SIAM J. OPTIM.
Vol. 20, No. 1, pp. 1–29

A DERIVATIVE-FREE ALGORITHM FOR INEQUALITY CONSTRAINED NONLINEAR PROGRAMMING VIA SMOOTHING OF AN ℓ_∞ PENALTY FUNCTION[*]

G. LIUZZI[†] AND S. LUCIDI[‡]

Abstract. In this paper we consider inequality constrained nonlinear optimization problems where the first order derivatives of the objective function and the constraints cannot be used. Our starting point is the possibility to transform the original constrained problem into an unconstrained or linearly constrained minimization of a nonsmooth exact penalty function. This approach shows two main difficulties: the first one is the nonsmoothness of this class of exact penalty functions which may cause derivative-free codes to converge to nonstationary points of the problem; the second one is the fact that the equivalence between stationary points of the constrained problem and those of the exact penalty function can only be stated when the penalty parameter is smaller than a threshold value which is not known a priori. In this paper we propose a derivative-free algorithm which overcomes the preceding difficulties and produces a sequence of points that admits a subsequence converging to a Karush–Kuhn–Tucker point of the constrained problem. In particular the proposed algorithm is based on a smoothing of the nondifferentiable exact penalty function and includes an updating rule which, after at most a finite number of updates, is able to determine a "right value" for the penalty parameter. Furthermore we present the results obtained on a real world problem concerning the estimation of parameters in an insulin-glucose model of the human body.

Key words. derivative-free optimization, constrained optimization, nonlinear programming, nondifferentiable exact penalty functions

AMS subject classifications. 65K05, 90C30, 90C56

DOI. 10.1137/070711451

1. Introduction.

We consider the following problem:

$$
\begin{array}{ll}
\min & f(x) \\
\text{s.t.} & g(x) \le 0, \\
 & Ax \le b,
\end{array}
\tag{1}
$$

where $x \in R^n$, $f : R^n \to R$, $g : R^n \to R^m$, $A \in R^{p\times n}$, $b \in R^p$, and we assume that f and g are twice continuously differentiable on R^n. We denote by $a_j^\top$, $j = 1, \dots, p$, the rows of matrix A and by

$$\mathcal{F} = \{x \in R^n : \ Ax \le b, \ g(x) \le 0\}$$

the feasible set of problem (1). We assume that the derivatives of the objective and nonlinear constraint functions can be neither calculated nor explicitly approximated. Indeed, in many engineering problems the analytic expressions of the functions defining the objective and constraints of the problem are not available and their values

[*] Received by the editors December 20, 2007; accepted for publication (in revised form) October 15, 2008; published electronically March 13, 2009. This work has been supported by MIUR-PRIN 2005 Research Program on New Problems and Innovative Methods in Nonlinear Optimization.

http://www.siam.org/journals/siopt/20-1/71145.html

[†] CNR - Consiglio Nazionale delle Ricerche, IASI - Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti," Viale Manzoni 30, 00185 Rome, Italy (liuzzi@iasi.cnr.it).

[‡] Università degli Studi di Roma "La Sapienza," Dipartimento di Informatica e Sistemistica "A. Ruberti," Via Ariosto 25, 00185 Rome, Italy (lucidi@dis.uniroma1.it).

are computed by means of complex simulation computer programs. For further motivations on the necessity of using derivative-free methods we refer the reader to the survey paper [14].

In the literature, some globally convergent derivative-free methods for the solution of problem (1) have been proposed. In [20] a pattern search algorithm is used within a sequential augmented Lagrangian approach. Essentially, the method embeds the pattern search algorithm proposed in [18], within the augmented Lagrangian method [6], which is the basis for the subroutine AUGLG in the LANCELOT optimization package. More recently, in [15] a generating set direct search augmented Lagrangian algorithm is proposed which explicitly handles the linear constraints.

In [1] the filter method proposed in [10] is adapted to include a pattern search minimization strategy. Basically, the method employs a "filter" for acceptance of the points produced by the pattern search local optimizer.

In [2] a so-called *extreme barrier* approach is employed. Namely the constrained problem is converted to an unconstrained one by setting the objective function value to infinity for infeasible points. To minimize this extreme barrier function, the authors propose an extension of the generalized pattern search class of algorithms which allows local exploration in an asymptotically dense set of directions. More recently, in [3] Audet and Dennis propose a mesh adaptive direct search algorithm which uses a progressive barrier strategy and allows infeasible starting points.

Similarly to [20, 15, 2] and [3], in this paper we propose an algorithm which is based on the idea of employing a derivative-free method to solve a linearly constrained reformulation of problem (1). Our approach differs from the preceding ones in that we use the fact that one can solve problem (1) by minimizing a nonsmooth exact penalty function over a set defined by the linear constraints of problem (1). However, nonsmooth exact penalty functions cannot be straightforwardly combined to a globally convergent derivative-free algorithm. Indeed, the following theoretical and computational aspects should be carefully taken into account.

- Ill-conditioning of merit functions. This aspect makes the minimization of such functions a difficult task, especially for derivative-free codes which use only evaluations of the objective function.
- Nondifferentiability of the penalty function. The lack of differentiability may have negative effects on the convergence of derivative-free methods to stationary points of the penalty function. Indeed, most of the unconstrained derivative-free methods require the objective function to be at least continuously differentiable.
- Equivalence between stationary points of the penalty function and KKT pairs of problem (1). An exact penalty function enjoys its exactness properties only if the penalty parameter is below a certain threshold value which is not known a priori. This aspect is crucial also in the case where derivatives are available.

As regards the first point, we introduce a new exact penalty function which penalizes only the nonlinear constraints and does this penalization in such a way to reduce as much as possible the ill-conditioning.

The nondifferentiability of the new penalty function is tackled by employing the smoothing technique proposed in [4, 31]. In particular, in order to find a stationary point of the penalty function by minimizing the smooth approximation, we adapted the method proposed in [21] to solve linearly constrained finite minimax problems.

As for the last point, the properties of the smooth approximation allow us to define a suitable updating rule for the penalty parameter ϵ. This rule, after a finite number of reductions, is able to find a right value for ϵ so as to convey the desirable exactness properties to the penalty function.

To conclude, we propose a globally convergent algorithm which is based on the derivative-free minimization of a smooth approximation of a nondifferentiable exact penalty function which does an ℓ_∞ penalization of the constraints. Moreover, this new algorithm exploits the structure of the problem by allowing an explicit handling of the linear constraints.

In regards to a possible practical interest of the proposed approach, we recall the encouraging results described in [12]. In fact, [12] reports an extensive numerical testing and comparison which point out significant computational advantages in using a smooth approximation of an exact ℓ_∞ penalty function in the field of derivative-free methods.

Even though the main aim of this paper is the definition of a new algorithm and the study of its theoretical properties, we also show that a rough implementation of the method is able to solve successfully a real world problem concerning the estimation of parameters in an insulin-glucose model of the human body. This result seems to confirm further the conclusion of [12].

This paper is organized as follows. In section 2, the exact penalty function approach is introduced and discussed. Section 3 is devoted to the description of a smooth approximation technique along with some preliminary properties. In section 4, the derivative-free method is presented and its global convergence is studied. Section 5 is devoted to the solution of the constrained parameter estimation problem. Finally, in section 6, we draw some conclusions.

We end this section by introducing some notations which will be used in the rest of this paper. Given a set $\mathcal{S}$, we denote by $\overset{\circ}{\mathcal{S}}$ and $\bar{\mathcal{S}}$, respectively, the interior and the closure of $\mathcal{S}$. By $\|\cdot\|$ we indicate the Euclidean norm. The following index sets will be used in this paper:

$$I_0(x) := \{i : g_i(x) = 0\}, \quad I_\pi(x) := \{i : g_i(x) \geq 0\}, \quad I_\nu(x) := \{i : g_i(x) < 0\},$$
$$J(x) := \{j : a_j^\top x = b_j\}.$$

2. An exact penalty function approach. As already said in the introduction, the first step of our approach is that of defining and using a penalty function which will be more tractable from a computational point of view. Namely, a penalty function which

(i) has a structure that presents fewer nonlinearities than previous exact penalty functions [9];

(ii) allows direct handling of linear and bound constraints thus penalizing only the nonlinear ones.

As regards point (i), nondifferentiable globally exact penalty functions were introduced in [9] for nonlinear programming problems of the form

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g(x) \leq 0. \end{array} \tag{2}$$

In order to prove the relevant exactness properties, it is necessary to introduce a compact relaxation of the feasible set $\mathcal{F}$.

Thus, following [9], given a vector $\alpha \in R^m$ such that $\alpha_i > 0$, $i = 1, \dots, m$, the set

$$\mathcal{D}_\alpha = \{x \in R^n : g(x) \leq \alpha\}$$

is considered. Then, on the interior of set $\mathcal{D}_\alpha$, the penalty function

$$Q(x;\epsilon) = f(x) + \frac{1}{\epsilon}\max\left\{0, \frac{g_1(x)}{\alpha_1 - g_1(x)}, \dots, \frac{g_m(x)}{\alpha_m - g_m(x)}\right\}$$

can be defined, where the terms $\alpha_i - g_i(x)$ make $Q(x;\epsilon)$ go to infinity when x approaches the boundary of $\mathcal{D}_\alpha$, thus guaranteeing the compactness of its level sets.

