3). *Suppose B_0 is symmetric and positive definite and there are positive constants $m \le M$ such that for all $k \ge 0$, y_k and s_k satisfy*

$$\frac{y_k^T s_k}{\|s_k\|^2} \ge m, \quad \frac{\|y_k\|^2}{y_k^T s_k} \le M. \tag{3.11}$$

Then there exist constants $\beta_1, \beta_2, \beta_3 > 0$ such that, for any positive integer t, (3.4) *holds for at least $\lceil t/2 \rceil$ values of $k \in \{1, \ldots, t\}$.*

By using Lemma 3.2 and Theorem 3.1, we can establish the following global convergence theorem for Algorithm 1.

THEOREM 3.3. *Let Assumption A hold and $\{x_k\}$ be generated by Algorithm* 1. *Then* (3.5) *holds.*

Proof. By Theorem 3.1, it suffices to show that there are infinitely many indices k satisfying (3.4).

If $\bar{K}$ is finite, then B_k remains constant after a finite number of iterations. Since B_k is symmetric and positive definite for each k, it is obvious that there are constants $\beta_1, \beta_2, \beta_2 > 0$ such that (3.4) holds for all k sufficiently large.

Consider the case where $\bar{K}$ is infinite. For the sake of contradiction, we suppose that (3.5) is not true. That is, there is a constant $\delta > 0$ such that $\|g_k\| \geq \delta$ for all k. It then follows from (3.2) that $y_k^T s_k \geq \epsilon \delta^\alpha \|s_k\|^2$ holds for all $k \in \bar{K}$. This together with (3.1) implies that for any $k \in \bar{K}$, we have

$$\frac{\|y_k\|^2}{y_k^T s_k} \leq \frac{L^2}{\epsilon \delta^\alpha}.$$

Applying Lemma 3.2 to the matrix subsequence $\{B_k\}_{k \in \bar{K}}$, it is clear that there are constants $\beta_1, \beta_2, \beta_3 > 0$ such that (3.4) holds for infinitely many k. The proof is then complete. □

Theorem 3.3 shows that there exists a subsequence of $\{x_k\}$ converging to a stationary point x^* of (2.1). If f is convex, then x^* is a global minimum of f. Since the sequence $\{f(x_k)\}$ converges, it is clear that every accumulation point of $\{x_k\}$ is a global optimal solution of (2.1). That is, we have the following corollary.

COROLLARY 3.4. *Let Assumption* A *hold and* $\{x_k\}$ *be generated by Algorithm* 1. *If* f *is convex, then the whole sequence* $\{g_k\}$ *converges to zero. Consequently, every accumulation point of* $\{x_k\}$ *is a global optimal solution of* (2.1).

In the case where f is nonconvex, Corollary 3.4 is not guaranteed. The following theorem shows that if some additional conditions are assumed, then the whole sequence $\{x_k\}$ converges to a local optimal solution of (2.1).

THEOREM 3.5. *Let* f *be twice continuously differentiable. Suppose that* $s_k \to 0$. *If there exists an accumulation point* x^* *of* $\{x_k\}$ *at which* $g(x^*) = 0$ *and* $G(x^*)$ *is positive definite, then the whole sequence* $\{x_k\}$ *converges to* x^*. *If in addition,* G *is Hölder continuous and the parameters in the line searches satisfy* $\sigma, \sigma_1 \in (0, 1/2)$, *then the convergence rate is superlinear.*

Proof. The assumptions particularly imply that x^* is a strict local optimal solution of (2.1). Since $\{f(x_k)\}$ converges, it follows that x^* is an isolated accumulation point of $\{x_k\}$. Then, by the assumption that $\{s_k\}$ converges to zero, the whole sequence $\{x_k\}$ converges to x^*. Hence $\{g_k\}$ tends to zero and, by the positive definiteness of $G(x^*)$, the matrices

$$A_k \triangleq \int_0^1 G(x_k + \tau s_k) d\tau$$

are uniformly positive definite for all k large enough. Moreover, by the mean-value theorem, we have $y_k = A_k s_k$. Therefore, there is a constant $\bar{m} > 0$ such that $y_k^T s_k \geq \bar{m} \|s_k\|^2$, which implies that when k is sufficiently large, the condition $y_k^T s_k / \|s_k\|^2 \geq \epsilon \|g_k\|^\alpha$ is always satisfied. This means that Algorithm 1 reduces to the ordinary BFGS method when k is sufficiently large. The superlinear convergence of Algorithm 1 then follows from the related theory in [1, 2, 17]. □

4. Numerical experiments. This section reports some numerical experiments with Algorithm 1. We tested the algorithm on some problems [14] taken from MATLAB with given initial points. These problems can be obtained at the website ftp://ftp.mathworks.com/pub/contrib/v4/optim/uncprobs/. We applied Algorithm 1, which will be called the CBFGS method (C stands for cautious), with the Armijo-type or the Wolfe-type line search or polynomial search, to these problems and compared it with the ordinary BFGS method. We used the condition $\|g(x_k)\| \leq 10^{-6}$ as the stopping criterion. For each problem, we chose the initial matrix $B_0 = I$, i.e., the unit matrix. For each problem, the parameters common to the two methods

were set identically. Specifically, we set $\sigma_1 = 0.1$ and $\sigma_2 = 0.9$ in the Wolfe-type line search (2.5), and $\sigma = 0.01$ in the Armijo-type line search (2.6). We let $\epsilon = 10^{-6}$ in the cautious update (2.10). We let $\rho = 0.5$ in the Armijo-type line search. Due to roundoff error, sometimes the directions generated by the algorithms may be not descent. We then use the steepest descent direction instead of the related BFGS (or CBFGS) direction if $g_k p_k > -10^{-14}$.

As to the parameter α in the cautious update (2.10), we first let $\alpha = 0.01$ if $\|g_k\| \geq 1$, and $\alpha = 3$ if $\|g_k\| < 1$. We call this choice Rule 1. Rule 1 is intended to make the cautious update closer to the original BFGS update. It is not difficult to see that the convergence theorems in section 3 remain true if we choose α according to this rule. Indeed, even if α varies in an interval $[\mu_1, \mu_2]$ with $\mu_1 > 0$, all the theorems in section 3 hold true. More generally, as an anonymous referee pointed out, the convergence theorems in section 3 remain true if $\epsilon\|g_k\|^\alpha$ is replaced by a general forcing function $\phi(\|g_k\|)$, which is strictly monotone with $\phi(0) = 0$. We also tested the cautious update (2.10) with $\alpha = 1$ always, which we call Rule 2.

Tables 1, 2, and 3 show the computational results, where the columns have the following meanings:

Problem:	the name of the test problem in MATLAB;
Dim:	the dimension of the problem;
CBFGS^1:	the number of iterations for the cautious BFGS method with Rule 1;
CBFGS^2:	the number of iterations for the cautious BFGS method with Rule 2;
BFGS:	the number of iterations for the BFGS method;
L-BFGS:	the number of iterations for the L-BFGS method [13];
off:	the number of k's such that $y_k^T s_k/\|s_k\|^2 < \epsilon\|g_k\|^\alpha$ (CBFGS),
	the number of k's such that $y_k^T s_k/\|s_k\|^2 < \epsilon$ (BFGS with Armijo search);
SD:	the number of iterations for which the steepest descent direction used;
fnum:	the number of function evaluations;
gnum:	the number of gradient evaluations.

We first tested the CBFGS method and the BFGS method through a MATLAB code we ourselves wrote. The results are given in Tables 1 and 2. Tables 1 and 2 show the performance of the CBFGS method and the BFGS method with the Armijo-type line search and the Wolfe-type line search, respectively. From the results, we see that the CBFGS method is compatible with the ordinary BFGS method. Moreover, for the problem "meyer," the BFGS method did not terminate regularly, while the CBFGS method did. However, this does not mean that the ordinary BFGS method with the Wolfe search fails to converge when applied to solve this problem. If we adjust the parameters appropriately, then starting from the same initial point, the ordinary BFGS method with the Wolfe search also converges to a station point of this "meyer."

We then edited a MATLAB code for the CBFGS method with a polynomial line search based on a standard low storage BFGS code called L-BFGS given by Kelley [13] (http://www.siam.org/catalog/mcc12/fr18.htm). The test results are given in Table 3. Table 3 shows that for the test problems, the CBFGS method converges to a stationary point of the test problem if the L-BFGS method does.

We observed that the condition in the cautious update was usually satisfied, which suggests that the ordinary BFGS method is generally "cautious," and hence it seldom fails in practice. The results also show that the choice of the parameter α affects the performance of the method. Moreover, if we choose it appropriately, then the condition in the cautious update is almost always satisfied and the CBFGS method essentially reduces to the ordinary BFGS method. However, when it was violated too

TABLE 1
Test results for CBFGS/BFGS methods with Armijo search.

		CBFGS[1]				CBFGS[2]				BFGS			
Problem	Dim	Iter	off	SD	fnum	Iter	off	SD	fnum	Iter	off	SD	fnum
badscb	2	42	0	1	91	-	-	-	-	42	0	1	91
badscp	2	193	7	2	332	277	107	3	452	474	259	34	1335
band	10	-	-	-	-	-	-	-	-	-	-	-	-
bard	3	23	0	0	34	23	0	0	34	26	5	0	40
bd	4	-	-	-	-	-	-	-	-	-	-	-	-
beale	2	15	0	0	24	15	0	0	24	15	0	0	24
biggs	6	42	2	0	52	42	0	0	52	42	2	0	52
box	3	30	0	0	40	30	0	0	40	30	0	0	40
bv	10	18	0	0	39	18	0	0	39	18	0	0	38
froth	2	10	0	0	22	10	0	0	22	13	0	0	51
gauss	3	4	0	0	7	4	0	0	7	4	0	0	7
gulf	3	1	0	0	4	1	0	0	4	1	0	0	2
helix	3	27	0	0	54	27	0	0	54	31	0	0	59
ie	10	11	0	0	14	11	0	0	14	10	0	0	12
ie	100	12	0	0	15	12	0	0	15	12	0	0	14
jensam	2	12	0	0	24	12	0	0	24	12	0	0	24
kowosb	4	28	0	0	32	28	0	0	32	29	0	0	32
lin	10	1	0	0	3	1	0	0	3	1	0	0	3
lin	100	1	0	0	3	1	0	0	3	1	0	0	3
lin1	10	2	0	0	21	2	0	0	21	2	0	0	21
lin0	10	2	0	0	19	2	0	0	19	2	0	0	19
meyer	3	6	0	2	57	2	0	0	19	6	0	2	57
osb1	5	50	0	2	121	32	28	29	191	50	0	2	121
osb2	11	53	0	0	80	53	0	0	80	58	0	0	84
pen1	10	152	3	0	215	152	3	0	215	154	3	0	216
pen1	100	279	1	0	406	279	1	0	406	299	2	0	441
pen2	10	917	0	0	1338	917	0	0	1338	571	0	0	851
rose	2	34	0	0	54	34	0	0	54	34	0	0	54
rosex	100	407	0	1	1148	407	0	0	1148	405	0	1	1134
sing	4	29	0	0	52	29	0	0	52	29	0	0	52
singx	400	191	0	0	633	191	0	0	633	182	0	0	590
trid	10	19	0	0	74	19	0	0	74	42	1	0	94
trid	100	112	0	0	637	121	0	0	637	145	0	0	650
trig	10	26	0	0	27	26	0	0	27	26	0	0	27
trig	100	48	0	0	51	48	0	0	51	48	0	0	51
vardim	10	13	0	0	35	13	0	0	35	13	0	0	35
watson	12	58	0	0	87	58	0	0	87	58	0	0	87
watson	20	55	0	0	91	55	0	0	91	55	0	0	91
wood	4	52	0	0	97	52	0	0	97	28	0	1	73

often, the CBFGS method's performance was worse than the BFGS method, even failing to converge.

5. Conclusion. We have proposed a cautious BFGS update and shown that the method converges globally with the Wolfe-type line search or the Armijo-type line search. The method retains the scale-invariance property of the original BFGS method, except for a minor scale dependence of the skipping condition in the cautious update (2.10). Moreover, the cautious update makes B_{k+1} inherit the positive definiteness of B_k no matter what line search is used. The established global convergence theorems do not rely on the convexity assumption on the objective function. The reported numerical results show that the BFGS method with the proposed cautious update is comparable to the ordinary BFGS method. Moreover, the conditions used

TABLE 2
Test results for CBFGS/BFGS methods with Wolfe search.

		CBFGS[1]			CBFGS[2]			BFGS		
Problem	Dim	Iter(off, SD)	fnum	gnum	Iter(off, SD)	fnum	gnum	Iter(SD)	fnum	gnum
badscb	2	5 (5, 0)	34	12	-	-	-	26(1)	114	175
badscp	2	319(199, 8)	1391	2196	180(20, 7)	782	1223	174(10)	587	827
band	10	-	-	-	-	-	-	-	-	-
bard	3	14(0, 0)	46	47	14(0, 0)	44	67	14(0)	46	47
bd	4	-	-	-	-	-	-	-	-	-
beale	2	14(0, 0)	38	55	14(0, 0)	38	55	14(0)	38	55
biggs	6	30(0, 0)	97	159	30(0, 0)	97	159	30(0)	97	159
box	3	16(1, 0)	68	111	20(0, 0)	75	123	20(0)	75	123
bv	10	15(0, 0)	42	54	15(0, 0)	42	54	15(0)	42	54
froth	2	11(2, 0)	27	34	8(0, 0)	24	31	8(0)	24	31
gauss	3	4(0, 0)	13	19	4(0, 0)	13	19	4(0)	13	19
gulf	3	1(0, 0)	2	2	1(0, 0)	2	2	1(0)	2	2
helix	3	30(2, 0)	103	147	28(0, 0)	94	137	28(0)	94	137
ie	10	11(0, 0)	17	20	11(0, 0)	17	20	11(0)	17	20
ie	100	13(0, 0)	19	22	13(0, 0)	19	22	13(0)	19	22
jensam	2	10(0, 0)	30	41	10(0, 0)	30	41	10(0)	30	41
kowosb	4	21(0, 0)	74	122	21(0, 0)	74	122	21(0)	74	122
lin	10	2(0, 0)	9	13	2(0, 0)	9	13	2(0)	9	13
lin	100	2(0, 0)	9	13	2(0, 0)	9	13	2(0)	9	13
lin1	10	5(3, 0)	46	38	2(0, 0)	13	11	2(0)	13	11
lin0	10	5(3, 0)	26	6	2(0, 0)	8	3	2(0)	8	3
meyer	3	11(9, 0)	72	76	-	-	-	-	-	-
osb1	5	44(0,2)	157	221	44(0,2)	157	221	44(0,2)	157	221
osb2	11	52(0, 0)	153	231	52(0, 0)	153	231	52(0)	153	231
pen1	10	104(2, 0)	374	597	34(0, 0)	152	253	34(0)	152	253
pen1	100	36(4, 0)	135	185	70(0, 0)	333	553	70(0)	333	553
pen2	10	852(0, 0)	2594	4135	852(0, 0)	2594	4135	852(0)	2594	4135
rose	2	28(0, 0)	93	141	28(0, 0)	93	141	28(0)	93	141
rosex	100	322(1, 1)	1348	1843	333(0, 2)	1409	1934	333(2)	1409	1934
sing	4	35(0, 0)	134	216	35(0, 0)	134	216	35(0)	134	216
singx	400	186(4, 0)	674	833	196(0, 0)	671	817	196(0)	671	817
trid	10	20(0, 0)	57	57	20(0, 0)	57	57	20(0)	57	57
trid	100	95(0, 0)	546	622	95(0, 0)	546	622	95(0)	546	622
trig	10	24(0, 0)	58	91	24(0, 0)	58	91	24(0)	58	91
trig	100	46(0, 0)	117	185	46(0, 0)	177	185	46(0)	117	185
vardim	10	6(2, 0)	30	25	6(0, 0)	42	65	6(0)	42	65
watson	12	41(0, 0)	172	280	41(0, 0)	172	280	41(0)	172	280
watson	20	47 (0, 0)	210	348	47(0, 0)	210	348	47(0)	210	348
wood	4	48(1, 0)	156	225	49(0, 0)	149	220	49(0)	149	220

in the cautious rule are generally satisfied and hence the cautious update essentially reduces to the ordinary update in most cases. This suggests that the ordinary BFGS method is generally "cautious," and hence the BFGS method seldom fails in practice. We hope that the results established in this paper contribute toward resolving the fundamental open problem of whether the BFGS method converges for nonconvex unconstrained optimization problems.

Acknowledgments. The authors would like to thank two anonymous referees for their helpful comments, which have made the paper clearer and more comprehensive than the earlier version.

TABLE 3
Test results for CBFGS/L-BFGS methods with polynomial search.

		CBFGS[1]			CBFGS[2]			BFGS		
Problem	Dim	Iter	off	fnum	Iter	off	fnum	Iter	off	fnum
badscb	2	-	-	-	-	-	-	-	-	-
badscp	2	-	-	-	-	-	-	-	-	-
band	10	-	-	-	-	-	-	-	-	-
bard	3	28	1	64	28	1	64	28	1	64
bd	4	491	3	1021	491	3	1021	-	-	-
beale	2	19	0	42	19	0	42	19	0	42
biggs	6	52	1	116	52	1	116	52	1	116
box	3	30	0	82	38	0	82	38	0	82
bv	10	17	0	42	17	0	42	17	0	42
froth	2	9	0	23	9	0	23	9	0	23
gauss	3	3	0	8	3	0	8	3	0	8
gulf	3	1	0	3	1	0	3	1	0	3
helix	3	37	0	83	37	0	83	37	0	83
ie	10	9	0	21	9	0	21	9	0	21
ie	100	10	0	23	10	0	23	10	0	23
jensam	2	12	0	31	12	0	31	12	0	31
kowosb	4	28	0	60	28	0	60	28	0	60
lin	10	1	0	4	1	0	4	1	0	4
lin	100	1	0	4	1	0	4	1	0	4
lin1	10	-	-	-	-	-	-	-	-	-
lin0	10	743	4	1501	743	4	1501	-	-	-
meyer	3	-	-	-	-	-	-	-	-	-
osb1	5	-	-	-	-	-	-	-	-	-
osb2	11	58	0	136	58	0	136	72	1	174
pen1	10	188	3	433	188	3	433	199	7	454
pen1	100	796	5	1639	796	5	1639	372	7	773
pen2	10	305	2	721	305	2	721	258	5	592
rose	2	39	0	91	39	0	91	39	0	91
rosex	100	39	0	91	39	0	91	39	0	91
sing	4	50	0	110	50	0	110	50	0	110
singx	400	55	0	116	55	0	116	69	1	148
trid	10	18	0	60	18	0	60	18	0	60
trid	100	127	0	467	127	0	467	131	2	526
trig	10	26	0	53	26	0	53	26	0	53
trig	100	49	0	103	49	0	103	49	0	103
vardim	10	407	2	833	407	2	833	176	3	377
watson	12	61	0	137	61	0	137	87	1	197
watson	20	58	0	133	58	0	133	89	1	204
wood	4	30	0	75	30	0	75	30	0	75

REFERENCES

[1] R. BYRD AND J. NOCEDAL, *A tool for the analysis of quasi-Newton methods with application to unconstrained minimization*, SIAM J. Numer. Anal., 26 (1989), pp. 727–739.

[2] R. BYRD, J. NOCEDAL, AND Y. YUAN, *Global convergence of a class of quasi-Newton methods on convex problems*, SIAM J. Numer. Anal., 24 (1987), pp. 1171–1190.

[3] J. E. DENNIS, JR. AND J. J. MORÉ, *A characterization of superlinear convergence and its application to quasi-Newton methods*, Math. Comp., 28 (1974), pp. 549–560.

[4] J. E. DENNIS, JR. AND J. J. MORÉ, *Quasi-Newton methods, motivation and theory*, SIAM Rev., 19 (1977), pp. 46–89.

[5] J. E. DENNIS, JR. AND R. B. SCHNABEL, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1983.

[6] L. C. W. DIXON, *Variable metric algorithms: Necessary and sufficient conditions for identical behavior on nonquadratic functions*, J. Optim. Theory Appl., 10 (1972), pp. 34–40.

[7] R. Fletcher, *Practical Methods of Optimization,* 2nd ed., John Wiley & Sons, Chichester, 1987.
[8] R. Fletcher, *An overview of unconstrained optimization*, in Algorithms for Continuous Optimization: The State of the Art, E. Spedicato, ed., Kluwer Academic Publishers, Boston, 1994, pp. 109–143.
[9] A. Griewank, *The global convergence of partitioned BFGS on problems with convex decompositions and Lipschitzian gradients*, Math. Program., 50 (1991), pp. 141–175.
[10] A. Griewank and Ph. L. Toint, *Local convergence analysis for partitioned quasi-Newton updates*, Numer. Math., 39 (1982), pp. 429–448.
[11] D. H. Li and M. Fukushima, *A globally and superlinearly convergent Gauss–Newton-based BFGS method for symmetric nonlinear equations*, SIAM J. Numer. Anal., 37 (1999), pp. 152-172.
[12] D. H. Li and M. Fukushima, *A modified BFGS method and its global convergence in nonconvex minimization*, J. Comput. Appl. Math., to appear.
[13] C. T. Kelley, *Iterative Methods for Optimization*, SIAM, Philadelphia, 1999.
[14] J. J. Moré, B. S. Garbow, and K. E. Hillstrome, *Testing unconstrained optimization software*, ACM Trans. Math. Software, 7 (1981), pp. 17–41.
[15] J. Nocedal, *Theory of algorithms for unconstrained optimization*, Acta Numer., 1 (1992), pp. 199–242.
[16] M. J. D. Powell, *On the convergence of the variable metric algorithm*, J. Inst. Math. Appl., 7 (1971), pp. 21–36.
[17] M. J. D. Powell, *Some global convergence properties of a variable metric algorithm for minimization without exact line searches*, in Nonlinear Programming, SIAM-AMS Proc. IX, R. W. Cottle and C. E. Lemke, eds., AMS, Providence, RI, 1976, pp. 53–72.
[18] Ph. L. Toint, *Global convergence of the partitioned BFGS algorithm for convex partially separable optimization*, Math. Program., 36 (1986), pp. 290–306.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1065–1080

AN ALTERNATIVE METHOD TO CROSSING MINIMIZATION ON HIERARCHICAL GRAPHS*

PETRA MUTZEL†

Abstract. A common method for drawing directed graphs is, as a first step, to partition the vertices into a set of k levels and then, as a second step, to permute the vertices within the levels such that the number of crossings is minimized. We suggest an alternative method for the second step, namely, removing the minimal number of edges such that the resulting graph is k-level planar. For the final diagram the removed edges are reinserted into a k-level planar drawing. Hence, instead of considering the k-level crossing minimization problem, we suggest solving the k-level planarization problem. In this paper we address the case $k = 2$. First, we give a motivation for our approach. Then, we address the problem of extracting a 2-level planar subgraph of maximum weight in a given 2-level graph. This problem is NP-hard. Based on a characterization of 2-level planar graphs, we give an integer linear programming formulation for the 2-level planarization problem. Moreover, we define and investigate the polytope $2\mathcal{LPS}(G)$ associated with the set of all 2-level planar subgraphs of a given 2-level graph G. We will see that this polytope has full dimension and that the inequalities occurring in the integer linear description are facet-defining for $2\mathcal{LPS}(G)$. The inequalities in the integer linear programming formulation can be separated in polynomial time; hence they can be used efficiently in a branch-and-cut method for solving practical instances of the 2-level planarization problem. Furthermore, we derive new inequalities that substantially improve the quality of the obtained solution. The separation problem for all the new classes of inequalities can be solved in polynomial time. We report on extensive computational results.

Key words. 2-level graphs, 2-level planarization, graph drawing, integer linear programming, polyhedral combinatorics, branch-and-cut

AMS subject classifications. 90C57, 68R10, 05C10, 05C85

PII. S1052623498334013

1. Introduction.

Directed graphs are widely used to represent structures in many fields such as economics, social sciences, and the mathematical and computer sciences. A good visualization of structural information allows the reader to focus on the informational content of the diagram.

A common method for drawing directed graphs was introduced by Sugiyama, Tagawa, and Toda [32] and Carpano [1]. In the first step, the vertices are partitioned into a set of k levels and, in the second step, the vertices within each level are permuted in such a way that the number of crossings is small.

From now on let us assume that we are given a *k-level graph* (*k-level hierarchy*), i.e., a graph $G = (V, E) = (V_1, V_2, \ldots, V_k, E)$ with vertex sets $V_1, \ldots, V_k$, $V = V_1 \cup V_2 \cup \ldots \cup V_k$, $V_i \cap V_j = \emptyset$ for $i \neq j$, and an edge set E connecting vertices in levels V_i and V_{i+1} $(1 \leq i \leq k-1)$. V_i is called the *ith level*. A k-level graph is drawn in such a way that the vertices in each level V_i are drawn on a horizontal line L_i with y-coordinate $k - i$, and the edges are drawn as straight lines. Contrary to the definitions of a hierarchy in [32], [14], we consider undirected edges, since their direction is irrelevant

*Received by the editors February 11, 1998; accepted for publication (in revised form) February 16, 1999; published electronically May 10, 2001. An extended abstract of this paper was published in *Proceedings of Graph Drawing* '96, Lecture Notes in Comput. Sci. 1190, Springer-Verlag, Berlin, New York, 1997, pp. 318–333. This work was partially supported by the DFG-grant Ju204/7-1 and by ESPRIT LTR Project 20244 – ALCOM-IT. This work was carried out while the author was at the Max-Planck-Institut für Informatik, Saarbrücken, Germany.

http://www.siam.org/journals/siopt/11-4/33401.html

†Institut für Computergraphik und Algorithmen, Technische Universität Wien, Favoritenstr. 9-11 E186, A-1040 Wien, Austria (mutzel@ads.tuwien.ac.at).

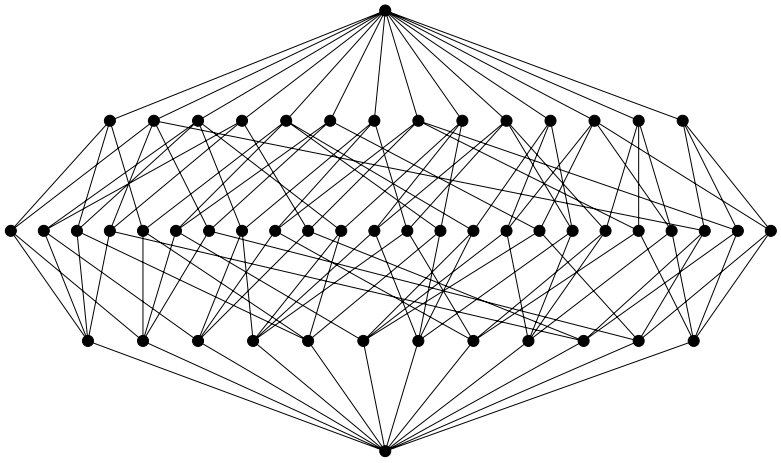


FIG. 1. *A k-level graph arising in practice* [9].

for the problem considered here. Essentially, a k-level hierarchy is a k-partite graph that is drawn in a special way. Figure 1 shows a k-level graph.

Even for 2-level graphs the straight line crossing minimization problem is NP-hard. Exact algorithms based on branch and bound have been suggested by various authors (see, e.g., [35] and [21]). For $k \geq 2$, a vast amount of heuristics has been published in the literature (see, e.g., [37, 32, 4, 26, 7] and [3]).

We suggest an alternative approach to crossing minimization, namely, to remove a minimal set of edges such that the remaining k-level graph can be drawn without edge crossings. In the final drawing, the removed edges are reinserted. Since the insertion of each edge may produce many crossings, the final drawing may be far from an edge-crossing minimal drawing.

Figure 2(a) shows a drawing of a graph obtained by 2-level planarization, whereas Figure 2(b) shows the same graph drawn with the minimal number of edge crossings (using the exact algorithm given in [21]). Although the drawing in Figure 2(a) has 34 crossings, that is, 41% more crossings than the drawing in Figure 2(b) (24 crossings), the reader will not recognize this fact. Indeed, anecdotal evidence suggests that the great majority of computer scientists are fooled by Figure 2 and see more crossings in Figure 2(b) than in Figure 2(a). This encourages us to study the k-level planarization problem.

Another motivation for studying k-level planarization arises from the fact that the k-level crossing minimization problem is a very hard problem that cannot be solved exactly or approximately (with some reasonable solution guarantees) in short computation time. Our experiments in [20, 21] showed that for sparse graphs, such as they occur in graph drawing, the heuristic methods used in practice are far from the optimum. We believe that the methods of polyhedral combinatorics that have been successfully applied for the maximum planar subgraph problem [18, 19, 27]—and for the straight line crossing minimization problem on two levels, where one level is fixed [21]—may be helpful for obtaining better approximation algorithms in practice. Recent work in this direction supports this conjecture [15, 13]. However, much more effort is needed to obtain efficient algorithms that can approach the k-level crossing minimization problem for practical instances.

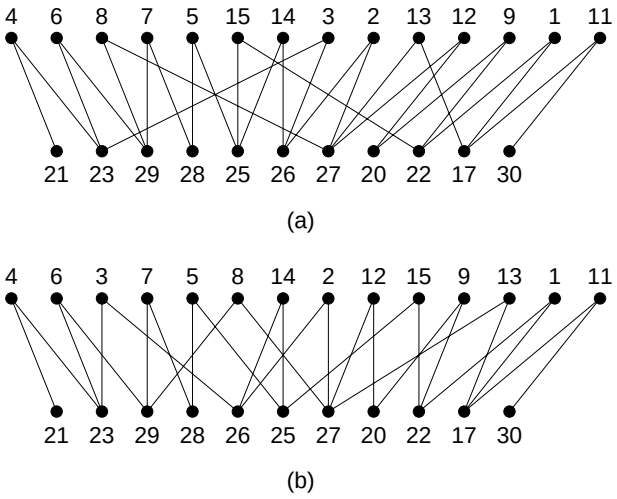


FIG. 2. *A graph* (a) *drawn using k-planarization and* (b) *drawn with the minimal number of crossings computed by the algorithm in* [21].

The k-level planarization problem, however, may be easier to attack. We build our hope on the fact that there is a fast polynomial time algorithm for recognizing k-level planar graphs (see [14, 16, 17] and [2]). Moreover, our computational results on 2-level graphs addressed in this paper support our conjecture. Additionally, there are many bipartite graphs out there, for which a nice 2-level drawing is useful.

In addition to the application in automatic graph drawing, the 2-level planarization problem comes up in computational biology. In DNA mapping, small fragments of DNA have to be ordered according to the given overlap data and some additional information. Waterman and Griggs [38] suggested combining the information derived by a digest mapping experiment with the information on the overlap between the DNA fragments. If the overlap data are correct, the maps can be represented as a 2-level planar graph. But, in practice, the overlap data may contain errors. Hence, Waterman and Griggs suggested solving the 2-level planarization problem (see also [36]). Furthermore, the 2-level planarization problem arises in global routing for row-based VLSI layout (see [25, 34]).

Section 2 reports on previously known results of the 2-level planarization problem. One of the characterizations of 2-level planar graphs leads directly to an integer linear programming formulation for the 2-level planarization problem. In section 3 we study the polytope associated with the set of all possible 2-level planar subgraphs of a given 2-level graph. From this we obtain new classes of inequalities that tighten the associated LP-relaxation. In order to get practical use out of these inequalities, we have to solve the "separation problem." This problem will be addressed in section 4, where we also discuss a branch-and-cut algorithm based on those results. Computational results with a branch-and-cut algorithm are presented in section 5.

2. Characterizing 2-level planar graphs. A 2-*level graph* is a graph $G = (L, U, E)$ with vertex sets L and U, called lower and upper level, and an edge set E connecting vertices in L with vertices in U. There are no edges between two vertices in the same level. A 2-*level planar graph* $G = (L, U, E)$ is a graph that can be drawn in such a way that all the vertices in L appear on a line (the lower line), the vertices in

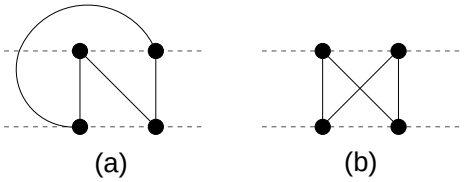


FIG. 3. (a) *A planar bipartite graph that is* (b) *not* 2*-level planar.*

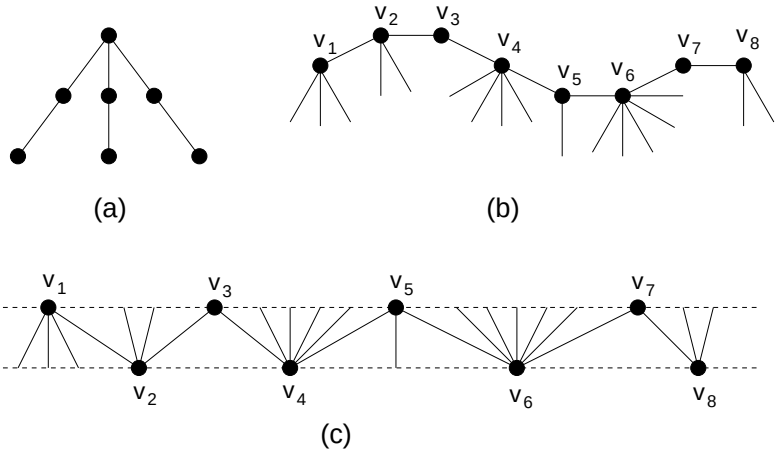


FIG. 4. (a) *Double claw.* (b) *Caterpillar.* (c) *Caterpillars can be embedded on* 2*-levels without any crossings.*

U appear on the upper line, and the edges are drawn as straight lines without crossing each other. The difference between a planar bipartite graph and a 2-level planar graph is obvious. For example, the graph shown in Figure 3 is a planar bipartite graph, but not a 2-level planar graph.

Given a 2-level graph $G = (L, U, E)$ with weights $w_e > 0$ on the edges, the 2*-level planarization problem* (or *maximum* 2*-level planar subgraph problem*) is to extract a 2-level planar subgraph $G' = (L, U, F)$, $F \subseteq E$ of maximum weight; i.e., the sum $\sum_{e \in F} w_e$ is maximum.

To our knowledge, only the unweighted ($w_e = 1$ for all $e \in E$) 2-level planarization problem has been considered in the literature so far. It was first independently mentioned in [12, 33] and [5]. Two of these authors introduced the problem in the context of graph drawing. All of them have given the following nice characterization of 2-level planar graphs based on forbidden subgraphs.

We will call the graph shown in Figure 4(a) a *double claw.* A *caterpillar* is a connected graph $G = (V, E)$ having edges on its backbone $(v_1, v_2, \ldots, v_l)$ and single edges (v_i, w), $w \in V \setminus \{v_1, v_2, \ldots, v_l\}$ (see Figure 4(b)).

THEOREM 2.1 (see [12, 33, 5]). *A* 2*-level graph is* 2*-level planar if and only if it contains no cycle and no double claw.*

Proof. A graph without any cycles is a set of trees. A tree without any double claws is a caterpillar. Caterpillars can be embedded on 2-levels without any crossings (see Figure 4(c)). On the other hand, a 2-level planar graph can contain neither a cycle nor a double claw. □

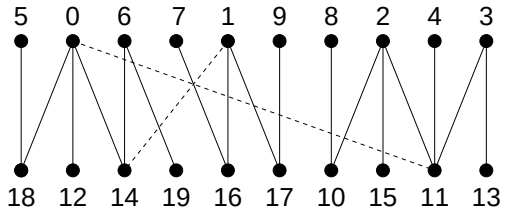


FIG. 5. *An acyclic* 2*-level graph for which the algorithm suggested in* [33] *leads to a nonoptimal solution.*

Note that this characterization is not correct for general nonbipartite graphs. For example, a triangle contains a cycle and can be drawn crossing-free with its vertices on two lines; however, it does not have a 2-level drawing, because vertices in the same level are not allowed to be connected.

The following alternative characterization leading to a simple linear time algorithm was given in [33]. It is useful in our branch-and-cut algorithm.

THEOREM 2.2 (see [33]). *A* 2*-level graph* G *is* 2*-level planar if and only if the graph* G^**, that is, the remainder of* G *after deleting all vertices of degree one, is acyclic and contains no vertices of degree at least three.*

However, the 2-level planarization problem is NP-hard even in the unweighted case and when each vertex in U has degree three and each vertex in L has degree two (by reduction from a Hamiltonian path problem) [6]. Therefore, Eades and Whitesides [6] suggested a heuristic based on the search for a longest path which will be used as a "backbone" of the caterpillar to be constructed.

Tomii, Kambayashi, and Shuzo [33] suggest an algorithm for acyclic 2-level graphs. The algorithm can be seen as an adaptive greedy algorithm. In each step, the edges are labeled according to some rule and the edge with the highest label is removed. However, this algorithm does not lead to the optimal solution as shown in Figure 5. The algorithm would remove the edge $(0, 14)$ in a first step. The remaining graph still contains two edge-disjoint double claws that have to be destroyed by removing two more edges, whereas the optimal solution would be to remove the two edges $(0, 11)$ and $(1, 14)$.

Recently, Shahrokhi et al. [31] presented a linear time algorithm for the 2-level planarization problem on 2-level acyclic graphs. For double claw free graphs, the 2-level planarization problem is equivalent to the maximum forest subgraph problem that can be solved via a simple greedy algorithm.

3. Polyhedral studies on the 2-level planarization problem. Based on the characterization of 2-level planar graphs in terms of forbidden subgraphs (see Theorem 2.1), it is straightforward to derive an integer linear programming formulation for the maximum 2-level planar subgraph problem. We introduce variables x_e for all edges $e \in E$ of the given 2-level graph $G = (L, U, E)$. We use the following notation: vectors x are column vectors, and their transposed vectors x^T are row vectors. If $w^T = (w_1, w_2, \ldots, w_m)$ and $x^T = (x_1, x_2, \ldots, x_m)$, then $w^T x = \sum_{i=1}^{m} w_i x_i$. We use the notation $x(C) = \sum_{e \in C} x_e$ for $C \subseteq E$.

For any set $P \subseteq E$ of edges we define an incidence vector $\chi^P \in R^{|E|}$ with the ith component $\chi^P(e_i)$ getting value 1 if $e_i \in P$, and 0 otherwise. Any vector $x^T = (x_{e_1}, x_{e_2}, \ldots, x_{e_{|E|}})$ that is the incidence vector of a 2-level planar graph satisfies the

following inequalities:

$$0 \le x_e \le 1 \quad \text{for all } e \in E, \tag{1}$$

$$x(C) \le |C| - 1 \quad \text{for all cycles } C \subseteq E, \tag{2}$$

$$x(T) \le |T| - 1 \quad \text{for all double claws } T \subseteq E, \tag{3}$$

$$x_e \text{ integral} \quad \text{for all } e \in E, \tag{4}$$

and vice versa: any vector $x^T = (x_{e_1}, x_{e_2}, \ldots, x_{e_{|E|}})$ satisfying inequalities (1), (2), (3), and (4) corresponds to a 2-level planar subgraph of G. Hence, solving the integer linear system $\{\max\ w^T x |$ constraints (1)–(4) hold for $x\}$ will give us the solution of the maximum 2-level planar subgraph problem for a given graph $G = (L, U, E)$ with weights w_e on the edges $e \in E$.

Since solving integer linear programs is, in general, NP-hard, we will have to drop the integrality constraints (4), which gives us a relaxation of the original integer linear programming formulation. In polyhedral combinatorics, we try to substitute the missing integrality constraints with additional inequalities. An excellent introduction to the theory of polyhedral combinatorics is given by Pulleyblank in [30].