In [9], it is shown that $Q(x;\epsilon)$ is a globally exact penalty function for problem (2); that is, a value $\epsilon^\star > 0$ for the penalty parameter exists such that for every $\epsilon \in (0, \epsilon^\star]$ the solution of problem (2) is equivalent to the solution of

$$\min Q(x;\epsilon) \quad \text{s.t.} \quad x \in \overset{\circ}{\mathcal{D}}_\alpha, \tag{3}$$

which, essentially, amounts to an unconstrained minimization of $Q(x;\epsilon)$, due to the fact that $\overset{\circ}{\mathcal{D}}_\alpha$ is an open set. More precisely, a value $\epsilon^\star > 0$ for the penalty parameter exists such that for every $\epsilon \in (0, \epsilon^\star]$, every local (global, stationary) point of problem (3) is a local (global Karush–Kuhn–Tucker (KKT)) point of problem (2) and conversely.

However, we note that the structure of $Q(x;\epsilon)$ is such that two contrasting effects, tied with the choice of parameters α and ϵ, may arise. Indeed, rewriting

$$Q(x;\epsilon) = f(x) + \max\left\{0, \frac{g_1(x)}{\epsilon(\alpha_1 - g_1(x))}, \dots, \frac{g_m(x)}{\epsilon(\alpha_m - g_m(x))}\right\},$$

two conflicting requirements become apparent. On the one hand, in order to limit the ill-conditioning of the penalty function near the boundary of the compact set $\mathcal{D}_\alpha$, sufficiently large α_i's should be chosen. On the other hand, the exactness properties follow only when the constraints are sufficiently penalized, that is, when the terms $\epsilon(\alpha_i - g_i(x))$ are sufficiently small on all $\mathcal{D}_\alpha$. Hence, choosing large α_i's requires very small values of ϵ thus increasing the ill-conditioning of the penalty function whenever at least one term $(\alpha_i - g_i(x))$ exists which is not excessively large. Besides, reasonable values for ϵ preclude the possibility of choosing large values for the α_i's, that is, the possibility of choosing sets $\mathcal{D}_\alpha$ having the boundary sufficiently away from the feasible region.

In order to overcome the preceding difficulties, we introduce the following new penalty function:

$$Z(x;\epsilon) = f(x) + \max\left\{0, \left(\frac{1}{\epsilon} + \frac{1}{\alpha_1 - g_1(x)}\right) g_1(x), \dots, \left(\frac{1}{\epsilon} + \frac{1}{\alpha_m - g_m(x)}\right) g_m(x)\right\},$$

where the terms $g_i(x)/\epsilon$ (needed to achieve the exactness properties) and $g_i(x)/(\alpha_i - g_i(x))$ (required to guarantee the compactness of the level sets) are split in such a way that they no longer interfere one another. Introducing the functions

$$\hat{g}_i(x;\epsilon) = \left(1 + \frac{\epsilon}{\alpha_i - g_i(x)}\right) g_i(x), \qquad i = 1, \dots, m,$$

$\hat{g}_0(x;\epsilon) = 0$, $Z(x;\epsilon)$ can be rewritten in the compact form

$$Z(x;\epsilon) = f(x) + \frac{1}{\epsilon}\max_{i=0,1,\dots,m}\{\hat{g}_i(x;\epsilon)\}.$$

As regards point (ii), the need for an "ad-hoc" handling of the constraints arises every time they can be partitioned into two subsets of "difficult" and "easy" constraints. A more traditional approach to problem (1) would be that of penalizing all the constraints by adding to the objective function a penalty term for every constraint. However, every penalty term increases the nonlinearities and the ill-conditioning of the penalty function. This, in turn, would surely result in a difficult problem to solve especially for a derivative-free method. On the other hand, many efficient derivative-free methods exist which are able to solve linearly and bound constrained optimization problems by explicitly handling the linear and bound constraints [19, 22, 30]. For this reason, instead of problem (2), we consider problem (1) and penalize only the nonlinear constraints.

To this aim, we shall prove that a threshold value $\epsilon^\star > 0$ exists such that, for all $\epsilon \in (0, \epsilon^\star]$, problem (1) is equivalent to

$$\begin{array}{ll} \min & Z(x;\epsilon) \\ \text{s.t.} & Ax \le b, \\ & x \in \overset{\circ}{\mathcal{D}}_\alpha\ . \end{array} \tag{4}$$

Note that the new structure of the penalty function along with the presence of the linear constraints and, hence, the fact that $Z(x;\epsilon)$ is to be minimized on the set $\{x \in R^n :\ Ax \le b,\ x \in \overset{\circ}{\mathcal{D}}_\alpha\}$, makes the analysis of [7, 9] not readily applicable. Therefore, in the following subsections, we analyze the theoretical properties and connections between problems (4) and (1), by adapting the analysis carried out in [7, 9].

In what follows we denote

$$\mathcal{S}_\alpha = \overset{\circ}{\mathcal{D}}_\alpha \cap \{x \in R^n :\ Ax \le b\}.$$

2.1. Definitions and assumptions. In order to state the equivalence between problems (1) and (4), we introduce the following assumptions which we require to hold true throughout this paper. They are standard assumptions in a constrained context.

Assumption 1. The set $\bar{\mathcal{S}}_\alpha$ is compact.

Assumption 2. At any point $x \in \bar{\mathcal{S}}_\alpha$, a vector $d \in T(x)$ exists such that

$$\nabla g_i(x)^\top d < 0 \qquad \forall\ i \text{ s.t. } g_i(x) \ge 0,$$

where

$$T(x) = \{d \in R^n : a_j^\top d \le 0\ \forall\ j \in J(x)\} \tag{5}$$

is the *cone of feasible directions* with respect to the linear inequality constraints.

The first assumption is needed to guarantee that the penalty function has compact level sets while the second one guarantees that the feasible region of problem (1) is not empty with a nonempty interior.

As concerns problem (1), the presence of the linear inequality constraints allows us to define necessary optimality conditions under somewhat weaker assumptions than usual. In particular, under Assumption 2 it is possible to state the following (KKT) necessary conditions for local optimality of problem (1).

PROPOSITION 1. *Let* $\bar{x} \in \mathcal{F}$ *be a local solution of problem* (1). *Then,*

$$\begin{aligned} &\nabla f(\bar{x}) + \nabla g(\bar{x})\bar{\lambda} + A^\top \bar{\mu} = 0, \\ &\bar{\lambda}^\top g(\bar{x}) = 0, \quad \bar{\lambda} \ge 0, \\ &\bar{\mu}^\top (Ax - b) = 0, \quad \bar{\mu} \ge 0, \end{aligned} \tag{6}$$

for some vectors $\bar{\lambda} \in R^m$ *and* $\bar{\mu} \in R^p$.

Proof. The proof follows by considering Propositions 3.3.11 and 3.3.12 in [5] along with the Motzkin theorem of the alternative [23]. □

As regards problem (4), we recall that the directional derivative $DZ(x, d; \epsilon)$ of $Z(x; \epsilon)$ at $x \in \mathcal{S}_\alpha$ along direction $d \in R^n$ exists and is given by (see, for instance, [4])

$$DZ(x, d; \epsilon) = \nabla f(x)^\top d + \frac{1}{\epsilon} \max_{i \in B(x;\epsilon)} \left\{ \nabla \hat{g}_i(x; \epsilon)^\top d \right\},$$

where

$$B(x; \epsilon) = \left\{ i \in \{0, 1, \dots, m\} : \hat{g}_i(x; \epsilon) = \max_{j=0,1,\dots,m} \{\hat{g}_j(x; \epsilon)\} \right\}$$

and

$$\nabla \hat{g}_i(x; \epsilon) = \left(1 + \frac{\epsilon \alpha_i}{(\alpha_i - g_i(x))^2} \right) \nabla g_i(x), \qquad i = 1, \dots, m.$$

Therefore, the usual definition of stationarity for problem (4) can be given as follows.

Definition 1. *A point $\bar{x} \in \mathcal{S}_\alpha$ is a stationary point of problem* (4) *if*

$$DZ(\bar{x}, d; \epsilon) \geq 0 \ \ \forall \ d \in T(\bar{x}).$$

By exploiting the particular structure of problem (4), that is, the expression of the penalty function $Z(x; \epsilon)$, it is possible to state a different characterization of its stationary points.

Proposition 2. *For any given $\epsilon > 0$, a point $\bar{x} \in \mathcal{S}_\alpha$ is a stationary point of problem* (4) *if and only if for each $i \in B(\bar{x}; \epsilon)$ there exists λ_i satisfying*

$$\lambda_i \geq 0, \qquad \sum_{i \in B(\bar{x};\epsilon)} \lambda_i = 1, \tag{7}$$

$$\left(\nabla f(\bar{x}) + \frac{1}{\epsilon} \sum_{i \in B(\bar{x};\epsilon)} \lambda_i \nabla \hat{g}_i(\bar{x}; \epsilon) \right)^\top d \geq 0 \quad \forall \ d \in T(\bar{x}). \tag{8}$$

Proof. First let us assume that $\bar{x} \in \mathcal{S}_\alpha$ is a stationary point of problem (4). Then (7) and (8) follows by considering Propositions 3.3.10 and 3.3.11 of [5].