We define the polytope $2\mathcal{LPS}(G)$ for a given 2-level graph $G = (L, U, E)$ as the convex hull over all incidence vectors of 2-level planar subgraphs of G. The vertices of this polytope correspond exactly to the 2-level planar subgraphs of G and vice versa. If we can describe the polytope $2\mathcal{LPS}(G)$ as the solution set of linear inequalities, we can optimize any given cost function over the set of all 2-level planar subgraphs of G. Of course, because of the NP-hardness of the problem we cannot expect to find such a description, but in practice a partial description may also suffice.

In a nonredundant description only facet-defining inequalities are present. An inequality is said to be *facet-defining* for a polytope $\mathcal{P}$ if it defines a face of maximal dimension of $\mathcal{P}$. An inequality $c^T x \le c_0$ is said to define a *face* of $\mathcal{P}$ if $c^T y \le c_0$ for all points $y \in \mathcal{P}$ and if there is at least one point y' in $\mathcal{P}$ with $c^T y' = c_0$.

So, our task is to find facet-defining inequalities for the polytope $2\mathcal{LPS}(G)$ for a given 2-level graph G. We will first investigate the inequalities given in the integer linear programming formulation. We will see that the linear inequalities (1) and (3) are facet-defining, but only a part of the inequalities (2) is facet-defining. But first we will determine the dimension of $2\mathcal{LPS}(G)$.

Let us consider the set $\mathcal{S}$ of all 2-level planar subgraphs of G. The set $\mathcal{S}$ is a *monotone system* (also called *independence system*), since the empty subgraph is 2-level planar and any subgraph of a 2-level planar graph is also 2-level planar. Hence, we easily get the following theorem using the theory for monotone systems.

THEOREM 3.1. *Let $G = (L, U, E)$ be a graph on two levels. The dimension of $2\mathcal{LPS}(G)$, the convex hull of incidence vectors of 2-level planar subgraphs of G, is $|E|$. The trivial inequalities $x_e \ge 0$ and $x_e \le 1$ are facet-defining for $2\mathcal{LPS}(G)$.*

Proof. It is a well-known fact that for a monotone system $(E, \mathcal{S})$ with ground set E the dimension of the associated polyhedron $P_{\mathcal{S}}$ is $|E| - (|E - \cup\mathcal{S}|)$ (a proof is contained, e.g., in [11]). Moreover, $x_e \ge 0$ defines a facet of $P_{\mathcal{S}}$ if and only if $e \in \cup\mathcal{S}$. Since every single edge is 2-level planar, we have $\cup\mathcal{S} = E$. Hence the dimension of the polyhedron $2\mathcal{LPS}(G)$ is $|E|$ and $x_e \ge 0$ is facet-defining for $2\mathcal{LPS}(G)$.

Let P_i be the 2-level planar graphs induced by the edge sets $\{e \cup e_i\}$ for a given edge $e \in E$ and $e_i \in E \setminus \{e\}$ for $i = 1, \ldots, |E| - 1$. The incidence vectors of the graph P induced by the edge e and the graphs P_i for $i = 1, 2, \ldots, |E| - 1$ are affinely independent and they satisfy $x_e = 1$. Hence we have shown that $x_e \le 1$ is facet-defining for $2\mathcal{LPS}(G)$. ∎

Next we will see that not all of the inequalities (2) are facet-defining for $2\mathcal{LPS}(G)$.

THEOREM 3.2. *Let $G = (L, U, E)$ be a 2-level graph. The cycle inequalities*

$$x(C) \leq |C| - 1,$$

where $C \subseteq E$ induces a cycle in G, are facet-defining for $2\mathcal{LPS}(G)$ if and only if C induces a cycle without chords in G.

Proof. Let $C \subseteq E$ be a cycle without chords in G. We will show that there are $|E|$ incidence vectors of 2-level planar subgraphs induced by the edge set F of G that are linearly independent and that satisfy $\chi^F(C) = |C| - 1$. Consider the graphs induced by the edge sets $F_i = C \setminus \{e_i\}$ for $e_i \in C$ for $i = 1, 2, \ldots, |C|$. Moreover, consider the graphs induced by the edge sets $H_j = F_1 \cup f_j$ for $f_j \in E \setminus C$, $j = 1, 2, \ldots, |E| - |C|$. Since the cycle C is chordless, adding any edge $f_j \in E \setminus C$ to F_1 still gives a 2-level planar graph, since neither a cycle nor a double claw destroying 2-level planarity can occur. All the $|E|$ incidence vectors of the 2-level planar graphs induced by F_i for $i = 1, 2, \ldots, |C|$ and H_j for $j = 1, 2, \ldots, |E| - |C|$ are linearly independent and they satisfy inequality (2) with equality. Hence the facet-defining property is shown.

Suppose now that $C = (v_1, v_2, \ldots, v_k, v_1)$ is a cycle with a chord $d = (v_h, v_l) \in E$, $d \notin C$, in G for some $h, l \in \{1, 2, \ldots, k\}$. There exists no 2-level planar graph containing the edge d and $|C| - 1$ edges of C. Hence, there exists no point x in $2\mathcal{LPS}(G)$ with $x_d = 1$ and $x(C) = |C| - 1$, which will prove our claim. ∎

In the following we will see that all the double claws contained in G are present in a nonredundant description of $2\mathcal{LPS}(G)$ as linear inequalities.

THEOREM 3.3. *Let $G = (L, U, E)$ be a 2-level graph. The double claw inequalities*

$$x(T) \leq |T| - 1,$$

where $T \subseteq E$ induces a double claw in G, are facet-defining for $2\mathcal{LPS}(G)$.

Proof. Let $T = \{e_1, \ldots, e_6\}$ and let $F_i = T \setminus e_i$ for $i = 1, \ldots, 6$. Obviously, the graphs induced by F_i are 2-level planar graphs and satisfy inequality (3) with equality. Moreover, consider the graphs induced by $H_j = T \cup f_j$ for $f_j \in E \setminus T$, $j = 1, 2, \ldots, |E| - |T|$. If H_j contains a cycle C, we can remove any edge in $C \cap T$ to get a 2-level planar graph induced by H'_j. In all the other cases there is always an edge we can remove from $H_j \cap T$ such that the remaining set H'_j induces a set of caterpillars. Clearly, the incidence vectors of the 2-level planar subgraphs induced by F_i, $i = 1, 2, \ldots, 6$, and H'_j, $j = 1, 2, \ldots, |E| - |T|$ of G are linearly independent and satisfy inequality (3) with equality. ∎

We can tighten the LP-relaxation of (1)–(3) by introducing new inequalities that are valid and tight in the sense that they are facet-defining for $2\mathcal{LPS}(G)$. First, we generalize the double claw inequalities to k-double claw inequalities. Considering a double claw as a claw having three paths of length two, a *generalized k-double claw* is a claw having k paths of length two (see Figure 6(a)). The only vertex contained in all k paths is called the *root node* of the generalized k-double claw.

THEOREM 3.4. *Let $G = (L, U, E)$ be a 2-level graph. The generalized k-double claw inequalities*

$$x(T) \leq k + 2, \tag{5}$$

where $T \subseteq E$ induces a k-double claw in G ($k \geq 3$), are facet-defining for $2\mathcal{LPS}(G)$.

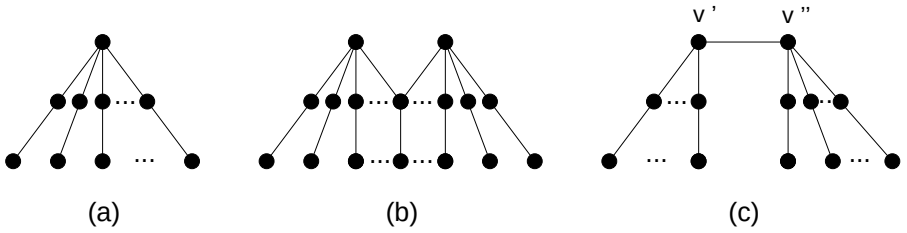


FIG. 6. (a) *Generalized k-double claw.* (b) *Combined k-double claw.* (c) *Node-split k-double claw.*

Proof. Obviously, the inequality is valid. We denote $x(T) \leq k+2$ by $c^T x \leq c_0$. Let us assume that there exists an inequality $a^T x \leq a_0$ with $\{x|c^T x = c_0\} \subseteq \{x|a^T x = a_0\}$. We show that $a_e = \lambda c_e$ and $a_0 = \lambda c_0$ for $\lambda > 0$. Let r be the root of the k-double claw and P denote the subgraph of $G = (V, E)$ induced by the edge set $F = \{(r, w)|w \in N(r) \cap V(T)\}$, where $N(r) = \{v|(r, v) \in E\}$ is the neighborhood of r. Adding any two edges $e_1 \neq e_2$ in $T \setminus F$ to P gives a 2-level planar subgraph P' induced by the edge set $F' = \{F \cup e_1 \cup e_2\}$ satisfying $c^T \chi^{F'} = c_0$; hence also $a^T \chi^{F'} = a_0$. Since we can substitute e_1 and e_2 with any of the edges in $T \setminus F'$ we get $a_e = a_f$ for all $e, f \in T \setminus F$. Inserting the edge $e_3 = (w_3, u_3) \in T \setminus F'$ with $w_3 \in N(r) \cap V(T)$ in P' while removing the edge $e'_3 = (r, w_3)$ gives $a_{e_3} = a_{e'_3}$ and finally $a_e = a_f$ for all $e, f \in T$.

For any edge $e \in E \setminus T$ we can find a 2-level planar subgraph induced by the edge set F'' with $e \in F''$ satisfying $c^T \chi^{F''} = c_0$. Hence $a_e = 0$ for all $e \in E \setminus T$. □

We can prove that the *combined k-double claws* give rise to a class of facet-defining inequalities for our polytope. A *combined k-double claw* $T = (T_1, T_2)$ consists of two k_i-double claws T_i, $i = 1, 2$, that share a single edge $e_0 = (l_0, u_0)$ not incident to the root nodes of T_1 and T_2 (see Figure 6(b)).

THEOREM 3.5. *The combined k-double claw inequalities*

$$x(T) \leq k_1 + k_2 + 3, \tag{6}$$

where $T = (T_1, T_2) \subseteq E$ induces a combined k-double claw in G with parameters $k_1 \geq 3$ and $k_2 \geq 3$, are facet-defining for $2\mathcal{LPS}(G)$ if and only if there exist no edges (r_1, w_2) and (r_2, w_1) in G, where r_i is the root of T_i and $w_i \in N(r_i) \setminus \{l_0, u_0\}$ for $i = 1, 2$.

Proof. Let $e_0 = (l_0, u_0)$ denote the edge contained in both k_i-double claws $G_i = (L_i, U_i, T_i)$ and let $T = T_1 \cup \{T_2 \setminus \{e_0\}\}$. We first show validity. Let us assume that there is a 2-level planar subgraph induced by the edge set F violating inequality (6). Let $r_i \in U_i$, and $e_i = (r_i, l_0)$ for $i = 1, 2$. The set $T_2 \cap F$ cannot contain more than $k_2 + 2$ edges. On the other hand, the set $T_1 \setminus \{e_0, e_1\}$ induces a $(k_1 - 1)$-claw and can contain at most $k_1 - 1 + 2 = k_1 + 1$ edges. Since $T = T_1 \setminus \{e_0, e_1\} \cup T_2 \cup \{e_1\}$, we have $e_1 \in F$; otherwise, (6) is not violated. Symmetrically, we also get $e_2 \in F$. Now, consider the k_2-double claw $T'_2 = T_2 \setminus \{e_0\} \cup \{e_1\} \supseteq F$. The set $T'_2 \cap F$ cannot contain more than k_2 edges in addition to e_1 and e_2. We get a symmetrical argument for $F \cap T'_1$, where $T'_1 = T_1 \setminus \{e_0\} \cup \{e_2\}$. Altogether F cannot contain more than $k_1 + k_2 + 3$ edges, since $F \subseteq T'_1 \cup (T'_2 \setminus \{e_1, e_2\}) \cup e_0$.

Now, let us assume that there is an inequality $a^T x \leq a_0$ with $\{x|c^T x = c_0\} \subseteq \{x|a^T x = a_0\}$, where $c^T x \leq c_0$ denotes inequality (6). The set $T''_1 = T_1 \setminus \{e_0, e_1\}$ induces a $(k_1 - 1)$-double claw which is not connected with T_2. Combining any 2-level planar subgraph of size $(k_1 - 1) + 2$ in T''_1 with any double claw free subgraph of

size k_2+2 of T_2 gives a 2-level planar subgraph of size k_1+k_2+3 in T. Because of Theorem 3.4, we have that $a_{e''} = a_{f''}$ for all $e'', f'' \in T_1''$, and $a_e = a_f$ for all $e, f \in T_2$. Symmetrical arguments for $T_2'' = T_2 \setminus \{e_0, e_1\}$ and T_1 together with the fact that $e_0 \in \{T_1 \cap T_2\}$ lead to $a_e = a_f$ for all $e, f \in T$.

We have already seen that zero-lifting is possible within the k_i-claws ($i = 1, 2$). The critical edges to add are those connecting T_1 and T_2. Let $e = (l_2, u_1)$ be such a critical edge with $l_2 \in L_2$, $u_1 \in U_1$. If $u_1 \neq r_1$, we can show that $a_e = 0$. But in the case that $u_1 = r_1$, there exists no 2-level planar subgraph induced by the edge set F with $c^T\chi^F = c_0$ containing the edge e. Hence, in this case, inequality (6) is not facet-defining for $2\mathcal{LPS}(G)$. □

The *node-splitting operation* at vertex v in a graph G substitutes the subgraph induced by the edge set $\{(v,w)|w \in N(v)\}$ with a new subgraph induced by $\{(v', w')|w' \in W'\} \cup \{(v'', w'')|w'' \in W''\} \cup \{(v', v'')\}$, where $N(v)$ is the set of adjacent vertices of v in G, $W', W'' \subseteq N(v)$ with $W' \cup W'' = N(v)$ and $W' \cap W'' = \emptyset$. The vertices v' and v'' are the *duplicates* of v. The resulting graph when splitting the root node of a k-double claw is called a *node-split k-double claw* with parameters k_1 and k_2 (see Figure 6(c)). The inequalities derived for those graphs contain a coefficient of two.

THEOREM 3.6. *Let $G = (L, U, E)$ be a 2-level graph. The node-split k-double claw inequalities*

$$x(T) + 2x_{(r_1,r_2)} \leq k_1 + k_2 + 4, \tag{7}$$

where $T \subseteq E$ induces a node-split k-double claw G' in G with parameters $k_1 \geq 2$ and $k_2 \geq 2$, are facet-defining for $2\mathcal{LPS}(G)$.

Proof. Let $e_0 = (r_1, r_2)$ and $T = T_1 \cup T_2 \cup \{e_0\}$, where T_1 and T_2 are the edge sets inducing the two components of $T \setminus \{e_0\}$. We first show validity. Let us assume that there exists a 2-level planar subgraph P induced by the edge set F violating the inequality (7). We know that $T_1 \cap F$ and $T_2 \cap F$ cannot contain more than k_1+2 and k_2+2 edges. If $e_0 \notin F$, the inequality cannot be violated by P. But if $e_0 \in F$, either T_1 contains at most k_1 edges, T_2 contains at most k_2 edges, or T_1 and T_2 contain at most k_1+1 and k_2+1 edges to ensure 2-level planarity of P. Hence, inequality (7) cannot be violated with P and validity is shown.

Now let us assume that there is an inequality $a^Tx \leq a_0$ with $\{x|c^Tx = c_0\} \subseteq \{x|a^Tx = a_0\}$, where $c^Tx \leq c_0$ denotes inequality (7). Let P be the 2-level planar subgraph induced by k_1+2 edges in T_1 and k_2+2 edges in T_2 (edge set $F = F_1 \cup F_2$, $F_i \in T_i$ for $i = 1, 2$) not containing e_0. If $k_i \geq 3$, then any edge in F_i can be substituted with an edge $e_i \in T_i \setminus F_i$ maintaining the 2-level planarity. Hence, in this case we have shown that $a_e = a_f$ for all $e, f \in T_i$. Otherwise, let us assume that $k_i = 2$. It is not hard to see that there is a 2-level planar subgraph P' containing e_0 and k_i+1 edges of T_i, $i = 1, 2$ (induced by the edge set F'). Any edge in $F' \cap T_i$ can be substituted with an edge $f \in T_i$, $f \notin F'$ without destroying 2-level planarity. Hence, $a_e = a_f$ for all $e, f \in T_i$. Taking the difference of $a^T\chi^F$ and $a^T\chi^{F'}$ yields $a_{e_0} = a_{e_1} + a_{e_2}$ for $e_i \in T_i$, $i = 1, 2$. Moreover, there is a 2-level planar subgraph induced by e_0, k_1+2 edges in T_1 and k_2 edges in T_2. Hence, we have shown that $a_{e_1} = a_{e_2}$ for all $e_1 \in T_1$, $e_2 \in T_2$ and $a_{e_0} = 2a_e$ for all $e \in T_1 \cup T_2$ if $k_1, k_2 \geq 2$. Hence, inequality (7) is facet-defining for $2\mathcal{LPS}(G')$.

It remains to show that $a_e = 0$ for all edges $e \in E \setminus T$ if $G' = (V', T)$ with $T \subseteq E$ and $V' \subseteq V$. Since zero-lifting is possible for double claw inequalities, we can restrict our attention to edges $e = (v, w)$ with $v \in G_1$ and $w \in G_2$, where G_1 and G_2 denote the graphs induced by the edge sets T_1 and T_2. For the two possible cases, we can

always find a 2-level planar graph containing the edge e_0, an additional edge $e \notin T$, and $k_1 + k_2 + 4$ edges of T in total. □

In the case that the given 2-level graph contains no double claw, the 2-level planarization problem is equivalent to the maximum forest problem. It is well known that this problem can be solved in polynomial time by a simple greedy algorithm. Moreover, the structure of the associated weighted forest polytope has been well studied (see, e.g., [8]). The inequalities of the weighted forest polytope are still valid for our polytope $2\mathcal{LPS}(G)$, even if the graph G contains double claws. Also, as we will see in our computational experiments, they are quite useful in practice.

LEMMA 3.7. *Let $G = (L, U, E)$ be a 2-level graph. The forest inequalities*

$$x(F) \leq V(F) - 1, \tag{8}$$

where $F \subseteq E$ and $V(F)$ is the number of vertices contained in the subgraph induced by F, are valid for $2\mathcal{LPS}(G)$.

The special case when F induces a complete bipartite subgraph of a 2-level graph G leads to forest inequalities that are facet-defining for $2\mathcal{LPS}(G)$. We will call them *crown inequalities*. For $|L'| = |U'| = 2$, the crown inequalities are equivalent to the cycle inequalities for $|C| = 4$. Hence, the crown inequalities are a generalization of this cycle inequality.

THEOREM 3.8. *Let $G = (L, U, E)$ be a 2-level graph containing a complete bipartite subgraph $G' = (L', U', E')$, $E' \subseteq E$. The crown inequalities*

$$x(E') \leq |L'| + |U'| - 1 \tag{9}$$

with $|L'| \geq 2$ and $|U'| \geq 3$ are facet-defining for $2\mathcal{LPS}(G)$.

Proof. The validity follows from Lemma 3.7. Let us assume that there is an inequality $a^T x \leq a_0$ with $\{x | c^T x = c_0\} \subseteq \{x | a^T x = a_0\}$, where $c^T x \leq c_0$ denotes inequality (9). Let $U' = \{u_1, \ldots, u_{|U'|}\}$, $L' = \{l_1, \ldots, l_{|L'|}\}$, $u \in U'$, and $l \in L'$. The edge set $F = \{(u, l_i) | l_i \in L', i = 1, \ldots, |L'|\} \cup \{(l, u_i) | u_i \in U' \setminus \{u\}, i = 1, \ldots, |L'|\}$ induces a 2-level planar subgraph satisfying $c^T \chi^F = c_0$; hence also $a^T \chi^F = c_0$. Removing the edge (l, u) from F and adding the edge $\{u_i, l_i\}$, where $u_i \neq u$, $l_i \neq l$, $u_i \in U'$, and $l_i \in L'$, will still leave a 2-level planar graph. Hence, we have $a_{(u,l)} = a_{(u_i,l_i)}$, and since we can choose u and l free among the vertices, we have $a_e = a_f$ for all $e \in E'$. On the other hand, it is always possible to add an extra edge $(l_i, v) \in E \setminus E'$ or $(u_i, v) \in E \setminus E'$ to F without losing 2-level planarity. Hence, $a_e = 0$ for all edges in $E \setminus E'$, and the theorem is proved. □

In the next section we show how the theoretical results obtained in this section can be used in an algorithm for solving practical instances of the 2-level planarization problem.

4. Separation problems and a branch-and-cut algorithm. According to results of Grötschel, Lovász, and Schrijver [10], Karp and Papadimitriou [23], and Padberg and Rao [28], we can optimize a linear objective function over a polytope in polynomial time if and only if we can solve the *separation problem* in polynomial time; i.e., given a vector $\bar{x} \in \mathbf{Q}^{|E|}$, decide whether $\bar{x} \in \mathcal{P}$ and, if $\bar{x} \notin \mathcal{P}$, find a vector $d \in \mathbf{Q}^{|E|}$ and a scalar $d_0 \in \mathbf{Q}$ such that the inequality $d^T \bar{x} \leq d_0$ is valid with respect to $\mathcal{P}$ and $d^T \bar{x} > d_0$.

It is well known that the separation problem restricted to the class of inequalities (2) can be solved in polynomial time.

LEMMA 4.1. *For the cycle inequalities* (2) *the separation problem can be solved in polynomial time by computing at most $|E|$ shortest path problems.*

Proof. Given a point $\bar{x} \in \mathbf{Q}^{|E|}$, we are searching for a cycle $C \subseteq E$ with $\bar{x}(C) > |C| - 1$. Let us write the inequality in a different way: $|C| - \bar{x}(C) < 1$ which corresponds to $\sum_{e \in C}(1 - x_e) < 1$. For any fixed $e_0 \in E$ we solve a shortest path problem on the graph given by $G - \{e_0\}$ with edge costs $z_e = 1 - x_e$ for $e \in E \setminus \{e_0\}$. Let W be the weight of the shortest path. We only have to test if $W + z_{e_0}$ is less than one. In this case we have found a cycle C violating inequality $\bar{x}(C) > |C| - 1$ of $\bar{x}$. If a violated inequality is not found for any $e_0 \in E$, we have a proof that all the inequalities of type (2) are satisfied at $\bar{x}$. Hence, we have solved the separation problem for (2) in polynomial time. ∎

The separation problem can also be solved for the double claw inequalities (3) and their generalization to k-double claw inequalities. This is surprising, since it is only obvious that the separation problem can be solved for fixed k.

THEOREM 4.2. *The separation problem for the double claw inequalities and the generalized k-double claw inequalities can be solved in polynomial time by computing a series of maximum bipartite matching problems on subgraphs of $G = (V, E)$.*

Proof. Obviously, all k-double claws for $k = 3$ can simply be enumerated in polynomial time. Our task is more difficult for the class of generalized k-double claw inequalities. Given a point $\bar{x} \in \mathbf{Q}^{|E|}$, we are searching for a $k \in \{3, \ldots, \lceil \frac{n}{2} \rceil - 1\}$ and a k-double claw $T \subseteq E$ with $\bar{x}(T) > k+2$. For all vertices $r \in V$ with $\delta(r) \geq 3$, we search for a generalized k-double claw with root node r violating inequality (5) at point $\bar{x}$ and $k \in \{3, \ldots, \lceil \frac{n}{2} \rceil - 1\}$. We define the set $N(r) = \{w_1, \ldots, w_t\} = \{w | (r, w) \in E\}$ and the set $F \subseteq E$ as $F = \{(w_i, u) \in E | i = 1, \ldots, t, \text{and } u \neq r\}$. Note that for any edge $e = (w_i, u) \in F$ belonging to a generalized double claw T rooted at r, the edge (r, w_i) must also belong to T. Furthermore, any matching M with $|M| > 2$ in the subgraph of G induced by F gives a generalized $|M|$-double claw T with $|M| \in \{3, \ldots, \lceil \frac{n}{2} \rceil - 1\}$ by adding the edges (w_i, r) to M if and only if the vertex w_i is covered by a matching edge in M. In order to violate the inequality, we must have $\sum_{e \in T} \bar{x}_e > k + 2$ which is equivalent to

$$\sum_{(w_i, u) \in M} \left(\bar{x}_{(w_i, u)} + \bar{x}_{(w_i, r)} - 1 \right) > 2,$$

where $M \subseteq E$ induces a matching of size $|M| = k$ in G.

We set $\bar{z}_{(w_i, u)} = \bar{x}_{(w_i, u)} + \bar{x}_{(w_i, r)} - 1$ for all $e = (w_i, u) \in F$ and search for a bipartite matching M^* of maximum weight in the graph induced by F with weights $\bar{z}_e$ on the edges $e \in F$. If the weight $\bar{z}_e$ of an edge $e \in F$ is positive, it may be useful to add it to T, since the violation will increase. Otherwise, if $\bar{z}_e < 0$, it is not useful to add it to T, since the violation will decrease; also, the edge will not be contained in the maximum weight matching M^*. We have the following: If the optimum value of the maximum weight matching M^* is more than two, we have found a generalized k-double claw inequality violated at $\bar{x}$ with $k = |M^*| \in \{3, \ldots, \lceil \frac{n}{2} \rceil - 1\}$; otherwise, no generalized k-double claw rooted at r and violating inequality (5) exists. ∎

Surprisingly, the following separation problem also can be solved in polynomial time.

THEOREM 4.3. *The separation problem for the classes of combined k-double claw inequalities and the node-split k-double claw inequalities can be solved in polynomial time by computing a series of maximum bipartite matching problems on subgraphs of $G = (V, E)$.*

Proof. The proof is similar to the proof of Theorem 4.2. For the class of combined k-double claw inequalities, we fix both root vertices r_1 and r_2 of the k-double claws

T_1 and T_2, and the only edge $e_0 = (w_0, u_0)$ contained in $T_1 \cap T_2$. We define w_0 to be the only vertex in $N(r_1) \cap N(r_2)$, $N(r_1) \cup N(r_2) = \{w_0, w_1, \ldots, w_t\}$ and the set $F = \{(w_i, u) \in E | i = 1, \ldots, t, w_i \neq w_0$, and $u \neq r_1, r_2, u_0\}$. Any matching M in the subgraph of G induced by F extended by e_0 defines a combined k-double claw T: for every edge $e = (w_i, u) \in M$, we add the edge (w_i, r) in the set $\{(w_i, r_1), (w_i, r_2)\}$ with highest x-value if w_i is adjacent to both roots; otherwise, we add the unique edge $(w_i, r) \in E$ with $r \in \{r_1, r_2\}$. Moreover, we add the edges $e_0, (w_0, r_1), (w_0, r_2)$ to get a combined k-double claw containing e_0 as special edge.

Let $c = \bar{x}_{e_0} + \bar{x}_{(w_0,r_1)} + \bar{x}_{(w_0,r_2)}$. In order to violate the inequality, we must have $\sum_{e \in T} \bar{x}_e > k_1 + k_2 + 3$, which is equivalent to

$$\sum_{(w_i,u) \in M} \left(\bar{x}_{(w_i,u)} + \bar{x}_{(w_i,r)} - 1\right) > 5 - c,$$

where $M \subseteq E$ induces a matching of size $k_1 + k_2 - 2$ in the graph induced by F. We set $\bar{z}_{(w_i,u)} = \bar{x}_{(w_i,u)} + \bar{x}_{(w_i,r)} - 1$ for all $e = (w_i, u) \in F$ and search for a bipartite matching M^* of maximum weight in the graph induced by F with weights $\bar{z}_e$ on the edges $e \in F$. If the optimum value of the maximum weight matching M^* is more than $5 - c$, we have found a combined k-double claw inequality violated at $\bar{x}$; otherwise, no generalized k-double claw rooted at r_1, r_2, containing e_0 as a special edge and violating inequality (6), exists.

The separation problem for the node-split k-double claw inequalities can be solved similarly after fixing the edge $e_0 = (r_1, r_2)$, where r_1 and r_2 denote the root nodes of T_1 and T_2, respectively. □

Padberg and Wolsey [29] have already shown that the separation problem for the inequalities occurring in the weighted forest polytope can be solved in polynomial time.

THEOREM 4.4 (see [29]). *The separation problem of the forest inequalities* (8) *can be solved by computing a minimum cut in a capacitated network G^* constructed from $G = (V, E)$. G^* contains $2(|V| + |E|)$ arcs and $|V| + 2$ vertices.*

We suggest a branch-and-cut algorithm for solving practical instances of the maximum 2-level planar subgraph problem. We will see that our branch-and-cut algorithm is able to find nearly optimal solutions for moderately sized problem instances in reasonable computation time.

We implemented a branch-and-cut algorithm based on the system ABACUS [22] using the separation routines mentioned above. In our algorithm we start with the linear system $\{\max w^T x | x_e \geq 0, x_e \leq 1$ for all $e \in E\}$. Let x^* denote the optimal solution of the LP-system. We solve the separation problem for inequalities (2), (3), (5), and (8) using Theorems 4.1–4.3. We add all the found inequalities to our system and optimize again. The algorithm stops if no violated inequalities of the above mentioned types are found. If x^* is integer, we know that x^* is the incidence vector of a 2-level planar graph. In this case we have found the optimal solution of the 2-level planarization problem. Otherwise, x^* gives us an upper bound to the value of a maximum 2-level planar subgraph of the given instance G. In this case, we branch by setting the value of a fractional variable to zero or one and try to solve the subproblems such as the root node.

In addition, we try to find good solutions to the problem. After each optimization process, we may find new solutions x^* to the problem, most of which are fractional. Fractional solutions x^* may give us a hint about good solutions to the problem.

TABLE 1
Computational results for graphs on 20 *vertices per level.*

$\|V_i\|$	$\|E\|$	Gar	Time	Cycles	2Claw	kClaw	Forest
20	20	0.00	0.01	0.18	0.52	0.00	0.00
20	25	0.00	0.03	0.81	1.65	0.15	0.00
20	30	0.00	0.17	2.36	4.60	0.87	0.05
20	35	0.00	0.52	5.04	16.08	4.52	0.16
20	40	0.00	5.81	11.39	55.69	22.20	1.09
20	45	0.03	25.56	19.38	116.73	92.80	3.93
20	50	0.67	100.38	32.62	185.72	234.21	9.13
20	55	0.53	81.17	25.14	194.21	297.81	8.11
20	60	0.37	56.04	23.69	167.63	277.66	5.78
20	65	0.32	54.25	23.22	188.07	320.39	5.66
20	70	0.13	25.97	18.47	103.06	159.25	1.79
20	75	0.13	21.69	18.33	76.45	139.69	0.92
20	80	0.03	12.37	17.01	61.09	67.38	0.11
20	85	0.10	20.28	18.16	75.91	111.53	0.34
20	90	0.02	7.47	13.46	29.05	34.66	0.09
20	95	0.00	3.94	11.66	15.81	16.77	0.12
20	100	0.00	4.08	10.83	13.90	15.72	0.02

Variables with value close to one will be in the optimal solution with high probability. We try to use this information in our heuristics that we apply in each iteration.

5. Computational results. For our experiments we used the branch-and-cut algorithm described above. The algorithm stops if either the optimal solution is found or if no violated cycle, double claw, generalized double claw, or forest inequality can be detected. Moreover, we put a time limit of five minutes (=300 seconds) on our program. In any case, the program gives a 2-level planar subgraph together with an upper bound of the optimal solution. The random bipartite graphs have been generated using the Stanford GraphBase [24]. We used the separation procedures described above to identify violated inequalities.

Table 1 shows computational results for 100 instances of 2-level graphs with 20 vertices at each level with increasing density. The columns show the number of vertices per level, the number of edges, and the average guarantee of the solution value, i.e., if Sol denotes the number of edges remaining in a found 2-level planar subgraph and UpBound denotes the value determined by the linear programming relaxation, then the solution guarantee Gar is $(\frac{\text{UpBound}-\text{Sol}}{\text{UpBound}}) \times 100\%$. Column 4 shows the time on a SUN Ultra 2/2x200 in seconds. Columns 5–8 show the average number of found violated cycle, double claw, generalized k-double claw, and forest inequalities.

The average quality of the solution value is visualized in Figure 7. The results are surprisingly good. On the average, the solution we found is very close (below 0.7% on average) to the optimal one. The results show that the problem is easy for very sparse graphs, gets harder with increasing density, and is again easy for dense graphs. This can be explained as follows: For very sparse graphs, the used inequalities for our relaxation give very good bounds; hence in most of the cases a branching phase is not necessary. However, for dense graphs there is a high probability that some solutions of cardinality $|V| - 1$ exist; again, the bound given by the cutting planes is good. Figure 8 shows the average running time in seconds averaged over 100 instances per size.

Furthermore, we ran 100 instances on a series of sparse graphs. The results are promising also for these cases (see Table 2). Our solution is at most 5% away from the

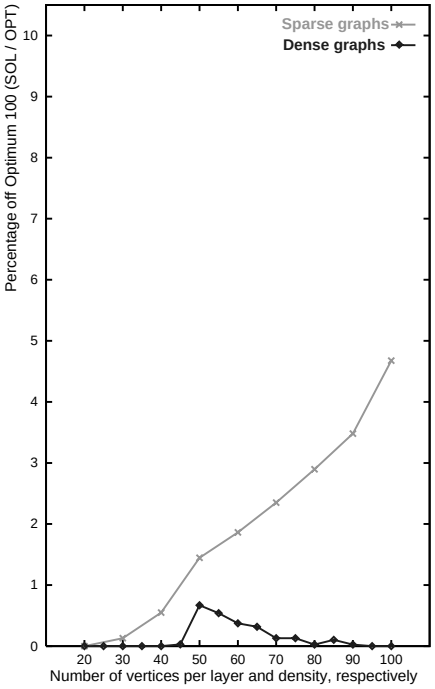


FIG. 7. *Results for* 100 *instances.*

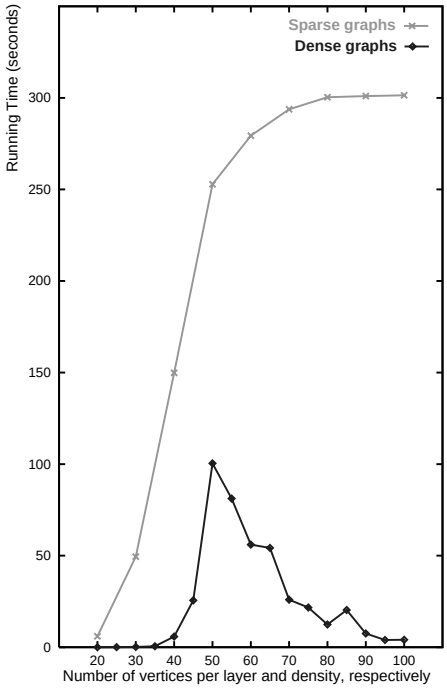


FIG. 8. *Average running times.*

TABLE 2
Computational results for sparse graphs.

$\|V_i\|$	$\|E\|$	Gar	Time	Cycles	2Claw	kClaw	Forest
20	40	0.00	5.97	11.39	55.69	22.20	1.09
30	60	0.13	49.41	15.17	136.79	83.24	2.25
40	80	0.55	149.85	17.46	227.30	137.60	2.99
50	100	1.45	252.81	20.26	309.59	178.53	3.51
60	120	1.86	279.38	22.63	395.82	230.21	1.81
70	140	2.35	293.73	25.65	441.42	222.76	1.64
80	160	2.90	300.37	28.68	534.30	248.17	1.02
90	180	3.48	300.97	31.18	589.81	241.98	0.33
100	200	4.67	300.39	35.26	687.14	237.27	0.31

optimal solution. Figures 7 and 8 visualize the average guarantee of the solution and also the running times for sparse graphs. Many practical instances in graph drawing have up to 30 vertices per layer. Hence, for these instances, regarding the running times, our algorithm is competitive with the classical heuristics used for crossing minimization in graph drawing.

Consider the graph shown in Figure 2. Our branch-and-cut algorithm solved the 2-level planarization problem for the given instance provably optimal within 0.01 seconds. During the run, 5 violated cycle constraints were found, 10 double claw inequalities, 1 generalized k-double claw inequality, and no forest inequality.

Acknowledgments. I started to get interested in the 2-level planarization problem when Peter Eades visited the Max-Planck-Institut für Informatik (Saarbrücken, Germany) in June 1995. Thanks to Peter for initiating my interest in this beautiful

problem. I am very grateful to René Weiskircher for providing the implementation of the branch-and-cut algorithm and for running the computational experiments.

REFERENCES

[1] M. J. Carpano, *Automatic display of hierarchized graphs for computer aided decision analysis*, IEEE Trans. Syst. Man and Cybern., SMC-10 (1980), pp. 705–715.

[2] G. Di Battista and E. Nardelli, *Hierarchies and planarity theory*, IEEE Trans. Syst. Man and Cybern., 18 (1988), pp. 1035–1046.

[3] S. Dresbach, *A new heuristic layout algorithm for DAGs*, in Operations Research Proceedings 1994, U. Derigs, A. Bachem, and A. Drexl, eds., Springer-Verlag, Berlin, 1994, pp. 121–126.

[4] P. Eades and D. Kelly, *Heuristics for drawing* 2*-layered networks*, Ars Combin., 21-A (1986), pp. 89–98.

[5] P. Eades, B. D. McKay, and N. C. Wormald, *On an edge crossing problem*, in Proceedings of the 9th Austral. Comp. Sci. Conference, Australian National University, Canberra, Australia, 1986, pp. 327–334.

[6] P. Eades and S. Whitesides, *Drawing graphs in two layers*, Theoret. Comput. Sci., 131 (1994), pp. 361–374.

[7] P. Eades and N. C. Wormald, *Edge crossings in drawings of bipartite graphs*, Algorithmica, 10 (1994), pp. 379–403.

[8] J. Edmonds, *Submodular functions, matroids and certain polyhedra*, in Combinatorial Structures and Their Applications, Gordon and Breach, London, 1970, pp. 69–87.

[9] A. Fukuda, *Face lattices*, personal communication, 1996.

[10] M. Grötschel, L. Lovász, and A. Schrijver, *The ellipsoid method and its consequences in combinatorial optimization*, Combinatorica, 1 (1981), pp. 169–197.

[11] M. Grötschel and M. W. Padberg, *Polyhedral theory*, in The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization, E. L. Lawler et al., eds., Wiley-Interscience, New York, 1985, pp. 1–10.

[12] F. Harary and A. Schwenk, *A new crossing number for bipartite graphs*, Util. Math., 1 (1972), pp. 203–209.

[13] P. Healy and A. Kuusik, *The vertex-exchange graph: A new concept for multi-level crossing minimization*, in Proceedings of Graph Drawing '99, Lecture Notes in Comput. Sci. 1731, J. Kratochvíl, ed., Springer-Verlag, Berlin, New York, 1999, pp. 205–216.

[14] L. S. Heath and S. V. Pemmaraju, *Recognizing leveled-planar dags in linear time*, in Proceedings of Graph Drawing '95, Lecture Notes in Comput. Sci. 1027, F. Brandenburg, ed., Springer-Verlag, Berlin, New York, 1996, pp. 300–311.

[15] M. Jünger, E. Lee, P. Mutzel, and T. Odenthal, *A polyhedral approach to the multilayer crossing number problem*, in Proceedings of Graph Drawing '97, Lecture Notes in Comput. Sci. 1353, G. Di Battista, ed., Springer-Verlag, Berlin, New York, 1997, pp. 13–24.

[16] M. Jünger, S. Leipert, and P. Mutzel, *Pitfalls of using PQ-trees in automatic graph drawing*, in Proceedings of Graph Drawing '97, Lecture Notes in Comput. Sci. 1353, G. Di Battista, ed., Springer-Verlag, Berlin, New York, 1997, pp. 193–204.

[17] M. Jünger, S. Leipert, and P. Mutzel, *Level planarity testing in linear time*, in Proceedings of Graph Drawing '98, Lecture Notes in Comput. Sci. 1547, S. Whitesides, ed., Springer-Verlag, Berlin, New York, 1998, pp. 224–237.

[18] M. Jünger and P. Mutzel, *Solving the maximum planar subgraph problem by branch and cut*, in Proceedings of the 3rd IPCO Conference, L. A. Wolsey and G. Rinaldi, eds., Erice, CIACO, Louvain-La-Neuve, 1993, pp. 479–492.

[19] M. Jünger and P. Mutzel, *Maximum planar subgraphs and nice embeddings: Practical layout tools*, Algorithmica, 16 (1996), pp. 33–59.

[20] M. Jünger and P. Mutzel, *Exact and heuristic algorithms for* 2*-layer straightline crossing minimization*, in Proceedings of Graph Drawing '95, Lecture Notes in Comput. Sci. 1027, F. Brandenburg, ed., Springer-Verlag, Berlin, New York, 1996, pp. 337–348.

[21] M. Jünger and P. Mutzel, 2*-layer straightline crossing minimization: Performance of exact and heuristic algorithms*, J. Graph Algorithms Appl., 1 (1996), pp. 1–25. Available online at http://www.cs.brown.edu/publications/jgaa/.

[22] M. Jünger and S. Thienel, *The ABACUS-system for branch and cut and price algorithms in integer programming and combinatorial optimization*, Softw. Pract. Exper., 30 (2000), pp. 1325–1352.

[23] R. M. Karp and C. H. Papadimitriou, *On linear characterizations of combinatorial optimization problems*, in Proceedings of the 21st Annual Symposium on the Foundations of Computer Science, IEEE, Piscataway, NJ, 1980, pp. 1–9.
[24] D. E. Knuth, *The Stanford GraphBase: a Platform for Combinatorial Computing*, Addison-Wesley, New York, 1993.
[25] T. Lengauer, *Combinatorial Algorithms for Integrated Circuit Layout*, John Wiley & Sons, Chichester, UK, 1990.
[26] E. Mäkinen, *Experiments on drawing* 2*-level hierarchical graphs*, Int. J. Comput. Math., 37 (1990), pp. 129–135.
[27] P. Mutzel, *The Maximum Planar Subgraph Problem*, Dissertation, Universität zu Köln, Germany, 1994.
[28] M. W. Padberg and M. R. Rao, *The Russian Method for Linear Inequalities* III: *Bounded Integer Programming*, GBA Working Paper, New York University, New York, 1981.
[29] M. W. Padberg and L. A. Wolsey, *Trees and cuts*, Ann. Discrete Math., 17 (1983), pp. 511–517.
[30] W. R. Pulleyblank, *Polyhedral combinatorics*, in Optimization, Handbooks Oper. Res. Management Sci. 1, G.L. Nemhauser, A. H. G. Rinnoy Kan, and M. J. Todd, eds., North-Holland, Amsterdam, 1989, pp. 371–446.
[31] F. Shahrokhi, O. Sýkora, L. A. Székely, and I. Vrťo, *On bipartite crossings, largest biplanar subgraphs, and the linear arrangement problem*, in Proceedings of the Workshop on Algorithms and Data Structures (WADS '97), Lecture Notes in Comput. Sci. 1272, F. Dehne et al., eds., Springer-Verlag, Berlin, New York, 1997, pp. 55–68.
[32] K. Sugiyama, S. Tagawa, and M. Toda, *Methods for visual understanding of hierarchical system structures*, IEEE Trans. Syst. Man Cybern., SMC-11 (1981), pp. 109–125.
[33] N. Tomii, Y. Kambayashi, and Y. Shuzo, *On Planarization Algorithms of* 2*-Level Graphs*, Technical report EC77-38, Institute of Electronic and Communication Engineers of Japan (IECEJ), Tokyo, Japan, 1977.
[34] J. D. Ullman, *Computational Aspects of VLSI*, Computer Science Press, Rockville, MD, 1984.
[35] V. Valls, R. Marti, and P. Lino, *A branch and bound algorithm for minimizing the number of crossing arcs in bipartite graphs*, J. Oper. Res., 90 (1996), pp. 303–319.
[36] M. Vingron, H.-P. Lenhof, and P. Mutzel, *Computational molecular biology*, in Annotated Bibliographies in Combinatorial Optimization, M. Dell'Amico, F. Maffioli, and S. Martello, eds., John Wiley & Sons, New York, 1997, pp. 445–471.
[37] J. N. Warfield, *Crossing theory and hierarchy mapping*, IEEE Trans. Syst. Man Cybern., SMC-7 (1977), pp. 505–523.
[38] M. S. Waterman and J. R. Griggs, *Interval graphs and maps of DNA*, Bull. Math. Biology, 48 (1986), pp. 189–195.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1081–1091

APPROXIMATE MAX-MIN RESOURCE SHARING FOR STRUCTURED CONCAVE OPTIMIZATION*

M. D. GRIGORIADIS†, L. G. KHACHIYAN†, L. PORKOLAB‡, AND J. VILLAVICENCIO§

Abstract. We present a Lagrangian decomposition algorithm which uses logarithmic potential reduction to compute an ε-approximate solution of the general max-min resource sharing problem with M nonnegative concave constraints on a convex set B. We show that this algorithm runs in $O(M(\varepsilon^{-2} + \ln M))$ iterations, a data independent bound which is optimal up to polylogarithmic factors for any fixed relative accuracy $\varepsilon \in (0,1)$. In the block-angular case, B is the product of K convex sets (blocks) and each constraint function is block separable. For such models, an iteration of our method requires a $\Theta(\varepsilon)$-approximate solution of K independent block maximization problems which can be computed in parallel.

Key words. approximation algorithm, covering problem, Lagrangian decomposition, logarithmic potential, packing problem, resource sharing, structured optimization

AMS subject classifications. 68Q25, 90C05, 90C27, 90C30, 90C06

PII. S1052623499358689

1. Introduction.

We consider the approximate solution of concave *max-min resource sharing* problems of the form

$$(\mathcal{P}) \qquad \lambda^* = \max\{\ \lambda \mid f(x) \geq \lambda e,\ x \in B\},$$

where $f : B \to \mathbb{R}^M$ is a given vector of M nonnegative continuous concave functions defined on a nonempty convex compact set B, called *block*, e is the vector of all ones and, with no loss of generality, $\lambda^* > 0$. We shall denote by $\mathbb{R}^M_+$ ($\mathbb{R}^M_{++}$) the nonnegative (positive) orthants of $\mathbb{R}^M$ and denote $\lambda(f) \doteq \min_{1 \leq m \leq M} f_m$ for any given $f \in \mathbb{R}^M_+$.

We shall be interested in computing an *ε-approximate solution* of this problem; i.e., for a given *relative tolerance* $\varepsilon \in (0,1)$,

$$(\mathcal{P}_\varepsilon) \qquad \text{compute } x \in B \text{ that satisfies } \ f(x) \geq (1-\varepsilon)\lambda^* e.$$

Our approach is based on the well-known duality relation

$$\lambda^* = \max_{x \in B} \min_{p \in P} p^T f(x) = \min_{p \in P} \max_{x \in B} p^T f(x), \tag{1.1}$$

where $P \doteq \{p \in \mathbb{R}^M_+ \mid e^T p = 1\}$. It follows that

$$\lambda^* = \min\{\Lambda(p) \mid p \in P\}, \qquad \textit{(Lagrangian dual)} \tag{1.2}$$

where

$$\Lambda(p) = \max\{p^T f(x) \mid x \in B\}. \qquad \textit{(Block problem)} \tag{1.3}$$

*Received by the editors July 2, 1999; accepted for publication (in revised form) October 13, 2000; published electronically May 10, 2001. This research was supported by the National Science Foundation under grant CCR-9618796.

http://www.siam.org/journals/siopt/11-4/35868.html

†Department of Computer Science, Busch Campus, Rutgers University, New Brunswick, NJ 08903 (grigoria@cs.rutgers.edu, leonid@cs.rutgers.edu).

‡Department of Computing, Imperial College, London, England (porkolab@doc.ic.ac.uk).

§Facultad de Matemáticas, Pontificia Universidad Católica de Chile, Santiago, Chile (jvillavi@puc.cl).

The exact optimality conditions for $\mathcal{P}$ can thus be stated as follows: *A pair $x \in B$, $p \in P$ is optimal if and only if $\Lambda(p) = \lambda(f(x))$.*

In its simplest form, *Lagrangian* or *price-directive decomposition* is an iterative strategy that solves $\mathcal{P}$ via its Lagrangian dual by computing a sequence of pairs p, x as follows. A *coordinator* uses the current $x \in B$ to compute some weights $p = p(f(x)) \in P$ corresponding to the coupling constraints $f(x) \geq \lambda e$, calls a *block solver* to compute a solution $\hat{x} \in B$ of (1.3) for this $p \in P$, and then makes a move from x to $(1-\tau)x + \tau\hat{x}$ with an appropriate step length $\tau \in (0, 1]$. We call each such Lagrangian decomposition iteration a *coordination step*.

We shall only require an *approximate block solver* ($\mathcal{ABS}$), one that solves (1.3) to a given optimization tolerance $t > 0$, defined below:

$$\mathcal{ABS}(p,t): \qquad \text{compute } \ \hat{x} = \hat{x}(p) \in B \text{ such that } p^T f(\hat{x}) \geq (1-t)\Lambda(p).$$

We shall eventually set $t = \Theta(\varepsilon)$ in our algorithm.

By analogy to $\mathcal{P}_\varepsilon$ and based on the fact that λ^* is the optimal value of the Lagrangian dual (1.2), we define the *ε-approximate dual problem* as follows:

$$(\mathcal{D}_\varepsilon) \qquad \text{compute } \ p \in P \ \text{ that satisfies } \Lambda(p) \leq (1+\varepsilon)\lambda^*.$$

The purpose of this paper is to present a simple approximation algorithm that, for a given relative accuracy $\varepsilon \in (0, 1)$, solves problems $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$ in

$$N = O\left(M\left(\varepsilon^{-2} + \ln M\right)\right) \tag{1.4}$$

coordination steps, each of which requires a call to $\mathcal{ABS}(p, \Theta(\varepsilon))$ and a coordination overhead of $O(M \ln(M/\varepsilon))$ arithmetic operations.

We note that our iteration bound (1.4) can be improved by using polynomial-time methods of nondifferentiable optimization for minimizing $\Lambda(p)$, such as the methods of inscribed ellipsoids [13], [9], volumetric centers [14], and cutting plane methods based on analytic centers, e.g., [1], [3], [4]. Specifically, problem $\mathcal{D}_\varepsilon$ can be solved in $O(M \ln(M/\varepsilon))$ iterations by the inscribed ellipsoid method of [13] or by the method of volumetric centers [14] and in $O(M \ln^2(M/\varepsilon))$ iterations by a cutting plane method based on analytic centers [1]. (The primal solution can be recovered in a postprocessing step by solving a linear program that maximizes λ over the convex hull of those primal iterates generated along the way.) The coordination tasks for the above methods, however, are much more complex, require all or a substantial part of the history of iterates, as well as a more accurate block solver. In particular, the theoretically fastest of these methods (volumetric centers) requires $O(\mathcal{M}(M))$ arithmetic operations per coordination step, where $\mathcal{M}(M)$ is the cost of matrix multiplication for M-order matrices. With the best currently known bound of $\mathcal{M}(M) = O(M^{2.38})$, and disregarding the large hidden constant factors, this is still substantially higher than the almost linear coordination overhead for the method suggested in this paper.

On the other hand, it is easy to see that for $f(x) = x$ and $B = P$, any algorithm that solves $\mathcal{P}_\varepsilon$ for $\varepsilon < 1$ by a sequence of block optimizations (1.3) must perform at least M coordination steps: no Lagrangian decomposition scheme for this instance can bring in more than one new vertex of the standard simplex B per iteration, whereas all the M vertices of B are needed to compute an approximate solution x with $\lambda(x) > 0$. This is true even for methods that use arbitrarily powerful coordinators which can compute p by taking into account previous iterates and block solutions, as well as other data (see [7] for further discussion). It follows that, for a fixed $\varepsilon \in (0, 1)$, the

bound (1.4) on coordination steps is optimal to within a factor of $\ln M$. A similar lower bound of $\Omega(M)$ iterations holds for min-max problems [7], for which a more recent data-dependent lower bound below $\Theta(M^{1/2})$ iterations is suggested in [10]. Information-theoretic lower bounds for matrix games can be found in [6].

The linear feasibility variant of $\mathcal{P}$, i.e., find $x \in B$ such that $f(x) = Ax \geq e$, often referred to as the *fractional covering* problem, is solved in [12] by Lagrangian decomposition using exponential potential reduction. Up to polylogarithmic factors, the number of iterations of that algorithm is proportional to M and ρ/ε^2, where the data-dependent quantity $\rho = \max_m \max_{x \in B} f_m(x)$ is the *width* of B *relative to* $Ax \geq e$. The problem is further decomposed in [12] in a way that reduces this linear dependence on ρ down to $\log \rho$. However, this introduces additional constraints on the block problems which, in general, become NP-hard.

The logarithmic potential reduction algorithm for max-min optimization presented in this paper circumvents the issue of width altogether and uses the approximate block solvers $\mathcal{ABS}$ on the original blocks (cf. [5] and [7]). A width-independent Lagrangian decomposition iteration bound for general min-max sharing based on logarithmic potential function reduction was given in [7]. This bound was recently matched in [2] using an exponential potential reduction technique. It would thus be worthwhile to develop a width-independent exponential function reduction method for the max-min case as well.

Although our iteration bound (1.4) is independent of the dimension and structure of problem P, the overall running time of our algorithm will depend upon these factors and can be significantly enhanced if the block B and coupling functions f have special structures. A prevalent example is the *block-angular* structure for which $B = B^1 \times \cdots \times B^K$ for $K > 1$ given nonempty convex compact sets (blocks) B^k, $k = 1, \ldots, K$, and $f_m(x) = \sum_{k=1}^K f_m^k(x^k)$, where $x = (x^1, x^2, \ldots, x^K)$ and $f_m^k(x^k)$, $m = 1, \ldots, M$, are given continuous nonnegative concave functions of $x^k \in B^k$. For such problems, $\Lambda(p) = \sum_{k=1}^K \Lambda^k(p)$, where

$$\Lambda^k(p) \doteq \max\{p^T f^k(x^k) \mid x^k \in B^k\}$$

and $\mathcal{ABS}(p,t)$ decomposes into K independent block solvers $\mathcal{ABS}^k(p,t)$, $k = 1, \ldots, K$, each operating to the same accuracy $t > 0$. Further special structures for $K = 1$ or within each B^k also arise in practice and can be exploited to facilitate this task. The running time of implementations of the algorithm can thus be estimated by combining (1.4), the coordination overhead, and the time complexity of the block solvers for specific classes of models for $\mathcal{P}$ (see, e.g., [5], [12], [7], [8]), but we shall not pursue such specializations here.

The paper is organized as follows. In section 2 we define the standard logarithmic barrier function and examine some of its properties. In section 3 we develop our algorithm for solving $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$ and prove its correctness. Finally, we analyze the coordination complexity of the algorithm in section 4. We will use the following notational abbreviations: $f \doteq f(x)$, $f' \doteq f(x')$, $\hat{f} \doteq f(\hat{x})$, $p \doteq p(f) \doteq p(f(x))$, for points x, x', $\hat{x} \in B$, respectively. The symbol e denotes the vector of all ones while e_i represents the ith unit vector.

2. Logarithmic potential function. We shall associate with the coupling inequalities $f \geq \lambda e$ the standard logarithmic potential function (see, e.g., Chapter 4 of

[11]) of the form

$$\Phi_t(\theta, f) = \ln\theta + \frac{t}{M}\sum_{m=1}^{M} \ln(f_m - \theta), \tag{2.1}$$

where $\theta \in \mathbb{R}$, $f = (f_1, f_2, \ldots, f_M)$ are variables and t is a fixed positive parameter, identical to that used for $\mathcal{ABS}(p,t)$. The function Φ_t is well defined for $0 < \theta < \lambda(f) = \min\{f_1, f_2, \ldots, f_M\}$. This implies that $f \in \mathbb{R}^M_{++}$, which will be the case for all iterates of the algorithm we shall present in section 3.

Similar to [7], [15], we define the *reduced potential function* as the maximum of $\Phi_t(\theta, f)$ over $\theta \in (0, \lambda(f))$ for a fixed $f \in \mathbb{R}^M_{++}$, i.e.,

$$\phi_t(f) = \max_{0<\theta<\lambda(f)} \Phi_t(\theta, f). \tag{2.2}$$

The maximizer $\theta(f)$ of $\Phi_t(\theta, f)$ can be determined from the first-order optimality condition

$$\frac{t\theta}{M}\sum_{m=1}^{M} \frac{1}{f_m - \theta} = 1, \tag{2.3}$$

which has a unique root since its left side is a strictly increasing function of θ. We can thus write the reduced potential function as

$$\phi_t(f) = \Phi_t(\theta(f), f).$$

It is easy to see that the smooth function $\theta(f)$ approximates the piecewise nonlinear concave function $\lambda(f)$ as follows:

$$\frac{\lambda(f)}{1+t} \le \theta(f) \le \frac{\lambda(f)}{1+t/M}, \tag{2.4}$$

a property which motivates our approach.

Next, we define the *logarithmic dual vector* $p = p(f)$ for a fixed $f \in \mathbb{R}^M_{++}$ to be

$$p_m(f) = \frac{t}{M}\frac{\theta(f)}{f_m - \theta(f)}, \qquad m = 1, \ldots, M, \tag{2.5}$$

where $p(f) \in P$ by (2.3). A useful consequence of this definition is the following identity.

PROPOSITION 1. $p(f)^T f = (1+t)\theta(f)$.

Proof. Denoting $\theta = \theta(f)$, we write

$$\begin{aligned} p^T f &= \frac{t\theta}{M}\sum_{m=1}^{M}\frac{f_m}{f_m - \theta} = \frac{t\theta}{M}\sum_{m=1}^{M}\left(1 + \frac{\theta}{f_m - \theta}\right) \\ &= t\theta + \theta\sum_{m=1}^{M} p_m = (1+t)\theta. \qquad \square \end{aligned}$$

A more important observation is that the accuracy with which the optimality criteria $\Lambda(p) = \lambda(f)$ are to be satisfied at a given point $x \in B$ can be approximated by the quantity

$$\nu \doteq \nu(x, \hat{x}) = \frac{p^T\hat{f} - p^T f}{p^T\hat{f} + p^T f}, \tag{2.6}$$

where $p \in P$ from (2.5), $f = f(x)$, and $\hat{f} = f(\hat{x})$ for an approximate block solution $\hat{x} \in B$ produced by $\mathcal{ABS}(p,t)$. The lemma below states that a pair x, p solves $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, respectively, whenever ν and t are of order ε.

LEMMA 1. *Suppose $\varepsilon \in (0,1)$ and $t = \varepsilon/6$. For a given point $x \in B$, let $p \in P$ be computed by (2.5) and $\hat{x}$ computed by $\mathcal{ABS}(p,t)$. If $\nu(x,\hat{x}) \le t$, then the pair x, p solves $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, respectively.*

Proof. Use (2.6) to rewrite the condition $\nu \le t$ as follows:

$$(1-t)p^T\hat{f} \le (1+t)p^T f.$$

Since $p^T\hat{f} \ge \Lambda(p)(1-t)$, $p^T f = (1+t)\theta$ by Proposition 1 and $\theta = \theta(f) < \lambda(f)$, we obtain

$$\Lambda(p) \le \frac{1+t}{(1-t)^2} p^T f \le \left(\frac{1+t}{1-t}\right)^2 \lambda(f) \le (1+\varepsilon)\lambda(f), \tag{2.7}$$

where the last inequality follows from the assumption $t = \varepsilon/6$ and $\varepsilon \in (0,1)$. Given that $\Lambda(p) \ge \lambda^*$, we have $\lambda^* \le (1+\varepsilon)\lambda(f) \le \lambda(f)/(1-\varepsilon)$, and thus $x \in B$ solves $\mathcal{P}_\varepsilon$. On the other hand, $\lambda(f) \le \lambda^*$, so that (2.7) implies $\Lambda(p) \le (1+\varepsilon)\lambda^*$, and hence $p \in P$ solves $\mathcal{D}_\varepsilon$. ∎

Remark 1. The bound of Lemma 1 is close to the best possible: for $\varepsilon \in (0, 1/2]$ we have $\nu \le 4\varepsilon$ for any pair $x \in B$, $p \in P$ that solves $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, respectively. To see this, first consider that

$$\Lambda(p) \le (1+\varepsilon)\lambda^* \le \frac{1+\varepsilon}{1-\varepsilon}\lambda(f), \tag{2.8}$$

which simplifies to $\Lambda(p) \le (1+4\varepsilon)\lambda(f)$ for $\varepsilon \in (0, 1/2]$. Then, use this inequality, (2.6), and the fact that $\lambda(f) \le p^T f$, to write

$$\nu \le \frac{\Lambda(p) - p^T f}{p^T\hat{f} + p^T f} \le \frac{4\varepsilon p^T f}{p^T\hat{f} + p^T f} \le 4\varepsilon. \quad ∎$$

3. The approximation algorithm. We shall now state our Algorithm $\mathcal{A}$ to compute solutions of both problems $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, as a direct implementation of the Lagrangian decomposition scheme stated in the introduction, section 1. The algorithm accepts as input f, B, $\varepsilon \in (0,1)$ and an initial point $x = x^0 \doteq \frac{1}{M}\sum_{m=1}^{M} \hat{x}^{(m)} \in B$, computed by $\hat{x}^{(m)} := \mathcal{ABS}(e_m, 1/2)$.

ALGORITHM $\mathcal{A}(f, B, \varepsilon, x)$.
$t := \varepsilon/6$
repeat
  Compute $\theta(f)$ from (2.3) and $p \in P$ from (2.5).
  $\hat{x} := \mathcal{ABS}(p,t)$.
  Compute $\nu \doteq \nu(x,\hat{x})$ from (2.6).
  if $\nu \le t$ **then return**(x,p)
  else $x := (1-\tau)x + \tau\hat{x}$, for an appropriate step length $\tau \in (0,1]$ **end**
end

Our subsequent analysis uses the step length

$$\tau = \frac{t\theta\nu}{2M(p^T\hat{f} + p^T f)}, \tag{3.1}$$

which is *strictly feasible*, i.e., $\tau \in (0,1)$. (To see this, substitute $t = \varepsilon/6$ in (3.1) and use the inequality $\theta/(p^T\hat{f} + p^T f) \le 1$, which is a straightforward consequence of Proposition 1.) In practice, one usually computes τ by performing a line search to maximize $\phi_t(x + \tau(\hat{x} - x))$. Our analysis remains valid for such step lengths.

Algorithm $\mathcal{A}$ is correct since $\tau \in (0,1)$ as indicated above and since, by Lemma 1, the pair $x \in B$, $p \in P$ solves $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, respectively, when the algorithm halts.

4. Analysis of the approximation algorithm. Our next task is to derive the iteration bound (1.4) for our algorithm as claimed in the introduction, section 1. We shall first establish several observations. In Lemma 2 we bound the error in the initial approximation $x^0 \in B$ as defined in section 3. Lemma 3 shows that each coordination step achieves a sizable guaranteed increase in the value of the reduced potential function $\phi_t(f)$. In contrast, Lemma 4 bounds the sum of such increases between any two, not necessarily consecutive, iterates. These observations ultimately lead us to the iteration bound stated in Theorem 1, which is further improved by employing a simple error-scaling technique akin to that used in [15] for structured min-max problems.

LEMMA 2. *$\lambda^* \le \Lambda(p) \le 2Mp^T f(x^0)$ for all $p \in P$.*

Proof. The left inequality is from (1.2)–(1.3). To show the right inequality, note that for any $p \in P$, (1.3) provides

$$\Lambda(p) = \max\{p^T f(x) \mid x \in B\} \le \sum_{m=1}^{M} p_m \max\{f_m(x) \mid x \in B\} = \sum_{m=1}^{M} p_m \Lambda(e_m).$$

Now, $\Lambda(e_m) \le 2f_m(\hat{x}^{(m)})$, where $\hat{x}^{(m)}$ is the approximate block solution computed by $\mathcal{ABS}(e_m, 1/2)$. Then, by using the concavity of the nonnegative functions f_m, we obtain

$$f_m(\hat{x}^{(m)}) \le \sum_{\ell=1}^{M} f_m(\hat{x}^{(\ell)}) \le M f_m\left(\frac{1}{M}\sum_{\ell=1}^{M} \hat{x}^{(\ell)}\right) = M f_m(x^0). \qquad \square$$

Next, we prove that the increase in the reduced potential $\phi_t(f)$ is sufficiently large at each iteration.

LEMMA 3. *For any two consecutive iterates x, x' of Algorithm $\mathcal{A}$:*

$$\phi_t(f') - \phi_t(f) \ge t\nu^2/4M.$$

Proof. From Algorithm $\mathcal{A}$ we have $x' = (1-\tau)x + \tau\hat{x}$. Denote $\Lambda \doteq \Lambda(p(f))$. By the concavity of the f_m and definition (2.5),

$$\begin{aligned} f'_m - \theta &\ge (1-\tau)f_m + \tau\hat{f}_m - \theta = (f_m - \theta)\left(1 + \tau\frac{\hat{f}_m - f_m}{f_m - \theta}\right) \\ &= (f_m - \theta)\left(1 + \frac{\tau M}{t\theta} p_m(\hat{f}_m - f_m)\right). \end{aligned} \tag{4.1}$$

In order to bound the last expression above, consider that by definition (3.1),

$$\left|\frac{\tau M}{t\theta} p_m(\hat{f}_m - f_m)\right| \le \frac{\tau M}{t\theta} p_m(\hat{f}_m + f_m) \le \frac{\tau M}{t\theta}(p^T\hat{f} + p^T f) = \frac{\nu}{2} \le \frac{1}{2}.$$

Accordingly, (4.1) gives $f'_m - \theta > 0,\ m = 1, \ldots, M$, so that $\lambda(f') > \theta$. From the definition (2.2) of $\phi_t(f')$,

$$\phi_t(f') = \max_{0<\xi<\lambda(f')} \Phi_t(\xi, f') \geq \Phi_t(\theta, f') = \ln\theta + \frac{t}{M}\sum_{m=1}^{M} \ln(f'_m - \theta), \tag{4.2}$$

where θ denotes the root $\theta(f)$ of (2.3). By combining inequalities (4.1) and (4.2) and by using the definition of $\phi_t(f) \equiv \Phi(\theta(f), f)$, we obtain

$$\phi_t(f') \geq \phi_t(f) + \frac{t}{M}\sum_{m=1}^{M} \ln\left(1 + \frac{\tau M}{t\theta} p_m(\hat{f}_m - f_m)\right). \tag{4.3}$$

We now use the inequality $\ln(1+z) \geq z - z^2$ for all $z \geq -1/2$ to write

$$\phi_t(f') - \phi_t(f) \geq \tau\frac{p^T\hat{f} - p^T f}{\theta} - \frac{\tau^2 M}{\theta^2 t}\left\|D(f - \hat{f})\right\|^2,$$

where $D = \text{diag}(p_1, \ldots, p_m)$ and $\|\cdot\|$ denotes the 2-norm. Furthermore, $\|D(f - \hat{f})\| \leq \|D(f + \hat{f})\|_1 = p^T f + p^T\hat{f}$, which implies that

$$\phi_t(f') - \phi_t(f) \geq \tau\frac{p^T\hat{f} - p^T f}{\theta} - \frac{\tau^2 M}{\theta^2 t}(p^T f + p^T\hat{f})^2.$$

This last inequality proves the claim for the value of τ given by (3.1). ∎

In contrast to the previous lemma, our third observation provides that the increase in the reduced potential $\phi_t(f)$ cannot be too large even after an arbitrary number of iterations.

LEMMA 4. *For any two points $x, x' \in B$ such that $\lambda(f) > 0$ and $\lambda(f') > 0$,*

$$\phi_t(f') - \phi_t(f) \leq (1+t)\ln\frac{\Lambda(p)}{p^T f},$$

where p is defined by (2.5).

Proof. Denote $\theta \doteq \theta(f)$, $\theta' \doteq \theta(f')$, and $\Lambda \doteq \Lambda(p) = \max\{p^T f(x) \mid x \in B\}$ as defined in (1.3). Then we can write

$$\begin{aligned}
\phi_t(f') - \phi_t(f) &= \ln\frac{\theta'}{\theta} + \frac{t}{M}\sum_{m=1}^{M}\ln\left(\frac{f'_m - \theta'}{f_m - \theta}\right) \\
&= \ln\frac{\theta'}{\theta} + \frac{t}{M}\sum_{m=1}^{M}\ln\left(\frac{M}{t\theta}p_m\,(f'_m - \theta')\right) \\
&= \ln\frac{\theta'}{\theta} + t\ln\frac{M}{t\theta} + \frac{t}{M}\sum_{m=1}^{M}\ln\left(p_m\,(f'_m - \theta')\right).
\end{aligned}$$

Next, using the concavity of $\ln(\cdot)$ in the last expression we obtain

$$\begin{aligned}
\phi_t(f') - \phi_t(f) &\leq \ln\frac{\theta'}{\theta} + t\ln\frac{M}{t\theta} + t\ln\left(\frac{1}{M}p^T(f' - \theta' e)\right) \\
&\leq \ln\frac{\theta'}{\theta} + t\ln\frac{1}{t\theta} + t\ln\left(\Lambda - \theta'\right),
\end{aligned}$$

which is further simplified as follows:

$$\phi_t(f') - \phi_t(f) \le \max_{\xi\in(0,\Lambda)} \left\{ \ln\frac{\xi}{\theta} + t\ln\frac{1}{t\theta} + t\ln(\Lambda-\xi) \right\}$$
$$= (1+t)\ln\left(\Lambda/(1+t)\theta\right) = (1+t)\ln\left(\Lambda/p^T f\right). \qquad \square$$

We shall apply this lemma to any two, not necessarily consecutive, iterates $x, x' \in B$ of Algorithm $\mathcal{A}$. Clearly, $\lambda(f(x)) > 0$ for the initial iterate, but also for each subsequent x, since every step τ taken by the algorithm is strictly feasible.

We are now in a position to address the coordination complexity of Algorithm $\mathcal{A}$ by combining the lower and upper bounds for the increase in $\phi_t(f)$ we have thus far obtained.

THEOREM 1. *For any given relative accuracy $\varepsilon \in (0,1)$, Algorithm $\mathcal{A}$ solves problems $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$ in*

$$N = O\left(M\left(\varepsilon^{-1}\ln M + \varepsilon^{-2}\right)\right)$$

coordination steps.

Proof. First, let N_0 be the number of iterations of Algorithm $\mathcal{A}$ required to obtain an iterate x^1 with a corresponding optimality error $\nu \le 1/2$, starting from the initial point x^0. For as long as $\nu > 1/2$, each iteration increases the reduced potential by at least $t/16M$ (Lemma 3). However, by Lemma 4, the total increase in the value of the reduced potential can be bounded as follows:

$$\phi_t(f^1) - \phi_t(f^0) \le (1+t)\ln\left(\Lambda(p^0)/p^{0T}f^0\right). \tag{4.4}$$

Since $t = \varepsilon/6$ and $\Lambda(p^0) \le 2Mp^{0T}f^0$ by Lemma 2, it follows that $N_0 = O\left(\varepsilon^{-1}M\ln M\right)$.

Next, suppose that the error is $\nu_\ell \le 1/2^\ell$ for some iterate $x^\ell \in B$ and let N_ℓ be the number of iterations required to halve this error, for $\ell = 1, 2, \ldots$. Again, Lemma 3 provides

$$\phi_t(f^{\ell+1}) - \phi_t(f^\ell) \ge N_\ell\ t\nu_\ell^2/16M. \tag{4.5}$$

To bound the left side of this inequality, consider that

$$(1-\nu_\ell)p^{\ell T}\hat{f}^\ell = (1+\nu_\ell)p^{\ell T}f^\ell,$$

directly from the definition of ν_ℓ in (2.6). And since $p^{\ell T}\hat{f}^\ell \ge (1-t)\Lambda(p^\ell)$ for $\mathcal{ABS}(p^\ell, t)$, we have

$$\frac{\Lambda(p^\ell)}{p^{\ell T}f^\ell} \le \frac{1+\nu_\ell}{(1-t)(1-\nu_\ell)}.$$

This inequality, along with the fact that $t \le \nu_\ell \le 1/2$, implies

$$\frac{\Lambda(p^\ell)}{p^{\ell T}f^\ell} \le \frac{1+\nu_\ell}{(1-\nu_\ell)^2} \le 1 + 10\nu_\ell.$$

Now, Lemma 4 for $x' \doteq x^{\ell+1}$ and $x \doteq x^\ell$ provides

$$\phi_t(f^{\ell+1}) - \phi_t(f^\ell) \le (1+t)\ln(1+10\nu_\ell) \le 10(1+t)\nu_\ell,$$

which, together with (4.5), results in the bound $N_\ell = O(M/t\nu_\ell)$. The total number of coordination steps N in the claim is obtained by summing the N_ℓ over $\ell = 0, 1, \ldots, \lceil \lg(1/\varepsilon) \rceil$. □

The coordination complexity of Algorithm $\mathcal{A}$ given by Theorem 1 is for a fixed value of the parameter t. The algorithm can be implemented and its coordination complexity improved by embedding Algorithm $\mathcal{A}$ within a sequence of *scaling phases* that gradually reduce t to the desired accuracy, much like implementations of path-following methods for convex programs. The sth scaling phase, $s = 0, 1, \ldots$, sets $\varepsilon_s := \varepsilon_{s-1}/2$, correspondingly $t_s := \varepsilon_s/6$, and uses the current approximate point x^{s-1} as its initial solution. For the initial scaling phase we invoke Algorithm $\mathcal{A}$ of section 3 (along with its own initialization step) to compute a pair of 1/4-approximate solutions x, p of problems $\mathcal{P}$ and $\mathcal{D}$, respectively. We then define $x^0 \doteq x$, $p^0 \doteq p$, and $\varepsilon_0 \doteq \frac{1}{4}$ to be the solution of the 0th scaling phase. The resulting coordination complexity of the overall scheme is analyzed below.

THEOREM 2. *For any given relative accuracy $\varepsilon \in (0, 1)$, the error scaling implementation of Algorithm $\mathcal{A}$ computes solutions x, p of problems $\mathcal{P}_\varepsilon$ and $\mathcal{D}_\varepsilon$, respectively, in*

$$N = O\left(M\left(\ln M + \varepsilon^{-2}\right)\right)$$

coordination steps.

Proof. Denote by $\mathcal{N}_s$ the number of coordination steps in the $s+1$st scaling phase, $s = 0, 1, \ldots$. By Theorem 1, $\mathcal{N}_0 = O(M \ln M)$. It remains to show that $\mathcal{N}_s = O\left(M/\varepsilon_s^2\right)$ for each subsequent scaling phase.

Our arguments are analogous to those used in the proof of Theorem 1. By Lemma 3, each iteration of the sth scaling phase increases the (current) reduced potential function by $t_s\nu_s^2/4M \geq t_s^3/4M = \Omega(\varepsilon_s^3/M)$.

By invoking Lemma 4 with $x \doteq x^s$ and $x' \doteq x^{s+1}$, the total increase of the potential in the sth phase can be bounded by

$$\phi_{t_s}(f^{s+1}) - \phi_{t_s}(f^s) \leq (1 + \varepsilon_s/6) \ln\left(\Lambda^s / p^{sT} f^s\right). \tag{4.6}$$

Furthermore, since x^s, p^s are $2\varepsilon_s$-approximate solutions of problem $\mathcal{P}$ and $\mathcal{D}$, respectively,

$$\Lambda^s \leq \frac{1 + 2\varepsilon_s}{1 - 2\varepsilon_s} \lambda\left(f^s\right) \leq (1 + 8\varepsilon_s) p^{sT} f^s. \tag{4.7}$$

The bound $N_s = O(M/\varepsilon_s^2)$ is deduced from (4.7) and the fact that $\ln(1 + 8\alpha) \leq 8\alpha$ for all $\alpha > 0$.

As before, the overall coordination complexity is obtained by adding the coordination bounds N_s for all scaling phases. □

Remark 2. We implicitly assumed in the foregoing that $p \in P$ was computed exactly from (2.5) at each coordination step. In fact, this is impractical since (2.5) requires the root $\theta(f)$ of (2.3), which can be computed only approximately. How accurate then does this computation need to be so that the iteration bounds given by Theorems 1 and 2 remain valid?

Let $\tilde{p}$ approximate the exactly computed p to a relative accuracy of $\delta \in (0, 1/2)$, i.e., $(1 - \delta)p \leq \tilde{p} \leq (1 + \delta)p$. Then it is easy to see that any solution provided by $\mathcal{ABS}(\tilde{p}, t))$ solves the block problem (1.3) to a relative accuracy of $O(t + \delta) = O(\varepsilon)$ for $\delta = O(\varepsilon)$. It follows that (2.5) need only provide the p_m's to a relative accuracy of $\delta = O(\varepsilon)$.

To estimate the required accuracy of $\theta(f)$, first note that (2.3) is homogeneous: $\theta(sf) = s\theta(f)$ and $p(sf) = p(f)$ for any positive scalar s. The M-vector f can thus be locally prescaled (for the computation of $\theta(f)$) so that $\lambda(f) = 1$. By (2.4) this implies that $\theta(f) \in [1/(1+t), M/(M+t)]$ and hence $|\theta(f) - 1| = O(\varepsilon)$. A simple analysis then shows that an absolute error of $O(\varepsilon^2/M)$ in the computation of $\theta(f)$ results in a relative error of $\delta = O(\varepsilon)$ in the value of each p_m.

In practice, binary search on the interval $\theta \in [1/(1+t), M/(M+t)]$ can be used to compute $\theta(f)$ from (2.3) to the required accuracy in $O(\ln(M/\varepsilon))$ steps. This would require $O(\ln(M/\varepsilon))$ computations of the sum $\sum_{m=1}^{M} 1/(f_m - \theta)$ per coordination step of Algorithm $\mathcal{A}$, resulting in $O(M \ln(M\varepsilon))$ arithmetic operations per iteration (or $O(\ln M \ln(M/\varepsilon))$ parallel time on $M/\ln M$ processors).

This bound can be improved by using Newton's method to compute $\max_\theta \Phi(\theta, f)$. An analogous analysis for the min-max problem in [7] shows that the amortized number of all Newton iterations per iteration of Algorithm $\mathcal{A}$ is $O(\ln\ln(M/\varepsilon))$. Since the cost of each Newton iteration is of the same order as that for binary search, the coordination task can be implemented to run in $O(M \ln\ln(M/\varepsilon))$ arithmetic operations (or in $O(\ln M \ln\ln(M/\varepsilon))$ parallel time on $M/\ln M$ processors). We conclude that the sequential running time of each coordination step is roughly linear in M.