On the contrary, let $\bar{x} \in \mathcal{S}_\alpha$ be a point which satisfies conditions (7) and (8); then we can write

$$\begin{aligned} 0 &\leq \left(\nabla f(\bar{x}) + \frac{1}{\epsilon} \sum_{i \in B(\bar{x};\epsilon)} \lambda_i \nabla \hat{g}_i(\bar{x}; \epsilon) \right)^\top d \\ &\leq \left(\nabla f(\bar{x})^\top d + \frac{1}{\epsilon} \max_{i \in B(\bar{x};\epsilon)} \{ \nabla \hat{g}_i(\bar{x}; \epsilon)^\top d \} \sum_{i \in B(\bar{x};\epsilon)} \lambda_i \right) \\ &= \left(\nabla f(\bar{x})^\top d + \frac{1}{\epsilon} \max_{i \in B(\bar{x};\epsilon)} \{ \nabla \hat{g}_i(\bar{x}; \epsilon)^\top d \} \right) \end{aligned}$$

for all $d \in T(\bar{x})$, which shows that $\bar{x}$ is a stationary point of problem (4). □

2.2. Exactness properties. The exactness properties of the penalty function $Z(x;\epsilon)$ heavily hinge on the following lemma, which is a slight modification of Theorem 2.2 of [13].

LEMMA 1. *Let $\hat{x} \in \{x \in R^n : \ Ax \le b\}$. Then, an open neighborhood $\mathcal{B}(\hat{x};\rho)$ of $\hat{x}$ and a direction $d \in T(\hat{x})$ exist such that, for all $i \in I_\pi(\hat{x})$, we have*

$$\nabla g_i(x)^\top d \le -1 \qquad \forall\ x \in \mathcal{B}(\hat{x};\rho) \cap \{x \in R^n : \ Ax \le b\}, \tag{9}$$

$$\nabla \hat{g}_i(x;\epsilon)^\top d \le -1 \qquad \forall\ x \in \mathcal{B}(\hat{x};\rho) \cap \mathcal{S}_\alpha, \ \forall\ \epsilon > 0. \tag{10}$$

Proof. By Assumption 2, a vector $\hat{z} \in T(\hat{x})$ exists such that $\nabla g_i(\hat{x})^\top \hat{z} < 0$ for all $i \in I_\pi(\hat{x})$. Hence, by continuity, a $\rho > 0$ exists such that

$$\nabla g_i(x)^\top \hat{z} \le -\gamma/2$$

for all $x \in \mathcal{B}(\hat{x};\rho) \cap \{x \in R^n : \ Ax \le b\}$ and $i \in I_\pi(\hat{x})$, where

$$-\gamma = \max_{i \in I_\pi(\hat{x})} \{\nabla g_i(\hat{x})^\top \hat{z}\} < 0.$$

Thus (9) follows by choosing $d = 2\hat{z}/\gamma$.

For $x \in \mathcal{B}(\hat{x};\rho) \cap \mathcal{S}_\alpha$ we can write

$$\nabla g_i(x)^\top d = \frac{(\alpha_i - g_i(x))^2}{(\alpha_i - g_i(x))^2 + \epsilon\alpha_i} \nabla \hat{g}_i(x;\epsilon)^\top d \le -1 \qquad \forall\ i \in I_\pi(\hat{x}),$$

so that, considering

$$\frac{(\alpha_i - g_i(x))^2 + \epsilon\alpha_i}{(\alpha_i - g_i(x))^2} > 1 \qquad \forall\ i \in I_\pi(\hat{x}), \ \forall\ \epsilon > 0$$

we have

$$\nabla \hat{g}_i(x;\epsilon)^\top d \le -1 \qquad \forall\ i \in I_\pi(\hat{x}), \ \forall\ \epsilon > 0 \tag{11}$$

which proves (10). □

The analysis of the exactness properties of $Z(x;\epsilon)$ follows the same reasonings used in [7] and [9]. For the sake of clarity, here we report only the statement of the main results and refer the interested reader to the appendix for a thorough development and analysis of the exactness properties.

The following propositions establish a connection between stationary points of the exact penalty function $Z(x;\epsilon)$ which are feasible and KKT pairs of problem (1).

PROPOSITION 3. *Let $\bar{x} \in \mathcal{F}$. Then, for any $\epsilon > 0$, if $\bar{x}$ is a critical point of $Z(x;\epsilon)$, there exist multipliers $\bar{\lambda} \in R^m$ and $\bar{\mu} \in R^p$ such that $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a KKT triple for problem* (1).

For sufficiently small values of the penalty parameter ϵ, a one-to-one correspondence between KKT pairs of problem (1) and critical points of the penalty function $Z(x;\epsilon)$ exist.

PROPOSITION 4. *There exists an $\epsilon^\star > 0$ such that, for all $\epsilon \in (0, \epsilon^\star]$, if $\bar{x} \in \mathcal{S}_\alpha$ is a critical point of $Z(x;\epsilon)$, there exist multipliers $\bar{\lambda} \in R^m$ and $\bar{\mu} \in R^p$ such that $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a KKT triple for problem* (1) *and conversely.*

Finally, the last proposition describes a connection between local and global minimum points of the penalty function and problem (1).

PROPOSITION 5. *There exists an $\epsilon^\star > 0$ such that, for all $\epsilon \in (0, \epsilon^\star]$,*

(a) *if $x_\epsilon \in \mathcal{S}_\alpha$ is a (strict) local unconstrained minimum point of $Z(x;\epsilon)$, then x_ϵ is a (strict) local constrained minimum point of problem* (1)*;*

(b) *if $x^\star \in \mathcal{S}_\alpha$ is a global unconstrained minimum point of $Z(x;\epsilon)$ on $\mathcal{S}_\alpha$, then $x^\star$ is a global solution to problem* (1) *and conversely.*

3. Smooth approximation and preliminary results. In this section we concentrate on the derivative-free approach to the solution of problem (1). As already argued, in order to solve this problem we resort to a reformulation of problem (1) by means of the exact penalty function $Z(x;\epsilon)$, so that, on the basis of the exactness properties studied in the preceding section, we can concentrate on the solution of the linearly constrained problem (4).

To tackle the nondifferentiability of function $Z(x;\epsilon)$, we adopt a smoothing technique [4, 31] which consists of solving a sequence of smooth problems approximating the nonsmooth one in the limit. Let $\mu > 0$ be a smoothing parameter, and define

$$Z(x;\mu,\epsilon) = f(x)+\mu \ln\left(\sum_{i=0}^{m} \exp\left(\frac{\hat{g}_i(x;\epsilon)}{\mu\epsilon}\right)\right) = f(x)+\mu \ln\left(1+\sum_{i=1}^{m} \exp\left(\frac{\hat{g}_i(x;\epsilon)}{\mu\epsilon}\right)\right).$$

We report some properties of $Z(x;\mu,\epsilon)$ [31] that exploit the fact that $\nabla \hat{g}_0(x;\epsilon) = 0$.

PROPOSITION 6.

(i) *For any given $x \in R^n$ and $\epsilon > 0$, $Z(x;\mu,\epsilon)$ is increasing with respect to μ and*

$$Z(x;\epsilon) \le Z(x;\mu,\epsilon) \le Z(x;\epsilon) + \mu \ln m. \tag{12}$$

(ii) *$Z(x;\mu,\epsilon)$ is twice continuously differentiable for all $\mu > 0$, $\epsilon > 0$, and*

$$\begin{aligned}\nabla_x Z(x;\mu,\epsilon) &= \nabla f(x) + \frac{1}{\epsilon}\sum_{i=0}^{m} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)\\ &= \nabla f(x) + \frac{1}{\epsilon}\sum_{i=1}^{m} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon),\end{aligned} \tag{13}$$

$$\begin{aligned}\nabla_x^2 Z(x;\mu,\epsilon) = \nabla^2 f(x) &+ \frac{1}{\epsilon}\sum_{i=1}^{m} \left(\lambda_i(x;\mu,\epsilon)\nabla^2 \hat{g}_i(x;\epsilon)\right)\\ &+ \frac{1}{\mu\epsilon^2}\sum_{i=1}^{m} \left(\lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)\nabla \hat{g}_i(x;\epsilon)^\top\right)\\ &- \frac{1}{\mu\epsilon^2}\left(\sum_{i=1}^{m} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)\right)\left(\sum_{i=1}^{m} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)\right)^\top,\end{aligned} \tag{14}$$

where

$$\lambda_i(x;\mu,\epsilon) = \frac{\exp\left(\dfrac{\hat{g}_i(x;\epsilon)}{\mu\epsilon}\right)}{1+\displaystyle\sum_{j=1}^{m} \exp\left(\dfrac{\hat{g}_j(x;\epsilon)}{\mu\epsilon}\right)} \in (0,1), \quad i = 0,1,\dots,m, \tag{15}$$

and $\sum_{i=0}^{m} \lambda_i(x;\mu,\epsilon) = 1$.

Thus, having introduced the smoothing function $Z(x;\mu,\epsilon)$, we consider the following smooth approximating problem:

$$\min_{x\in\mathcal{S}_\alpha} Z(x;\mu,\epsilon), \tag{16}$$

where the approximating parameter μ and the penalty parameter ϵ will be adaptively reduced during the optimization process.