Acknowledgment. The authors gratefully acknowledge the valuable comments offered by the two anonymous referees.

REFERENCES

[1] D. S. Atkinson and P. M. Vaidya, *A cutting plane algorithm for convex programming that uses analytic centers. Nondifferentiable and large-scale optimization,* Math. Programming Ser. B, 69 (1995), pp. 1–43.

[2] N. Garg and J. Könemann, *Faster and simpler algorithms for multicommodity flow and other fractional packing problems*, in Proceedings of the 39th Annual Symposium on Foundations of Computer Science, IEEE, Piscataway, NJ, 1998, pp. 300–309.

[3] J. L. Goffin, A. Haurie, and J. P. Vial, *Decomposition and nondifferentiable optimization with the projective algorithm*, Management Sci., 38 (1992), pp. 284–302.

[4] J. L. Goffin, A. Haurie, J. P. Vial, and D. L. Zhu, *Using central prices in the decomposition of linear programs*, Eur. J. Oper. Res., 64 (1993), pp. 393–409.

[5] M. D. Grigoriadis and L. G. Khachiyan, *Fast approximation schemes for convex programs with many blocks and coupling constraints*, SIAM J. Optim., 4 (1994), pp. 86–107.

[6] M. D. Grigoriadis and L. G. Khachiyan, *A sublinear-time randomized approximation algorithm for matrix games*, Oper. Res. Lett., 18 (1995), pp. 53–58.

[7] M. D. Grigoriadis and L. G. Khachiyan, *Coordination complexity of parallel price-directive decomposition*, Math. Oper. Res., 21 (1996), pp. 321–340.

[8] M. D. Grigoriadis and L. G. Khachiyan, *Approximate minimum-cost multicommodity flows in $O(\varepsilon^{-2}NMK)$ time*, Math. Programming, 75 (1996), pp. 477–482.

[9] L. G. Khachiyan, *Optimal algorithms in convex programming, decomposition, and sorting*, in Computers and Decision Problems, Ju. Zhuravlev, ed., Nauka, Moscow, 1989, pp. 161–205 (in Russian).

[10] P. Klein and N. E. Young, *On the number of iterations of Dantzig-Wolfe optimization and packing-covering approximation algorithms*, in Proceedings of Integer Programing and Combinatorial Optimization, G. Cornuéjols, R. E. Burkard, and G. J. Woeginger, eds., Lecture Notes in Comput. Sci., Springer-Verlag, Berlin, 1999, pp. 320–327.

[11] Y. Nesterov and A. Nemirovskii, *Interior Point Polynomial Algorithms in Convex Programming*, SIAM, Philadelphia, 1994.

[12] 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.

[13] S. P. Tarasov, L. G. Khachiyan, and I. I. Erlich, *The method of inscribed ellipsoids*, Soviet Math. Doklady, 37 (1988), pp. 226–230.

[14] P. M. Vaidya, *A new algorithm for minimizing convex functions over convex sets*, Math. Programming Ser. A, 73 (1996), pp. 291–341.

[15] J. Villavicencio and M. D. Grigoriadis, *Approximate Lagrangian decomposition with a modified Karmarkar logarithmic potential*, in Network Optimization, P. Pardalos, D. W. Hearn, and W. W. Hager, eds., Lecture Notes in Econ. and Math. Systems 450, Springer-Verlag, Berlin, 1997, pp. 471–485.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1092–1118

A COMPUTATIONALLY EFFICIENT FEASIBLE SEQUENTIAL QUADRATIC PROGRAMMING ALGORITHM*

CRAIG T. LAWRENCE† AND ANDRÉ L. TITS‡

Abstract. A sequential quadratic programming (SQP) algorithm generating feasible iterates is described and analyzed. What distinguishes this algorithm from previous feasible SQP algorithms proposed by various authors is a reduction in the amount of computation required to generate a new iterate while the proposed scheme still enjoys the same global and fast local convergence properties. A preliminary implementation has been tested and some promising numerical results are reported.

Key words. sequential quadratic programming, SQP, feasible iterates, feasible SQP, FSQP

AMS subject classifications. 49M37, 65K05, 65K10, 90C30, 90C53

PII. S1052623498344562

1. Introduction.

Consider the inequality-constrained nonlinear programming problem

$$\text{(P)} \qquad \begin{array}{ll} \min & f(x) \\ \text{s.t.} & g_j(x) \le 0, \quad j = 1, \dots, m, \end{array}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \dots, m$, are continuously differentiable. Sequential quadratic programming (SQP) algorithms are widely acknowledged to be among the most successful algorithms available for solving (P). For an excellent recent survey of SQP algorithms, and the theory behind them, see [2].

Denote the feasible set for (P) by

$$X \stackrel{\Delta}{=} \{\ x \in \mathbb{R}^n \mid g_j(x) \le 0,\ j = 1, \dots, m\ \}.$$

In [19, 8, 16, 17, 1], variations on the standard SQP iteration for solving (P) are proposed which generate iterates lying within X. Such methods are sometimes referred to as "feasible SQP" (FSQP) algorithms. It was observed that requiring feasible iterates has both algorithmic and application-oriented advantages. Algorithmically, feasible iterates are desirable because

- the QP subproblems are always consistent, i.e., a feasible solution always exists, and
- the objective function may be used directly as a merit function in the line search.

In an engineering context, feasible iterates are important because

- often $f(x)$ is undefined outside of the feasible region X,
- trade-offs between design alternatives (all requiring that "hard" constraints be satisfied) may then be meaningfully explored, and
- the optimization process may be stopped after a few iterations, yielding a feasible point.

*Received by the editors September 14, 1998; accepted for publication (in revised form) January 11, 2001; published electronically May 16, 2001. This work was supported in part by the National Science Foundation under grant DMI-9813057.
http://www.siam.org/journals/siopt/11-4/34456.html
†Alphatech, Inc., Arlington, VA 22201 (craigl@dc.alphatech.com).
‡Department of Electrical and Computer Engineering and Institute for Systems Research, University of Maryland, College Park, MD 20742 (andre@eng.umd.edu).

The last feature is critical for real-time applications, where a feasible point may be required before the algorithm has had time to "converge" to a solution. On the flip side, it can be argued that requiring an initial feasible point for (P) may be taxing; in particular the objective function value may increase excessively in "phase I." It has been observed, however, that the "cost of feasibility" is typically small (see [17]).

An important function associated with the problem (P) is the *Lagrangian* L: $\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, which is defined by

$$L(x, \lambda) \triangleq f(x) + \sum_{i=1}^{m} \lambda^i g_i(x).$$

Given a feasible estimate x of the solution of (P) and a symmetric matrix H that approximates the Hessian of the Lagrangian $L(x, \lambda)$, where λ is a vector of nonnegative Lagrange multiplier estimates, the standard SQP search direction, denoted $d^0(x, H)$, or d^0 for short, solves the quadratic program (QP)

$$QP^0(x, H) \qquad \begin{aligned} \min \quad & \tfrac{1}{2}\langle d^0, Hd^0\rangle + \langle \nabla f(x), d^0\rangle \\ \text{s.t.} \quad & g_j(x) + \langle \nabla g_j(x), d^0\rangle \le 0, \quad j = 1, \ldots, m. \end{aligned}$$

Positive definiteness of H is often assumed as it ensures existence and uniqueness of such a solution. With appropriate merit function, line search procedure, Hessian approximation rule, and (if necessary) Maratos effect [15] avoidance scheme, the SQP iteration is known to be globally and locally superlinearly convergent (see, e.g., [2]).

A *feasible direction* at a point $x \in X$ is defined as any vector d in $\mathbb{R}^n$ such that $x + td$ belongs to X for all t in $[0, \bar{t}\,]$, for some positive $\bar{t}$. Note that the SQP direction d^0, a direction of descent for f, may not be a feasible direction at x, though it is at worst tangent to the active constraint surface. Thus, in order to generate feasible iterates in the SQP framework, it is necessary to "tilt" d^0 into the feasible set. A number of approaches has been considered in the literature for generating feasible directions and, specifically, tilting the SQP direction.

Early feasible direction algorithms (see, e.g., [29, 19]) were first-order methods, i.e., only first derivatives were used and no attempt was made to accumulate and use second-order information. Furthermore, search directions were often computed via linear programs instead of QPs. As a consequence, such algorithms converged linearly at best. Polak proposed several extensions to these algorithms (see [19], section 4.4) which took second-order information into account when computing the search direction. A few of the search directions proposed by Polak could be viewed as tilted SQP directions (with proper choice of the matrices encapsulating the second-order information in the defining equations). Even with second-order information, though, it is not possible to guarantee superlinear convergence of these algorithms because no mechanism was included for controlling the *amount* of tilting.

A straightforward way to tilt the SQP direction is, of course, to perturb the right-hand side of the constraints in $QP^0(x, H)$. Building on this observation, Herskovits and Carvalho [8] and Panier and Tits [16] independently developed similar FSQP algorithms in which the size of the perturbation was a function of the norm of $d^0(x, H)$ at the current feasible point x. Thus, their algorithms required the solution of $QP^0(x, H)$ in order to *define* the perturbed QP. Both algorithms were shown to be superlinearly convergent. On the other hand, as a by-product of the tilting scheme, global convergence proved to be more elusive. In fact, the algorithm in [8] is not globally convergent, while the algorithm in [16] has to resort to a first-order search

direction far from a solution in order to guarantee global convergence. Such a hybrid scheme could give slow convergence if a poor initial point is chosen.

The algorithm developed by Panier and Tits in [17], and analyzed under weaker assumptions by Qi and Wei in [22], has enjoyed a great deal of success in practice as implemented in the FFSQP/CFSQP [28, 13] software packages. We will refer to their algorithm throughout this paper as **FSQP**. In [17], instead of directly perturbing $QP^0(x, H)$, tilting is accomplished by replacing d^0 with the convex combination $(1-\rho)d^0+\rho d^1$, where d^1 is an (essentially) arbitrary feasible descent direction. To preserve the local convergence properties of the SQP iteration, ρ is selected as a function $\rho(d^0)$ of d^0 in such a way that d approaches d^0 fast enough (in particular, $\rho(d^0) = O(\|d^0\|^2)$) as the solution is approached. Finally, in order to avoid the Maratos effect and guarantee a superlinear rate of convergence, a second-order correction d^{C} (denoted $\tilde{d}$ in [17]) is used to "bend" the search direction. That is, an Armijo-type search is performed along the arc $x + td + t^2 d^{\mathrm{C}}$, where d is the tilted direction. In [17], the directions d^1 and d^{C} are both computed via QPs but it is pointed out that d^{C} could instead be taken as the solution of a linear least squares problem without affecting the asymptotic convergence properties.

From the point of view of computational cost, the main drawback of algorithm **FSQP** is the need to solve three QPs (or two QPs and a linear least squares problem) at each iteration. Clearly, for many problems it would be desirable to reduce the number of QPs at each iteration while preserving the generation of feasible iterates as well as the global and local convergence properties. This is especially critical in the context of those large-scale nonlinear programs for which the time spent solving the QPs dominates that used to evaluate the functions.

With that goal in mind, consider the following perturbation of $QP^0(x, H)$. Given a point x in X, a symmetric positive definite matrix H, and a nonnegative scalar η, let $(d(x, H, \eta), \gamma(x, H, \eta))$ solve the QP

$$QP(x, H, \eta) \qquad \begin{aligned} \min \quad & \tfrac{1}{2}\langle d, Hd\rangle + \gamma \\ \text{s.t.} \quad & \langle \nabla f(x), d\rangle \le \gamma, \\ & g_j(x) + \langle \nabla g_j(x), d\rangle \le \gamma \cdot \eta, \quad j = 1, \ldots, m, \end{aligned}$$

where γ is an additional, scalar variable.

The idea is that, away from KKT points of (P), $\gamma(x, H, \eta)$ will be negative and thus $d(x, H, \eta)$ will be a descent direction for f (due to the first constraint) as well as, if η is strictly positive, a feasible direction (due to the m other constraints). Note that when η is set to one the search direction is a special case of those computed in Polak's second-order feasible direction algorithms (again, see section 4.4 of [19]). Further, it is not difficult to show that when η is set to zero, we recover the SQP direction, i.e., $d(x, H, 0) = d^0(x, H)$. Large values of the parameter η, which we will call the *tilting parameter*, emphasize feasibility, while small values of η emphasize descent.

In [1], Birge, Qi, and Wei propose a feasible SQP algorithm based on $QP(x, H, \eta)$. Their motivation for introducing the right-hand side constraint perturbation and the tilting parameters (they use a vector of parameters, one for each constraint) is, like ours, to obtain a feasible search direction. Specifically, motivated by the high cost of function evaluations in the application problems they are targeting, their goal is to ensure that a full step of one is accepted in the line search as early on as is possible (so that costly line searches are avoided for most iterations). To this end, their tilting parameters start out positive and, if anything, increase when a step of one is not accepted. A side effect of such an updating scheme is that the algorithm cannot

achieve a superlinear rate of convergence, as the authors point out in Remark 5.1 of [1].

In the present paper, our goal is to compute a feasible descent direction which approaches the true SQP direction fast enough so as to ensure superlinear convergence. Furthermore, we would like to do this with as little computation per iteration as possible. While computationally rather expensive, algorithm **FSQP** of [17] has the convergence properties and practical performance we seek. We thus start by reviewing its key features. For x in X, define

$$I(x) \triangleq \{\ j \mid g_j(x) = 0\ \},$$

the index set of active constraints at x. In **FSQP**, in order for the line-search (with the objective function f used directly as the merit function) to be well defined, and in order to preserve global and fast local convergence, the sequence of search directions $\{d_k\}$ generated by algorithm **FSQP** is constructed so that the following properties hold:

P1. $d_k = 0$ if x_k is a KKT point for (P),
P2. $\langle \nabla f(x_k), d_k \rangle < 0$ if x_k is not a KKT point,
P3. $\langle \nabla g_j(x_k), d_k \rangle < 0$ for all $j \in I(x_k)$ if x_k is not a KKT point, and
P4. $d_k = d_k^0 + O(\|d_k^0\|^2)$.

We will show in section 3 that given any symmetric positive definite matrix H_k and nonnegative scalar η_k, $d(x_k, H_k, \eta_k)$ automatically satisfies P1 and P2. Furthermore, it satisfies P3 if η_k is strictly positive. Ensuring that P4 holds requires a bit more care.

In the algorithm proposed in this paper, at iteration k, the search direction is computed via solving $QP(x_k, H_k, \eta_k)$ and the tilting parameter η_k is iteratively adjusted to ensure that the four properties are satisfied. The resulting algorithm will be shown to be locally superlinearly convergent and globally convergent without resorting to a first-order direction far from the solution. Further, the generation of a new iterate requires only the solution of one QP and two closely related linear least squares problems. In contrast with the algorithm presented in [1], our tilting parameter starts out positive and asymptotically approaches zero.

There has been a great deal of interest recently in interior point algorithms for nonconvex nonlinear programming (see, e.g., [5, 6, 26, 4, 18, 25]). Such algorithms generate feasible iterates and typically require only the solution of linear systems of equations in order to generate new iterates. SQP-type algorithms, however, are often at an advantage over such methods in the context of applications where the number of variables is not too large but evaluations of objectives/constraint functions and of their gradients are highly time consuming. Indeed, because these algorithms use quadratic programs as successive models, away from a solution, progress between (expensive) function evaluations is often significantly better than that achieved by algorithms making use of mere linear systems of equations as models. On the other hand, for problems with large numbers of variables and inexpensive function evaluations, interior-point methods should be expected to perform more efficiently than SQP-type methods.

In section 2, we present the details of our new FSQP algorithm. In section 3, we show that under mild assumptions our iteration is globally convergent, as well as locally superlinearly convergent. The algorithm has been implemented and tested and we show in section 4 that the numerical results are quite promising. Finally, in section 5, we offer some concluding remarks and discuss some extensions to the

algorithm that are currently being explored.

Most of the ideas and results included in the present paper, in particular the algorithm of section 2, already appeared in [14].

2. Algorithm. We begin by making a few assumptions that will be in force throughout.

Assumption 1. The set X is nonempty.

Assumption 2. The functions $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \dots, m$, are continuously differentiable.

Assumption 3. For all $x \in X$ with $I(x) \neq \emptyset$, the set $\{\nabla g_j(x) \mid j \in I(x)\}$ is linearly independent.

A point $x^* \in \mathbb{R}^n$ is said to be a KKT point for the problem (P) if there exist scalars (*KKT multipliers*) $\lambda^{*,j}$, $j = 1, \dots, m$, such that

$$\left\{ \begin{array}{l} \nabla f(x^*) + \displaystyle\sum_{j=1}^{m} \lambda^{*,j} \nabla g_j(x^*) = 0, \\ g_j(x^*) \leq 0, \quad j = 1, \dots, m, \\ \lambda^{*,j} g_j(x^*) = 0 \text{ and } \lambda^{*,j} \geq 0, \quad j = 1, \dots, m. \end{array} \right. \tag{2.1}$$

It is well known that, under our assumptions, a necessary condition for optimality of a point $x^* \in X$ is that it be a KKT point.

Note that, with $x \in X$, $QP(x, H, \eta)$ is always consistent: $(0, 0)$ satisfies the constraints. Indeed, $QP(x, H, \eta)$ always has a unique solution (d, γ) (see Lemma 1 below) which, by convexity, is its unique KKT point; i.e., there exist multipliers μ and λ^j, $j = 1, \dots, m$, which, together with (d, γ), satisfy

$$\left\{ \begin{array}{l} \left[\begin{array}{c} Hd \\ 1 \end{array} \right] + \mu \left[\begin{array}{c} \nabla f(x) \\ -1 \end{array} \right] + \displaystyle\sum_{j=1}^{m} \lambda^j \left[\begin{array}{c} \nabla g_j(x) \\ -\eta \end{array} \right] = 0, \\ \langle \nabla f(x), d \rangle \leq \gamma, \\ g_j(x) + \langle \nabla g_j(x), d \rangle \leq \gamma \cdot \eta, \quad j = 1, \dots, m, \\ \mu \left(\langle \nabla f(x), d \rangle - \gamma \right) = 0 \text{ and } \mu \geq 0, \\ \lambda^j \left(g_j(x) + \langle \nabla g_j(x), d \rangle - \gamma \cdot \eta \right) = 0 \text{ and } \lambda^j \geq 0, \quad j = 1, \dots, m. \end{array} \right. \tag{2.2}$$

A simple consequence of the first equation in (2.2), which will be used throughout our analysis, is an affine relationship amongst the multipliers, namely

$$\mu + \eta \cdot \sum_{j=1}^{m} \lambda^j = 1. \tag{2.3}$$

Parameter η will be assigned a new value at each iteration, η_k at iteration k, to ensure that $d(x_k, H_k, \eta_k)$ has the necessary properties. Strict positivity of η_k is sufficient to guarantee that properties P1–P3 are satisfied. As it turns out, however, this is not enough to ensure that, away from a solution, there is adequate tilting into the feasible set. For this, we will force η_k to be bounded away from zero away from KKT points of (P). Finally, P4 requires that η_k tend to zero sufficiently fast as $d^0(x_k, H_k)$ tends to zero, i.e., as a solution is approached. In [16], a similar effect is achieved by first computing $d^0(x_k, H_k)$ but, of course, we want to avoid that here.

Given an estimate I_k^{E} of the active set $I(x_k)$, we can compute an estimate $d^{\mathrm{E}}(x_k, H_k, I_k^{\mathrm{E}})$ of $d^0(x_k, H_k)$ by solving the *equality*-constrained QP

$$LS^{\mathrm{E}}(x_k, H_k, I_k^{\mathrm{E}}) \qquad \begin{array}{ll} \min & \frac{1}{2}\langle d^{\mathrm{E}}, H_k d^{\mathrm{E}}\rangle + \langle \nabla f(x_k), d^{\mathrm{E}}\rangle \\ \text{s.t.} & g_j(x_k) + \langle \nabla g_j(x_k), d^{\mathrm{E}}\rangle = 0, \quad j \in I_k^{\mathrm{E}}, \end{array}$$

which is equivalent (after a change of variables) to solving a linear least squares problem. Let I_k be the set of active constraints, not including the "objective descent" constraint $\langle \nabla f(x_k), d_k\rangle \le \gamma_k$, for $QP(x_k, H_k, \eta_k)$, i.e.,

$$I_k \triangleq \{\ j \mid g_j(x_k) + \langle \nabla g_j(x_k), d_k\rangle = \gamma_k \cdot \eta_k\ \}.$$

We will show in section 3 that $d^{\mathrm{E}}(x_k, H_k, I_{k-1}) = d^0(x_k, H_k)$ for all k sufficiently large. Furthermore, we will prove that, when d_k is small, choosing

$$\eta_k \propto \|d^{\mathrm{E}}(x_k, H_k, I_{k-1})\|^2$$

is sufficient to guarantee global and local superlinear convergence. Proper choice of the proportionality constant (C_k in the algorithm statement below), while not important in the convergence analysis, is critical for satisfactory numerical performance. This will be discussed in section 4.

In [17], given x, H, and a feasible descent direction d, the Maratos correction d^{C} (denoted $\bar{d}$ in [17]) is taken as the solution of the QP

$$QP^{\mathrm{C}}(x, d, H) \qquad \begin{array}{ll} \min & \frac{1}{2}\langle d + d^{\mathrm{C}}, H(d + d^{\mathrm{C}})\rangle + \langle \nabla f(x), d + d^{\mathrm{C}}\rangle \\ \text{s.t.} & g_j(x + d) + \langle \nabla g_j(x), d + d^{\mathrm{C}}\rangle \le -\|d\|^{\tau}, \quad j = 1, \ldots, m, \end{array}$$

if it exists and has norm less than $\min\{\|d\|, C\}$, where τ is a given scalar satisfying $2 < \tau < 3$ and C a given large scalar. Otherwise, d^{C} is set to zero. (Indeed, a large d^{C} is meaningless and may jeopardize global convergence.) In section 1, it was mentioned that a linear least squares problem could be used instead of a QP to compute a version of the Maratos correction d^{C} with the same asymptotic convergence properties. Given that our goal is to reduce the computational cost per iteration, it makes sense to use such an approach here. Thus, at iteration k, we take the correction d_k^{C} to be the solution $d^{\mathrm{C}}(x_k, d_k, H_k, I_k)$, if it exists and is not too large (specifically, if its norm is no larger than that of d_k), of the equality-constrained QP (equivalent to a least squares problem after a change of variables)

$$LS^{\mathrm{C}}(x_k, d_k, H_k, I_k) \qquad \begin{array}{ll} \min & \langle d_k + d^{\mathrm{C}}, H_k(d_k + d^{\mathrm{C}})\rangle + \langle \nabla f(x_k), d_k + d^{\mathrm{C}}\rangle \\ \text{s.t.} & g_j(x_k + d_k) + \langle \nabla g_j(x_k), d^{\mathrm{C}}\rangle = -\|d_k\|^{\tau} \qquad \forall j \in I_k, \end{array}$$

where $\tau \in (2, 3)$, a direct extension of an alternative considered in [16]. In making use of the best available metric, such an objective, as compared to the pure least squares objective $\|d^{\mathrm{C}}\|^2$, should yield a somewhat better iterate without significantly increasing computational requirements (or affecting the convergence analysis). Another advantage of using metric H_k is that, asymptotically, the matrix underlying $LS^{\mathrm{C}}(x_k, d_k, H_k, I_k)$ will be the same as that underlying $LS^{\mathrm{E}}(x_k, H_k, I_{k-1})$, resulting in computational savings. In the case that $LS^{\mathrm{C}}(x_k, d_k, H_k, I_k)$ is inconsistent, or the computed solution d_k^{C} is too large, we will simply set d_k^{C} to zero.

The proposed algorithm is as follows. Parameters α, β are used in the Armijo-like search, τ is the "bending" exponent in LS^{C}, and ϵ_ℓ, $\underline{C}$, $\overline{C}$, and $\overline{D}$ are used in the

update rule for η_k. The algorithm is dubbed **RFSQP**, where "R" reflects the reduced amount of work per iteration.

ALGORITHM **RFSQP**.

Parameters: $\alpha \in (0, \frac{1}{2})$, $\beta \in (0,1)$, $\tau \in (2,3)$, $\epsilon_\ell > 0$, $0 < \underline{C} \leq \overline{C}$, $\bar{D} > 0$.
Data: $x_0 \in X$, H_0 positive definite, $\eta_0 > 0$.
Step 0 - *Initialization.* **set** $k \leftarrow 0$.
Step 1 - *Computation of search arc.*
(i) **compute** $(d_k, \gamma_k) = (d(x_k, H_k, \eta_k), \gamma(x_k, H_k, \eta_k))$, the active set I_k, and associated multipliers $\mu_k \in \mathbb{R}$, $\lambda_k \in \mathbb{R}^m$.
if $(d_k = 0)$ **then stop**.
(ii) **compute** $d_k^{\mathrm{C}} = d^{\mathrm{C}}(x_k, d_k, H_k, I_k)$ if it exists and satisfies $\|d_k^{\mathrm{C}}\| \leq \|d_k\|$. Otherwise, **set** $d_k^{\mathrm{C}} = 0$.
Step 2 - *Arc search.* **compute** t_k, the first value of t in the sequence $\{1, \beta, \beta^2, \dots\}$ that satisfies

$$f(x_k + td_k + t^2 d_k^{\mathrm{C}}) \leq f(x_k) + \alpha t \langle \nabla f(x_k), d_k \rangle,$$
$$g_j(x_k + td_k + t^2 d_k^{\mathrm{C}}) \leq 0, \quad j = 1, \dots, m.$$

Step 3 - *Updates.*
(i) **set** $x_{k+1} \leftarrow x_k + t_k d_k + t_k^2 d_k^{\mathrm{C}}$.
(ii) **compute** H_{k+1}, a new symmetric positive definite estimate to the Hessian of the Lagrangian.
(iii) **select** $C_{k+1} \in [\underline{C}, \overline{C}]$.
∗ **if** $(\|d_k\| < \epsilon_\ell)$ **then** if $LS^{\mathrm{E}}(x_{k+1}, H_{k+1}, I_k)$ has a unique solution and unique associated multipiers, **compute** $d_{k+1}^{\mathrm{E}} = d^{\mathrm{E}}(x_{k+1}, H_{k+1}, I_k)$, and the associated multipliers $\lambda_{k+1}^{\mathrm{E}} \in \mathbb{R}^{|I_k|}$. In such case,
· **if** $(\|d_{k+1}^{\mathrm{E}}\| \leq \bar{D}$ and $\lambda_{k+1}^{\mathrm{E}} \geq 0)$ **then set**

$$\eta_{k+1} \leftarrow C_{k+1} \cdot \|d_{k+1}^{\mathrm{E}}\|^2.$$

· **else set** $\eta_{k+1} \leftarrow C_{k+1} \cdot \|d_k\|^2$.
∗ **else set** $\eta_{k+1} \leftarrow C_{k+1} \cdot \epsilon_\ell^2$.
(iv) **set** $k \leftarrow k+1$ and **go to** *Step* 1.

3. Convergence analysis. Much of our analysis, especially the local analysis, will be devoted to establishing the relationship between $d(x, H, \eta)$ and the SQP direction $d^0(x, H)$. Given x in X and H symmetric positive definite, d^0 is a KKT point for $QP^0(x, H)$ (thus its unique solution $d^0(x, H)$) if and only if there exists a multiplier vector λ^0 such that

$$\text{(3.1)} \qquad \begin{cases} Hd^0 + \nabla f(x) + \sum_{j=1}^{m} \lambda^{0,j} \nabla g_j(x) = 0, \\ g_j(x) + \langle \nabla g_j(x), d^0 \rangle \leq 0, \quad j = 1, \dots, m, \\ \lambda^{0,j} \cdot \big(g_j(x) + \langle \nabla g_j(x), d^0 \rangle\big) = 0 \text{ and } \lambda^{0,j} \geq 0, \quad j = 1, \dots, m. \end{cases}$$

Further, given $I \subseteq \{1, \dots, m\}$, an estimate d^{E} is a KKT point for $LS^{\mathrm{E}}(x, H, I)$ (thus its unique solution $d^{\mathrm{E}}(x, H, I)$) if and only if there exists a multiplier vector λ^{E} such

that

$$\begin{cases} Hd^{\mathrm{E}} + \nabla f(x) + \sum_{j \in I} \lambda^{\mathrm{E},j} \nabla g_j(x) = 0, \\ g_j(x) + \langle \nabla g_j(x), d^{\mathrm{E}} \rangle = 0, \quad j \in I. \end{cases} \tag{3.2}$$

Note that the components of λ^{E} for $j \notin I$ play no role in the optimality conditions.

3.1. Global convergence. In this section we establish that, under mild assumptions, **RFSQP** generates a sequence of iterates $\{x_k\}$ with the property that all accumulation points are KKT points for (P). We begin by establishing some properties of the tilted SQP search direction $d(x, H, \eta)$.

LEMMA 1. *Suppose Assumptions* 1–3 *hold. Then, given H symmetric positive definite, $x \in X$, and $\eta \geq 0$, $d(x, H, \eta)$ is well defined and $(d(x, H, \eta), \gamma(x, H, \eta))$ is the unique KKT point of $QP(x, H, \eta)$. Furthermore, $d(x, H, \eta)$ is bounded over compact subsets of $X \times \mathcal{P} \times \mathbb{R}^+$, where $\mathcal{P}$ is the set of symmetric positive definite $n \times n$ matrices and $\mathbb{R}^+$ the set of nonnegative real numbers.*

Proof. First note that the feasible set for $QP(x, H, \eta)$ is nonempty, since $(d, \gamma) = (0, 0)$ is always feasible. Now consider the cases $\eta = 0$ and $\eta > 0$ separately. From (2.2) and (3.1), it is clear that, if $\eta = 0$, (d, γ) is a solution to $QP(x, H, 0)$ if and only if d is a solution of $QP^0(x, H)$ and $\gamma = \langle \nabla f(x), d \rangle$. It is well known that, under our assumptions, $d^0(x, H)$ is well defined, unique, and continuous. The claims follow. Suppose now that $\eta > 0$. In that case, (d, γ) is a solution of $QP(x, H, \eta)$ if and only if d solves the unconstrained problem

$$\min \frac{1}{2}\langle d, Hd \rangle + \max \left\{ \langle \nabla f(x), d \rangle, \ \frac{1}{\eta} \cdot \max_{j=1,\dots,m} \{ g_j(x) + \langle \nabla g_j(x), d \rangle \} \right\} \tag{3.3}$$

and

$$\gamma = \max \left\{ \langle \nabla f(x), d \rangle, \ \frac{1}{\eta} \cdot \max_{j=1,\dots,m} \{ g_j(x) + \langle \nabla g_j(x), d \rangle \} \right\}.$$

Since the function being minimized in (3.3) is strictly convex and radially unbounded, it follows that $(d(x, H, \eta), \gamma(x, H, \eta))$ is well defined and unique as a global minimizer for the convex problem $QP(x, H, \eta)$ and thus unique as a KKT point for that problem. Boundedness of $d(x, H, \eta)$ over compact subsets of $X \times \mathcal{P} \times \mathbb{R}^+$ follows from the first equation in (2.2), our regularity assumptions, and (2.3), which shows (since $\eta > 0$) that the multipliers are bounded. ∎

LEMMA 2. *Suppose Assumptions* 1–3 *hold. Then, given H symmetric positive definite and $\eta \geq 0$,*

(i) *$\gamma(x, H, \eta) \leq 0$ for all $x \in X$, and moreover $\gamma(x, H, \eta) = 0$ if and only if $d(x, H, \eta) = 0$;*

(ii) *$d(x, H, \eta) = 0$ if and only if x is a KKT point for* (P)*, and moreover, if either (thus both) of these conditions holds, then the multipliers λ and μ for $QP(x, H, \eta)$ and λ^* for* (P) *are related by $\mu = (1 + \eta \sum_j \lambda^{*,j})^{-1}$ and $\lambda = \mu \lambda^*$.*

Proof. To prove (i), first note that since $(d, \gamma) = (0, 0)$ is always feasible for $QP(x, H, \eta)$, the optimal value of the QP is nonpositive. Further, since $H > 0$, the quadratic term in the objective is nonnegative, which implies $\gamma(x, H, \eta) \leq 0$. Now suppose that $d(x, H, \eta) = 0$; then feasibility of the first QP constraint implies that $\gamma(x, H, \eta) = 0$. Finally, suppose that $\gamma(x, H, \eta) = 0$. Since $x \in X$, $H > 0$, and $\eta \geq 0$,

it is clear that $d = 0$ is feasible and achieves the minimum value of the objective. Thus, uniqueness gives $d(x, H, \eta) = 0$ and part (i) is proved.

Suppose now that $d(x, H, \eta) = 0$. Then $\gamma(x, H, \eta) = 0$ and by (2.2) there exist a multiplier vector λ and a scalar multiplier $\mu \geq 0$ such that

$$\begin{cases} \mu \nabla f(x) + \sum_{j=1}^{m} \lambda^j \nabla g_j(x) = 0, \\ g_j(x) \leq 0 \quad \forall j = 1, \ldots, m, \\ \lambda^j g_j(x) = 0 \text{ and } \lambda^j \geq 0 \quad \forall j = 1, \ldots, m. \end{cases} \tag{3.4}$$

We begin by showing that $\mu > 0$. Proceeding by contradiction, suppose $\mu = 0$; then by (2.3) we have

$$\sum_{j=1}^{m} \lambda^j > 0. \tag{3.5}$$

Note that

$$\begin{aligned} I &\triangleq \{\, j \mid g_j(x) + \langle \nabla g_j(x), d(x, H, \eta) \rangle = \gamma(x, H, \eta) \cdot \eta \,\} \\ &= \{\, j \mid g_j(x) = 0 \,\} = I(x). \end{aligned}$$

Thus, by the complementary slackness condition of (2.2) and the optimality conditions (3.4),

$$0 = \sum_{j=1}^{m} \lambda^j \nabla g_j(x) = \sum_{j \in I(x)} \lambda^j \nabla g_j(x).$$

By Assumption 3, this sum vanishes only if $\lambda^j = 0$ for all $j \in I(x)$, contradicting (3.5). Thus $\mu > 0$. It is now immediate that x is a KKT point for (P) with multipliers $\lambda^{*,j} = \lambda^j / \mu$, $j = 1, \ldots, m$.

Finally, to prove the necessity portion of part (ii) note that if x is a KKT point for (P), then (2.1) shows that $(d, \gamma) = (0, 0)$ is a KKT point for $QP(x, H, \eta)$, with $\mu = (1 + \eta \sum_j \lambda^{*,j})^{-1}$ and $\lambda^j = \lambda^{*,j}(1 + \eta \sum_j \lambda^{*,j})^{-1}$, $j = 1, \ldots, m$. Uniqueness of such points (Lemma 1) yields the result. ∎

The next two lemmas establish that the line search in Step 2 of Algorithm **RFSQP** is well defined.

LEMMA 3. *Suppose Assumptions* 1–3 *hold. Suppose $x \in X$ is not a KKT point for* (P)*, H is symmetric positive definite, and $\eta > 0$. Then*

(i) $\langle \nabla f(x), d(x, H, \eta) \rangle < 0$, *and*

(ii) $\langle \nabla g_j(x), d(x, H, \eta) \rangle < 0$ *for all* $j \in I(x)$.

Proof. Both follow immediately from Lemma 2 and the fact that $d(x, H, \eta)$ and $\gamma(x, H, \eta)$ must satisfy the constraints in $QP(x, H, \eta)$. ∎

LEMMA 4. *Suppose Assumptions* 1–3 *hold. Then, if $\eta_k = 0$, x_k is a KKT point for* (P) *and the algorithm will stop in Step* 1(i) *at iteration k. On the other hand, whenever the algorithm does not stop in Step* 1(i)*, the line search is well defined; i.e., Step* 2 *yields a step t_k equal to β^{j_k} for some finite j_k.*

Proof. Suppose that $\eta_k = 0$. Then $k > 0$ and, by Step 3(iii), either $d_k^{\mathrm{E}} = 0$ with $\lambda_k^{\mathrm{E}} \geq 0$, or $d_{k-1} = 0$. The latter case cannot hold, as the stopping criterion in Step

1(i) would have stopped the algorithm at iteration $k-1$. On the other hand, if $d_k^{\mathrm{E}} = 0$ with $\lambda_k^{\mathrm{E}} \geq 0$, then in view of the optimality conditions (3.2), and the fact that x_k is always feasible for (P), we see that x_k is a KKT point for (P) with multipliers

$$\begin{cases} \lambda_k^{\mathrm{E},j}, & j \in I_{k-1}, \\ 0 & \text{otherwise.} \end{cases}$$

Thus, by Lemma 2, $d_k = 0$ and the algorithm will stop in Step 1(i). The first claim is thus proved. Also, we have established that $\eta_k > 0$ whenever Step 2 is reached. The second claim now follows immediately from Lemma 3 and Assumption 2. ∎

The previous lemmas imply that the algorithm is well defined. In addition, Lemma 2 shows that if Algorithm **RFSQP** generates a finite sequence terminating at the point x_N, then x_N is a KKT point for the problem (P). We now concentrate on the case in which an infinite sequence $\{x_k\}$ is generated, i.e., the algorithm never satisfies the termination condition in Step 1(i). Note that, in view of Lemma 4, we may assume throughout that

$$\eta_k > 0 \quad \forall k. \tag{3.6}$$

Before proceeding, we make an assumption concerning the estimates H_k of the Hessian of the Lagrangian.

Assumption 4. There exist positive constants σ_1 and σ_2 such that, for all k,

$$\sigma_1 \|d\|^2 \leq \langle d, H_k d \rangle \leq \sigma_2 \|d\|^2 \quad \forall d \in \mathbb{R}^n.$$

LEMMA 5. *Suppose Assumptions* 1–4 *hold. Then the sequence* $\{\eta_k\}$ *generated by Algorithm* **RFSQP** *is bounded. Further, the sequence* $\{d_k\}$ *is bounded on subsequences on which* $\{x_k\}$ *is bounded.*

Proof. The first claim follows from the update rule in Step 3(iii) of Algorithm **RFSQP**. The second claim then follows from Lemma 1 and Assumption 4. ∎

Given an infinite index set $\mathcal{K}$, we will use the notation

$$x_k \xrightarrow{k \in \mathcal{K}} x^*$$

to mean

$$x_k \to x^* \quad \text{as} \quad k \to \infty, \ k \in \mathcal{K}.$$

LEMMA 6. *Suppose Assumptions* 1–3 *hold. Suppose* $\mathcal{K}$ *is an infinite index set such that* $x_k \xrightarrow{k \in \mathcal{K}} x^* \in X$, $\{\eta_k\}$ *is bounded on* $\mathcal{K}$, *and* $d_k \xrightarrow{k \in \mathcal{K}} 0$. *Then* $I_k \subseteq I(x^*)$, *for all* $k \in \mathcal{K}$, k *sufficiently large, and the QP multiplier sequences* $\{\mu_k\}$ *and* $\{\lambda_k\}$ *are bounded on* $\mathcal{K}$. *Further, given any accumulation point* $\eta^* \geq 0$ *of* $\{\eta_k\}_{k \in \mathcal{K}}$, $(0,0)$ *is the unique solution of* $QP(x^*, H^*, \eta^*)$.

Proof. In view of Assumption 2 $\{\nabla f(x_k)\}_{k \in \mathcal{K}}$ must be bounded. Lemma 2(i) and the first constraint in $QP(x_k, H_k, \eta_k)$ give

$$\langle \nabla f(x_k), d_k \rangle \leq \gamma_k \leq 0 \quad \forall k \in \mathcal{K}.$$

Thus, $\gamma_k \xrightarrow{k \in \mathcal{K}} 0$. To prove the first claim, let $j' \notin I(x^*)$. There exists $\delta_{j'} > 0$ such that $g_{j'}(x_k) \leq -\delta_{j'} < 0$, for all $k \in \mathcal{K}$, k sufficiently large. In view of Assumption 2, and since $d_k \xrightarrow{k \in \mathcal{K}} 0$, $\gamma_k \xrightarrow{k \in \mathcal{K}} 0$, and $\{\eta_k\}$ is bounded on $\mathcal{K}$, it is clear that

$$g_{j'}(x_k) + \langle \nabla g_{j'}(x_k), d_k \rangle - \gamma_k \cdot \eta_k \leq -\frac{\delta_{j'}}{2} < 0,$$

i.e., $j' \notin I_k$ for all $k \in \mathcal{K}$, k sufficiently large, proving the first claim.

Boundedness of $\{\mu_k\}_{k\in\mathcal{K}}$ follows from nonnegativity and (2.3). To prove that of $\{\lambda_k\}_{k\in\mathcal{K}}$, using complementary slackness and the first equation in (2.2), write

$$H_k d_k + \mu_k \nabla f(x_k) + \sum_{j\in I(x^*)} \lambda_k^j \nabla g_j(x_k) = 0. \tag{3.7}$$

Proceeding by contradiction, suppose that $\{\lambda_k\}_{k\in\mathcal{K}}$ is unbounded. Without loss of generality, assume that $\|\lambda_k\|_\infty > 0$ for all $k \in \mathcal{K}$ and define for all $k \in \mathcal{K}$

$$\nu_k^j \triangleq \frac{\lambda_k^j}{\|\lambda_k\|_\infty} \in [0,1].$$

Note that, for all $k \in \mathcal{K}$, $\|\nu_k\|_\infty = 1$. Dividing (3.7) by $\|\lambda_k\|_\infty$ and taking limits on an appropriate subsequence of $\mathcal{K}$, it follows from Assumptions 2 and 4 and boundedness of $\{\mu_k\}$ that

$$\sum_{j\in I(x^*)} \nu^{*,j} \nabla g_j(x^*) = 0$$

for some $\nu^{*,j}$, $j \in I(x^*)$, where $\|\nu^*\|_\infty = 1$. As this contradicts Assumption 3, it is established that $\{\lambda_k\}_{k\in\mathcal{K}}$ is bounded.

To complete the proof, let $\mathcal{K}' \subseteq \mathcal{K}$ be an infinite index set such that $\eta_k \overset{k\in\mathcal{K}'}{\longrightarrow} \eta^*$ and assume without loss of generality that $H_k \overset{k\in\mathcal{K}'}{\longrightarrow} H^*$, $\mu_k \overset{k\in\mathcal{K}'}{\longrightarrow} \mu^*$, and $\lambda_k \overset{k\in\mathcal{K}'}{\longrightarrow} \lambda^*$. Taking limits in the optimality conditions (2.2) shows that, indeed, $(d,\gamma) = (0,0)$ is a KKT point for $QP(x^*, H^*, \eta^*)$ with multipliers μ^* and λ^*. Finally, uniqueness of such points (Lemma 1) proves the result. ∎

LEMMA 7. *Suppose Assumptions* 1–4 *hold. Then, if* $\mathcal{K}$ *is an infinite index set such that* $d_k \overset{k\in\mathcal{K}}{\longrightarrow} 0$*, all accumulation points of* $\{x_k\}_{k\in\mathcal{K}}$ *are KKT points for* (P).

Proof. Suppose that $\mathcal{K}' \subseteq \mathcal{K}$ is an infinite index set on which $x_k \overset{k\in\mathcal{K}'}{\longrightarrow} x^* \in X$. In view of Assumption 4 and Lemma 5, assume without loss of generality that $H_k \overset{k\in\mathcal{K}'}{\longrightarrow} H^*$, a positive definite matrix, and $\eta_k \overset{k\in\mathcal{K}'}{\longrightarrow} \eta^* \geq 0$. In view of Lemma 6, $(0,0)$ is the unique solution of $QP(x^*, H^*, \eta^*)$. It follows from Lemma 2 that x^* is a KKT point for (P). ∎

We now state and prove the main result of this subsection.

THEOREM 1. *Under Assumptions* 1–4*, Algorithm* **RFSQP** *generates a sequence* $\{x_k\}$ *for which all accumulation points are KKT points for* (P).

Proof. Suppose $\mathcal{K}$ is an infinite index set such that $x_k \overset{k\in\mathcal{K}}{\longrightarrow} x^*$. In view of Lemma 5 and Assumption 4, we may assume without loss of generality that $d_k \overset{k\in\mathcal{K}}{\longrightarrow} d^*$, $\eta_k \overset{k\in\mathcal{K}}{\longrightarrow} \eta^* \geq 0$, and $H_k \overset{k\in\mathcal{K}}{\longrightarrow} H^* > 0$. The cases $\eta^* = 0$ and $\eta^* > 0$ are considered separately.

Suppose first that $\eta^* = 0$. Then, by Step 3(iii), there exists an infinite index set $\mathcal{K}' \subseteq \mathcal{K}$ such that either $d_k^{\mathrm{E}} \overset{k\in\mathcal{K}'}{\longrightarrow} 0$ with $\lambda_k^{\mathrm{E}} \geq 0$, for all $k \in \mathcal{K}'$, or $d_{k-1} \overset{k\in\mathcal{K}'}{\longrightarrow} 0$. If the latter case holds, it is then clear that $x_{k-1} \overset{k\in\mathcal{K}'}{\longrightarrow} x^*$, since $\|x_k - x_{k-1}\| \leq 2\|d_{k-1}\| \overset{k\in\mathcal{K}'}{\longrightarrow} 0$. Thus, by Lemma 7, x^* is a KKT point for (P). Now suppose instead that $d_k^{\mathrm{E}} \overset{k\in\mathcal{K}'}{\longrightarrow} 0$ with $\lambda_k^{\mathrm{E}} \geq 0$ for all $k \in \mathcal{K}'$. From the second set of equations in (3.2), one can easily see that $I_{k-1} \subseteq I(x^*)$ for all $k \in \mathcal{K}'$, k sufficiently large, and using an argument very similar to that used in Lemma 6, one can show that $\{\lambda_k^{\mathrm{E}}\}_{k\in\mathcal{K}'}$ is a bounded sequence.

Thus, taking limits in (3.2) on an appropriate subsequence of $\mathcal{K}'$ shows that x^* is a KKT point for (P).

Now consider the case $\eta^* > 0$. We show that $d_k \overset{k\in\mathcal{K}}{\longrightarrow} 0$. Proceeding by contradiction, without loss of generality suppose there exists $\underline{d} > 0$ such that $\|d_k\| \geq \underline{d}$ for all $k \in \mathcal{K}$. From nonpositivity of the optimal value of the objective function in $QP(x_k, H_k, \eta_k)$ (since $(0, 0)$ is always feasible) and Assumption 4, we see that

$$\gamma_k \leq -\frac{1}{2}\sigma_1 \underline{d}^2 < 0 \quad \forall k \in \mathcal{K}.$$

Further, in view of (3.6) and since $\eta^* > 0$, there exists $\underline{\eta} > 0$ such that

$$\eta_k > \underline{\eta} \quad \forall k \in \mathcal{K}.$$

From the constraints of $QP(x_k, H_k, \eta_k)$, it follows that

$$\langle \nabla f(x_k), d_k \rangle \leq -\frac{1}{2}\sigma_1 \underline{d}^2 < 0 \quad \forall k \in \mathcal{K}$$

and

$$g_j(x_k) + \langle \nabla g_j(x_k), d_k \rangle \leq -\frac{1}{2}\sigma_1 \underline{d}^2 \underline{\eta} < 0 \quad \forall k \in \mathcal{K},$$

$j = 1, \ldots, m$. Hence, using Assumption 2, it is easily shown that there exists $\delta > 0$ such that for all $k \in \mathcal{K}$, k large enough,

$$\begin{aligned} \langle \nabla f(x_k), d_k \rangle &\leq -\delta, \\ \langle \nabla g_j(x_k), d_k \rangle &\leq -\delta \quad \forall j \in I(x^*) \\ g_j(x_k) &\leq -\delta \quad \forall j \in \{1, \ldots, m\} \setminus I(x^*). \end{aligned}$$

The rest of the contradiction argument establishing $d_k \overset{k\in\mathcal{K}}{\longrightarrow} 0$ follows exactly the proof of Proposition 3.2 in [16]. Finally, it then follows from Lemma 7 that x^* is a KKT point for (P). ∎

3.2. Local convergence. While the details are often quite different, overall the analysis in this section is inspired by and occasionally follows that of Panier and Tits in [16, 17]. The key result is Proposition 1 which states that, under appropriate assumptions, the arc search eventually accepts the full step of one. With this and the fact, to be established along the way, that tilted direction d_k approaches the standard SQP direction sufficiently fast, superlinear convergence follows from a classical analysis given by Powell [20, sections 2–3]. As a first step, we strengthen the regularity assumptions.

Assumption 2′. The functions $f : \mathbb{R}^n \to \mathbb{R}$ and $g_j : \mathbb{R}^n \to \mathbb{R}$, $j = 1, \ldots, m$, are three times continuously differentiable.

A point x^* is said to satisfy the *second-order sufficiency conditions with strict complementary slackness* for (P) if there exists a multiplier vector $\lambda^* \in \mathbb{R}^m$ such that

- the pair (x^*, λ^*) satisfies (2.1), i.e., x^* is a KKT point for (P),
- $\nabla^2_{xx} L(x^*, \lambda^*)$ is positive definite on the subspace

$$\{h \mid \langle \nabla g_j(x^*), h \rangle = 0 \ \forall j \in I(x^*)\},$$

- and $\lambda^{*,j} > 0$ for all $j \in I(x^*)$ (strict complementary slackness).

In order to guarantee that the entire sequence $\{x_k\}$ converges to a KKT point x^*, we make the following assumption. (Recall that we have already established, under weaker assumptions, that *every* accumulation point of $\{x_k\}$ is a KKT point for (P).)

Assumption 5. The sequence $\{x_k\}$ has an accumulation point x^* which satisfies the second-order sufficiency conditions with strict complementary slackness.

It is well known that Assumption 5 guarantees that the entire sequence converges. For a proof see, e.g., Proposition 4.1 in [16].

LEMMA 8. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then the entire sequence generated by Algorithm* **RFSQP** *converges to a point* x^* *satisfying the second-order sufficiency conditions with strict complementary slackness.*

From this point forward, λ^* will denote the (unique) multiplier vector associated with KKT point x^* for (P). It is readily checked that, for any symmetric positive definite H, $(0, \lambda^*)$ is the KKT pair for $QP^0(x^*, H)$.

As announced, as a first main step, we show that our sequence of tilted SQP directions approaches the true SQP direction sufficiently fast. (This is achieved in Lemmas 9–18.) In order to do so, define d_k^0 to be equal to $d^0(x_k, H_k)$, where x_k and H_k are as computed by Algorithm **RFSQP**. Further, for each k, define λ_k^0 as a multiplier vector such that (d_k^0, λ_k^0) satisfy (3.1) and let $I_k^0 \triangleq \{\ j \mid g_j(x_k) + \langle \nabla g_j(x_k), d_k^0 \rangle = 0\ \}$. The following lemma is proved in [17] (with reference to [16]) under identical assumptions.

LEMMA 9. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then*

(i) $d_k^0 \to 0$,

(ii) $\lambda_k^0 \to \lambda^*$,

(iii) *for all* k *sufficiently large, the following two equalities hold:*

$$I_k^0 = \{\ j \mid \lambda_k^{0,j} > 0\ \} = I(x^*).$$

We next establish that the entire tilted SQP direction sequence converges to 0. In order to do so, we establish that $d(x, H, \eta)$ is continuous in a neighborhood of (x^*, H^*, η^*), for any $\eta^* \geq 0$ and H^* symmetric positive definite. Complicating the analysis is the fact that we have yet to establish that the sequence $\{\eta_k\}$ does, in fact, converge. Given $\eta^* \geq 0$, define the set

$$N^*(\eta^*) \triangleq \left\{ \begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix}, \begin{pmatrix} \nabla g_j(x^*) \\ -\eta^* \end{pmatrix}, j \in I(x^*) \right\}.$$

LEMMA 10. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then, given any* $\eta^* \geq 0$, *the set* $N^*(\eta^*)$ *is linearly independent.*

Proof. Let H^* be symmetric positive definite. Note that, in view of Lemma 2, $d(x^*, H^*, \eta^*) = 0$. Now suppose the claim does not hold; i.e., suppose there exist scalars λ^j, $j \in \{0\} \cup I(x^*)$, not all zero, such that

$$\lambda^0 \begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix} + \sum_{j \in I(x^*)} \lambda^j \begin{pmatrix} \nabla g_j(x^*) \\ -\eta^* \end{pmatrix} = 0. \tag{3.8}$$

In view of Assumption 3, $\lambda^0 \neq 0$ and the scalars λ^j are unique modulo a scaling factor. This uniqueness, the fact that $d(x^*, H^*, \eta^*) = 0$, and the first n scalar equations in the optimality conditions (2.2) imply that $\mu^* = 1$ and

$$\lambda^{*,j} = \begin{cases} \dfrac{\lambda^j}{\lambda^0}, & j \in I(x^*), \\ 0 & \text{else}, \end{cases}$$

$j = 1, \ldots, m$, are KKT multipliers for $QP(x^*, H^*, \eta^*)$. Thus, in view of (2.3),

$$\eta^* \cdot \sum_{j \in I(x^*)} \frac{\lambda^j}{\lambda^0} = 0.$$

But this contradicts (3.8), which gives

$$\eta^* \cdot \sum_{j \in I(x^*)} \frac{\lambda^j}{\lambda^0} = -1;$$

hence $N^*(\eta^*)$ is linearly independent. □

LEMMA 11. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Let* $\eta^* \geq 0$ *be an accumulation point of* $\{\eta_k\}$. *Then, given any symmetric positive definite* H, $(d^*, \gamma^*) = (0, 0)$ *is the unique solution of* $QP(x^*, H, \eta^*)$ *and the second-order sufficiency conditions hold, with strict complementary slackness.*

Proof. In view of Lemma 2, $QP(x^*, H, \eta^*)$ has $(d^*, \gamma^*) = (0, 0)$ as its unique solution. Define the Lagrangian function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}$ for $QP(x^*, H, \eta^*)$ as

$$\begin{aligned} \mathcal{L}(d, \gamma, \mu, \lambda) \quad = \quad & \frac{1}{2}\langle d, Hd\rangle + \gamma + \mu\left(\langle \nabla f(x^*), d\rangle - \gamma\right) \\ & + \sum_{j=1}^{m} \lambda^j \left(g_j(x^*) + \langle \nabla g_j(x^*), d\rangle - \gamma\eta^*\right). \end{aligned}$$

Suppose $\hat{\mu} \in \mathbb{R}$ and $\hat{\lambda} \in \mathbb{R}^m$ are KKT multipliers such that (2.2) holds with $d = 0$, $\gamma = 0$, $\mu = \hat{\mu}$, and $\lambda = \hat{\lambda}$. Let $j = 0$ be the index for the first constraint in $QP(x^*, H, \eta^*)$, i.e., $\langle \nabla f(x^*), d\rangle \leq \gamma$. Note that since $(d^*, \gamma^*) = (0, 0)$, the active constraint index set I^* for $QP(x^*, H, \eta^*)$ is equal to $I(x^*) \cup \{0\}$. (Note that we define I^* as including 0, while I_k was defined as a subset of $\{1, \ldots, m\}$.) Thus the set of active constraint gradients for $QP(x^*, H, \eta^*)$ is $N^*(\eta^*)$.

Now consider the Hessian of the Lagrangian for $QP(x^*, H, \eta^*)$, i.e., the second derivative with respect to the first two variables (d, γ),

$$\nabla^2 \mathcal{L}(0, 0, \hat{\lambda}, \hat{\mu}) = \begin{bmatrix} H & 0 \\ 0 & 0 \end{bmatrix},$$

and given an arbitrary $h \in \mathbb{R}^{n+1}$, decompose it as $h = (y^T, \alpha)^T$, where $y \in \mathbb{R}^n$ and $\alpha \in \mathbb{R}$. Then clearly,

$$\langle h, \nabla^2 \mathcal{L}(0, 0, \hat{\lambda}, \hat{\mu}) h\rangle \geq 0 \quad \forall h$$

and for $h \neq 0$, $h^T \nabla^2 \mathcal{L}(0, 0, \hat{\lambda}, \hat{\mu}) h = y^T H y$ is zero if and only if $y = 0$ and $\alpha \neq 0$. Since for such h

$$\begin{pmatrix} \nabla f(x^*) \\ -1 \end{pmatrix}^T \begin{pmatrix} 0 \\ \alpha \end{pmatrix} = -\alpha \neq 0,$$

it then follows that $\nabla^2 \mathcal{L}(0, 0, \hat{\lambda}, \hat{\mu})$ is positive definite on $N^*(\eta^*)^{\perp}$, the tangent space to the active constraints for $QP(x^*, H, \eta^*)$ at $(0, 0)$. Thus, it is established that the second-order sufficiency conditions hold.

Finally, it follows from Lemma 2(ii) that $\hat{\mu} > 0$ and $\hat{\lambda} = \hat{\mu}\lambda^*$ which, together with Assumption 5, implies strict complementarity for $QP(x^*, H, \eta^*)$ at $(0, 0)$. ∎

LEMMA 12. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then, if* $\mathcal{K}$ *is a subsequence on which* $\{\eta_k\}$ *converges, say, to* $\eta^* \geq 0$, *then* $\mu_k \overset{k\in\mathcal{K}}{\longrightarrow} \hat{\mu} > 0$ *and* $\lambda_k \overset{k\in\mathcal{K}}{\longrightarrow} \hat{\mu}\lambda^*$, *where* $\hat{\mu} = (1 + \eta^* \sum_j \lambda^{*,j})^{-1}$. *Finally,* $d_k \to 0$ *and* $\gamma_k \to 0$.

Proof. First, proceed by contradiction to show that the first two claims hold and that, in addition,

$$(d_k, \gamma_k) \overset{k\in\mathcal{K}}{\longrightarrow} (0, 0); \tag{3.9}$$

i.e., suppose that on some infinite index set $\mathcal{K}' \subseteq \mathcal{K}$ either μ_k is bounded away from $\hat{\mu}$, or λ_k is bounded away from $\hat{\mu}\lambda^*$, or (d_k, γ_k) is bounded away from zero. In view of Assumption 4, there is no loss of generality is assuming that $H_k \overset{k\in\mathcal{K}'}{\longrightarrow} H^*$ for some symmetric positive definite H^*. In view of Lemmas 10 and 11, we may thus invoke a result due to Robinson (Theorem 2.1 in [23]) to conclude that, in view of Lemma 2(ii),

$$(d_k, \gamma_k) \overset{k\in\mathcal{K}'}{\longrightarrow} (0, 0), \quad \mu_k \overset{k\in\mathcal{K}'}{\longrightarrow} \hat{\mu}, \quad \lambda_k \overset{k\in\mathcal{K}'}{\longrightarrow} \hat{\mu}\lambda^*,$$

a contradiction. Hence the first two claims hold, as does (3.9). Next, proceeding again by contradiction, suppose that $d_k \not\to 0$. Then, since $\{H_k\}$ and $\{\eta_k\}$ are bounded, there exists an infinite index set $\mathcal{K}$ on which $\{H_k\}$ and $\{\eta_k\}$ converge and d_k is bounded away from zero. This contradicts (3.9). Thus $d_k \to 0$. It immediately follows from the first constraint in $QP(x_k, H_k, \eta_k)$ that $\gamma_k \to 0$. ∎

LEMMA 13. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then, for all* k *sufficiently large,* $I_k = I(x^*)$.

Proof. Since $\{\eta_k\}$ is bounded and, in view of Lemma 12, $(d_k, \gamma_k) \to (0, 0)$, Lemma 6 implies that $I_k \subseteq I(x^*)$, for all k sufficiently large. Now suppose it does not hold that $I_k = I(x^*)$ for all k sufficiently large. Thus, there exists $j' \in I(x^*)$ and an infinite index set $\mathcal{K}$ such that $j' \notin I_k$, for all $k \in \mathcal{K}$. Now, in view of Lemma 5, there exists an infinite index set $\mathcal{K}' \subseteq \mathcal{K}$ and $\eta^* \geq 0$ such that $\eta_k \overset{k\in\mathcal{K}'}{\longrightarrow} \eta^*$. Since $j' \in I(x^*)$, Assumption 5 guarantees $\lambda^{*,j'} > 0$. Further, Lemma 12 shows that $\lambda_k^{j'} \overset{k\in\mathcal{K}'}{\longrightarrow} \hat{\mu}\lambda^{*,j'} > 0$. Therefore, $\lambda_k^{j'} > 0$ for all k sufficiently large, $k \in \mathcal{K}'$, which, by complementary slackness, implies $j' \in I_k$ for all $k \in \mathcal{K}'$ large enough. Since $\mathcal{K}' \subseteq \mathcal{K}$, this is a contradiction, and the claim is proved. ∎

Now define

$$R_k \triangleq [\ \nabla g_j(x_k)\ :\ j \in I(x^*)\ ],$$
$$g_k \triangleq [\ g_j(x_k)\ :\ j \in I(x^*)\ ]^T,$$

and, given a vector $\lambda \in \mathbb{R}^m$, define the notation

$$\lambda^+ \triangleq [\ \lambda^j\ :\ j \in I(x^*)\ ]^T.$$

Note that, in view of Lemma 9(iii), for k large enough, the optimality conditions (3.1) yield

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^0 \\ (\lambda_k^0)^+ \end{pmatrix} = -\begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix}. \tag{3.10}$$

The following well-known result will be used.

LEMMA 14. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then the matrix*

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix}$$

is invertible for all k large enough and its inverse remains bounded as $k \to \infty$.

LEMMA 15. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. For all k sufficiently large,* d_k^{E} *and* λ_k^{E} *are uniquely defined, and* $d_k^{\mathrm{E}} = d_k^0$.

Proof. In view of Lemma 13, the optimality conditions (3.2), and Lemma 14, for all k large enough, the estimate d_k^{E} and its corresponding multiplier vector λ_k^{E} are well defined as the unique solution of

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^{\mathrm{E}} \\ (\lambda_k^{\mathrm{E}})^+ \end{pmatrix} = - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix}. \tag{3.11}$$

The claim then follows from (3.10). ∎

LEMMA 16. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then*

(i) $\eta_k \to 0$,

(ii) $\mu_k \to 1$ *and* $\lambda_k \to \lambda^*$,

(iii) *for all k sufficiently large,* $I_k = \{\ j \mid \lambda_k^j > 0\ \}$.

Proof. Claim (i) follows from Step 3(iii) of Algorithm **RFSQP**, since in view of Lemma 12, Lemma 15, and Lemma 9, $\{d_k\}$ and $\{d_k^{\mathrm{E}}\}$ both converge to 0. In view of (i), Lemma 12 establishes that $\mu_k \to 1$, and $\lambda_k \to \lambda^*$; hence claim (ii) is proved. Finally, claim (iii) follows from claim (ii), Lemma 13, and Assumption 5. ∎

We now focus our attention on establishing relationships between d_k, d_k^{C}, and the true SQP direction d_k^0.

LEMMA 17. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then*

(i) $\eta_k = O(\|d_k^0\|^2)$,

(ii) $d_k = d_k^0 + O(\|d_k^0\|^2)$,

(iii) $\gamma_k = O(\|d_k^0\|)$.

Proof. In view of Lemma 15, for all k sufficiently large, d_k^{E} and λ_k^{E} exist and are uniquely defined, and $d_k^{\mathrm{E}} = d_k^0$. Lemmas 12 and 9 ensure that Step 3(iii) of Algorithm **RFSQP** chooses $\eta_k = C_k \cdot \|d_k^{\mathrm{E}}\|^2$ for all k sufficiently large; thus (i) follows. It is clear from Lemma 13 and the optimality conditions (2.2) that d_k and λ_k satisfy

$$\begin{aligned} \begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k \\ \lambda_k^+ \end{pmatrix} &= - \begin{pmatrix} \mu_k \cdot \nabla f(x_k) \\ g_k - \eta_k \cdot \gamma_k \cdot \mathbf{1}_{|I(x^*)|} \end{pmatrix} \\ &= - \begin{pmatrix} \nabla f(x_k) \\ g_k \end{pmatrix} + \eta_k \cdot \begin{pmatrix} \left(\sum_{j \in I(x^*)} \lambda_k^j \right) \cdot \nabla f(x_k) \\ \gamma_k \cdot \mathbf{1}_{|I(x^*)|} \end{pmatrix} \end{aligned} \tag{3.12}$$

for all k sufficiently large, where $\mathbf{1}_{|I(x^*)|}$ is a vector of $|I(x^*)|$ ones. It thus follows from (3.10), Assumption 2, and Lemmas 12, 14, and 16 that

$$d_k = d_k^0 + O(\eta_k),$$

and in view of claim (i), claim (ii) follows. Finally, since (from the QP constraint and Lemma 2) $\langle \nabla f(x_k), d_k \rangle \le \gamma_k < 0$, it is clear that $\gamma_k = O(\|d_k\|) = O(\|d_k^0\|)$. ∎

LEMMA 18. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then* $d_k^{\mathrm{C}} = O(\|d_k^0\|^2)$.

Proof. Let

$$c_k \triangleq [-g_j(x_k + d_k) - \|d_k\|^\tau \ : \ j \in I(x^*)]^T.$$

Expanding $g_j(\cdot)$, $j \in I(x^*)$, about x_k we see that, for some $\xi^j \in (0,1)$, $j \in I(x^*)$,

$$c_k = \Bigg[\overbrace{-g_j(x_k) - \langle \nabla g_j(x_k), d_k \rangle}^{=-\eta_k \cdot \gamma_k} + \frac{1}{2}\langle d_k, \nabla^2 g_j(x_k + \xi^j d_k) d_k \rangle - \|d_k\|^\tau \ : \ j \in I(x^*) \Bigg]^T.$$

Since $\tau > 2$, from Lemma 17 and Assumption 2′ we conclude that $c_k = O(\|d_k^0\|^2)$. Now, for all k sufficiently large, in view of Lemma 13, d_k^{C} is well defined and satisfies

$$g_j(x_k + d_k) + \langle \nabla g_j(x_k), d_k^{\mathrm{C}} \rangle = -\|d_k\|^\tau, \quad j \in I(x^*); \tag{3.13}$$

thus

$$R_k^T d_k^{\mathrm{C}} = c_k. \tag{3.14}$$

Now, the first-order KKT conditions for $LS^{\mathrm{C}}(x_k, d_k, H_k, I_k)$ tell us there exists a multiplier ${\lambda^{\mathrm{C}}}_k \in \mathbb{R}^{|I(x^*)|}$ such that

$$\begin{cases} H_k(d_k + d_k^{\mathrm{C}}) + \nabla f(x_k) + R_k {\lambda^{\mathrm{C}}}_k = 0, \\ R_k^T d_k^{\mathrm{C}} = c_k. \end{cases}$$

Also, from the optimality conditions (3.12) we have

$$H_k d_k + \nabla f(x_k) = q_k - R_k \lambda_k^+,$$

where

$$q_k \triangleq \eta_k \cdot \left(\sum_{j \in I(x^*)} \lambda_k^j \right) \cdot \nabla f(x_k).$$

In view of Lemma 17, $q_k = O(\|d_k^0\|^2)$. So, d_k^{C} and ${\lambda^{\mathrm{C}}}_k$ satisfy

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^{\mathrm{C}} \\ {\lambda^{\mathrm{C}}}_k \end{pmatrix} = \begin{pmatrix} R_k \lambda_k^+ - q_k \\ c_k \end{pmatrix}$$

or equivalently, with $\lambda_k' = {\lambda^{\mathrm{C}}}_k - \lambda_k^+$,

$$\begin{bmatrix} H_k & R_k \\ R_k^T & 0 \end{bmatrix} \begin{pmatrix} d_k^{\mathrm{C}} \\ \lambda_k' \end{pmatrix} = \begin{pmatrix} -q_k \\ c_k \end{pmatrix} = O(\|d_k^0\|^2).$$

The result then follows from Lemma 14. ∎

In order to prove the key result that the full step of one is eventually accepted by the line search, we now assume that the matrices $\{H_k\}$ suitably approximate the Hessian of the Lagrangian at the solution. Define the projection

$$P_k \triangleq I - R_k(R_k^T R_k)^{-1} R_k^T.$$

Assumption 6.

$$\lim_{k\to\infty} \frac{\|P_k(H_k - \nabla^2_{xx}L(x^*,\lambda^*))P_k d_k\|}{\|d_k\|} = 0.$$

The following technical lemma will be used.

LEMMA 19. *Suppose Assumptions* 1, 2′, *and* 3–5 *hold. Then there exist constants* $\nu_1,\ \nu_2,\ \nu_3 > 0$ *such that*

(i) $\langle \nabla f(x_k), d_k\rangle \le -\nu_1\|d_k^0\|^2$,

(ii) *for all* k *sufficiently large,*

$$\sum_{j\in I(x^*)} \lambda_k^j g_j(x_k) \le -\nu_2\|g_k\|,$$

(iii) $d_k = P_k d_k + d_k^1$, *where, for all* k *sufficiently large,*

$$\|d_k^1\| \le \nu_3\|g_k\| + O(\|d_k^0\|^3).$$

Proof. To show part (i), note that in view of the first QP constraint, negativity of the optimal value of the QP objective, and Assumption 4,

$$\begin{aligned}\langle \nabla f(x_k), d_k\rangle &\le \gamma_k \\ &\le -\tfrac{1}{2}\langle d_k, H_k d_k\rangle \\ &\le -\tfrac{\sigma_1}{2}\|d_k\|^2 = -\tfrac{\sigma_1}{2}\|d_k^0\|^2 + O(\|d_k^0\|^4).\end{aligned}$$

The proof of part (ii) is identical to that of Lemma 4.4 in [16]. To show (iii), note that from (3.12) for all k sufficiently large, d_k satisfies

$$R_k^T d_k = -g_k - \gamma_k\eta_k\cdot \mathbf{1}_{|I(x^*)|}.$$

Thus, we can write $d_k = P_k d_k + d_k^1$, where

$$d_k^1 = -R_k(R_k^T R_k)^{-1}(g_k + \gamma_k\eta_k\cdot \mathbf{1}_{|I(x^*)|}).$$

The result follows from Assumption 3 and Lemma 17(i),(iii). ∎

PROPOSITION 1. *Suppose Assumptions* 1, 2′, *and* 3–6 *hold. Then,* $t_k = 1$ *for all* k *sufficiently large.*

Proof. Following [16], consider an expansion of $g_j(\cdot)$ about $x_k + d_k$ for $j \in I(x^*)$, for all k sufficiently large,

$$\begin{aligned} g_j(x_k + d_k + d_k^{\mathrm{C}}) &= g_j(x_k + d_k) + \langle \nabla g_j(x_k + d_k), d_k^{\mathrm{C}}\rangle + O(\|d_k^0\|^4) \\ &= g_j(x_k + d_k) + \langle \nabla g_j(x_k), d_k^{\mathrm{C}}\rangle + O(\|d_k^0\|^3) \\ &= -\|d_k\|^\tau + O(\|d_k^0\|^3) \\ &= -\|d_k^0\|^\tau + O(\|d_k^0\|^3),\end{aligned}$$

where we have used Assumption 2′, Lemmas 17 and 18, boundedness of all sequences, and (3.13). As $\tau < 3$, it follows that $g_j(x_k + d_k + d_k^{\mathrm{C}}) \le 0$, $j \in I(x^*)$, for all k sufficiently large. The same result trivially holds for $j \notin I(x^*)$. Thus, for k large

enough, the full step of one satisfies the feasibility condition in the arc search test. It remains to show that the "sufficient decrease" condition is satisfied as well.

First, in view of Assumption 2′ and Lemmas 17 and 18,

$$\begin{aligned} f(x_k + d_k + d_k^{\mathrm{C}}) &= f(x_k) + \langle \nabla f(x_k), d_k \rangle + \langle \nabla f(x_k), d_k^{\mathrm{C}} \rangle \\ &\quad + \tfrac{1}{2}\langle d_k, \nabla^2 f(x_k) d_k \rangle + O(\|d_k^0\|^3). \end{aligned} \tag{3.15}$$

From the top equation in optimality conditions (2.2), equation (2.3), Lemma 17(i), and boundedness of all sequences, we obtain

$$H_k d_k + \nabla f(x_k) + \sum_{j=1}^{m} \lambda_k^j \nabla g_j(x_k) = O(\|d_k^0\|^2). \tag{3.16}$$

The last line in (2.2) and Lemma 17(i),(iii) yield

$$\lambda_k^j \langle \nabla g_j(x_k), d_k \rangle = -\lambda_k^j g_j(x_k) + O(\|d_k^0\|^3). \tag{3.17}$$

Taking the inner product of (3.16) with d_k, then adding and subtracting the quantity $\sum_j \lambda_k^j \langle \nabla g_j(x_k), d_k \rangle$, using (3.17), and finally multiplying the result by $\frac{1}{2}$ gives

$$\begin{aligned} \tfrac{1}{2}\langle \nabla f(x_k), d_k \rangle &= -\frac{1}{2}\langle d_k, H_k d_k \rangle - \sum_{j=1}^{m} \lambda_k^j \langle \nabla g_j(x_k), d_k \rangle \\ &\quad - \frac{1}{2}\sum_{j=1}^{m} \lambda_k^j g_j(x_k) + O(\|d_k^0\|^3). \end{aligned} \tag{3.18}$$

Further, Lemmas 17 and 18 and (3.16) give

$$\langle \nabla f(x_k), d_k^{\mathrm{C}} \rangle = -\sum_{j=1}^{m} \lambda_k^j \langle \nabla g_j(x_k), d_k^{\mathrm{C}} \rangle + O(\|d_k^0\|^3). \tag{3.19}$$

Combining (3.15), (3.18), and (3.19) and using the fact that, for k large enough, $\lambda_k^j = 0$ for all $j \notin I(x^*)$ (Lemma 9(iii)), we obtain

$$\begin{aligned} & f(x_k + d_k + d_k^{\mathrm{C}}) - f(x_k) \\ &= \frac{1}{2}\langle \nabla f(x_k), d_k \rangle - \frac{1}{2}\langle d_k, H_k d_k \rangle - \frac{1}{2}\sum_{j \in I(x^*)} \lambda_k^j g_j(x_k) \\ &\quad - \sum_{j \in I(x^*)} \lambda_k^j \langle \nabla g_j(x_k), d_k \rangle - \sum_{j \in I(x^*)} \lambda_k^j \langle \nabla g_j(x_k), d_k^{\mathrm{C}} \rangle \\ &\quad + \frac{1}{2}\langle d_k, \nabla^2 f(x_k) d_k \rangle + O(\|d_k^0\|^3). \end{aligned}$$

With this in hand, arguments identical to those used following (4.9) in [16] show that

$$f(x_k + d_k + d_k^{\mathrm{C}}) - f(x_k) - \alpha \langle \nabla f(x_k), d_k \rangle < 0$$

for all k sufficiently large. Thus the "sufficient decrease" condition is satisfied. ∎

A consequence of Lemmas 17 and 18 and Proposition 1 is that the algorithm generates a convergent sequence of iterates satisfying

$$x_{k+1} - x_k = d_k^0 + O(\|d_k^0\|^2).$$

Two-step superlinear convergence follows.

THEOREM 2. *Suppose Assumptions* 1, 2′, *and* 3–6 *hold. Then Algorithm* **RFSQP** *generates a sequence* $\{x_k\}$ *which converges* 2*-step superlinearly to* x^*, *i.e.,*

$$\lim_{k\to\infty} \frac{\|x_{k+2} - x^*\|}{\|x_k - x^*\|} = 0.$$

The proof is not given as it follows step by step, with minor modifications, that of [20, sections 2–3].

Finally, note that Q-superlinear convergence would follow if Assumption 6 were replaced with the stronger assumption

$$\lim_{k\to\infty} \frac{\|P_k(H_k - \nabla^2_{xx} L(x^*, \lambda^*))d_k\|}{\|d_k\|} = 0.$$

(See, e.g., [2].)

4. Implementation and numerical results. Our implementation of **RFSQP** (in C) differs in a number of ways from the algorithm stated in section 2. (It is readily checked that none of the differences significantly affect the convergence analysis of section 3.) Just like in the existing C implementation of **FSQP** (CFSQP: see [13]) the distinctive character of linear (affine) constraints and of simple bounds is exploited (provided the nature of these constraints is made explicit). Thus the general form of the problem description tackled by our implementation is

$$\begin{array}{lll} \min & f(x) & \\ \text{s.t.} & g_j(x) \le 0, & j = 1, \dots, m_n, \\ & \langle a_j, x\rangle + b_j \le 0, & j = 1, \dots, m_a, \\ & x^\ell \le x \le x^u, & \end{array}$$

where $a_j \in \mathbb{R}^n$, $b_j \in \mathbb{R}$, $j = 1, \dots, m_a$, and x^ℓ, $x^u \in \mathbb{R}^n$ with $x^\ell < x^u$ (componentwise). The details of the implementation are spelled out below. Many of them, including the update rule for H_k, are exactly as in CFSQP.

In the implementation of $QP(x_k, H_k, \eta_k)$, no "tilting" is performed in connection with the linear constraints and simple bounds, since clearly the untilted SQP direction is feasible for these constraints. In addition, each nonlinear constraint is assigned its own tilting parameter η_k^j, $j = 1, \dots, m_n$. Thus $QP(x_k, H_k, \eta_k)$ is replaced with

$$\begin{array}{lll} \min & \frac{1}{2}\langle d, H_k d\rangle + \gamma & \\ \text{s.t.} & \langle \nabla f(x_k), d\rangle \le \gamma, & \\ & g_j(x) + \langle \nabla g_j(x), d\rangle \le \gamma \cdot \eta_k^j, & j = 1, \dots, m_n, \\ & \langle a_j, x_k + d\rangle + b_j \le 0, & j = 1, \dots, m_a, \\ & x^\ell - x_k \le d \le x^u - x_k. & \end{array}$$

The η_k^j's are updated independently, based on independently adjusted C_k^j's. In the algorithm description and in the analysis, all that was required of C_k was that it remain bounded and bounded away from zero. In practice, though, performance of the algorithm is critically dependent upon the choice of C_k. In the implementation, an adaptive scheme was chosen in which the new values C_{k+1}^j are selected in Step 3 based on their previous values C_k^j, on the outcome of the arc search in Step 2, and on a preselected parameter $\delta_c > 1$. Specifically, (i) if the full step of one was accepted ($t_k = 1$), then all C^j are left unchanged; (ii) if the step of one was not accepted even though all trial points were feasible, then, for all j, C_k^j is decreased to $\min\{\delta_c C_k^j, \overline{C}\}$; (iii) if some infeasibility was encountered in the arc search, then, for all j such that g_j caused a step reduction at some trial point, C_k^j is increased to $\max\{C_k^j/\delta_c, \underline{C}\}$ and, for all other j, C_k^j is kept constant. Here, g_j is said to cause a step reduction if, for some trial point x, g_j is violated (i.e., $g_j(x) > 0$) but all constraints checked at x before g_j were found to be satisfied at that point. (See below for the order in which constraints are checked in the arc search.)