Considering problem (16), it is important to study the connections that $Z(x;\mu,\epsilon)$ has with the original constrained problem. In particular, $Z(x;\mu,\epsilon)$ should be able, when minimized, to drive the algorithm away from infeasible points with respect to the nonlinear inequality constraints. Indeed, the following proposition states an important result needed to prove convergence of the algorithm. Namely, on every $\hat{x}\in\mathcal{S}_\alpha$ a sufficiently small neighborhood of $\hat{x}$ exists such that in every infeasible point belonging to this neighborhood a direction $d\in T(\hat{x})$ exists such that the directional derivative of the smooth approximating function along direction d is negative and uniformly bounded away from zero, for ϵ sufficiently small.

Proposition 7. *Let $\hat{x}\in\mathcal{S}_\alpha$ and $\mu_{MAX}>0$ be any given scalar. Then, $\epsilon(\hat{x})>0$ and $\sigma(\hat{x})>0$ exist such that for all $x\in\mathcal{B}(\hat{x};\sigma(\hat{x}))\cap\mathcal{S}_\alpha$ and satisfying $g(x)\not\leq 0$, and for all $\epsilon\in(0,\epsilon(\hat{x})]$ a direction $d\in T(\hat{x})$ exists such that*

$$\nabla Z(x;\mu,\epsilon)^\top d \leq -\frac{1}{2\epsilon(m+1)}$$

for all $\mu\in(0,\mu_{MAX}]$.

Proof. Let $\mathcal{B}(\hat{x};\rho)$ and d be the neighborhood and the direction considered in Lemma 1. By continuity, we can find a neighborhood $\mathcal{B}(\hat{x};\sigma(\hat{x}))\subseteq\mathcal{B}(\hat{x};\rho)$ such that for $i\notin I_\pi(\hat{x})$ and $x\in\mathcal{B}(\hat{x};\sigma(\hat{x}))$, we have $g_i(x)<0$; it follows that $I_\pi(x)\subseteq I_\pi(\hat{x})$ and $I_\nu(\hat{x})\subseteq I_\nu(x)$ for $x\in\mathcal{B}(\hat{x};\sigma(\hat{x}))$.

Now let $x\in\mathcal{B}(\hat{x};\sigma(\hat{x}))\cap\mathcal{S}_\alpha$ be an infeasible point with respect to the nonlinear inequality constraints. Then, there must exist at least an index $i\in I_\pi(x)$ such that $g_i(x)>0$, which implies $I_\pi(x)\neq\phi$. By recalling expression (13), we can write

$$\begin{aligned}\nabla Z(x;\mu,\epsilon)^\top d &= \nabla f(x)^\top d\\ &\quad+\frac{1}{\epsilon}\left(\sum_{i\in I_\nu(\hat{x})}\lambda_i(x;\mu,\epsilon)\nabla\hat{g}_i(x;\epsilon)^\top d+\sum_{i\in I_\pi(\hat{x})}\lambda_i(x;\mu,\epsilon)\nabla\hat{g}_i(x;\epsilon)^\top d\right).\end{aligned}$$

By Lemma 1 we have that $\nabla\hat{g}_i(x;\epsilon)^\top d\leq -1$, $i\in I_\pi(\hat{x})$, so that we can write

$$\begin{aligned}\nabla Z(x;\mu,\epsilon)^\top d &\leq \nabla f(x)^\top d\\ &\quad+\frac{1}{\epsilon}\left(\sum_{i\in I_\nu(\hat{x})}\lambda_i(x;\mu,\epsilon)\nabla\hat{g}_i(x;\epsilon)^\top d-\sum_{i\in I_\pi(\hat{x})}\lambda_i(x;\mu,\epsilon)\right).\end{aligned} \tag{17}$$

Let $\bar{\imath}\in I_\pi(\hat{x})$ be an index such that $\hat{g}_{\bar{\imath}}(x;\epsilon)=\max_{i\in I_\pi(\hat{x})}\{\hat{g}_i(x;\epsilon)\}$. It is easily seen that $\sum_{i\in I_\pi(\hat{x})}\lambda_i(x;\mu,\epsilon)\geq\lambda_{\bar{\imath}}(x;\mu,\epsilon)$, and, since

$$\frac{\exp(\hat{g}_i(x;\epsilon)/\mu\epsilon)}{\exp(\hat{g}_{\bar{\imath}}(x;\epsilon)/\mu\epsilon)}\leq 1 \qquad \forall\ i\in\{1,\dots,m\},$$

then

$$\lambda_{\bar{\imath}}(x;\mu,\epsilon) \geq \frac{1}{1+1+\displaystyle\sum_{i=1,i\neq\bar{\imath}}^{m} \frac{\exp(\hat{g}_i(x;\epsilon)/\mu\epsilon)}{\exp(\hat{g}_{\bar{\imath}}(x;\epsilon)/\mu\epsilon)}} \geq \frac{1}{1+m}.$$

Hence we get

$$\sum_{i\in I_\pi(\hat{x})} \lambda_i(x;\mu,\epsilon) \geq 1/(1+m). \tag{18}$$

By considering (17) and (18), we get

$$\begin{aligned}\nabla Z(x;\mu,\epsilon)^\top d \leq{}& \nabla f(x)^\top d \\ &+ \frac{1}{\epsilon}\left(\sum_{i\in I_\nu(\hat{x})} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)^\top d - \frac{1}{1+m}\right).\end{aligned} \tag{19}$$

Now, since $I_\nu(\hat{x}) \subseteq I_\nu(x)$, for $x \in \mathcal{B}(\hat{x};\sigma(\hat{x}))$, by expression (15), it follows that, for any given $\mu > 0$ and $x \in \mathcal{B}(\hat{x};\sigma(\hat{x})) \cap \mathcal{S}_\alpha$ not feasible,

$$\lim_{\epsilon\to 0^+} \lambda_i(x;\mu,\epsilon) = 0, \qquad\qquad i \in I_\nu(\hat{x}).$$

Hence, by the boundedness of $\nabla \hat{g}_i(x;\epsilon)^\top d$ and $\nabla f(x)^\top d$, an $\epsilon(\hat{x}) > 0$ exists such that for all $\epsilon \in (0, \epsilon(\hat{x})]$ we have

$$\sum_{i\in I_\nu(\hat{x})} \lambda_i(x;\mu,\epsilon)\nabla \hat{g}_i(x;\epsilon)^\top d < \frac{1}{4(m+1)} \qquad \forall\ \mu \in (0, \mu_{MAX}] \tag{20}$$

$$\nabla f(x)^\top d < \frac{1}{4\epsilon(m+1)}. \tag{21}$$

The result follows from (20), (21), and (19). □

In order to guarantee the global convergence of the algorithm in the case where first derivatives are unavailable, it is necessary to get alternative information by sampling the objective function along a suitable set of search directions. Specifically, we follow the approach proposed in [22], which uses a set of search directions that positively span a "ν-approximation" of the cone of feasible directions; or in other words, the cone of feasible directions with respect to the ν-active linear constraints.

Formally, for any $\nu > 0$ and $x \in \mathcal{S}_\alpha$, we define the set of indices of ν-active linear constraints by

$$J(x;\nu) = \{j : a_j^\top x \geq b_j - \nu\},$$

and the ν-approximation of the cone of feasible directions by

$$T(x;\nu) = \{d \in R^n : a_j^\top d \leq 0\ \forall j \in J(x;\nu)\}.$$

The following proposition (see [22]) describes some properties of sets $J(x;\nu)$ and $T(x;\nu)$.

PROPOSITION 8. *Let $\{x_k\}$ be a sequence of iterates converging towards a point $\bar{x} \in \mathcal{S}_\alpha$. Then, there exists a value $\nu^* > 0$ (depending on $\bar{x}$ only) such that for every $\nu \in (0, \nu^*]$ there exists $\bar{k}_\nu$ such that*

$$J(x_k; \nu) = J(\bar{x}), \tag{22}$$

$$T(x_k; \nu) = T(\bar{x}) \tag{23}$$

for all $k \geq \bar{k}_\nu$.

Proof. See the proof of Proposition 1 in [22]. □

The first step toward defining a derivative-free method for the solution of problem (16) is to associate a suitable set of search directions with each point x_k produced by the algorithm. This set should have the property that the local behavior of the objective function in each direction in the set provides sufficient information to overcome the lack of the gradient. Formally, we introduce the following assumption.

Assumption 3. Let $\{x_k\}$ be sequence of points belonging to $\mathcal{S}_\alpha$, $\{r_k\}$ a sequence of positive scalars, and $\{D_k\}$ a sequence of sets of search directions defined as

$$D_k = \{p_k^i : \|p_k^i\| = 1, \quad i = 1, \dots, r_k\} \qquad \forall\ k.$$

Then, for some constant $\bar{\nu} > 0$,

$$cone\{D_k \cap T(x_k; \nu)\} = T(x_k; \nu) \quad \forall \nu \in [0, \bar{\nu}].$$

Moreover, $\bigcup_{k=0}^{\infty} D_k$ is a finite set and r_k is bounded.

Assumption 3 is quite a standard assumption in a derivative-free context and is needed to guarantee that the search directions are well defined and able to capture sufficiently well the local geometry of the feasible set. An example on how to compute a set of directions satisfying the above assumption can be found in the paper [19].

The proposition which follows is essential to prove convergence of the proposed algorithm to a KKT point. In particular, it points out the minimal requirements on the sampling of the smoothed penalty function $Z(x; \mu, \epsilon)$ along the directions p_k^i, $i = 1, \dots, r_k$, and on the updating of both the smoothing parameter μ and penalty parameter ϵ which are able to guarantee that feasibility of problem (1) is attained in a finite number of steps.

PROPOSITION 9. *Let $\{\mu_k\}$ be a sequence of smoothing parameters and $\{\epsilon_k\}$ a sequence of penalty parameters. Let $\{x_k\}$ be a sequence of points and $\bar{x}$ be a limit point of a subsequence $\{x_k\}_K$, for some infinite set $K \subseteq \{0, 1, \dots\}$, such that $\bar{x} \in \mathcal{S}_\alpha$. Let $\{D_k\}$, with $D_k = \{p_k^1, \dots, p_k^{r_k}\}$, be a sequence of sets of directions which satisfy Assumption 3 and $J_k = \{i \in \{1, \dots, r_k\} : p_k^i \in T(x_k; \nu)\}$ with $\nu \in (0, \min\{\bar{\nu}, \nu^\star\}]$, where $\nu^\star$ and $\bar{\nu}$ are defined in Proposition 8 and Assumption 3, respectively. Suppose that the following conditions hold:*

(i) *for each $k \in K$ and $i \in J_k$, there exist y_k^i and scalars $\xi_k^i > 0$ such that*

$$y_k^i + \xi_k^i p_k^i \in \mathcal{S}_\alpha, \qquad Z(y_k^i + \xi_k^i p_k^i; \mu_k, \epsilon_k) \geq Z(y_k^i; \mu_k, \epsilon_k) - o(\xi_k^i); \tag{24}$$

(ii) *and, furthermore, $\{\mu_k\}_K$ is a bounded sequence and*

$$\lim_{k \to \infty, k \in K} \epsilon_k = 0, \qquad \lim_{k \to \infty, k \in K} \frac{\max_{i \in J_k}\{\xi_k^i, \|x_k - y_k^i\|\}}{\mu_k \epsilon_k} = 0. \tag{25}$$

It results that a $\bar{k} \geq 0$ exists such that, for all $k \in K$ and k satisfying $k \geq \bar{k}$, x_k is feasible for problem (1).

Proof. As a first step, we show that for every bounded sequence $\{d_k\} \subset T(x_k; \nu)$ of directions, an infinite set $\bar{K} \subseteq K$ exists such that

$$\lim_{k\to\infty, k\in\bar{K}} \epsilon_k \nabla Z(x_k; \mu_k, \epsilon_k)^\top d_k \geq 0.$$

By assumption the limit point $\bar{x}$ belongs to $\mathcal{S}_\alpha$, thus an open neighborhood $\mathcal{B}(\bar{x})$ of $\bar{x}$ exists which is strictly contained within $\mathcal{S}_\alpha$. Therefore, by points (i) and (ii), we have that, for $k \in K$ sufficiently large and for all $i \in J_k$, x_k, y_k^i, and $y_k^i + \xi_k^i p_k^i$ belong to $\mathcal{B}(\bar{x})$.

By applying the mean-value theorem to (24), we can write

$$(26)\quad -o(\xi_k^i) \leq Z(y_k^i + \xi_k^i p_k^i; \mu_k, \epsilon_k) - Z(y_k^i; \mu_k \epsilon_k) = \xi_k^i \nabla Z(u_k^i; \mu_k, \epsilon_k)^\top p_k^i, \qquad i \in J_k,$$

where $u_k^i = y_k^i + t_k^i \xi_k^i p_k^i$, with $t_k^i \in (0,1)$. By using the mean-value theorem again and the Cauchy–Schwarz inequality, we can write

$$\begin{aligned}\xi_k^i \nabla Z(u_k^i; \mu_k, \epsilon_k)^\top p_k^i &= \xi_k^i \nabla Z(x_k; \mu_k, \epsilon_k)^\top p_k^i + \xi_k^i (u_k^i - x_k)^\top \nabla^2 Z(\tilde{u}_k^i; \mu_k, \epsilon_k) p_k^i \\ &\leq \xi_k^i \nabla Z(x_k; \mu_k, \epsilon_k)^\top p_k^i + \xi_k^i \|u_k^i - x_k\| \|\nabla^2 Z(\tilde{u}_k^i; \mu_k, \epsilon_k) p_k^i\|,\end{aligned}$$

where $\tilde{u}_k^i = x_k + \tilde{t}_k^i (u_k^i - x_k)$, with $\tilde{t}_k^i \in (0,1)$. By considering expression (14) of $\nabla^2 Z(\tilde{u}_k^i; \mu_k, \epsilon_k)$ and the triangle inequality, we get that

$$\begin{aligned}&\xi_k^i \nabla Z(u_k^i; \mu_k, \epsilon_k)^\top p_k^i \leq \xi_k^i \nabla Z(x_k; \mu_k, \epsilon_k)^\top p_k^i \\ &+\xi_k^i \|u_k^i - x_k\| \left\{ \left\|\nabla^2 f(\tilde{u}_k^i) p_k^i\right\| + \frac{1}{\epsilon_k} \left\| \sum_{j=1}^m \lambda_j(\tilde{u}_k^i; \mu_k, \epsilon_k) \nabla^2 \hat{g}_j(\tilde{u}_k^i; \epsilon_k) p_k^i \right\| \right. \\ &+ \frac{1}{\mu_k \epsilon_k^2} \left\| \sum_{j=1}^m \lambda_j(\tilde{u}_k^i; \mu_k, \epsilon_k) \nabla \hat{g}_j(\tilde{u}_k^i; \epsilon_k) \nabla \hat{g}_j(\tilde{u}_k^i; \epsilon_k)^\top p_k^i - \left(\sum_{j=1}^m \lambda_j(\tilde{u}_k^i; \mu_k, \epsilon_k) \nabla \hat{g}_j(\tilde{u}_k^i; \epsilon_k) \right) \right. \\ &\left.\left. \cdot \left(\sum_{j=1}^m \lambda_j(\tilde{u}_k^i; \mu_k, \epsilon_k) \nabla \hat{g}_j(\tilde{u}_k^i; \epsilon_k) \right)^\top p_k^i \right\| \right\}.\end{aligned}$$

Since $\{x_k\}_K$ converges, it follows from Assumption 3 and (15) that, for all i and j, $\{x_k\}_K$, $\{\tilde{u}_k^i\}$, $\{\lambda_j(\tilde{u}_k^i; \mu_k, \epsilon_k)\}$, $\{p_k^i\}$ are bounded sequences. Therefore, by the continuity assumption on $f(x)$ and $g(x)$, we can find positive constants c_1, c_2, and c_3 such that

$$(27)\quad \xi_k^i \nabla Z(u_k^i; \mu_k, \epsilon_k)^\top p_k^i \leq \xi_k^i \nabla Z(x_k; \mu_k, \epsilon_k)^\top p_k^i + \xi_k^i \left(c_1 + \frac{1}{\epsilon_k} c_2 + \frac{1}{\mu_k \epsilon_k^2} c_3 \right) \|u_k^i - x_k\|.$$

By (24), (26), and (27), we obtain

$$\nabla Z(x_k; \mu_k, \epsilon_k)^\top p_k^i + \left(c_1 + \frac{1}{\epsilon_k} c_2 + \frac{1}{\mu_k \epsilon_k^2} c_3 \right) \|u_k^i - x_k\| \geq -\frac{o(\xi_k^i)}{\xi_k^i}$$

from which, taking into account (13), we can write

$$\left(\nabla f(x_k) + \frac{1}{\epsilon_k}\sum_{j=1}^{m}\lambda_j(x_k;\mu_k,\epsilon_k)\nabla \hat{g}_j(x_k;\epsilon_k)\right)^{\top} p_k^i + \left(c_1 + \frac{c_2}{\epsilon_k} + \frac{c_3}{\mu_k\epsilon_k^2}\right)\|u_k^i - x_k\| \geq -\frac{o(\xi_k^i)}{\xi_k^i}. \tag{28}$$

Since $u_k^i = y_k^i + t_k^i\xi_k^i p_k^i$, with $t_k^i \in (0,1)$, and, by Assumption 3, p_k^i, $i \in J_k$, are bounded, we have that

$$\left(c_1 + \frac{1}{\epsilon_k}c_2 + \frac{1}{\mu_k\epsilon_k^2}c_3\right)\|u_k^i - x_k\| \leq \left(c_1 + \frac{1}{\epsilon_k}c_2 + \frac{1}{\mu_k\epsilon_k^2}c_3\right)(\|y_k^i - x_k\| + \xi_k^i) \quad \forall i \in J_k,$$

and from (28) we obtain

$$\left(\nabla f(x_k) + \frac{1}{\epsilon_k}\sum_{j=1}^{m}\lambda_j(x_k;\mu_k,\epsilon_k)\nabla \hat{g}_j(x_k;\epsilon_k)\right)^{\top} p_k^i + \left(c_1 + \frac{c_2}{\epsilon_k} + \frac{c_3}{\mu_k\epsilon_k^2}\right)(\|y_k^i - x_k\| + \xi_k^i) \geq -\frac{o(\xi_k^i)}{\xi_k^i}. \tag{29}$$