It was stressed in section 2 that the Maratos correction can be computed using an inequality-constrained QP such as QP^{C}, instead of LS^{C}. This was done in our numerical experiments, in order to more meaningfully compare the new algorithm with CFSQP, in which an inequality-constrained QP is indeed used. The implementation of QP^{C} and LS^{E} involves index sets of "almost active" constraints and of binding constraints. First we define

$$I_k^n = \{\ j \mid g_j(x_k) + \langle \nabla g_j(x_k), d_k\rangle - \gamma_k \cdot \eta_k^j > -\sqrt{\epsilon_m}\ \},$$
$$I_k^a = \{\ j \mid \langle a_j, x_k + d_k\rangle + b_j > -\sqrt{\epsilon_m}\ \},$$

where ϵ_m is the machine precision. Next, the *binding* sets are defined as

$$I_k^{b,n} = \{\ j \mid \lambda_k^j > 0\ \}, \qquad I_k^{b,a} = \{\ j \mid \lambda_k^{a,j} > 0\ \},$$
$$I_k^{b,l} = \{\ j \mid \zeta_k^{l,j} > 0\ \}, \qquad I_k^{b,u} = \{\ j \mid \zeta_k^{u,j} > 0\ \},$$

where $\lambda_k \in \mathbb{R}^{m_n}$ is now the QP multiplier corresponding to the nonlinear constraints and where $\lambda_k^a \in \mathbb{R}^{m_a}$, $\zeta_k^u \in \mathbb{R}^n$, and $\zeta_k^l \in \mathbb{R}^n$ are the QP multipliers corresponding to the affine constraints, the upper bounds, and the lower bounds, respectively. Of course, no bending is required from d_k^{C} in connection with affine constraints and simple bounds; hence if $I_k^n = \emptyset$, we simply set $d_k^{\mathrm{C}} = 0$. Otherwise the following modification of QP^{C} is used:

$$\begin{aligned}
\min \quad & \langle d_k + d^{\mathrm{C}}, H_k(d_k + d^{\mathrm{C}})\rangle + \langle \nabla f(x_k), d_k + d^{\mathrm{C}}\rangle & \\
\text{s.t.} \quad & g_j(x_k + d_k) + \langle \nabla g_j(x_k), d^{\mathrm{C}}\rangle \le -\min\{10^{-2}\|d_k\|, \|d_k\|^\tau\}, & j \in I_k^n,\\
& \langle a_j, x_k + d_k + d^{\mathrm{C}}\rangle + b_j \le 0, & j \in I_k^a,\\
& d^{\mathrm{C},j} \le x^u - x_k^j - d_k^j, & j \in I_k^{b,u},\\
& d^{\mathrm{C},j} \ge x^l - x_k^j - d_k^j, & j \in I_k^{b,l}.
\end{aligned}$$

Since not all simple bounds are included in the computation of d_k^{C}, it is possible that $x_k + d_k + d_k^{\mathrm{C}}$ will not satisfy all bounds. To take care of this, we simply "clip" d_k^{C} so that the bounds are satisfied. Specifically, for the upper bounds, we perform the

following:

for $j \notin I_k^{b,u}$ **do**
  if $(d_k^{\mathrm{C},j} \geq x^u - x_k^j - d_k^j)$ **then**
    $d_k^{\mathrm{C},j} \leftarrow x^u - x_k^j - d_k^j$
end

The same procedure, mutatis mutandis, is executed for the lower bounds. We note that such a procedure has no effect on the convergence analysis of section 3 since, locally, the active set is correctly identified and a full step along $d_k + d_k^{\mathrm{C}}$ is always accepted. The least squares problem LS^{E} used to compute d_k^{E} is modified similarly. Specifically, in the implementation, d_k^{E} is only computed if $m_n > 0$, in which case we use

$$\begin{aligned}
\min \quad & \tfrac{1}{2}\langle d^{\mathrm{E}}, H_k d^{\mathrm{E}}\rangle + \langle \nabla f(x_k), d^{\mathrm{E}}\rangle & \\
\text{s.t.} \quad & g_j(x_k) + \langle \nabla g_j(x_k), d^{\mathrm{E}}\rangle = 0, & j \in I_{k-1}^{b,n}, \\
& \langle a_j, x_k + d^{\mathrm{E}}\rangle + b_j = 0, & j \in I_{k-1}^{b,a}, \\
& d^{\mathrm{E},j} = x^u - x_k^j, & j \in I_{k-1}^{b,u}, \\
& d^{\mathrm{E},j} = x^l - x_k^j, & j \in I_{k-1}^{b,l}.
\end{aligned}$$

The implementation of the arc search (Step 2) is as in CFSQP. Specifically, feasibility is checked before sufficient decrease, and testing at a trial point is aborted as soon as infeasibility is detected. As in CFSQP, all linear and bound constraints are checked first, then nonlinear constraints are checked in an order maintained as follows: (i) at the start of the arc search from a given iterate x_k, the order is reset to be the natural numerical order; (ii) within an arc search, as a constraint is found to be violated at a trial point, its index is moved to the beginning of the list, with the order of the others left unchanged.

An aspect of the algorithm which was intentionally left vague in sections 2 and 3 was the updating scheme for the Hessian estimates H_k. In the implementation, we use the BFGS update with Powell's modification [21]. Specifically, define

$$\delta_{k+1} \triangleq x_{k+1} - x_k,$$

$$y_{k+1} \triangleq \nabla_x L(x_{k+1}, \lambda_k) - \nabla_x L(x_k, \lambda_k),$$

where, in an attempt to better approximate the true multipliers, if $\mu_k > \sqrt{\epsilon_m}$ we normalize as follows:

$$\lambda_k^j \leftarrow \frac{\lambda_k^j}{\mu_k}, \quad j = 1, \ldots, m_n.$$

A scalar $\theta_{k+1} \in (0, 1]$ is then defined by

$$\theta_{k+1} \triangleq \begin{cases} 1 & \text{if } \langle \delta_{k+1}, y_{k+1}\rangle \geq 0.2 \cdot \langle \delta_{k+1}, H_k \delta_{k+1}\rangle, \\ \dfrac{0.8 \cdot \langle \delta_{k+1}, H_k \delta_{k+1}\rangle}{\langle \delta_{k+1}, H_k \delta_{k+1}\rangle - \langle \delta_{k+1}, y_{k+1}\rangle} & \text{otherwise.} \end{cases}$$

Defining $\xi_{k+1} \in \mathbb{R}^n$ as

$$\xi_{k+1} \triangleq \theta_{k+1} \cdot y_{k+1} + (1 - \theta_{k+1}) \cdot H_k \delta_{k+1},$$

the rank two Hessian update is

$$H_{k+1} = H_k - \frac{H_k \delta_{k+1} \delta_{k+1}^T H_k}{\langle \delta_{k+1}, H_k \delta_{k+1} \rangle} + \frac{\xi_{k+1} \xi_{k+1}^T}{\langle \delta_{k+1}, \xi_{k+1} \rangle}.$$

Note that while it is not clear whether the resultant sequence $\{H_k\}$ will, in fact, satisfy Assumption 6, this update scheme is known to perform very well in practice.

All QPs and linear least squares subproblems were solved using QPOPT [7]. For comparison's sake, QPOPT was also used to solve the QP subproblems in CFSQP. While the default QP solver for CFSQP is the public domain code QLD (see [24]), we opted for QPOPT because it allows "warm starts" and thus is fairer to CFSQP in the comparison with the implementation of **RFSQP** (since more QPs are solved with the former). For all QPs in both codes, the active set in the solution at a given iteration was used as initial guess for the active set for the same QP at the next iteration.

In order to guarantee that the algorithm terminates after a finite number of iterations with an approximate solution, the stopping criterion of Step 1 is changed to

$$\textbf{if } (\|d_k\| \leq \epsilon) \textbf{ stop}, \tag{4.1}$$

where $\epsilon > 0$ is small. Finally, the following parameter values were selected:

$$\begin{array}{lll} \alpha = 0.1, & \beta = 0.5, & \tau = 2.5, \\ \epsilon_\ell = \sqrt{\epsilon}, & \underline{C} = 1 \times 10^{-3}, & \overline{C} = 1 \times 10^{3}, \\ \delta_c = 2, & \bar{D} = 10 \cdot \epsilon_\ell. & \end{array}$$

Further, we always set $H_0 = I$, and $C_0^j = 1$ and $\eta_0^j = \epsilon C_0^j (= \epsilon)$, $j = 1, \ldots, m_n$. All experiments were run on a Sun Microsystems Ultra 5 workstation.

For the first set of numerical tests, we selected a number of problems from [9] which provided feasible initial points and contained no equality constraints. The results are reported in Table 1, where the performance of our implementation of **RFSQP** is compared with that of CFSQP (with QPOPT as QP solver). The column labeled # lists the problem number as given in [9]; the column labeled `ALGO` is self-explanatory. The next three columns give the size of the problem following the conventions of this section. The columns labeled `NF`, `NG`, and `IT` give the number of objective function evaluations, nonlinear constraint function evaluations, and iterations required to solve the problem, respectively. Finally, $f(x^*)$ is the objective function value at the final iterate and ϵ is as above. The value of ϵ was chosen in order to obtain approximately the same precision as reported in [9] for each problem.

The results reported in Table 1 are encouraging. The performance of our implementation of Algorithm **RFSQP** in terms of number of iterations and function evaluations is essentially identical to that of CFSQP (Algorithm **FSQP**). The expected payoff of using **RFSQP** instead of **FSQP**, however, is that on large problems the CPU time expended in linear algebra, specifically in solving the QP and linear least squares subproblems, should be much less. To assess this, we next carried out comparative tests on the COPS suite of problems [3].

The first five problems from the COPS set [3] were considered, as these problems either do not involve nonlinear equality constraints or are readily reformulated without such constraints. (Specifically, in problem "Sphere" the equality constraint was changed to a "$\leq$" constraint; and in "Chain" the equality constraint (with $L = 4$) was replaced with two inequalities, with the left-hand side constrained to be between the

TABLE 1
Numerical results on Hock–Schittkowski problems.

#	ALGO	n	m_a	m_n	NF	NG	IT	$f(x^*)$	ϵ
12	**RFSQP**	2	0	1	7	14	7	-3.0000000E+01	1.E–6
	CFSQP				7	14	7	-3.0000000E+01	
29	**RFSQP**	3	0	1	11	20	10	-2.2627417E+01	1.E–5
	CFSQP				11	20	10	-2.2627417E+01	
30	**RFSQP**	3	0	1	18	35	18	1.0000000E+00	1.E–7
	CFSQP				18	35	18	1.0000000E+00	
31	**RFSQP**	3	0	1	9	36	8	6.0000000E+00	1.E–5
	CFSQP				9	19	7	6.0000000E+00	
33	**RFSQP**	3	0	2	4	11	4	-4.0000000E+00	1.E–8
	CFSQP				4	11	4	-4.0000000E+00	
34	**RFSQP**	3	0	2	8	34	8	-8.3403245E–01	1.E–8
	CFSQP				7	28	7	-8.3403244E–01	
43	**RFSQP**	4	0	3	9	51	9	-4.4000000E+01	1.E–5
	CFSQP				10	46	8	-4.4000000E+01	
66	**RFSQP**	3	0	2	8	30	8	5.1816327E–01	1.E–8
	CFSQP				8	30	8	5.1816327E–01	
84	**RFSQP**	5	0	6	4	37	4	-5.2803351E+06	1.E–8
	CFSQP				4	30	4	-5.2803351E+06	
93	**RFSQP**	6	0	2	13	54	12	1.3507596E+02	1.E–5
	CFSQP				16	62	13	1.3507596E+02	
113	**RFSQP**	10	3	5	12	120	12	2.4306210E+01	1.E–3
	CFSQP				12	108	12	2.4306377E+01	
117	**RFSQP**	15	0	5	20	205	19	3.2348679E+01	1.E–4
	CFSQP				20	219	19	3.2348679E+01	

values $L = 4$ and $L = 5$; the solution was always at 5.) All these problems are nonconvex. "Sawpath" was discarded because it involves few variables and many constraints, which is not the situation at which **RFSQP** is targeted. The results obtained with various instances of the other four problems are presented in Table 2. The format of that table is identical to that of Table 1 except for the additional column labeled NQP. In that column we list the total number of QP iterations in the solution of the two major QPs, as reported by QPOPT. (Note that QPOPT reports zero iteration when the result of the first step onto the working set of linear constraints happens to be optimal. To be "fair" to **RFSQP** we thus do not count the work involved in solving LS^{E} either. We also do not count the QP iterations in solving QP^{C}, the "correction" QP, because it is invoked identically in both algorithms.)

The results show a typical significantly lower number of QP iterations with **RFSQP** and, as in the case of the Hock–Schittkowski problems, a roughly comparable behavior of the two algorithms in terms of number of function evaluations. The abnormal terminations on *Sphere*-50 and *Sphere*-100 are both due to QPOPT's failure to solve a QP—the "tilting" QP in the case of CFSQP.

5. Conclusions. We have presented here a new SQP-type algorithm generating feasible iterates. The main advantage of this algorithm is a reduction in the amount of computation required in order to generate a new iterate. While this may not be very important for applications where function evaluations dominate the actual amount of work to compute a new iterate, it is very useful in many contexts. In any case, we saw in the previous section that preliminary results seem to indicate that decreasing the amount of computation per iteration did not come at the cost of increasing the number of function evaluations required to find a solution.

TABLE 2
Numerical results on COPS problems.

P	ALGO	n	m_a	m_n	NF	NG	IT	NQP	$f(x^*)$	ϵ
Polygon-10	**RFSQP**	18	8	36	17	798	18	51	.749137	1.E–4
	CFSQP				16	740	18	91	.749137	
Polygon-20	**RFSQP**	38	18	171	27	5552	28	142	.776859	1.E–4
	CFSQP				42	8177	44	350	.776859	
Polygon-40	**RFSQP**	78	38	741	267	208706	107	571	.783062	1.E–4
	CFSQP				243	126592	106	1689	.783062	
Polygon-50	**RFSQP**	98	48	1176	1023	1232889	273	938	.783062	1.E–4
	CFSQP				591	345458	154	2771	.783873	
Sphere-20	**RFSQP**	60	0	20	1462	35114	280	302	150.882	1.E–4
	CFSQP				1812	20920	352	745	150.882	
Sphere-30	**RFSQP**	90	0	30	8318	280532	1016	1065	359.604	1.E–4
	CFSQP				6494	74797	837	1743	359.604	
Sphere-40	**RFSQP**	120	0	40	1445	70960	311	406	660.675	1.E–4
	CFSQP				795	28328	246	587	660.675	
Sphere-50	**RFSQP**	150	0	50					failure	1.E–4
	CFSQP				2300	80467	560	1568	1055.18	
Sphere-100	**RFSQP**	300	0	50	516	119252	506	3589	4456.06	1.E–4
	CFSQP								failure	
Chain-50	**RFSQP**	50	0	2	154	917	165	171	4.81198	1.E–4
	CFSQP				247	1034	201	401	4.81198	
Chain-100	**RFSQP**	100	0	2	822	3171	394	401	4.81190	1.E–4
	CFSQP				837	2440	408	828	4.81190	
Chain-150	**RFSQP**	150	0	2	868	4108	485	510	4.81189	1.E–4
	CFSQP				1037	3486	541	1104	4.81189	
Chain-200	**RFSQP**	200	0	2	1218	5805	645	739	4.81189	1.E–4
	CFSQP				1534	5367	785	1648	4.81188	
Cam-50	**RFSQP**	50	1	102	49	13109	75	287	-214.640	1.E–4
	CFSQP				12	6288	39	604	-214.761	
Cam-100	**RFSQP**	100	1	202	12	22436	58	621	-414.067	1.E–4
	CFSQP				14	21558	61	1341	-428.415	
Cam-200	**RFSQP**	200	1	402	9	70824	90	842	-827.255	1.E–4
	CFSQP				16	73120	98	2859	-855.698	
Cam-400	**RFSQP**	400	1	802	15	243905	155	3403	-1678.65	1.E–4
	CFSQP				16	238373	156	6298	-1710.27	

A number of significant extensions of Algorithm **RFSQP** is being examined. It is not too difficult to extend the algorithm to handle mini-max problems. The only real issue that arises is how to handle the mini-max objectives in the least squares subproblems. Several possibilities, each with the desired global and local convergence properties, are being examined. Another extension that is important for engineering design is the incorporation of a scheme to efficiently handle very large sets of constraints and/or objectives. We will examine schemes along the lines of those developed in [12, 27]. Further, work remains to be done to exploit the close relationship between the two least squares problems and the quadratic program. A careful implementation should be able to use these relationships to great advantage computationally. For starters, updating the Cholesky factors of H_k instead of H_k itself at each iteration would save a factorization in each of the subproblems. Finally, it is possible to extend the class of problems (P) which are handled by the algorithm to include nonlinear equality constraints. Of course, we will not be able to generate feasible iterates for such constraints, but a scheme such as that studied in [11] could

be used in order to guarantee asymptotic feasibility while maintaining feasibility for all inequality constraints.

While this paper was under final review, the authors became aware of [10], where a related algorithm is proposed, for which similar properties are claimed. No numerical results are reported in that paper.

Acknowledgments. The authors wish to thank the review editor, Margaret H. Wright, and two anonymous referees for their extensive and most helpful comments on an earlier version of the paper. They also wish to thank Sasan Bakhtiari for his help in setting up the numerical experiments.

REFERENCES

[1] J. Birge, L. Qi, and Z. Wei, *A variant of the Topkis-Veinott method for solving inequality constrained optimization problems*, J. Appl. Math. Optim., 41 (2000), pp. 309–330.

[2] P. T. Boggs and J. W. Tolle, *Sequential Quadratic Programming*, Acta Numerica, Cambridge Univ. Press, Cambridge, UK, 1995, pp. 1–51.

[3] A. S. Bondarenko, D. M. Bortz, and J. J. Moré, *COPS, Large-Scale Nonlinearly Constrained Optimization Problems*, Technical report ANL/MCS-TM-237, Argonne National Laboratory, Argonne, IL, 1999.

[4] R. H. Byrd, J. C. Gilbert, and J. Nocedal, *A trust region method based on interior point techniques for nonlinear programming*, Math. Program., 89 (2000), pp. 149–185.

[5] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang, *On the formulation and theory of the Newton interior-point method for nonlinear programming*, J. Optim. Theory Appl., 89 (1996), pp. 507–541.

[6] D. M. Gay, M. L. Overton, and M. H. Wright, *A primal-dual interior method for nonconvex nonlinear programming*, in Advances in Nonlinear Programming, Y. Yuan, ed., Kluwer, Dordrecht, The Netherlands, 1998, pp. 31–56.

[7] P. E. Gill, W. Murray, and M. A. Saunders, *User's Guide for QPOPT* 1.0: *A Fortran Package for Quadratic Programming*, Technical report, Stanford University, Stanford, CA, 1995.

[8] J. N. Herskovits and L. A. V. Carvalho, *A successive quadratic programming based feasible directions algorithm*, in Proceedings of the Seventh International Conference on Analysis and Optimization of Systems—Antibes, A. Bensoussan and J. L. Lions, eds., Lecture Notes in Control and Inform. Sci. 83, Springer-Verlag, Berlin, 1986, pp. 93–101.

[9] W. Hock and K. Schittkowski, *Test Examples for Nonlinear Programming Codes*, Lecture Notes in Econ. and Math. Systems 187, Springer-Verlag, Berlin, 1981.

[10] M. M. Kostreva and X. Chen, *A superlinearly convergent method of feasible directions*, Appl. Math. Comput., 116 (2000), pp. 231–244.

[11] C. T. Lawrence and A. L. Tits, *Nonlinear equality constraints in feasible sequential quadratic programming*, Optim. Methods Softw., 6 (1996), pp. 265–282.

[12] C. T. Lawrence and A. L. Tits, *Feasible sequential quadratic programming for finely discretized problems from SIP*, in Semi-Infinite Programming, R. Reemtsen and J.-J. Rückmann, eds., Nonconvex Optim. Appl. 25, Kluwer, Boston, 1998, pp. 159–193.

[13] C. T. Lawrence, J. L. Zhou, and A. L. Tits, *User's Guide for CFSQP Version* 2.5: *A C Code for Solving (Large Scale) Constrained Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality Constraints*, Institute for Systems Research, University of Maryland, College Park, MD, 1997.

[14] C. T. Lawrence, *A Computationally Efficient Feasible Sequential Quadratic Programming Algorithm*, Ph.D. thesis, University of Maryland, College Park, MD, 1998.

[15] N. Maratos, *Exact Penalty Functions for Finite Dimensional and Control Optimization Problems*, Ph.D. thesis, Imperial College of Science and Technology, London, 1978.

[16] 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.

[17] 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.

[18] E. R. Panier, A. L. Tits, and J. N. Herskovits, *A QP-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization*, SIAM J. Control Optim., 26 (1988), pp. 788–811.

[19] E. Polak, *Computational Methods in Optimization*, Academic Press, New York, 1971.
[20] M. J. D. Powell, *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.
[21] M. J. D. Powell, *A fast algorithm for nonlinearly constrained optimization calculations*, in Numerical Analysis, G. A. Watson, ed., Lecture Notes in Math. 630, Springer-Verlag, Berlin, 1978, pp. 144–157.
[22] L. Qi and Z. Wei, *On the constant positive linear dependence condition and its application to SQP methods*, SIAM J. Optim., 10 (2000), pp. 963–981.
[23] S. M. Robinson, *Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear-programming algorithms*, Math. Programming, 7 (1974), pp. 1–16.
[24] K. Schittkowski, *QLD: A Fortran Code for Quadratic Programming, User's Guide*, Mathematisches Institut, Universität Bayreuth, Germany, 1986.
[25] T. Urban, A. L. Tits, and C. T. Lawrence, *A Primal-Dual Interior-Point Method for Nonconvex Optimization with Multiple Logarithmic Barrier Parameters and with Strong Convergence Properties*, Technical report 98-27, Institute for Systems Research, University of Maryland, College Park, MD, 1998.
[26] R. J. Vanderbei and D. F. Shanno, *An interior point algorithm for nonconvex nonlinear programming*, Comput. Optim. Appl., 13 (1999), pp. 231–252.
[27] J. L. Zhou and A. L. Tits, *An SQP algorithm for finely discretized continuous minimax problems and other minimax problems with many objective functions*, SIAM J. Optim., 6 (1996), pp. 461–487.
[28] J. L. Zhou, A. L. Tits, and C. T. Lawrence, *User's Guide for FSQP Version* 3.7: *A FORTRAN Code for Solving Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality and Linear Constraints*, ISR TR-92-107r2, Institute for Systems Research, University of Maryland, College Park, MD, 1997.
[29] G. Zoutendijk, *Methods of Feasible Directions*, Elsevier Science, Amsterdam, The Netherlands, 1960.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1119–1144

A NONLINEAR LAGRANGIAN APPROACH TO CONSTRAINED OPTIMIZATION PROBLEMS*

X. Q. YANG† AND X. X. HUANG‡

Abstract. In this paper we study nonlinear Lagrangian functions for constrained optimization problems which are, in general, nonlinear with respect to the objective function. We establish an equivalence between two types of zero duality gap properties, which are described using augmented Lagrangian dual functions and nonlinear Lagrangian dual functions, respectively. Furthermore, we show the existence of a path of optimal solutions generated by nonlinear Lagrangian problems and show its convergence toward the optimal set of the original problem. We analyze the convergence of several classes of nonlinear Lagrangian problems in terms of their first and second order necessary optimality conditions.

Key words. augmented Lagrangian, nonlinear Lagrangian, zero duality gap, optimal path, necessary optimality condition, smooth approximate variational principle

AMS subject classifications. 90C30, 49J52, 49M35

PII. S1052623400371806

1. Introduction.

It is well known that unconstrained optimization methods, such as the Lagrangian dual and penalty methods, have been extensively studied in order to solve constrained optimization problems. A zero duality gap can be guaranteed if conventional Lagrangian functions are used to define the dual problem under convexity or generalized convexity assumptions. Nevertheless, for a nonconvex constrained optimization problem, a nonzero duality gap may occur between the original problem and the conventional Lagrangian dual problem. To overcome this drawback, various approaches have been proposed in the literature. The convex conjugate framework in [16] was extended in [3, 13] for nonconvex optimization problems. In [17], a general augmented Lagrangian function was introduced, and it was shown that the general augmented dual problem constructed with an appropriately selected perturbation function yields a zero duality gap result. Recently, nonlinear Lagrangian functions were introduced using increasing functions for solving constrained optimization problems. A zero duality gap result is established between a nonconvex constrained optimization problem and the dual problem defined by using a nonlinear Lagrangian function in [10, 14, 18, 19]. In passing, we mention that exact penalization-type results were established for the augmented Lagrangian function in [17], for nonlinear Lagrangian functions under generalized calmness-type conditions for scalar optimization problems in [19], and for vector optimization problems in [12].

Noting the fact that, for nonconvex constrained optimization problems, both zero duality gap results in terms of augmented Lagrangian dual functions in [17] and nonlinear Lagrangian dual functions in [19] were established under very mild conditions, it is interesting to investigate whether there is a connection between these two

*Received by the editors May 5, 2000; accepted for publication (in revised form) November 11, 2000; published electronically May 16, 2001. This work was partially supported by the Research Grants Council of Hong Kong (grant PolyU B-Q359).

http://www.siam.org/journals/siopt/11-4/37180.html

†Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong (mayangxq@polyu.edu.hk).

‡Department of Mathematics and Computer Science, Chongqing Normal University, Chongqing 400047, China. Current address: Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong (mahuangx@polyu.edu.hk).

results. Therefore, the first goal of this paper is to establish an equivalence between zero duality gap properties, which are described using a class of augmented Lagrangian functions with specially structured perturbation functions, and nonlinear Lagrangian functions, respectively.

Recently, a wide class of penalty and barrier methods was studied in [2], including a number of specific functions in the literature (see [5, 9]). For convex programming problems, the existence of a path of optimal solutions generated by these penalty methods was established and its convergence toward the optimal set of the original problem was given. Hence, the second goal of this paper is to show, for nonconvex inequality constrained optimization problems, the existence of a path of optimal solutions generated by a general nonlinear Lagrangian function and to show its convergence toward the optimal set of the original problem. Moreover, we illustrate that this result can be specialized to convex programming problems, and thus a parallel result to that in [2] is obtained.

We then investigate the convergence analysis of nonlinear Lagrangian methods in terms of first and second order necessary optimality conditions, where the multipliers are independent of vectors in the tangential subspace of the active constraints. This follows the usual method, as in [1, 22]. Thus we need to derive, for example, corresponding second order necessary conditions for nonlinear Lagrangian problems. However, for cases where nonlinear Lagrangian functions are not twice differentiable, the derivation of this type of second order optimality condition of nonlinear Lagrangian problems is by no means an easy task. For example, one of the nonlinear Lagrangian functions to be considered is of the minimax type. Thus, the resulting problem is an unconstrained minimax optimization problem or, more generally, a convex composite optimization problem. Second order necessary conditions for convex composite optimization problems were established in [4, 7, 13, 23]. However, in these conditions the multipliers depend on the choice of the vector in the tangential subspace of the active constraints. These second order conditions are not applicable in our cases. Nevertheless, we are able to derive the required first and second order necessary conditions for these nonlinear Lagrangian problems by means of a higher order smooth approximation and the smooth approximate variational principle in [6, 8].

The outline of the paper is as follows. In section 2, we review the zero duality gap properties, which are obtained using augmented Lagrangian functions and nonlinear Lagrangian functions. In section 3, we show that if the dual problem which is constructed with an augmented Lagrangian and a specially structured perturbation function yields a zero duality gap, then the dual problem defined by nonlinear Lagrangian dual functions also yields a zero duality gap, and vice versa. In section 4, we show the existence of a path of optimal solutions generated by nonlinear Lagrangian problems and show its convergence to the optimal set of the original problem. In section 5, we carry out convergence analysis of this method for several classes of nonlinear Lagrangians in terms of first and second order necessary optimality conditions.

2. Zero duality gaps. In this section, we introduce some definitions and recall the zero duality gap properties, which are described by augmented Lagrangian functions and nonlinear Lagrangian functions, respectively. Consider the following inequality constrained optimization problem (P):

$$\begin{array}{ll} \inf & f(x) \\ \text{s.t.} & x \in X, \quad g_j(x) \le 0,\ j = 1, \ldots, q, \end{array}$$

where $X \subset R^p$ is a nonempty and closed set, and $f,\ g_j : X \to R^1\ (j = 1, \ldots, q)$ are real-valued functions. Denote by M_P the infimum of (P) and by X_0 the feasible set of (P):

$$X_0 = \{x \in X\ :\ g_j(x) \le 0 \quad \forall j = 1, \ldots, q\}.$$

In this paper, we assume that $X_0 \neq \emptyset$.

Throughout this paper, we also assume that

$$f(x) \ge 0 \quad \forall x \in X.$$

Note that this assumption is not very restrictive. Otherwise, we may replace the objective function $f(x)$ with $1+e^{f(x)}$, which satisfies the assumption; $\inf_{x\in X} f(x) > 0$ also holds; and the resulting constrained optimization problem has the same set of (local) solutions as that of (P).

Let $c : R^1_+ \times R^q \to R^1$ be a real-valued function. c is said to be *increasing* on $R^1_+ \times R^q$ if, for any $y^1, y^2 \in R^1_+ \times R^q$, $y^2 - y^1 \in R^{q+1}_+$ implies that $c(y^1) \le c(y^2)$. We will consider increasing and lower semicontinuous (l.s.c.) functions c defined on $R^1_+ \times R^q$, which enjoy the following properties:

(A) There exist positive real numbers $a_j, j = 1, \ldots, q$, such that, for any $y = (y_0, y_1, \ldots, y_q) \in R^1_+ \times R^q$, we have

$$c(y) \ge \max\{y_0, a_1 y_1, \ldots, a_q y_q\}.$$

(B) For any $y_0 \in R^1_+$,

$$c(y_0, 0, \ldots, 0) = y_0.$$

Let $y^+ = \max\{y, 0\}$ for $y \in R$. The following are some examples of function c (see [18]):

$$c(y) = \max\{y_0, y_1, \ldots, y_q\},$$
$$c(y) = \left(y_0^k + \sum_{j=1}^{q} y_j^{+k}\right)^{1/k}, \quad k \in (0, +\infty).$$

The convergence analysis of optimality conditions for nonlinear Lagrangian dual problems defined by these functions (see below) will be given in section 5.

Let c be an increasing function defined as above, and

$$F(x, d) = (f(x), d_1 g_1(x), \ldots, d_q g_q(x)) \quad \forall x \in X,\ d = (d_1, \ldots, d_q) \in R^q_+.$$

The function defined by

$$L(x, d) = c(F(x, d))$$

is called a *nonlinear Lagrangian* corresponding to c.

The *nonlinear Lagrangian dual function* for (P) corresponding to c is defined by

$$\phi(d) = \inf_{x\in X} L(x, d), \quad d \in R^q_+.$$

The *nonlinear Lagrangian dual problem* (D_N) for (P) corresponding to c is defined by

$$\sup_{d\in R^q_+} \phi(d).$$

Denote by M_N the supremum of problem (D_N). It can be easily verified [18, 19] that the following weak duality result holds:

$$M_N \le M_P. \tag{1}$$

DEFINITION 2.1. *Let c be an increasing function satisfying properties* (A) *and* (B). *The zero duality gap property with respect to c between* (P) *and* (D_N) *is said to hold if $M_N = M_P$.*

DEFINITION 2.2 (see [2]). *Let $X \subset R^p$ be unbounded. The function $h : X \to R^1$ is said to be* 0*-coercive on X if*

$$\lim_{x\in X, \|x\|\to+\infty} h(x) = +\infty.$$

Let

$$\begin{aligned} G(x) &= \max\{g_1(x), \dots, g_q(x)\}, \quad x \in X, \\ h(x) &= \max\{f(x), G(x)\}, \quad x \in X. \end{aligned} \tag{2}$$

THEOREM 2.3. *Suppose that h, defined by* (2), *is* 0*-coercive if X is unbounded. If the functions $f, g_1, \dots, g_q$ are l.s.c., then the zero duality gap property with respect to c between* (P) *and* (D_N) *holds.*

Proof. It is clear that $L(x, d)$ is an increasing function of d. The result follows from Theorem 4.2 in section 4. □

Let us recall the definition of the augmented Lagrangian function for (P) (for details, see Chapter 11, section K^* in [17]). Let $\varphi : R^p \to R^1 \bigcup\{+\infty\}$:

$$\varphi(x) = \begin{cases} f(x) & \text{if } x \in X_0; \\ +\infty & \text{otherwise.} \end{cases}$$

Let $\overline{f} : R^p \times R^q \to R^1 \bigcup\{+\infty\}$ be a perturbation function [17, p. 519] such that $\overline{f}(x, 0) = \varphi(x),\ x \in R^p$. Let σ be an *augmenting function*, namely, a proper, l.s.c., and convex function with the unique minimum at 0 and $\sigma(0) = 0$. The corresponding *augmented Lagrangian* $\overline{l} : R^p \times R^q \times (0, +\infty) \to R^1 \bigcup\{+\infty, -\infty\}$ with parameter $r > 0$ is defined by

$$\overline{l}(x, y, r) = \inf\{\overline{f}(x, u) + r\sigma(u) - \langle y, u\rangle : u \in R^q\},$$

where $\langle y, u\rangle$ denotes the inner product of y and u.

The corresponding *augmented Lagrangian dual function* is

$$\psi(y, r) = \inf\{\overline{l}(x, y, r) : x \in R^p\},$$

and the *augmented Lagrangian dual problem* (D_A) is

$$\sup_{(y,r)\in R^q\times(0,+\infty)} \psi(y, r).$$

Let M_A denote the supremum of the dual problem (D_A). The following weak duality for (P) and (D_A) holds (see [17]):

$$M_A \le M_P. \tag{3}$$

DEFINITION 2.4. *Let $\overline{f} : R^p \times R^q \to R^1 \bigcup \{+\infty\}$ be a perturbation function and σ be an augmenting function. The zero duality gap property with respect to $\overline{f}$ and σ between* (P) *and* (D_A) *is said to hold if $M_A = M_P$.*

DEFINITION 2.5 (see [17]). *A function $h : R^p \times R^q \to R^1 \bigcup \{+\infty, -\infty\}$ with values $h(x,u)$ is said to be level-bounded in x and locally uniform in u if, for each $\overline{u} \in R^q$ and $\alpha \in R^1$, there exists a neighborhood $V(\overline{u})$ of $\overline{u}$, along with a bounded set $D \subset R^p$, such that $\{x \in R^p : h(x,v) \le \alpha\} \subset D \ \forall v \in V(\overline{u})$.*

THEOREM 2.6 (see [17]). *Assume that the perturbation function $\overline{f} : R^p \times R^q \to R^1 \bigcup \{+\infty\}$ is proper and l.s.c., and that $\overline{f}(x,u)$ is level-bounded in x and locally uniform in u. Let σ be an augmenting function. Suppose further that there exist $\overline{y} \in R^q$ and $\overline{r} > 0$ such that*

$$\inf\{\overline{f}(x,u) + \overline{r}\sigma(u) - \langle \overline{y}, u\rangle : x \in R^p, u \in R^q\} > -\infty. \tag{4}$$

Then $M_A = M_P$.

3. Equivalence of zero duality gaps. In this section, we establish an equivalence of zero duality gap properties between a class of augmented Lagrangian dual problems and the nonlinear Lagrangian dual problem.

Denote the indicator function of a set $D \subset R^q$ by

$$\delta_D(y) = \begin{cases} 0 & \text{if } y \in D; \\ +\infty & \text{otherwise.} \end{cases}$$

It is easy to check that (P) is equivalent to the following problem:

$$\inf_{x \in X} f(x) + \delta_{R^q_-}(g_1(x), \ldots, g_q(x))$$

in the sense that the two problems have the same sets of (locally) optimal solutions and optimal values. Let

$$H(x) = (g_1(x), \ldots, g_q(x)),$$

$$\overline{f}(x,u) = f(x) + \delta_{R^q_-}(H(x) + u) + \delta_X(x). \tag{5}$$

Then, for $x \in R^p$, $\overline{f}(x,0) = \varphi(x)$. Thus, $\overline{f}(x,u)$ is a perturbation function.

LEMMA 3.1. *Let the perturbation function be defined by* (5)*, σ an augmenting function, and $v = (v_1, \ldots, v_q)$. Then*

$$\overline{l}(x,y,r) = \begin{cases} f(x) + \displaystyle\sum_{j=1}^{q} y_j g_j(x) + \inf_{v \ge 0} \left\{ \sum_{j=1}^{q} y_j v_j + r\sigma(-g_1(x) - v_1, \ldots, -g_q(x) - v_q) \right\} & \text{if } x \in X, \\ +\infty & \text{otherwise.} \end{cases}$$

Proof. Let $x \in X$.

$$\begin{aligned}\bar{l}(x,y,r) &= \inf\{\overline{f}(x,u) + r\sigma(u) - \langle y,u\rangle : u \in R^q\} \\ &= \inf_{v\geq 0}\left\{ f(x) + \sum_{j=1}^{q} y_j(g_j(x)+v_j) + r\sigma(-g_1(x)-v_1,\ldots,-g_q(x)-v_q)\right\} \\ &= f(x) + \sum_{j=1}^{q} y_j g_j(x) + \inf_{v\geq 0}\left\{\sum_{j=1}^{q} y_j v_j + r\sigma(-g_1(x)-v_1,\ldots,-g_q(x)-v_q)\right\}.\end{aligned}$$

Let $x \notin X$. It is clear that $\overline{f}(x,u) = +\infty$. Thus $\bar{l}(x,y,r) = +\infty$. □

The following proposition summarizes some properties of augmented Lagrangian $\bar{l}$, where $\overline{f}$ is defined by (5), and the nonlinear Lagrangian L.

LEMMA 3.2. *Let the perturbation function $\overline{f}(x,u)$ be defined by* (5)*. Then, the following properties of augmented Lagrangian function $\bar{l}$ hold:*

(I) $\bar{l}(x,y,r) \leq f(x)\ \forall x \in X_0,\ y \in R^q,\ r > 0$, *and* $\bar{l}(x,0,r) = f(x)\ \forall x \in X_0$, $r > 0$.

(II) $\bar{l}(x,0,r) \geq f(x)\ \forall x \in X$.

(III) *For any* $x \in X\backslash X_0,\ y \in R^q,\ \ \bar{l}(x,y,r) \to +\infty$ *as* $r \to +\infty$,

and the following properties of nonlinear Lagrangian function L hold:

(I′) $L(x,d) = f(x)\ \forall x \in X_0$.

(II′) $L(x,d) \geq f(x)\ \forall x \in X$.

(III′) *For any* $x \in X\backslash X_0,\ \ L(x,d) \to +\infty$ *as* $d \to +\infty$.

Here the notation $d = (d_1,\ldots,d_q) \to +\infty$ *means that* $d_j \to +\infty$ *for each* $j \in \{1,\ldots,q\}$.