Now, let $\{d_k\}$ be the generic and bounded sequence of directions such that $\{d_k\} \subset T(x_k;\nu)$. By Assumption 3 and by Corollary 10.2 of [19], we know that, for every index $k \in \bar{K}$, $\beta_k^i \geq 0$, $i \in J_k$, and $\bar{c} > 0$ exist such that

$$d_k = \sum_{i \in J_k}\beta_k^i p_k^i \quad \text{and} \quad |\beta_k^i| \leq \bar{c}\|d_k\|. \tag{30}$$

By multiplying (29) by $\epsilon_k\beta_k^i$, $i \in J_k$, and summing up, we get, for every index $k \in \bar{K}$,

$$\left(\epsilon_k\nabla f(x_k) + \sum_{j=1}^{m}\lambda_j(x_k;\mu_k,\epsilon_k)\nabla \hat{g}_j(x_k;\epsilon_k)\right)^{\top} \sum_{i \in J_k}\beta_k^i p_k^i \geq -\sum_{i \in J_k}\epsilon_k\left(\left(c_1 + \frac{1}{\epsilon_k}c_2 + \frac{1}{\mu_k\epsilon_k^2}c_3\right)(\|y_k^i - x_k\| + \xi_k^i) + \frac{o(\xi_k^i)}{\xi_k^i}\right)\beta_k^i. \tag{31}$$

Recalling the boundedness of sequences $\{x_k\}$, $\{\lambda_j(x_k;\mu_k,\epsilon_k)\}$, $j = 1,\dots,m$, an infinite set $\bar{K} \subseteq K$ exists such that

$$\lim_{\substack{k \to \infty \\ k \in \bar{K}}} x_k = \bar{x}, \tag{32}$$

$$\lim_{\substack{k \to \infty \\ k \in \bar{K}}} \lambda_j(x_k;\mu_k,\epsilon_k) = \bar{\lambda}_j, \qquad j = 1,\dots,m. \tag{33}$$

By Proposition 8, for all $k \in \bar{K}$ sufficiently large, we have that $T(x_k;\nu) = T(\bar{x})$, so that, considering the boundedness of sequence $\{d_k\}$, a direction $\bar{d} \in T(\bar{x})$ exists such that

$$\lim_{\substack{k \to \infty \\ k \in \bar{K}}} d_k = \bar{d}. \tag{34}$$

Furthermore, given the fact that r_k is bounded, a finite set $J \subseteq \{1, 2, \dots\}$ exists such that, for $k \in \bar{K}$ and sufficiently large, $J_k = J$.

Now, taking the limit for $k \to \infty$, $k \in \bar{K}$ in (31), and recalling assumption (ii), the boundedness of β_k^i, and the expression of $\nabla \hat{g}_j(x; \epsilon)$, we obtain

$$\lim_{k\to\infty, k\in\bar{K}} \epsilon_k \nabla Z(x_k; \mu_k, \epsilon_k)^\top d_k = \left(\sum_{j=1}^{m} \bar{\lambda}_j \nabla g_j(\bar{x}) \right)^\top \bar{d} \geq 0. \tag{35}$$

Let us now suppose by contradiction that an infinite set $\hat{K} \subseteq K$ exists such that $\lim_{k\to\infty, k\in\hat{K}} x_k = \bar{x}$ and $g(x_k) \not\leq 0$ for all $k \in \hat{K}$. By virtue of Proposition 7, given the fact that (25) holds and recalling that, by assumption, $\nu \in (0, \min\{\bar{\nu}, \nu^\star\}]$ so that, by Proposition 8, $T(x_k; \nu) = T(\bar{x})$, we have that a $\hat{k} \in \hat{K}$ and a direction $\hat{d} \in T(\bar{x})$ exist such that

$$\epsilon_k \nabla Z(x_k; \mu_k, \epsilon_k)^\top \hat{d} \leq -\frac{1}{2(m+1)}.$$

By setting $d_k = \hat{d}$, for all $k \in \hat{K}$, the above relation constitutes a contradiction with (35) thus completing the proof. □

4. A derivative-free method and global convergence result.

In this section we define an algorithm for the solution of problem (1). The main tools are the nondifferentiable exact penalty function $Z(x; \epsilon)$ defined in section 2 along with its exactness properties and the smooth approximating function introduced in section 3. Hence, it would be plausible to employ the algorithm proposed in [21]. Roughly speaking, the latter algorithm inexactly solves a sequence of problems (16) when the smoothing parameter μ is driven to zero at a suitable rate. The convergence analysis carried out in [21] guarantees that a subsequence exists which converges towards a stationary point of problem (4). However, it should be noted that the mentioned result is unsatisfactory in that a proper value for the penalty parameter ϵ is not known a priori (namely, ϵ should be smaller than the threshold $\epsilon^\star$ introduced in Propositions 4 and 5). This implies that a stationary point of problem (4) might have no connections with a solution of problem (1).

In this section, by exploiting Proposition 9, we define a derivative-free algorithm for problem (1) which hinges on a suitable automatic updating rule of the penalty parameter that in a finite number of steps is able to find a value below the mentioned threshold $\epsilon^\star$.

The method that we propose uses as a building block the algorithm proposed in [21]. For the sake of clarity, we report a single iteration of the method therein proposed and refer to it as iteration map $\mathcal{M}$.

```
Iteration map $\mathcal{M}(x, \mu, \tilde{\alpha}^0, \epsilon, q_1) \mapsto (\breve{x}, \breve{\mu}, \breve{\alpha}^0, \breve{\alpha}^{max})$

Data. $\gamma > 0,\ \theta \in (0,1)$.
Step 1. (Computation of search directions)
    Choose a set of directions $D = \{p^1, \dots, p^r\}$ satisfying Assumption 3.
Step 2. (Minimization on the cone$\{D\}$)
    Step 2.1. (Initialization)
        Set $i = 1$, $y^i = x$, $\tilde{\alpha}^i = \tilde{\alpha}^0$.
    Step 2.2. (Computation of the initial stepsize)
        Compute the maximum steplength $\bar{\alpha}^i$ such that $A(y^i + \bar{\alpha}^i p^i) \le b$
        and set $\hat{\alpha}^i = \min\{\bar{\alpha}^i, \tilde{\alpha}^i\}$.
    Step 2.3. (Test on the search direction)
        If $\Big(\hat{\alpha}^i > 0$ and $Z(y^i + \hat{\alpha}^i p^i; \mu, \epsilon) < Z(y^i; \mu, \epsilon) - \gamma(\hat{\alpha}^i)^2$
            and $y^i + \hat{\alpha}^i p^i \in \mathcal{S}_\alpha \Big)$, then
                compute $\alpha^i =$ expansion step$(\bar{\alpha}^i, \hat{\alpha}^i, y^i, p^i)$
                and set $\tilde{\alpha}^{i+1} = \alpha^i$;
        otherwise set $\alpha^i = 0$ and $\tilde{\alpha}^{i+1} = \theta\tilde{\alpha}^i$.
    Step 2.4. (New point)
        Set $y^{i+1} = y^i + \alpha^i p^i$.
    Step 2.5. (Test on the minimization on the cone$\{D\}$)
        If $i = r$, go to Step 3;
        otherwise set $i = i + 1$ and go to Step 2.2.
Step 3. (Iterate outputs)
    Set $\breve{x} = y^{i+1}$.
    Set $\breve{\alpha}^0 = \tilde{\alpha}^{i+1}$ and $\breve{\alpha}^{max} = \max_{i=1,\dots,r+1}\{\tilde{\alpha}^i\}$; choose $\breve{\mu} = \min\{\mu, (\breve{\alpha}^{max})^{q_1}\}\}$.
    return $(\breve{x}, \breve{\mu}, \breve{\alpha}^0, \breve{\alpha}^{max})$.
```

Therefore the expansion step in Step 2.3 is defined as follows:

```
Expansion step $(\bar{\alpha}^i, \hat{\alpha}^i, y^i, p^i) \mapsto \alpha$
Data. $\gamma > 0$, $\delta \in (0,1)$.
Step 1. Set $\alpha = \hat{\alpha}^i$.
Step 2. Let $\tilde{\alpha} = \min\{\bar{\alpha}^i, (\alpha/\delta)\}$.
Step 3. If $y^i + \tilde{\alpha} p^i \notin \mathcal{S}_\alpha$ or $\alpha = \bar{\alpha}^i$ or

            $Z\left(y^i + \tilde{\alpha} p^i; \mu, \epsilon\right) \ge Z(y^i; \mu, \epsilon) - \gamma\left(\tilde{\alpha}\right)^2$

    return $\alpha$.
Step 4. Set $\alpha = \tilde{\alpha}$ and go to Step 2.
```

We note that the iteration map $\mathcal{M}$ takes, as input arguments, current values for the iterate x, the smoothing parameter μ, the initial step size $\tilde{\alpha}^0$, the penalty parameter ϵ, and exponent q_1 and returns, as output arguments, the newly computed iterate $\check{x}$, smoothing parameter $\check{\mu}$, initial stepsize for subsequent calls $\check{\alpha}^0$, and maximum stepsize $\check{\alpha}^{max}$. For a thorough description of the iteration map $\mathcal{M}$ we refer the interested reader to [21].