It follows from Lemma 3.2 that $\bar{l}(x,0,r)$ behaves very similarly to $L(x,re)$, where $e = (1,\ldots,1) \in R^q_+$. For any $x \in R^p$, let

$$J^+(x) = \{j \in \{1,\ldots,q\} : g_j(x) > 0\},\ J(x) = \{j \in \{1,\ldots,q\} : g_j(x) = 0\}.$$

PROPOSITION 3.3. *Let augmenting function σ be a finite and l.s.c. function which attains its minimum* 0 *at* $0 \in R^q$. *Let the perturbation function $\overline{f}(x,u)$ defining the augmented Lagrangian be selected as* (5)*. If $M_A = M_P$, then $M_N = M_P$.*

Proof. If $M_N = M_P$ fails to hold by weak duality (1) of the nonlinear Lagrangian, then there exists $\epsilon_0 > 0$ such that $M_N \leq M_P - \epsilon_0$.

By the assumption, we get

$$M_A = \sup_{(y,r)\in R^q\times(0,+\infty)} \inf_{x\in X} \bar{l}(x,y,r) = M_P.$$

Then, for $\frac{\epsilon_0}{4} > 0$, there exist $\bar{y} \in R^q$ and $\bar{r} > 0$ such that $\bar{l}(x,\bar{y},\bar{r}) \geq M_P - \frac{\epsilon_0}{4}\ \forall x \in X$. That is, for any $x \in X$,

(6)
$$f(x) + \sum_{j=1}^{q} \bar{y}_j g_j(x) + \inf_{v\geq 0}\left\{\sum_{j=1}^{q} \bar{y}_j v_j + \bar{r}\sigma(-g_1(x)-v_1,\ldots,-g_q(x)-v_q)\right\} \geq M_P - \frac{\epsilon_0}{4}.$$

Let $d_n = (d_{1,n},\ldots,d_{q,n}) \to +\infty$. Thus,

$$\inf_{x\in X} L(x,d_n) = q(d_n) \leq M_P - \epsilon_0.$$

There then exists $x_n \in X$, such that

$$0 \le f(x_n) \le L(x_n, d_n) \le M_P - \frac{\epsilon_0}{2} \tag{7}$$

and

$$0 < \max\{a_1 d_{1,n} g_1(x_n), \ldots, a_q d_{q,n} g_q(x_n)\} \le L(x_n, d_n) \le M_P - \frac{\epsilon_0}{2}. \tag{8}$$

Equation (7) implies

$$\begin{aligned} f(x) + \sum_{j=1}^{q} \bar{y}_j g_j(x) + \sum_{j=1}^{q} \bar{y}_j v_j + \bar{r}\sigma(-g_1(x) - v_1, \ldots, -g_q(x) - v_q) \\ \ge M_P - \frac{\epsilon_0}{4} \quad \forall v \ge 0. \end{aligned} \tag{9}$$

Let $x = x_n$ in (9), $v_{j,n} = -g_j(x_n)$ if $g_j(x_n) \le 0$, and $v_{j,n} = 0$ if $g_j(x_n) > 0$, $j = 1, \ldots, q$. We get

$$f(x_n) + \sum_{j \in J^+(x_n)} \bar{y}_j g_j(x_n) + \bar{r}\sigma(-v^*_{1,n}, \ldots, -v^*_{q,n}) \ge M_P - \frac{\epsilon_0}{4}, \tag{10}$$

where $v^*_{j,n} = g_j(x_n), j \in J^+(x_n)$, and $v^*_{j,n} = 0$ otherwise.

By the assumption on σ, we know that σ is locally Lipschitz around $0 \in R^q$.

Equation (8) and $d_n \to +\infty$ yield that $0 < \max_{j \in J^+(x_n)}\{g_j(x_n)\} \to 0$ as $n \to +\infty$.

Therefore, there exist $\beta > 0$ and $n_0 > 0$ such that for $n \ge n_0$,

$$\sigma(-v^*_{1,n}, \ldots, -v^*_{q,n}) \le \beta \sum_{j=1}^{q} |v^*_j|.$$

Consequently, the facts above and (10) jointly yield

$$\begin{aligned} f(x_n) + \left(\sum_{j \in J^+(x_n)} (|\bar{y}_j| + \bar{r}\beta) \right) \max_{j \in J^+(x_n)} g_j(x_n) \\ \ge f(x_n) + \sum_{j \in J^+(x_n)} (\bar{y}_j + \bar{r}\beta) g_j(x_n) \\ = f(x_n) + \sum_{j \in J^+(x_n)} \bar{y}_j g_j(x_n) + \bar{r}\beta \sum_{j=1}^{m} |v^*_j| \\ \ge f(x_n) + \sum_{j \in J^+(x_n)} \bar{y}_j g_j(x_n) + \bar{r}\sigma(-v^*_{1,n}, \ldots, -v^*_{q,n}) \\ \ge M_P - \frac{\epsilon_0}{4}. \end{aligned}$$

Let $\gamma = \sum_{j=1}^{q} |\bar{y}_j| + q\bar{r}\beta$. Then

$$f(x_n) + \gamma \max_{j \in J^+(x_n)}\{g_j(x_n)\} \ge M_P - \frac{\epsilon_0}{4}. \tag{11}$$

On the other hand, let $\lambda_n = \min_{1 \le j \le q}\{a_j d_{j,n}\}$. It follows from (8) that

$$\lambda_n \max\{g_1(x_n), \ldots, g_q(x_n)\} \le L(x_n, d_n) \le M_P - \epsilon_0/2.$$

Thus,

$$\max_{j\in J^+(x_n)}\{g_j(x_n)\} \le \frac{M_P-\epsilon_0/2}{\lambda_n}.$$

By (11), we have

$$\begin{aligned} M_P-\frac{\epsilon_0}{4} &\le f(x_n)+\frac{\gamma}{\lambda_n}\left(M_P-\frac{\epsilon_0}{2}\right)\\ &\le M_P-\frac{\epsilon_0}{2}+\frac{\gamma}{\lambda_n}\left(M_P-\frac{\epsilon_0}{2}\right), \end{aligned}$$

where the last inequality follows from (7).

Noticing that $\lambda_n \to +\infty$ as $n\to\infty$ and letting $n\to\infty$, we obtain

$$M_P-\frac{\epsilon_0}{4} \le M_P-\frac{\epsilon_0}{2},$$

which is a contradiction. □

PROPOSITION 3.4. *Let function c defining the nonlinear Lagrangian L be continuous. If $M_P = M_N$, then $M_P = M_A$.*

Proof. By the weak duality (3) of the augmented Lagrangian, $M_A \le M_P$. Suppose to the contrary that there exists $\epsilon_0 > 0$ such that

$$M_A = \sup_{(y,r)\in R^q\times(0,+\infty)} \inf_{x\in X} \bar{l}(x,y,r) \le M_P-\epsilon_0.$$

Thus,

$$\inf_{x\in X}\bar{l}(x,y,r) \le M_P-\epsilon_0 \quad \forall (y,r)\in R^q\times(0,+\infty).$$

In particular,

$$\inf_{x\in X}\bar{l}(x,0,r) \le M_P-\epsilon_0 \quad \forall r\in(0,+\infty).$$

Let $r_n\to+\infty$. There then exists $n_0>0$ such that, for $n\ge n_0$ and some $x_n\in X$, $\bar{l}(x_n,0,r_n)\le M_P-\frac{\epsilon_0}{2}$. Thus,

$$f(x_n)+\inf_{v\in R^q_+}\{r_n\sigma(-g_1(x_n)-v_1,\dots,-g_q(x_n)-v_q)\} \le M_P-\frac{\epsilon_0}{2}.$$

Furthermore, there exists $v_n=(v_{1,n},\dots,v_{q,n})\in R^q_+$ such that

$$f(x_n)+r_n\sigma(-g_1(x_n)-v_{1,n},\dots,-g_q(x_n)-v_{q,n}) \le M_P-\frac{\epsilon_0}{4}, \quad n\ge n_0. \tag{12}$$

Noticing that $f(x_n)\ge 0\ \forall n$, we deduce from (12) that

$$\sigma(-g_1(x_n)-v_{1,n},\dots,-g_q(x_n)-v_{q,n}) \le \frac{M_P-\epsilon_0/4}{r_n}.$$

Thus

$$\limsup_{n\to+\infty}\sigma(-g_1(x_n)-v_{1,n},\dots,-g_q(x_n)-v_{q,n})=0.$$

Since σ is a convex function with a unique minimum at 0 with $\sigma(0) = 0$, it follows that

$$g_j(x_n) + v_{j,n} \to 0 \quad \text{as } n \to +\infty, (j = 1, \ldots, q).$$

Let $\epsilon_n = \max_{1 \le j \le q} g_j(x_n)$. Then $\epsilon_n > 0$ and $\epsilon_n \to 0$ as $n \to +\infty$. It follows from (12) and $f(x_n) \ge 0$ that

$$0 \le f(x_n) \le M_P - \frac{\epsilon_0}{4}, \quad n \ge n_0. \tag{13}$$

Without loss of generality, we assume that

$$f(x_n) \to t_0 \ge 0 \quad \text{as } n \to +\infty. \tag{14}$$

The combination of (13) and (14) yields $0 \le t_0 \le M_P - \frac{\epsilon_0}{4}$. Let $d = (d_1, \ldots, d_q) \in R^q_+$. Then, by the monotonicity of c,

$$c(f(x_n), d_1 g_1(x_n), \ldots, d_q g_q(x_n)) \le c(f(x_n), d\epsilon_n, \ldots, d\epsilon_n).$$

Taking the upper limit as $n \to +\infty$ and applying the continuity of c, we obtain

$$\limsup_{n \to +\infty} c(f(x_n), d_1 g_1(x_n), \ldots, d_q g_q(x_n)) \le c(t_0, 0, \ldots, 0) = t_0 \le M_P - \frac{\epsilon_0}{4}.$$

Hence, for each $d \in R^q_+$, $\exists n(d) > 0$ such that

$$c(f(x_{n(d)}), d_1 g_1(x_{n(d)}), \ldots, d_q g_q(x_{n(d)})) \le M_P - \frac{\epsilon_0}{8}.$$

It follows that

$$\inf_{x \in X} c(f(x), d_1 g_1(x), \ldots, d_q g_q(x)) \le M_P - \frac{\epsilon_0}{8}.$$

As $d \in R^q_+$ is arbitrary, we conclude that $M_N \le M_P - \frac{\epsilon_0}{8}$, which contradicts the assumption $M_N = M_P$. The proof is complete. □

The relationships are summarized below between the zero duality properties of the augmented Lagrangian dual problem (D_A), with the perturbation function $\overline{f}(x, u)$ selected as (5), and the nonlinear Lagrangian dual problem (D_N).

THEOREM 3.5. *Consider the problem* (P)*, the nonlinear Lagrangian dual problem* (D_N)*, and the augmented Lagrangian dual problem* (D_A)*. If the function* c *defining the nonlinear Lagrangian* L *is continuous, the perturbation function* $\overline{f}(x, u)$ *defining the augmented Lagrangian is selected as* (5)*, and the augmenting function* σ *is finite, l.s.c., and convex, attaining its minimum* 0 *at* $0 \in R^q$*, then the following two statements are equivalent:*

(i) $M_A = M_P$*;*

(ii) $M_N = M_P$.

The following example verifies Theorem 3.5.

Example 3.1. Consider the problem

$$\begin{array}{ll} \inf & f(x) \\ \text{s.t.} & x \in X,\ g(x) \le 0, \end{array}$$

where $X = [0, +\infty)$, $f(x) = 1/(x+1)$ $\forall x \in X$; $g(x) = x - 1$ if $0 \le x \le 1$; $g(x) = 1/\sqrt{x} - 1/x$ if $1 < x < +\infty$. Then $M_P = 1/2$.

Let $c(y_1, y_2) = \max\{y_1, y_2\}$ $\forall y_1 \geq 0, y_2 \in R^1$. It is easy to check that $M_N = 0$. Hence $M_N < M_P$.

Let

$$\overline{f}(x,u) = f(x) + \delta_{R^1_-}(g(x)+u) + \delta_X(x)$$

be defined as in (5). Let $\sigma(u) = 1/2u^2, u \in R^1$. Then $M_A = 0$. Indeed, by Lemma 3.1,

$$\bar{l}(x,y,r) = f(x) + yg(x) + \inf_{v\geq 0}\{yv + r/2(g(x)+v)^2\} \quad \forall x \in X, y \in R^1, r > 0. \tag{15}$$

By the definition of M_A, for any $\epsilon > 0$, there exist $\bar{y} \in R^1$ and $\bar{r} > 0$ such that

$$M_A < \bar{l}(x,\bar{y},\bar{r}) + \epsilon \quad \forall x \in X. \tag{16}$$

The combination of (15) and (16) yields

$$M_A < f(x) + \bar{y}(g(x)+v) + \bar{r}/2(g(x)+v)^2 + \epsilon \quad \forall x \in X,\ v \geq 0. \tag{17}$$

Setting $v = 0$ in (17) gives us

$$M_A < f(x) + \bar{y}g(x) + \bar{r}/2g^2(x) + \epsilon \quad \forall x \in X. \tag{18}$$

Note that, for any $x \in (1, +\infty)$, (18) becomes

$$M_A < \frac{1}{x+1} + \left(\frac{1}{\sqrt{x}} - \frac{1}{x}\right)\bar{y} + \bar{r}/2\left(\frac{1}{\sqrt{x}} - \frac{1}{x}\right)^2 + \epsilon. \tag{19}$$

Taking the limit in (19) as $x \to +\infty$, we obtain $M_A \leq \epsilon$. By the arbitrariness of $\epsilon > 0$, we deduce that $M_A \leq 0$. However, it is obvious that $M_A \geq 0$. Hence $M_A = 0$. Consequently, $M_A < M_P$. Thus, Theorem 3.5 is verified.

It is worth noting that the following conditions in Theorems 2.3 and 2.6 are not satisfied:

(i) The condition $\lim_{x\in X, \|x\|\to+\infty} \max\{f(x), g(x)\} \to +\infty$ in Theorem 2.3 does not hold.

(ii) $\overline{f}(x,u)$ is not level-bounded in x and locally uniform in u. In fact, for any sufficiently small $\epsilon > 0$, we cannot find a bounded set $D_0 \subset R^1$ such that $\{x \in X : \overline{f}(x,u) \leq 1\} \subset D_0$ holds for all u satisfying $|u| < \epsilon$.

The following examples show that, if the perturbation function is not defined by (5), then Theorem 3.5 may not hold.

Example 3.2. Consider the same problem as in Example 3.1. Then $M_N < M_P$. But if we let $\varphi(x) = f(x)$, if $x \in X_0$, and $\varphi(x) = +\infty$ otherwise. Define $\overline{f}(x,u) = \varphi(x)$; if $x \in X_0$ and $u = 0$, $\overline{f}(x,u) = +\infty$ otherwise. It is then easy to check that $\overline{f}(x,u)$ is a perturbation function, but is different from (5). On the other hand, the augmented Lagrangian $\bar{l}(x,y,r) = f(x)$ $\forall x \in X_0,\ y \in R^1,\ r > 0$, and $\bar{l}(x,y,r) = +\infty,\ x \notin X_0$. Thus $M_A = M_P$.

Example 3.3. Let $p = q = 1$. Let $X = [0, +\infty)$, $f(x) = x,\ x \in X$, and $g(x) = x - 1, x \in X$. Then we have

$$\sigma(u) = |u| \quad \forall u \in R^1,$$
$$\overline{f}(x,u) = \begin{cases} f(x) - u^2 & \text{if } g(x) \leq u, x \in X; \\ +\infty & \text{otherwise.} \end{cases}$$

It is easy to verify that

$$\overline{f}(x,0) = \begin{cases} f(x) & \text{if } x \in X_0 = [0,1]; \\ +\infty & \text{otherwise.} \end{cases}$$

Let us look at the augmented Lagrangian function

$$\bar{l}(x,y,r) = \inf\{f(x) - (v + g_1(x))^2 + r|v + g_1(x)| - y(g_1(x) + v) : v \geq 0\} \equiv -\infty.$$

Thus, (4) does not hold and $M_A = -\infty$. However, $M_P = 0$. It follows that $M_A < M_P$. On the other hand, $M_N = 0$. Hence $M_N = M_P$.

4. A nonlinear Lagrangian method. Let $d \in R^q_+$. Consider the following unconstrained optimization problem (Q_d):

$$\inf_{x \in X} L(x,d),$$

where $L(x,d)$ is a nonlinear Lagrangian function. Under certain conditions, we show the existence of a path of optimal solutions generated by unconstrained optimization problems (Q_{d^k}) (where $\{d^k\} \subset R^q_+$ and $d^k \to +\infty$ as $k \to +\infty$) and show its convergence to the optimal set of (P).

Let S denote the optimal solution set of (P), S_d the optimal solution set of (Q_d), and v_d the optimal value of (Q_d).

LEMMA 4.1 (see [12]). *Let $d \in R^q_+$. If the functions defining* (P) *are l.s.c., then $L(\cdot, d)$ is l.s.c. on X.*

THEOREM 4.2. *Consider the problem* (P). *Let $h(x)$ defined by* (2) *be* 0*-coercive on X if X is unbounded. Then S is nonempty and compact. For each $d \in R^q_+ + e$, S_d is nonempty and compact. Furthermore, for each selection $x_d \in S_d$ as $d \to +\infty$, $\{x_d\}$ is bounded, its limit points belong to S, and $\lim_{d \to +\infty} v_d = M_P$.*

Proof. Let $\overline{x} \in X_0$. By the 0-coercivity and l.s.c. of h,

$$X_1 = \{x \in X_0 : f(x) \leq f(\overline{x})\} = \{x \in X : h(x) \leq f(\overline{x})\} \cap X_0$$

is nonempty and compact. It follows that S is nonempty. In addition, $S \subset X_1$; therefore, S is bounded. As $S = \bigcap_{x \in X_0}[\{x^* \in X : f(x^*) \leq f(x)\} \bigcap X_0]$ is closed by the lower semicontinuity of f, S is nonempty and compact.

Let $h_1(x) = \max\{f(x), [\min_{1 \leq j \leq q} a_j] g(x)\}$. Then

$$L(x,d) \geq \max\{f(x), a_1 d_1 g_1(x), \ldots, a_q d_q g_q(x)\} \geq h_1(x) \quad \forall x \in X,\ d \in R^q_+ + e.$$

It is easy to see that $h_1(x)$ is l.s.c. and 0-coercive. Let $X_2 = \{x \in X : h_1(x) \leq f(\overline{x})\}$. Then X_2 is nonempty and compact. For each $d \in R^q_+ + e$, let $X^d = \{x \in X : L(x,d) \leq L(\overline{x},d)\}$. By Lemma 3.2(I′), we have $X^d = \{x \in X : L(x,d) \leq f(\overline{x})\}$. Moreover, since $L(x,d) \geq h_1(x)\ \forall x \in X$, it follows that $X^d \subseteq X_2$ is nonempty and compact. Hence, S_d is nonempty and bounded. It follows from Lemma 4.1 that $L(\cdot, d)$ is l.s.c. on X. Thus, S_d is closed. So S_d is nonempty and compact for any $d \in R^q_+ + e$. Moreover,

$$S_d \subseteq X^d \subseteq X_2 \quad \forall d \in R^q_+ + e.$$

It follows that, for each selection $x_d \in S_d$, $\{x_d\}$ is bounded. Suppose that x^* is a limit point of $\{x_d\}$, namely, $\exists d^k = (d^k_1, \ldots, d^k_m) \to +\infty$ and $x_{d^k} \to x^*$ as $k \to +\infty$. Arbitrarily fix an $x \in X_0$. Then we have

$$\max\{f(x_{d^k}), a_1 d^k_1 g_1(x_{d^k}), \ldots, a_q d^k_q g_q(x_{d^k})\} \leq L(x_{d^k}, d^k) \leq L(x, d^k) = f(x). \tag{20}$$

Thus,

$$f(x_{d^k}) \le f(x) \tag{21}$$

and

$$\left[\min_{1\le j\le q} a_j\right] \cdot \left[\min_{1\le j\le q} d_j^k\right] \cdot g(x_{d^k}) \le f(x). \tag{22}$$

Equation (22) implies

$$g(x_{d^k}) \le \frac{f(x)}{\left[\min\limits_{1\le j\le q} a_j\right] \cdot \left[\min\limits_{1\le j\le q} d_j^k\right]}.$$

Taking the lower limit and using the lower semicontinuity of g, we have $g(x) \le 0$, i.e., $x \in X_0$. Taking the lower limit in (21) and applying the lower semicontinuity of f, we obtain $f(x^*) \le f(x)$. By the arbitrariness of $x \in X_0$, we conclude that $x^* \in S$.

Furthermore, arbitrarily taking $\{d^k\} \subset R_+^q + e$ with $d^k \to +\infty$ as $k \to +\infty$, suppose that $x_{d^k} \to x^* \in S$. It follows from (20) (setting $x = x^*$) that $f(x_{d^k}) \le v_{d^k} \le f(x^*)$. Therefore,

$$v = f(x^*) \le \liminf_{k\to+\infty} f(x_{d^k}) \le \liminf_{k\to+\infty} v_{d^k}$$

and $\limsup_{k\to+\infty} v_{d^k} \le f(x^*) = M_P$. Consequently, $\lim_{k\to+\infty} v_{d^k} = M_P$. Thus $\lim_{d\to+\infty} v_d = M_P$. □

Remark 4.1. It is clear that if f is 0-coercive on X, then h is also 0-coercive. Theorem 4.2 holds if the 0-coercivity of h is replaced with the 0-coercivity of f.

As a byproduct, we apply Theorem 4.2 to obtain a corollary for the case that (P) is a convex programming problem, which is parallel to [2, Theorem 2.2]. In the following, we assume that f, g_j are finite, l.s.c., and convex functions defined on a nonempty, closed, and convex set $X \subseteq R^p$. Let $F : R^p \to R^1 \bigcup\{+\infty\}$ be an extended real-valued convex function. The *recession function* F^∞ of F is defined by

$$epi(F^\infty) = [epi(F)]^\infty,$$

where $epi(F) = \{(x, r) \in R^p \times R^1 : F(x) \le r\}$ is the epigraph of F. It is known [2] that

$$F^\infty(y) = \inf\left\{\liminf_{k\to+\infty} \frac{F(t_k x_k)}{t_k} : t_k \to +\infty, x_k \to y\right\},$$

where $\{t_k\}$ and $\{x_k\}$ are sequences in R^1 and R^p, respectively.

LEMMA 4.3. *Let f, g_j be finite, l.s.c., and convex functions defined on a nonempty, closed, and convex set X. If the optimal solution set S of* (P) *is nonempty and compact, then $h(x)$ is a finite, l.s.c., convex, and* 0*-coercive function on X.*

Proof. Let us set

$$\hat{f}(x) = \begin{cases} f(x) & \text{if } x \in X; \\ +\infty & \text{otherwise,} \end{cases}$$

$$\hat{g}_j(x) = \begin{cases} g_j(x) & \text{if } x \in X; \\ +\infty & \text{otherwise.} \end{cases}$$

Then (P) is equivalent to the following convex programming problem (P'):

$$\min\{\hat{f}(x) : x \in C\},$$

where $C = \{x \in R^p : \hat{g}_j(x) \le 0, j = 1, \ldots, q\}$.

It follows from the assumptions and [2] that S is nonempty and compact if and only if

$$\hat{f}_\infty(w) \le 0,\ (\hat{g}_j)_\infty(w) \le 0,\ j = 1, \ldots, q, \quad w \in R^p \Rightarrow w = 0. \tag{23}$$

Since S is nonempty and compact, (23) holds.

Now we show by contradiction that h is 0-coercive. Suppose that there exists $\{x_k\} \subset X$ such that $\|x_k\| \to +\infty$ and $h(x_k) \le M$ for some $M > 0$. Then $f(x_k) \le M\ \forall k$ and $g_j(x_k) \le M\ \forall j, k$. Since $\{\frac{x_k}{\|x_k\|}\}$ is bounded, without loss of generality we assume that $w_k = \frac{x_k}{\|x_k\|} \to w$ as $k \to +\infty$. Clearly, $w \ne 0$ since $\|w\| = 1$. It follows from the definition of a recession function that

$$\hat{f}_\infty(w) \le \liminf_{k\to+\infty} \frac{f(\|x_k\|w_k)}{\|x_k\|} \le \lim_{k\to+\infty} \frac{M}{\|x_k\|} = 0, \tag{24}$$

$$(\hat{g}_j)_\infty(w) \le \liminf_{k\to+\infty} \frac{g_j(\|x_k\|w_k)}{\|x_k\|} \le \lim_{k\to+\infty} \frac{M}{\|x_k\|} = 0. \tag{25}$$

Thus, $w \ne 0$, and (24) and (25) contradict (23). □

Remark 4.2. Let f, g_j, X be as in Lemma 4.3. If X is unbounded, then S is nonempty and compact if and only if h is 0-coercive. This can be regarded as a characterization of the nonemptiness and compactness of the optimal solution set S of the constrained convex programming problem (P).

COROLLARY 4.4. *Let X be a nonempty, closed, and convex subset of R^p. Let f, g_j be finite, l.s.c., and convex functions on X. If S is nonempty and compact, then for each $d \in R^q_+ + e$, S_d is nonempty and compact. Furthermore, for each selection $x_d \in S_d$, $\{x_d\}$ is bounded and its limit points belong to S and $\lim_{d\to+\infty} v_d = M_P$.*

Proof. The proof follows from Theorem 4.2 and Lemma 4.3. □

Next we apply Theorem 4.2 to develop a method to seek a so-called ϵ-quasi-solution of (P) when (P) may not have an optimal solution.

Let $\epsilon > 0$. The following various definitions of approximate solutions are cited from [15].

DEFINITION 4.5. *$x^* \in X_0$ is called an ϵ-solution of* (P) *if*

$$f(x^*) \le f(x) + \epsilon \quad \forall x \in X_0.$$

DEFINITION 4.6. *$x^* \in X_0$ is called an ϵ-quasi-solution of* (P) *if*

$$f(x^*) \le f(x) + \epsilon\|x - x^*\| \quad \forall x \in X_0.$$

Remark 4.3. An ϵ-quasi-solution is also a local ϵ-solution. In fact, x^* is an ϵ-solution of f on $\{x \in X_0 : \|x - x^*\| \le 1\}$.

DEFINITION 4.7. *Let $\epsilon > 0$. If $x^* \in X_0$ is both an ϵ-solution and an ϵ-quasi-solution of* (P)*, we say that x^* is a regular ϵ-solution of* (P).

Vavasis [20] gave an algorithm for seeking a local approximate solution via the Ekeland variational principle to a problem that contains only box constraints. Specifically, the following optimization problem (P'') is considered:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & \alpha_i \le x_i \le \beta_i, \quad i = 1, \ldots, p, \end{array}$$

where $\alpha_i, \beta_i, i = 1, \ldots, p$, are real numbers and $x = (x_1, \ldots, x_p)$. The algorithm in [20] attempted to find a feasible solution x^*, such that $\| \bigtriangledown f(x^*)\| \leq \epsilon$, which is a necessary condition for x^* to be an ϵ-quasi-solution of (P''), where $\epsilon > 0$ is a given precision value.

In the following, we give a model algorithm to find an ϵ-quasi-solution by using a nonlinear Lagrangian. Let $\epsilon > 0$ and $x_0 \in X$. Define

$$f^1(x) = f(x) + \epsilon\|x - x_0\|, \quad x \in X.$$

Consider the following optimization problem (P_ϵ):

$$\begin{array}{ll} \min & f^1(x) \\ \text{s.t.} & x \in X,\ g_j(x) \leq 0,\ j = 1, \ldots, q, \end{array}$$

and the following unconstrained optimization problem (Q_d^ϵ):

$$\min \overline{L}(x, d) \text{ s.t. } x \in X,$$

where $\overline{L}(x, d) = c(f^1(x), d_1 g_1(x), \ldots, d_q g_q(x))\ \forall x \in X,\ d = (d_1, \ldots, d_q) \in R_+^q$, and c is defined as in section 2.

Let $\overline{S}_\epsilon$ and $\overline{S}_d^\epsilon$ denote the optimal solution sets of (P_ϵ) and (Q_d^ϵ), respectively. Let $\overline{v}_\epsilon$ and $\overline{v}_d^\epsilon$ denote the optimal values of (P_ϵ) and (Q_d^ϵ), respectively.

THEOREM 4.8. *Let $f(x)$ be 0-coercive on X if X is unbounded. We have the following:*

(i) *$\overline{S}_\epsilon$ is a nonempty and compact set and, for each $d \in R_+^q + e$, $\overline{S}_d^\epsilon$ is a nonempty and compact set.*

(ii) *Let $\overline{x}_d \in \overline{S}_d^\epsilon,\ d \in R_+^q$. Then $\{\overline{x}_d\}$ is bounded, every limit point belongs to $\overline{S}_\epsilon$, and $\lim_{d \to +\infty} \overline{v}_d^\epsilon = \overline{v}_\epsilon$.*

(iii) *Furthermore, any $x^* \in \overline{S}_\epsilon$ is an ϵ-quasi-solution of* (P).

(iv) *If $x_0 \in X_0$, then*

$$f(x^*) \leq f(x_0) - \epsilon\|x_0 - x^*\|. \tag{26}$$

Proof. It is clear that f^1 is 0-coercive on X if X is unbounded. Applying Theorem 4.2 by replacing f with f^1, (P) with (P_ϵ), and (Q_d) with (Q_d^ϵ), we conclude that $\overline{S}_\epsilon$ is nonempty and compact; that for each $d \in R_+^q + e$, $\overline{S}_d^\epsilon$ is nonempty and compact; that for each selection $\overline{x}_d \in \overline{S}_d^\epsilon$, $\{\overline{x}_d\}$ is bounded; and that each limit point of $\{\overline{x}_d\}$ belongs to $\overline{S}_\epsilon$ and $\lim_{d \to +\infty} \overline{v}_d^\epsilon = \overline{v}_\epsilon$. Thus (i) and (ii) hold.

Furthermore, for $x^* \in \overline{S}_\epsilon$, we have

$$f(x^*) + \epsilon\|x^* - x_0\| \leq f(x) + \epsilon\|x - x_0\|\ \forall x \in X_0. \tag{27}$$

It follows that

$$f(x^*) \leq f(x) + \epsilon(\|x - x_0\| - \|x^* - x_0\|) \leq f(x) + \epsilon\|x - x^*\|\ \forall x \in X_0.$$

That is, x^* is an ϵ-quasi-solution of (P). Thus, (iii) holds. Moreover, if $x_0 \in X_0$, then by (27) (taking $x = x_0$), we get (26). The proof is complete. ∎

Remark 4.4. The last assertion (26) tells us that even if we already obtained an ϵ-quasi-solution x_0 of (P), it is still possible to apply Theorem 4.8 to seek a "better" ϵ-quasi-solution x^* of (P) (if the resulting $x^* \neq x_0$).

5. Convergence analysis of the nonlinear Lagrangian method in terms of necessary optimality conditions. In this section, we investigate the convergence of first and second order necessary optimality conditions that are obtained from nonlinear Lagrangian problems. Specifically, we shall consider the following classes of nonlinear Lagrangians:

(i) $L^{\infty}(x,d) = \max\{f(x), d_1 g_1(x), \ldots, d_q g_q(x)\}$, $x \in X$;

(ii) $L^k(x,d) = (f(x)^k + \sum_{j=1}^q d_j^k g_j^+(x)^k)^{1/k}$, $x \in X$, where $2 \le k < \infty$;

(iii) $L^k(x,d)$ is as in (ii) with $0 < k < 2$,

where properties (A) and (B) are satisfied with $a_j = 1, j = 1, \ldots, q$.

Throughout this section, we further assume

(A1) $X = R^p$;

(A2) $\beta = \inf_{x \in R^p} f(x) > 0$;

(A3) f, g_j, $j = 1, \ldots, q$, are $C^{1,1}$, namely, they are differentiable and their gradients are locally Lipschitz; and

(A4) $\max\{f(x), g_1(x), \ldots, g_q(x)\} \to +\infty$ as $\|x\| \to +\infty$.

Let f be a $C^{1,1}$ function. We denote by $\partial^2 f(x)$ the generalized Hessian of f at x; see [11, 23]. It is noted that the set-valued mapping $x \to \partial^2 f(x)$ is upper semicontinuous.

We consider the following type of optimality conditions which were derived in [11, 21]. It is worth noting that in these conditions the multipliers do not depend on the choice of vectors in the tangential subspace of the active constraints.

DEFINITION 5.1. *Let $x^* \in X_0$. The first order necessary condition of* (P) *is said to hold at x^* if there exist λ, $\mu_j \ge 0$, $j \in J(x^*)$, such that*

$$\lambda \bigtriangledown f(x^*) + \sum_{j \in J(x^*)} \mu_j \bigtriangledown g_j(x^*) = 0. \tag{28}$$

The second order necessary condition of (P) *is said to hold at x^* if* (28) *holds and, for any $u^* \in R^p$ satisfying*

$$\bigtriangledown g_j(x^*)^{\top} u^* = 0, \quad j \in J(x^*), \tag{29}$$

there exist $F \in \partial^2 f(x^*)$, $G_j \in \partial^2 g_j(x^*)$, $j \in J(x^*)$, such that

$$u^{*T} \left(\lambda F + \sum_{j \in J(x^*)} \mu_j G_j \right) u^* \ge 0. \tag{30}$$

We need the following lemma.

LEMMA 5.2. *Let $k \in (0, +\infty]$, $z \in X_0$, and $d_n = (d_{1,n}, \ldots, d_{q,n}) (\in R_+^q) \to +\infty$ as $n \to +\infty$. If the sequence $\{x_n\} \subset X$ satisfies $L^k(x_n, d_n) \le f(z)$ $\forall n$, then $\{x_n\}$ is bounded and its limit points belong to X_0.*

Proof. It is known that $\max\{f(x_n), d_{1,n} g_1(x_n), \ldots, d_{q,n} g_q(x_n)\} \le L^k(x_n, d_n)$. Thus,

$$\max\{f(x_n), d_{1,n} g_1(x_n), \ldots, d_{q,n} g_q(x_n)\} \le f(z). \tag{31}$$

Suppose that $\{x_n\}$ is unbounded. Without loss of generality, assume that $\|x_n\| \to +\infty$. By assumption (A4), we get

$$\max\{f(x_n), g_1(x_n), \ldots, g_q(x_n)\} \to +\infty \text{ as } n \to +\infty. \tag{32}$$

Since $d_{j,n} \to +\infty$ as $n \to +\infty$ $(j = 1, \ldots, q)$, we see that $d_{j,n} > 1$ $(j = 1, \ldots, q)$ when n is sufficiently large. Hence, for sufficiently large n,

$$\max\{f(x_n), g_1(x_n), \ldots, g_q(x_n)\} \le \max\{f(x_n), d_{1,n}g_1(x_n), \ldots, d_{q,n}g_q(x_n)\}.$$

This fact, combined with (32), contradicts (31). So the sequence $\{x_n\}$ is bounded.

Now we show that any limit point of $\{x_n\}$ belongs to X_0. Without loss of generality, we assume that $x_n \to x^*$. Suppose that $x^* \notin X_0$. There exists $\gamma_0 > 0$ such that $\max\{g_1(x^*), \ldots, g_q(x^*)\} \ge \gamma_0 > 0$. It follows that $\max\{g_1(x_n), \ldots, g_q(x_n)\} \ge \gamma_0/2$ for sufficiently large n. Moreover, it follows from (31) that

$$\begin{aligned} f(z) &\ge L^k(x_n, d_n) \ge \max\{d_{1,n}g_1(x_n), \ldots, d_{q,n}g_q(x_n)\} \\ &\ge \min_{1\le j\le q}\{d_{j,n}\}\max\{g_1(x_n), \ldots, g_q(x_n)\} \ge \frac{\gamma_0}{2}\min_{1\le j\le q}\{d_{j,n}\}, \end{aligned}$$

which is impossible, as $n \to +\infty$. □

Define

$$J^*(\overline{x}) = \begin{cases} J^+(\overline{x}) \cup J(\overline{x}) & \text{if } k \in (0,2), \\ J(\overline{x}) & \text{if } k \in [2,\infty), \\ J^+(\overline{x}) & \text{if } k = \infty. \end{cases}$$

LEMMA 5.3 (see [22]). *Suppose that $\{\nabla g_j(x)\}_{j\in J^*(x)}$ is linearly independent for any $x \in X_0$ and that $\overline{x}_n \to x^*$ as $n \to +\infty$ and $x^* \in X_0$. Then, for $u^* \in R^p$ satisfying* (29)*, there exists a sequence $\{u_n\} \subset R^p$ such that $\bigtriangledown g_j(\overline{x}_n)^\top u_n = 0$, $j \in J^*(x^*)$, and $u_n \to u^*$.*

As shown in [1, 22], if $x \in X_0$ and $x_n \to x$, then, for sufficiently large n,

$$J(x_n) \subseteq J(x), \quad J^+(x_n) \subseteq J(x). \tag{33}$$

We shall carry out the convergence analysis by considering the following two cases.

Case 1. $2 \le k < +\infty$.

Case 2. $k = +\infty$ or $k \in (0, 2)$.

5.1. Case 1. $2 \le k < +\infty$. When $2 \le k < +\infty$, the nonlinear Lagrangian function $L^k(x, d)$ is $C^{1,1}$. Thus, the first and second order necessary optimality conditions of (Q_{d_n}) can be easily derived.

Let $d_n = (d_{1,n}, \ldots, d_{q,n})(\in R^q_+) \to +\infty$ as $n \to +\infty$.

Let $\overline{x}_n$ be a local minimum of (Q_{d_n}). Thus, the first order necessary condition for $\overline{x}_n$ to be a local minimum of (Q_{d_n}) can be written as $\nabla L^k(\overline{x}_n, d_n) = 0$, or

$$a_n^{\frac{1}{k}-1}\left[f^{k-1}(\overline{x}_n) \bigtriangledown f(\overline{x}_n) + \sum_{j\in J^+(\overline{x}_n)} d_{j,n}^k (g_j^+(\overline{x}_n))^{k-1} \bigtriangledown g_j(\overline{x}_n)\right] = 0, \tag{34}$$

where $a_n = [L^k(\overline{x}_n, d_n)]^k$.

The second order necessary condition is that, for every $u \in R^p$, $u^\top M u \ge 0$ for some $M \in \partial^2 L^k(\overline{x}_n, d_n)$; thus there exist $F_n \in \partial^2 f(\overline{x}_n), G_{j,n} \in \partial^2 g_j(\overline{x}_n), j \in J^+(\overline{x}_n)$, such that

$$\left(\frac{1}{k}-1\right)a_n^{\frac{1}{k}-2}\left[\alpha(n)(\triangledown f(\overline{x}_n)^\top u)^2+\sum_{j\in J^+(\overline{x}_n)}\beta_{j,1}(n)(\triangledown g_j(\overline{x}_n)^\top u)^2\right.$$
$$+\sum_{j\in J^+(\overline{x}_n)}\beta_{j,2}(n)(\triangledown f(\overline{x}_n)^\top u)(\triangledown g_j(\overline{x}_n)^\top u)$$
$$\left.+\sum_{i\in J^+(\overline{x}_n)}\sum_{j\in J^+(\overline{x}_n)}\beta_{j,3}(n)(\triangledown g_i(\overline{x}_n)^\top u)(\triangledown g_j(\overline{x}_n)^\top u)\right]$$
$$+a_n^{\frac{1}{k}-1}(k-1)\left[\xi(n)(\triangledown f(\overline{x}_n)^\top u)^2+\sum_{j\in J(\overline{x}_n)}\eta_{j,1}(n)[(\triangledown g_j(\overline{x}_n)^\top u)^+]^2\right.$$
$$\left.+\sum_{j\in J^+(\overline{x}_n)}\eta_{j,2}(n)(\triangledown g_j(\overline{x}_n)^\top u)^2\right]$$
$$+a_n^{\frac{1}{k}-1}u^T\left[f^{k-1}(\overline{x}_n)F_n+\sum_{j\in J^+(\overline{x}_n)}d_{j,n}^k(g_j^+(\overline{x}_n))^{k-1}G_{j,n}\right]u\geq 0, \tag{35}$$

where $\alpha(n)$, $\beta_{i,1}(n)$, $\beta_{i,2}(n)$, $\beta_{i,3}(n)$, $\xi(n)$, $\eta_{i,1}(n)$, and $\eta_{i,2}(n)$ are real numbers.