On the basis of the iteration map $\mathcal{M}$ so far described, we can define our derivative-free method for the solution of problem (1).

```
**Algorithm.** DeFCon
**Data.** $\tilde{x} \in \mathcal{S}_\alpha$, $\tilde{\alpha}_0^0 > 0$, $\epsilon_0 > 0$, $\mu_{max} > 0$, $\gamma > 0$, $\delta \in (0,1)$, $\theta \in (0,1)$, $0 < q_1 < q_2 < 1$, and $\tau \in (0,1)$.
**Step 0.** Set $\mu_0 = \mu_{max}$, $x_0 = \tilde{x}$, $j = 0$, and $\epsilon = \epsilon_j$.
**Step 1.** Set $k = 0$.
**Step 2.** (Main iteration)
    Compute $\mathcal{M}(x_k, \mu_k, \tilde{\alpha}_k^0, \epsilon, q_1) \mapsto (x_{k+1}, \mu_{k+1}, \tilde{\alpha}_{k+1}^0, \tilde{\alpha}_{k+1}^{max})$.
    Set $k = k + 1$.
**Step 3.** (Penalty parameter testing)
    If $\dfrac{(\tilde{\alpha}_k^{max})^{q_2}}{\mu_k} < \min\{\epsilon, \max\{0, g_1(x_k), \dots, g_m(x_k)\}\}$, then set $\epsilon = \tau \dfrac{(\tilde{\alpha}_k^{max})^{q_2}}{\mu_k}$.
        If $Z(\tilde{x}; \mu_k, \epsilon) \le Z(x_k; \mu_k, \epsilon)$, then set $x_0 = \tilde{x}$ else $x_0 = x_k$.
        Set $\epsilon_{j+1} = \epsilon$, $j = j + 1$, $\mu_0 = \mu_k$ and go to Step 1.
    Else go to Step 2.
```

As mentioned earlier in this section, the crucial aspect of Algorithm DeFCon resides in Step 3, that is, in the penalty parameter testing and updating formula.

We remark that the requirement that $0 < q_1 < q_2 < 1$ is essential to prove convergence. In particular, $q_1 \in (0,1)$ is required to prove convergence of the method proposed in [21]; $q_2 \in (0,1)$ is needed to prove that the penalty parameter is updated only a finite number of times. Finally, $q_1 < q_2$ is essential to prove that a feasible point is obtained in the limit by Algorithm DeFCon.

The quantity $(\tilde{\alpha}_k^{max})^{q_2}$ can be viewed as a stationarity measure of the current iterate with respect to the smoothing function (see [16]). Then, on the basis of the analysis carried out in [16], we can say that $(\tilde{\alpha}_k^{max})^{q_2}/\mu_k$ roughly measures the stationarity of the current iterate with respect to problem (4). Thus, the rationale behind the penalty parameter updating is to decrease ϵ whenever an improvement of the quality of the solution of problem (4) does not correspond to a reduction of the infeasibility of the current iterate with respect to problem (1). Note, in particular, that if x_k is feasible with respect to the nonlinear inequality constraints, then the penalty parameter is left unchanged.

The following proposition is an important result in that it guarantees that the penalty parameter ϵ is reduced finitely many times.

Proposition 10. *Let $J = \{0, 1, \dots\}$ be the index set generated by Algorithm DeFCon at Step* 3*. Then, J is finite.*

Proof. Suppose that, each time that the penalty parameter satisfies the condition tested at Step 3 and before incrementing the counter j, the following quantities are stored: $\sigma_j^{max} = \tilde{\alpha}_k^{max}$, $\tilde{\sigma}_j^i = \tilde{\alpha}_j^i$, $\sigma_j^i = \alpha_j^i$ for all $i = 1, \dots, r_{k-1} + 1$, $w_j^i = y_{k-1}^i$ and $d_j^i = p_{k-1}^i$ for all $i = 1, \dots, r_{k-1}$, and $t_j = r_{k-1}$, $z_j = x_k$, $\rho_j = \mu_k$. For the sake of completeness, we set $\rho_{-1} = \mu_{max}$.

Reasoning by contradiction, we suppose that J is infinite. By the test of Step 3 of Algorithm DeFCon we have that

$$\frac{(\sigma_j^{max})^{q_2}}{\rho_j} \le \epsilon_j;$$

hence

$$\epsilon_{j+1} = \tau \frac{(\sigma_j^{max})^{q_2}}{\rho_j} \le \tau \epsilon_j$$

so that we get

$$\lim_{j\to\infty} \epsilon_j = 0. \tag{36}$$

Let $\{z_j\}$ be the sequence produced by Algorithm DeFCon. Then, by Assumption 1, an accumulation point $\bar{z} \in \bar{\mathcal{S}}_\alpha$ exists. Let us relabel $\{z_j\}$ the subsequence which converges to $\bar{z}$. Suppose, furthermore, that $\bar{z} \in \partial\mathcal{D}_\alpha$. By the instructions at Step 3 of Algorithm DeFCon, iteration map $\mathcal{M}$, and by point (i) of Proposition 6, we get that

$$Z(z_j;\epsilon_j) \le Z(z_j;\rho_j,\epsilon_j) \le Z(x_0^{(j)};\mu_0^{(j)},\epsilon_j) \le Z(x_0^{(j)};\rho_{j-1},\epsilon_j), \tag{37}$$

where we denote by $\{x_k^{(j)}\}$ and $\{\mu_k^{(j)}\}$ the sequences produced by Algorithm DeFCon when $\epsilon = \epsilon_j$.

Furthermore, by the second test at Step 3 of Algorithm DeFCon and by relation (12), we get

$$Z(x_0^{(j)};\rho_{j-1},\epsilon_j) \le Z(\tilde{x};\rho_{j-1},\epsilon_j) \le Z(\tilde{x};\epsilon_j) + \rho_{j-1}\ln m. \tag{38}$$

Hence, by (37) and (38) and multiplying by ϵ_j, we obtain

$$\epsilon_j Z(z_j;\epsilon_j) \le \epsilon_j Z(\tilde{x};\epsilon_j) + \epsilon_j\rho_{j-1}\ln m. \tag{39}$$

Since, when $j \to \infty$, $z_j \to \bar{z} \in \partial\mathcal{D}_\alpha$, an index $i \in \{1,\dots,m\}$ must exist such that $g_i(z_j) \to \alpha_i$ so that $\hat{g}_i(z_j;\epsilon_j) \to +\infty$. Therefore, we get that

$$\lim_{j\to\infty} \epsilon_j Z(z_j;\epsilon_j) = \lim_{j\to\infty} \max\{0, \hat{g}_1(z_j;\epsilon_j),\dots,\hat{g}_m(z_j;\epsilon_j)\} = +\infty. \tag{40}$$

Noting that, by the expression of $\hat{g}_i(x;\epsilon)$, $\lim_{j\to\infty} \hat{g}_i(\tilde{x};\epsilon_j) = g_i(\tilde{x})$, $i = 1,\dots,m$, we can write that

$$\lim_{j\to\infty} \epsilon_j Z(\tilde{x};\epsilon_j) + \epsilon_j\rho_{j-1}\ln m = \max\{0, g_1(\tilde{x}),\dots,g_m(\tilde{x})\} < +\infty. \tag{41}$$

Thus, (39), (40), and (41) prove that $\bar{z}$ cannot be on $\partial\mathcal{D}_\alpha$; therefore $\bar{z} \in \mathcal{S}_\alpha$.

Now, the test and the instructions at Step 3 of Algorithm DeFCon yield that for every index j,

$$\frac{(\sigma_j^{max})^{q_2}}{\rho_j} < \min\Big\{\epsilon, \max\{0, g_1(z_j),\dots,g_m(z_j)\}\Big\} \le \epsilon = \epsilon_j, \tag{42}$$

which, recalling that the sequence $\{\rho_j\}$ is bounded above, implies that $\lim_{j\to\infty}(\sigma_j^{max})^{q_2} = 0$; hence

$$\lim_{j\to\infty} \sigma_j^{max} = 0. \tag{43}$$

Now we show that $\lim_{j\to\infty} w_j^i = \bar z$ for all $i = 1,\dots,t_j$. To this aim, recalling the definition of σ_j^{max} and the instructions at Step 2.3 of iteration map $\mathcal{M}$, we can write

$$\|w_j^i - z_j\| \le t_j\sigma_j^{max} \qquad \forall\ i \in \{1,\dots,t_j\},$$

which, by (43) and by the boundedness of t_j, by Assumption 3, yield

$$\lim_{j\to\infty} \|w_j^i - z_j\| = 0 \quad \forall\ i \in \{1,\dots,t_j\}. \tag{44}$$