We have the following convergence result.

THEOREM 5.4. *Suppose that $\{\nabla g_j(x)\}_{j\in J(x)}$ is linearly independent for any $x\in X_0$. Let $2\leq k<+\infty$ and $d_n\in R_+^q$ be such that $d_n\to+\infty$. Let $\overline{x}_n$ be generated by some descent method for (Q_{d_n}) starting from a point $z\in X_0$ and $\overline{x}_n$ satisfy first order necessary condition* (34) *and second order necessary condition* (35). *Then $\{\overline{x}_n\}$ is bounded and every limit point of $\{\overline{x}_n\}$ is a point of X_0 satisfying first order necessary optimality condition* (28) *and second order necessary optimality condition* (30) *of* (P).

Proof. It follows from Lemma 5.2 that $\{\overline{x}_n\}$ is bounded and every limit point of $\{\overline{x}_n\}$ belongs to X_0. Without loss of generality, we assume that $\overline{x}_n\to x^*$. Let

$$a_n=[L^k(\overline{x}_n,d_n)]^k>0;\quad b_n=a_n^{\frac{1}{k}-1}\left(f^{k-1}(\overline{x}_n)+\sum_{j\in J^+(\overline{x}_n)}d_{j,n}^k g_j^+(\overline{x}_n)^{k-1}\right)>0.$$

Thus,

$$\frac{a_n^{\frac{1}{k}-1}f^{k-1}(\overline{x}_n)}{b_n}+\sum_{j\in J^+(\overline{x}_n)}\frac{a_n^{\frac{1}{k}-1}d_{j,n}^k(g_j^+(\overline{x}_n))^{k-1}}{b_n}=1.$$

Without loss of generality, we assume that

$$\frac{a_n^{\frac{1}{k}-1}f^{k-1}(\overline{x}_n)}{b_n}\to\lambda, \tag{36}$$

$$\frac{a_n^{\frac{1}{k}-1}d_{j,n}^k(g_j^+(\overline{x}_n))^{k-1}}{b_n}\to\mu_j,\quad j\in J(x^*). \tag{37}$$

Then by (33),

$$\lambda\geq 0,\ \mu_j\geq 0,\ j\in J(x^*),\ \text{and}\ \lambda+\sum_{j\in J(x^*)}\mu_j=1. \tag{38}$$

Dividing (34) by b_n and taking the limit, we obtain

$$\lambda \bigtriangledown f(x^*) + \sum_{j \in J(x^*)} \mu_j \bigtriangledown g_j(x^*) = 0.$$

Since $\{\bigtriangledown g_j(x^*)\}_{j \in J(x^*)}$ is linearly independent, it follows that $\lambda > 0$.

By Lemma 5.3, we deduce that, for any $u^* \in R^p$ satisfying (29), we can find $u_n \in R^p$ such that

$$\bigtriangledown g_j(\overline{x}_n)^\top u_n = 0, \ j \in J(x^*) \tag{39}$$

and

$$u_n \to u^*. \tag{40}$$

Furthermore, for every u_n satisfying (39) and (40), we can find $F_n \in \partial^2 f(\overline{x}_n), G_{j,n} \in \partial^2 g_j(\overline{x}_n), j \in J^+(\overline{x}_n)$, such that (35) holds with u replaced by u_n.

Substituting (39) into (34), we get

$$\bigtriangledown f(\overline{x}_n)^\top u_n = 0. \tag{41}$$

Substituting (39)–(41) into (35), we have

$$a_n^{\frac{1}{k}-1} u_n^\top \left(f^{k-1}(\overline{x}_n) F_n + \sum_{j \in J^+(\overline{x}_n)} d_{j,n}^k (g_j^+(\overline{x}_n))^{k-1} G_{j,n} \right) u_n \geq 0. \tag{42}$$

Since $\overline{x}_n \to x^*$ as $n \to \infty$, $\partial^2 f(\cdot), \partial^2 g_j(\cdot)$ are upper semicontinuous at x^* and $\partial^2 f(x^*), \partial^2 g_j(x^*)$ are compact, without loss of generality we can assume that

$$F_n \to F \in \partial^2 f(x^*), \ G_{j,n} \to G_j \in \partial^2 g_j(x^*), \ j \in J(x^*). \tag{43}$$

Dividing (42) by b_n and taking the limit, applying (36), (37), (40), and (43), we obtain

$$u^{*T} \left(\lambda F + \sum_{j \in J(x^*)} \mu_j G_j \right) u^* \geq 0 \ \text{ and } \lambda > 0.$$

5.2. Case 2. $k = +\infty$ or $k \in (0, 2)$. When $k = +\infty$, problem (Q_{d_n}) is a minimax optimization problem and thus a convex composite optimization problem. However, the second order necessary conditions for a convex composite optimization problem given in [4, 23] are not applicable, as the multipliers depend on the choice of the vector in the tangential subspace of the active constraints. When $k \in (0, 2)$, function $g_j^+(x)^k$ and thus $L^k(x, d)$ is not $C^{1,1}$. Thus, the existing optimality conditions in the literature are not applicable. However, we are able to derive optimality conditions for (Q_{d_n}) by applying the smooth approximate variational principle, which is due to Borwein and Preiss [6] (see also [8, Theorem 5.2]).

LEMMA 5.5 (approximate smooth variational principle [8, Theorem 5.2]). *Let X be a Hilbert space. Let $g : X \to (-\infty, +\infty]$ be l.s.c. and bounded below with $dom(g) \neq \emptyset$. Let $\overline{x}$ be a point such that $g(\overline{x}) < \inf_{x \in X} g(x) + \epsilon$, where $\epsilon > 0$. Then, for any $\lambda > 0$, there exist y_ϵ, z_ϵ with $\|y_\epsilon - z_\epsilon\| < \lambda, \|z_\epsilon - \overline{x}\| < \lambda$, $g(y_\epsilon) < \inf_{x \in X} g(x) + \epsilon$, and having the property that the function $y \to g(y) + (\epsilon/\lambda^2)\|y - z_\epsilon\|^2$ has a unique minimum over X at $y = y_\epsilon$.*

Remark 5.1. If the Hilbert space X in Lemma 5.5 is replaced with a nonempty and closed subset X_1, then the conclusion also holds. As a matter of fact, if $g : X_1 \to (-\infty, +\infty]$ is l.s.c. and bounded below on X_1, we can define a function $\overline{g} : X \to (-\infty, +\infty]$ as follows: $\overline{g}(x) = g(x)$ if $x \in X_1$ and $\overline{g}(x) = +\infty$ otherwise. It is easy to verify that $\overline{g}$ is l.s.c. and bounded below on X. Applying Lemma 5.3 to $\overline{g}$, the conclusion for g follows.

Next we present first and second order necessary conditions for $\overline{x}$ to be a local minimum of $L^k(x, d)$ under the linear independence assumption. The proof is given in the appendix.

PROPOSITION 5.6. *Let $k \in (0, 2)$ or $k = +\infty$. Let $\overline{x}$ be a local minimum of $L^k(x, d)$ and $\{\bigtriangledown g_j(\overline{x})\}_{j \in J^*(\overline{x})}$ be linearly independent. Then there exist $\lambda > 0$, $\mu_j \geq 0$, $j \in J^*(\overline{x})$, with $\lambda + \sum_{j \in J^*(\overline{x})} \mu_j = 1$ such that*

$$\lambda \bigtriangledown f(\overline{x}) + \sum_{j \in J^*(\overline{x})} \mu_j \bigtriangledown g_j(\overline{x}) = 0.$$

Furthermore, for each $u \in R^p$ satisfying

$$\bigtriangledown g_j(\overline{x})^\top u = 0, \quad j \in J^*(\overline{x}), \tag{44}$$

there exist $F \in \partial^2 f(\overline{x})$, $G_j \in \partial^2 g_j(\overline{x})$, $j \in J^*(\overline{x})$, such that

$$u^T \left(\lambda F + \sum_{j \in J^*(\overline{x})} \mu_j G_j \right) u \geq 0.$$

THEOREM 5.7. *Suppose that $\{\nabla g_j(x)\}_{j \in J^*(x)}$ is linearly independent for any $x \in X_0$. Let $k \in (0, 2)$ or $k = +\infty$. Let $d_n (\in R^q_+) \to +\infty$ as $n \to +\infty$. Let $\overline{x}_n$ be generated by some descent method for (Q_{d_n}) starting from a point $z \in X_0$. Then $\{\overline{x}_n\}$ is bounded and every limit point of $\{\overline{x}_n\}$ is a point of X_0 satisfying first order necessary condition* (28) *and second order necessary condition* (30) *of* (P)*, respectively.*

Proof. It follows from Lemma 5.2 that $\{\overline{x}_n\}$ is bounded and every limit point of $\{\overline{x}_n\}$ belongs to X_0. Without loss of generality, suppose that $\overline{x}_n \to x^* \in X_0$ and that $J^+(\overline{x}_n) \cup J(\overline{x}_n) \subset J(x^*)$ for sufficiently large n. That $\{\bigtriangledown g_j(x^*)\}_{j \in J(x^*)}$ is linearly independent implies that $\{\bigtriangledown g_j(\overline{x}_n)\}_{j \in J^+(\overline{x}_n) \cup J(\overline{x}_n)}$ is linearly independent when n is sufficiently large. In other words, the assumptions in Proposition 5.6 hold (with $\overline{x}$ replaced by $\overline{x}_n$) when n is sufficiently large. Thus, we assume that $\{\bigtriangledown g_j(\overline{x}_n)\}_{j \in J^+(\overline{x}_n) \cup J(\overline{x}_n)}$ is linearly independent for all n.

The first order necessary optimality conditions in Proposition 5.6 can be written as

$$\lambda_n \bigtriangledown f(\overline{x}_n) + \sum_{j \in J(x^*)} \mu_{j,n} \bigtriangledown g_j(\overline{x}_n) = 0, \tag{45}$$

where $\lambda_n > 0, \mu_{j,n} \geq 0, j \in J(x^*)$, with $\mu_{j,n} = 0 \ \forall j \in J(x^*) \backslash J(\overline{x}_n)$ and $\lambda_n + \sum_{j \in J(x^*)} \mu_j = 1$. Without loss of generality, we assume that $\lambda_n \to \lambda, \mu_{j,n} \to \mu_j, j \in J(x^*)$, as $n \to +\infty$. Taking the limit in (45) gives us

$$\lambda \bigtriangledown f(x^*) + \sum_{j \in J(x^*)} \mu_j \bigtriangledown g_j(x^*) = 0.$$

By the linear independence of $\{\bigtriangledown g_j(x^*)\}_{j \in J(x^*)}$, we see that $\lambda > 0$. That is, (28) holds.

Let $u^* \in R^p$ satisfy (29). Since $\{\bigtriangledown g_j(x^*)\}_{j\in J(x^*)}$ is linearly independent and $\overline{x}_n \to x^*$, by Lemma 5.3, we obtain $\overline{u}_n \in R^p$ such that

$$\bigtriangledown g_j(\overline{x}_n)^T \overline{u}_n = 0, \qquad j \in J(x^*), \tag{46}$$

and $\overline{u}_n \to u^*$.

Thus, if $\overline{x}_n$ satisfies any one of the second order necessary conditions in Proposition 5.6, then, for every $\overline{u}_n$ satisfying (46), there exist $F_n \in \partial^2 f(\overline{x}_n), G_{j,n} \in \partial^2 g_j(\overline{x}_n), j \in J(x^*)$,

$$\overline{u}_n{}^T \left(\lambda_n F_n + \sum_{j\in J(x^*)} \mu_{j,n} G_{j,n} \right) \overline{u}_n \geq 0, \tag{47}$$

where $\lambda_n, \mu_{j,n}$ are as in (45).

By the upper semicontinuity of $\partial^2 f(\cdot), \partial^2 g_j(\cdot)$ and the nonemptiness and compactness of $\partial^2 f(x^*), \partial^2 g_j(x^*)(j = 1, \ldots, q)$, without loss of generality we assume that

$$F_n \to F \in \partial^2 f(x^*),\ G_{j,n} \to G_j \in \partial^2 g_j(x^*),\ j \in J(x^*),$$

as $n \to +\infty$. Taking the limit in (47), we get

$$u^{*T} \left(\lambda F + \sum_{j\in J(x^*)} \mu_j G_j \right) u^* \geq 0,$$

where $\lambda > 0$. Thus, (30) follows. The proof is complete. □

Appendix. Proof of Proposition 5.6. We consider the following two cases.

Case 1. $k = \infty$. In this case, $J^*(\overline{x}) = J^+(\overline{x})$. Since $\overline{x} \in X, f(\overline{x}) > 0$. Thus, it follows that $L^\infty(\overline{x}, d) = \max\{f(\overline{x}), d_j g_j(\overline{x})\}_{j\in J^+(\overline{x})}$. Since $\overline{x}$ is a local minimum of $L^\infty(x, d)$, there exists $\delta > 0$ such that

$$L^\infty(\overline{x}, d) \leq L^\infty(x, d) = \max\{f(x), d_j g_j(x)\}_{j\in J^+(\overline{x})} \quad \forall x \in U_\delta,$$

where $U_\delta = \{x \in R^p : \|x - \overline{x}\| \leq \delta\}$ $(X = R^p)$.

Let $m > 0$ be an integer and

$$s_m(x) = \left[f^m(x) + \sum_{j\in J^+(\overline{x})} d_j^m g_j^m(x) \right]^{\frac{1}{m}}, \ x \in U_\delta,$$
$$\epsilon_m = \left[(q+1)^{\frac{1}{m}} - 1 \right] L^\infty(\overline{x}, d).$$

Then $0 \leq s_m(x) - L^\infty(x, d)$ $\forall x \in U_\delta$ and $s_m(\overline{x}) \leq [(q+1)^{\frac{1}{m}}]L^\infty(\overline{x}, d)$. Thus,

$$\begin{aligned} s_m(\overline{x}) &\leq L^\infty(\overline{x}, d) + [(q+1)^{\frac{1}{m}} - 1]L^\infty(\overline{x}, d) \\ &\leq L^\infty(x, d) + [(q+1)^{\frac{1}{m}} - 1]L^\infty(\overline{x}, d) \\ &\leq s_m(x) + [(q+1)^{\frac{1}{m}} - 1]L^\infty(\overline{x}, d) \\ &= s_m(x) + \epsilon_m \quad \forall x \in U_\delta. \end{aligned}$$

Note that $\epsilon_m \downarrow 0$ as $m \to +\infty$. Without loss of generality, we assume that $2\epsilon_m^{1/4} < \delta\ \forall m$. Applying Lemma 5.5 by setting $\lambda = \epsilon_m^{1/4}$, we obtain $\overline{x}'_m, \overline{x}''_m \in U_\delta$ such that

$$\|\overline{x}'_m - \overline{x}''_m\| < \epsilon_m^{1/4} \quad \text{and} \quad \|\overline{x}''_m - \overline{x}\| < \epsilon_m^{1/4}$$

and $\overline{x}'_m$ is a unique minimum of the problem

$$\min\ v_m(x) = s_m(x) + \epsilon_m^{1/2}\|x - \overline{x}''_m\|^2 \quad \text{s.t. } x \in U_\delta. \tag{48}$$

Note that $\|\overline{x}'_m - \overline{x}\| \le \|\overline{x}'_m - \overline{x}''_m\| + \|\overline{x}''_m - \overline{x}\| \le 2\epsilon_m^{1/4} < \delta$. It follows that $\overline{x}'_m \in \text{int}U_\delta$. Applying the first order necessary optimality condition to problem (48), we get $\bigtriangledown v_m(\overline{x}'_m) = 0$. That is,

(49)

$$a_m^{\frac{1}{m}-1}\left[f^{m-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m) + \sum_{j\in J^+(\overline{x})} d_j^m g_j^{m-1}(\overline{x}'_m) \bigtriangledown g_j(\overline{x}'_m)\right] + 2\epsilon_m^{1/2}(\overline{x}'_m - \overline{x}''_m) = 0,$$

where $a_m = [s_m(\overline{x}'_m)]^m$.

Let

$$b_m = a_m^{\frac{1}{m}-1}\left[f^{m-1}(\overline{x}'_m) + \sum_{j\in J^+(\overline{x})} d_j^m g_j^{m-1}(\overline{x}'_m)\right].$$

It is clear that there exists $\alpha > 0$ such that $b_m \ge \alpha > 0\ \forall m$. Without loss of generality, we can assume that

$$\frac{a_m^{\frac{1}{m}-1} f^{m-1}(\overline{x}'_m)}{b_m} \to \lambda,\ \frac{a_m^{\frac{1}{m}-1} d_j^m g_j^{m-1}(\overline{x}'_m)}{b_m} \to \mu_j,\ j \in J^+(\overline{x}). \tag{50}$$

Thus

$$\lambda \ge 0,\ \mu_j \ge 0,\ j \in J^+(\overline{x}),\ \text{ and } \lambda + \sum_{j\in J^+(\overline{x})} \mu_j = 1.$$

Dividing (50) by b_m and taking the limit as $m \to +\infty$, it follows from (50) that

$$\lambda \bigtriangledown f(\overline{x}) + \sum_{j\in J^+(\overline{x})} \mu_j \bigtriangledown g_j(\overline{x}) = 0.$$

Since $\{\bigtriangledown g_j(\overline{x})\}_{j\in J^+(\overline{x})}$ is linearly independent, it follows that $\lambda > 0$.

Now we apply the second order necessary optimality condition to (48). For any $u \in R^p$, there exists $V_m \in \partial^2 v_m(\overline{x}'_m)$ such that $u^\top V_m u \ge 0$. That is, there exist $F_m \in \partial^2 f(\overline{x}'_m)$ and $G_{j,m} \in \partial^2 g_j(\overline{x}'_m),\ j \in J^+(\overline{x})$, such that

$$\begin{aligned}
&\left(\frac{1}{m} - 1\right) a_m^{\frac{1}{m}-2}\left(f^{m-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u + \sum_{j\in J^+(\overline{x})} d_j^m g_j^{m-1}(\overline{x}'_m) \bigtriangledown g_j(\overline{x}'_m)^\top u\right)^2 \\
&+(m-1)a_m^{\frac{1}{m}-1}\left(f^{m-2}(\overline{x}'_m)(\bigtriangledown f(\overline{x}'_m)^\top u)^2 + \sum_{j\in J^+(\overline{x})} d_j^m g_j^{m-2}(\overline{x}'_m)(\bigtriangledown g_j(\overline{x}'_m)^\top u)^2\right) \\
&+a_m^{\frac{1}{m}-1} u^\top\left(f^{m-1}(\overline{x}'_m)F_m + \sum_{j\in J^+(\overline{x})} d_j^m (g_j^+(\overline{x}'_m))^{m-1} G_{j,m}\right) u + 2\epsilon_m^{1/2} u^T u \ge 0.
\end{aligned} \tag{51}$$

Since $\{\bigtriangledown g_j(\overline{x})\}_{j\in J^+(\overline{x})}$ is linearly independent and $\overline{x}'_m \to \overline{x}$, from Lemma 5.3, for any $\overline{u} \in R^p$ satisfying (44), there exists a sequence $\{u_m\}$, such that

$$\bigtriangledown g_j(\overline{x}'_m)^\top u_m = 0, \quad j \in J^+(\overline{x}), \tag{52}$$

and $u_m \to \overline{u}$.

The combination of (51) (setting $u = u_m$) and (52) yields

$$\left(\frac{1}{m}-1\right) a_m^{\frac{1}{m}-2}\left(f^{m-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 + (m-1)a_m^{\frac{1}{m}-1} f^{m-2}(\overline{x}'_m)(\bigtriangledown f(\overline{x}'_m)^\top u_m)^2$$
$$+a_m^{\frac{1}{m}-1} u_m^T \left[f^{m-1}(\overline{x}'_m)F_m + \sum_{j\in J^+(\overline{x})} d_j^m g_j^{m-1}(\overline{x}'_m) G_{j,m}\right] u_m + 2\epsilon_m^{1/2} u_m^T u_m \geq 0. \tag{53}$$

From (50) (setting $u = u_m$) and (52), we have

$$\left|\left(\frac{1}{m}-1\right) a_m^{\frac{1}{m}-2}\left(f^{m-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 / b_m\right|$$
$$= 4\epsilon_m[(\overline{x}'_m - \overline{x}''_m)^\top u_m]^2 \left(1-\frac{1}{m}\right)/(a_m^{1/m} b_m) \leq \frac{4\epsilon_m^{\frac{3}{2}}}{(\alpha\beta)}\|u_m\|^2.$$

Therefore,

$$\left(\frac{1}{m}-1\right) a_m^{1/m-2}\left(f^{m-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 / b_m \to 0 \quad \text{as } m \to \infty.$$

The first formula in (50) guarantees that, when m is sufficiently large,

$$a_m^{\frac{1}{m}-1} f^{m-1}(\overline{x}'_m)/b_m > \lambda/2 > 0.$$

Thus, the combination of (50) (letting $u = u_m$) and (52) also yields

$$(m-1)a_m^{\frac{1}{m}-1} f^{m-2}(\overline{x}'_m)\left(\bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 / b_m$$
$$= \frac{1}{f(\overline{x}'_m)}(m-1)4\epsilon_m[(\overline{x}'_m - \overline{x}''_m)^\top u_m]^2 / [(a_m^{\frac{1}{m}-1} f^{m-1}(\overline{x}'_m)/b_m) b_m^2]$$
$$\leq \frac{1}{\beta\alpha^2}\|u_m\|^2 4(m-1)\epsilon_m^{3/2}/(\lambda/2).$$

Noting that

$$4(m-1)\epsilon_m^{3/2} \leq 4(m-1)\left((q+1)^{1/m}-1\right)^{3/2}[L^\infty(\overline{x}, d)]^{3/2},$$

we deduce that

$$(m-1)a_m^{\frac{1}{m}-1} f^{m-2}(\overline{x}'_m)\left(\bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 / b_m \to 0 \quad \text{as } m \to \infty.$$

Since $\partial^2 f(\cdot)$, $\partial^2 g_j(\cdot)$ are upper semicontinuous at $\overline{x}$ and $\partial^2 f(\overline{x}), \partial^2 g_j(\overline{x})$ are nonempty and compact, we obtain $F \in \partial^2 f(\overline{x})$, $G_j \in \partial^2 g_j(\overline{x})$, $j \in J^+(\overline{x})$, such that

$$F_m \to F,\ G_m \to G,\ j \in J^+(\overline{x}) \ \text{ as } m \to \infty.$$

Thus, dividing (53) by b_m and taking the limit, we have

$$\overline{u}^T \left(\lambda F + \sum_{j \in J^+(\overline{x})} \mu_j G_j \right) \overline{u} \geq 0 \quad \text{and} \quad \lambda > 0.$$

Case 2. $k \in (0,2)$. In this case, $J^*(\overline{x}) = J^+(\overline{x}) \bigcup J(\overline{x})$. Since $\overline{x}$ is a local minimum of $L^k(x,d)$, there exists $\delta > 0$ such that $L^k(\overline{x},d) \leq L^k(x,d)\ \forall x \in U_\delta$. Then

$$\left(f^k(\overline{x}) + \sum_{j \in J^+(\overline{x}) \cup J(\overline{x})} d_j^k g_j^{+k}(\overline{x}) \right)^{1/k} \leq \left(f^k(x) + \sum_{j \in J^+(\overline{x}) \cup J(\overline{x})} d_j^k g_j^{+k}(x) \right)^{1/k}.$$

Let

$$t_m(x) = \left(f^k(x) + \frac{1}{2^k} \sum_{j \in J^+(\overline{x}) \cup J(\overline{x})} \left(d_j g_j(x) + \sqrt{d_j^2 g_j^2(x) + 1/m} \right)^k \right)^{1/k}.$$

It is not hard to prove that $0 \leq t_m(\overline{x}) - L^k(\overline{x},d) \leq \epsilon_m$ and $L^k(x,d) \leq t_m(x)\ \forall x \in U_\delta$, where

$$\epsilon_m = \begin{cases} \frac{q}{k} L^k(\overline{x},d)^{\frac{1}{k}-1} \frac{1}{m^{k/2}} & \text{if } k \in (0,1]; \\ \frac{1}{2\sqrt{m}} q^{1/k} & \text{if } k \in (1,2). \end{cases}$$

Thus,

$$t_m(\overline{x}) \leq L^k(\overline{x},d) + \epsilon_m \leq L^k(x,d) + \epsilon_m \leq t_m(x) + \epsilon_m \quad \forall x \in U_\delta.$$

Since $\epsilon_m \downarrow 0$ as $m \to +\infty$, without loss of generality we assume that $2\epsilon_m^{1/4} < \delta\ \forall m$. Applying Lemma 5.5 by setting $\lambda = \epsilon_m^{1/4}$, there exist $\overline{x}'_m, \overline{x}''_m \in U_m$ with $\|\overline{x}'_m - \overline{x}''_m\| < \epsilon_m^{1/4}$, and $\|\overline{x}''_m - \overline{x}\| < \epsilon_m^{1/4}$, such that $\overline{x}'_m$ is the unique minimum of the optimization problem

$$\min\ w_m(x) = t_m(x) + \epsilon_m^{1/2} \|x - \overline{x}''_m\|^2 \quad \text{s.t. } x \in U_\delta. \tag{54}$$

Applying the first order necessary optimality condition to $w_m(x)$ and noticing that $\overline{x}'_m \in \text{int}U_\delta$, we have $\bigtriangledown w_m(\overline{x}'_m) = 0$. That is,

$$\begin{aligned} a_m^{\frac{1}{k}-1} \Bigg(& f^{k-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m) \\ & + \frac{1}{2^k} \sum_{j \in J^+(\overline{x}) \cup J(\overline{x})} d_j c_m^{k-1} (1 + d_j g_j(\overline{x}'_m)(d_j^2 g_j^2(\overline{x}'_m) + 1/m)^{-1/2}) \bigtriangledown g_j(\overline{x}'_m) \Bigg) \\ & + \epsilon_m^{1/2} (\overline{x}'_m - \overline{x}''_m) = 0, \end{aligned} \tag{55}$$

where

$$a_m = (t_m(\overline{x}'_m))^k;\ c_m = d_j g_j(\overline{x}'_m) + \sqrt{d_j^2 g_j^2(\overline{x}'_m) + 1/m}.$$

Let

$$b_m = a_m^{\frac{1}{k}-1}\left[f^{k-1}(\overline{x}'_m) + \frac{1}{2^k}\sum_{j\in J^+(\overline{x})\cup J(\overline{x})} d_j c_m^{k-1}\left(1 + d_j g_j(\overline{x}'_m)\left(d_j^2 g_j^2(\overline{x}'_m) + \frac{1}{m}\right)^{-1/2}\right)\right].$$

Without loss of generality, we assume that

$$\begin{aligned} &\frac{a_m^{\frac{1}{k}-1} f^{k-1}(\overline{x}'_m)}{b_m} \to \lambda, \\ &c_{j,m}/b_m \to \mu_j, \quad j \in J^+(\overline{x}) \cup J(\overline{x}), \end{aligned} \tag{56}$$

where, for $j \in J^+(\overline{x}) \cup J(\overline{x})$,

$$c_{j,m} = \frac{a_m^{\frac{1}{k}-1}}{2^k} d_j c_m^{k-1}\left(1 + d_j g_j(\overline{x}'_m)\left(d_j^2 g_j^2(\overline{x}'_m) + \frac{1}{m}\right)^{-1/2}\right).$$

It is easy to see that $\mu_j = 0, j \in J(\overline{x})$, if $k > 1$. Thus we obtain $\lambda \geq 0, \mu_j \geq 0$ with $\lambda + \sum_{j\in J^*(\overline{x})} \mu_j = 1$.

Dividing (55) by b_m and taking the limit, we get

$$\lambda \bigtriangledown f(\overline{x}) + \sum_{j\in J^+(\overline{x})\cup J(\overline{x})} \mu_j \bigtriangledown g_j(\overline{x}) = 0.$$

Applying the second order necessary optimality condition to (54), we know that, for every $u \in R^p$, there exist $F_m \in \partial^2 f(\overline{x}'_m), G_{j,m} \in \partial^2 g_j(\overline{x}'_m), j \in J^+(\overline{x}) \cup J(\overline{x})$ such that

$$\begin{aligned} &\left(\frac{1}{k} - 1\right) a_m^{\frac{1}{k}-2}\left(f^{k-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u + \sum_{j\in J^+(\overline{x})\cup J(\overline{x})} \alpha_j(m) \bigtriangledown g_j(\overline{x}'_m)^\top u\right)^2 \\ &+ a_m^{\frac{1}{k}-1}\left((k-1) f^{k-2}(\overline{x}'_m)(\bigtriangledown f(\overline{x}'_m)^\top u)^2 + \sum_{j\in J^+(\overline{x})\cup J(\overline{x})} \theta_j(m)(\bigtriangledown g_j(\overline{x}'_m)^\top u)^2\right) \\ &+ a_m^{\frac{1}{k}-1} u^\top \left(f^{k-1}(\overline{x}'_m) F_m + \frac{1}{2^k}\sum_{j\in J^+(\overline{x})\cup J(\overline{x})} d_j \left[d_j g_j(\overline{x}'_m) + \sqrt{d_j^2 g_j^2(\overline{x}'_m) + \frac{1}{m}}\right]^{k-1}\right. \\ &\left.\left(1 + d_j g_j(\overline{x}'_m)\sqrt{d_i^2 g_j^2(\overline{x}'_m) + \frac{1}{m}}\right) G_{j,m}\right) u \geq 0, \end{aligned} \tag{57}$$

where $\alpha_j(n), \theta_j(n)$ are real numbers. Since $\{\bigtriangledown g_j(\overline{x})\}_{j\in J^*(\overline{x})}$ is linearly independent, i.e., $\{\bigtriangledown g_j(\overline{x})\}_{j\in J^+(\overline{x})\cup J(\overline{x})}$ is linearly independent, and $\overline{x}'_m \to \overline{x}$, by Lemma 5.3, we conclude that, for every $\overline{u} \in R^p$ satisfying (44), there exists $u_m \in R^p$, such that

$$\bigtriangledown g_j(\overline{x}'_m)^\top u_m = 0, \quad j \in J^*(\overline{x}), \tag{58}$$

and $u_m \to \overline{u}$.

Furthermore, for every u_m satisfying (58), we obtain $F_m \in \partial^2 f(\overline{x}'_m)$, $G_{j,m} \in \partial^2 g_j(\overline{x}'_m)$, $j \in J^+(\overline{x}) \bigcup J(\overline{x})$, such that (57) holds (with u replaced by u_m).

The combination of (58) and (55) gives us

$$a_m^{\frac{1}{k}-1} f^{k-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m = -\epsilon_m^{\frac{1}{2}}(\overline{x}'_m - \overline{x}''_m)^\top u_m.$$

Thus

$$\left|\left(\frac{1}{k}-1\right) a_m^{\frac{1}{k}-2} \left(f^{k-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2\right| \le \frac{1}{q^*}\epsilon_m^{\frac{3}{4}}\|u_m\|^2$$

and

$$\left|(k-1)a_m^{\frac{1}{k}-1} f^{k-2}(\overline{x}'_m)(\bigtriangledown f(\overline{x}'_m)^\top u_m)^2\right| \le \frac{1-k}{q^*}\epsilon_m^{\frac{3}{4}}\|u_m\|^2.$$

Noting that $b_m \ge 1$, we obtain, as $m \to +\infty$,

$$\frac{1}{b_m}\left(\frac{1}{k}-1\right) a_m^{\frac{1}{k}-2} \left(f^{k-1}(\overline{x}'_m) \bigtriangledown f(\overline{x}'_m)^\top u_m\right)^2 \to 0, \tag{59}$$

$$\frac{1}{b_m}(k-1)a_m^{\frac{1}{k}-1} f^{k-2}(\overline{x}'_m)(\bigtriangledown f(\overline{x}'_m)^\top u_m)^2 \to 0. \tag{60}$$

By the upper semicontinuity of $x \to \partial^2 f(x),\ x \to \partial^2 g_j(x)(j = 1, \ldots, q)$ and the nonemptiness and compactness of $\partial^2 f(\overline{x})$ and $\partial^2 g_j(\overline{x})$, without loss of generality we can assume that $F_m \to F \in \partial^2 f(\overline{x}),\ G_{j,m} \to G_j \in \partial^2 g_j(\overline{x}),\ j \in J^+(\overline{x}) \cup J(\overline{x})$.

Letting $u = u_m$ in (57) and substituting (58) into it, dividing (57) by b_m and taking the limit, and applying (56), (59), and (60), we obtain

$$\overline{u}^T \left(\lambda F + \sum_{j \in J^+(\overline{x}) \cup J(\overline{x})} \mu_j G_j\right) \overline{u} \ge 0,$$

where $\lambda > 0$. □

Acknowledgments. The authors are grateful to the two referees for their detailed comments and suggestions which have improved the presentation of this paper.

REFERENCES

[1] A. Auslender, *Penalty methods for computing points that satisfy second order necessary conditions*, Math. Programming, 17 (1979), pp. 229–238.

[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] E. J. Balder, *An extension of duality-stability relations to nonconvex optimization problems*, SIAM J. Control Optim., 15 (1977), pp. 329–343.

[4] A. Ben-Tal and J. Zowe, *Necessary and sufficient optimality conditions for a class of nonsmooth minimization problems*, Math. Programming, 24 (1982), pp. 70–91.

[5] D. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[6] J. M. Borwein and D. Preiss, *A smooth variational principle with applications to subdifferentiability and differentiability*, Trans. Amer. Math. Soc., 303 (1987), pp. 517–527.

[7] C. Charalambous, *On conditions for optimality of the nonlinear l_1 problem*, Math. Programming, 17 (1979), pp. 123–135.

[8] F. H. Clarke, Y. S. Ledyaev, and P. R. Wolenski, *Proximal analysis and minimization principles*, J. Math. Anal. Appl., 196 (1995), pp. 722–735.

[9] A. Fiacco and G. McCormick, *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Wiley, New York, 1968.

[10] C. J. Goh and X. Q. Yang, *A nonlinear Lagrangian theory for nonconvex optimization*, J. Optim. Theory Appl., 109 (2001), pp. 99–121.
[11] J. B. Hiriart-Urruty, J. J. Strodiot, and V. Hien Nguyen, *Generalized Hessian matrix and second-order optimality conditions for problems with $C^{1,1}$ data*, Appl. Math. Optim., 11 (1984), pp. 43–56.
[12] X. X. Huang and X. Q. Yang, *Nonlinear Lagrangian for Multiobjective Optimization and Application to Duality and Exact Penalization*, preprint, Department of Applied Mathematics, The Hong Kong Polytechnic University, Kowloon, Hong Kong, 2001.
[13] 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.
[14] D. Li, *Zero duality gap for a class of nonconvex optimization problems*, J. Optim. Theory Appl., 85 (1995), pp. 309–324.
[15] P. Loridan, *Necessary conditions for ϵ-optimality*, Math. Programming Stud., 19 (1982), pp. 140–152.
[16] R. T. Rockafellar, *Conjugate Duality and Optimization*, SIAM, Philadelphia, PA, 1974.
[17] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*, Springer-Verlag, Berlin, 1998.
[18] A. M. Rubinov, B. M. Glover, and X. Q. Yang, *Modified Lagrangian and penalty functions in continuous optimization*, Optimization, 46 (1999), pp. 327–351.
[19] A. M. Rubinov, B. M. Glover, and X. Q. Yang, *Decreasing functions with applications to penalization*, SIAM J. Optim., 10 (1999), pp. 289–313.
[20] S. A. Vavasis, *Black-box complexity of local minimization*, SIAM J. Optim., 3 (1993), pp. 60–79.
[21] X. Q. Yang, *Second-order conditions of $C^{1,1}$ optimization with applications*, Numer. Funct. Anal. Optim., 14 (1993), pp. 621–632.
[22] X. Q. Yang, *An exterior point method for computing points that satisfy second order necessary conditions for a $C^{1,1}$ optimization problem*, J. Math. Anal. Appl., 87 (1994), pp. 118–133.
[23] X. Q. Yang, *Second-order global optimality conditions for convex composite optimization*, Math. Programming, 81 (1998), pp. 327–347.

SIAM J. OPTIM.
Vol. 11, No. 4, pp. 1145–1146

CORRIGENDUM: ON THE CONSTANT POSITIVE LINEAR DEPENDENCE CONDITION AND ITS APPLICATION TO SQP METHODS*

LIQUN QI† AND ZENGXIN WEI‡

Abstract. We correct an assertion in the quoted paper [Qi and Wei, *SIAM J. Optim.*, 10 (2000), pp. 963–981].

Key words. feasible SQP method, superlinear convergence, strict complementarity

AMS subject classifications. 90C30, 60K05

PII. S1052623401383327

In section 6.2 of [3], after hypotheses (H7)–(H9), an additional hypothesis (H10) needs to be added as follows:

(H10) strict complementarity holds at x^*.

The reason this hypothesis is needed is that the end of the proof of Proposition 6.1 of [3] claims to follow, step by step, the proof of Proposition 3.6 of [2]. The proof of Proposition 3.6 of [2] refers to that of Proposition 4.8 of [1], which invokes Lemma 4.4 of [1]. Lemma 4.4 of [1], however, is proved under a strict complementarity condition, which requires all components of $u^*_{I_i}$ to be strictly positive. This cannot be guaranteed in [3] without the additional hypothesis (H10).

Because of this correction, the last sentence of the abstract of [3] should be changed as follows: "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, strict complementarity, and a condition slightly weaker than the linear independence constraint qualification." Corresponding changes should also be made in the introduction and section 6.1.

The following changes to clarify notation should be made in the proof of Proposition 6.1:

1. In line 8, "$u^*_{\bar{I}_i}$" is replaced by "$u^*_{\bar{I}_i}(i)$";
2. Line 9 should begin, "Let $u^*_j(i) = 0$ if $j \notin \bar{I}_i$. Then $u^*(i) \in M(x^*)$."
3. In line 13, "$u^*(i)$" replaces "u^*."

It might be possible to remove (H10) if the proof of Proposition 6.1 is further modified. At this time, we have not explored such an alternative approach.

Acknowledgments. The above difficulty with [3] was pointed out by André Tits. We are thankful to him for his comments and appreciate his helpfulness. We are also grateful to Margaret Wright for her help.

*Received by the editors January 11, 2001; accepted for publication January 11, 2001; published electronically May 16, 2001.
http://www.siam.org/journals/siopt/11-4/38332.html
†Department of Applied Mathematics, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (maqilq@polyu.edu.hk).
‡Department of Mathematics, Guangxi University, Nanning, Guangxi, China (zxwei@gxu.edu.cn).

REFERENCES

[1] 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.

[2] E.R. Panier and A.L. Tits, *On combining feasibility, descent and superlinear convergence in inequality constrained optimization*, Math. Program., 59 (1993), pp. 261–276.

[3] L. Qi and Z. Wei, *On the constant positive linear dependence condition and its application to SQP methods*, SIAM J. Optim., 10 (2000), pp. 963–981.