Hence, by the fact that $z_j \to \bar z$, we obtain that

$$w_j^i \to \bar z \quad \forall\ i = 1,\dots,t_j. \tag{45}$$

By (45), (43), and the fact that $\bar z \in \mathcal{S}_\alpha$, we have that, for sufficiently large values of j, $w_j^i + \tilde\sigma_j^i d_j^i \in \mathcal{S}_\alpha$ and $w_j^i + \sigma_j^i d_j^i \in \mathcal{S}_\alpha$. We recall that point (ii) of Proposition 6 in [21] holds. Therefore, by the instructions of Step 3 and (44), we obtain that, for sufficiently large values of j, either

$$w_j^i + \frac{\sigma_j^i}{\delta} d_j^i \in \mathcal{S}_\alpha \text{ and } Z\left(w_j^i + \frac{\sigma_j^i}{\delta} d_j^i;\rho_j,\epsilon_j\right) \ge Z(w_j^i;\rho_j,\epsilon_j) - \gamma\left(\frac{\sigma_j^i}{\delta}\right)^2$$

or

$$w_j^i + \tilde\sigma_j^i d_j^i \in \mathcal{S}_\alpha \text{ and } Z\left(w_j^i + \tilde\sigma_j^i d_j^i;\rho_j,\epsilon_j\right) \ge Z(w_j^i;\rho_j,\epsilon_j) - \gamma\left(\tilde\sigma_j^i\right)^2$$

are satisfied. Now, setting $\xi_j^i = \frac{\sigma_j^i}{\delta}$ in the first case and $\xi_j^i = \tilde\sigma_j^i$ in the second one, we have, for sufficiently large values of j,

$$w_j^i + \xi_j^i d_j^i \in \mathcal{S}_\alpha \text{ and } Z\left(w_j^i + \xi_j^i d_j^i;\rho_j,\epsilon_j\right) \ge Z(w_j^i;\rho_j,\epsilon_j) - \gamma\left(\xi_j^i\right)^2. \tag{46}$$

From the updating formula for y^i in Step 2.4 of iteration map $\mathcal{M}$, we note that

$$\|w_j^i - z_j\| \le \sum_{l=1}^{i-1} \sigma_j^l \le \delta \sum_{l=1}^{i-1} \xi_j^l \le \delta t_j \max_{l=1,\dots,t_j}\{\xi_j^l\}, \tag{47}$$

from which we get that

$$\max_{i=1,\dots,t_j}\{\xi_j^i, \|z_j - w_j^i\|\} \le \max\{1,\delta t_j\} \max_{i=1,\dots,t_j}\{\xi_j^i\} \le t_j \max_{i=1,\dots,t_j}\{\tilde\sigma_j^i,\sigma_j^i\}. \tag{48}$$

From (42), we have that, for every index j,

$$\max_{i=1,\dots,t_j}\{(\tilde\sigma_j^i)^{q_2},(\sigma_j^i)^{q_2}\} = (\sigma_j^{max})^{q_2} < \rho_j\epsilon_j,$$

which implies that

$$(\epsilon_j\rho_j)^{1/q_2} > \sigma_j^{max}, \tag{49}$$

so that, by (48) and (49), we obtain $\max_{i=1,\dots,t_j}\{\xi_j^i, \|z_j - w_j^i\|\} < t_j(\epsilon_j\rho_j)^{1/q_2}$, from which, recalling that $0 < q_2 < 1$, we get

$$\lim_{j\to\infty} \frac{\max_{i=1,\dots,t_j}\{\xi_j^i, \|z_j - w_j^i\|\}}{\epsilon_j\rho_j} = 0. \tag{50}$$

By considering (36), (46), and (50), we have that the hypotheses of point (ii) of Proposition 9 are satisfied so that a $\bar{\jmath} \geq 0$ exists such that, for all $j \geq \bar{\jmath}$, z_j is feasible for problem (1).

On the other hand, by the instruction of Step 3 of Algorithm DeFCon, we have that, for every index j,

$$\frac{(\sigma_j^{max})^{q_2}}{\rho_j} < \max\{0, g_1(z_j), \ldots, g_m(z_j)\},$$

which means that $z_j \in \mathcal{S}_\alpha$ and $g(z_j) \not\leq 0$, for every index j. The latter contradicts what has just been proved, namely, that z_j is feasible for problem (1) for j sufficiently large, thus completing the proof. ▯

The previous proposition guarantees that, after finitely many times, the test at Step 3 of Algorithm DeFCon is never satisfied so that the penalty parameter ϵ stays fixed at its last value, say $\epsilon_{\bar{\jmath}}$, and $\{\epsilon_j\}$, $\{z_j\}$, $\{\rho_j\}$ are all finite sequences. Therefore, from now on, we shall assume that $\epsilon = \epsilon_{\bar{\jmath}}$.

The following proposition describes some properties concerning the sequences of points and of objective function values generated by Algorithm DeFCon and the sampling technique adopted.

PROPOSITION 11 (see [21]). *Let $\{x_k\}$, $\{\mu_k\}$ be the sequences generated by Algorithm* DeFCon *when $\epsilon = \epsilon_{\bar{\jmath}}$. Then*

(a) *$\{x_k\}$ is well defined;*

(b) *the sequence $\{x_k\}$ is bounded;*

(e) *the following limits hold:*

$$\lim_{k\to\infty} \max_{i=1,\ldots,r_k} \left\{\alpha_k^i\right\} = 0, \tag{51}$$

$$\lim_{k\to\infty} \max_{i=1,\ldots,r_k} \left\{\tilde{\alpha}_k^i\right\} = 0, \tag{52}$$

$$\lim_{k\to\infty} \max_{i=1,\ldots,r_k} \left\|x_k - y_k^i\right\| = 0. \tag{53}$$

As shown in [21], by carrying out the convergence analysis, a significant role is played by the index set K defined as follows:

$$K = \{k : \mu_{k+1} < \mu_k\}. \tag{54}$$

Indeed, the following proposition shows that every accumulation point of the sequence $\{x_k\}_K$ is a KKT point for problem (1).

PROPOSITION 12. *Let $\{x_k\}$ be the sequence generated by Algorithm DeFCon when $\epsilon = \epsilon_{\bar{\jmath}}$. Then the sequence $\{x_k\}$ is bounded and every accumulation point $\bar{x}$ of $\{x_k\}_K$, where K is defined by* (54)*, is a KKT point for problem* (1).

Proof. First of all we prove that $\bar{x}$ is feasible for problem (1). Since ϵ is no longer updated, from the instruction of Step 3 of Algorithm DeFCon we know that, for every index $k \in K$,

$$0 \leq \min\left\{\epsilon_{\bar{\jmath}}, \max\{0, g_1(x_{k+1}), \ldots, g_m(x_{k+1})\}\right\} \leq \frac{(\tilde{\alpha}_{k+1}^{max})^{q_2}}{\mu_{k+1}},$$

and, by the instruction at Step 3 of iteration $\mathcal{M}$,

$$\frac{(\tilde{\alpha}_{k+1}^{max})^{q_2}}{\mu_{k+1}} = (\tilde{\alpha}_{k+1}^{max})^{q_2 - q_1},$$

which, taking the limit for $k \to \infty, k \in K$, recalling the results of Proposition 11, and the fact that $q_1 < q_2$, implies that

$$\lim_{k\to\infty, k\in K} \max\{0, g_1(x_{k+1}), \dots, g_m(x_{k+1})\} = 0. \tag{55}$$

Now let $\bar{x}$ be an accumulation point of sequence $\{x_{k+1}\}_K$; that is, an infinite index set $\hat{K} \subseteq K$ exists such that

$$\lim_{k\to\infty, k\in\hat{K}} x_{k+1} = \bar{x}.$$

On account of relation (55), we have that

$$\lim_{k\to\infty, k\in\hat{K}} \max\{0, g_1(x_{k+1}), \dots, g_m(x_{k+1})\} = 0,$$

which means that $\bar{x}$ is such that $\bar{x} \in \mathcal{F}$.

By employing (53) of Proposition 11 and the definition of iteration $\mathcal{M}$, we have that

$$\lim_{k\to\infty} \|x_k - x_{k+1}\| = 0.$$

Hence, we know that

$$\lim_{k\to\infty, k\in\hat{K}} x_k = \lim_{k\to\infty, k\in\hat{K}} x_{k+1},$$

so that $\lim_{k\to\infty, k\in\hat{K}} x_k = \bar{x} \in \mathcal{F}$.

Finally, we show that $\bar{x}$ is a KKT point for problem (1). By the instructions of iteration map $\mathcal{M}$, every point $x_k \in \mathcal{S}_\alpha$: whose closure is compact by Assumption 1. Hence, the sequence $\{x_k\}$ is bounded and therefore it admits limit points.

Now let $\bar{x}$ be any accumulation point of the subsequence $\{x_k\}_K$, where K is defined by (54). By the first part of the proof, we know that $\bar{x} \in \mathcal{F}$. Furthermore, by Corollary 1 of [21], we have that $\bar{x}$ is a stationary point of the exact penalty function $Z(x; \epsilon)$, so that, by Proposition 3, $\bar{x}$ is a KKT point for problem (1). □

5. Case study: Constrained parameter estimation for glucose kinetics model. The aim of this paper is mainly theoretical. Thus, the development of an efficient code based on the proposed algorithm and the analysis of its numerical performance are beyond the scope of the present paper. We refer to [12] for a numerical experimentation on standard test problems. In [12] an asynchronous parallel generating set search approach has been used to minimize many different penalty functions. The influence of the penalty function has been analyzed from a computational point of view. In particular, Griffin and Kolda employ as penalty functions the nondifferentiable ℓ_1, ℓ_2, and ℓ_∞ plus their smoothed versions s_1, s_2, and s_∞. All of these penalization techniques are compared against the standard ℓ_2^2 differentiable penalty function. An extensive numerical experimentation is carried out on a large set of test problems from the CUTEr collection [11]. The conclusion of [12] is that the use of a smooth approximation s_∞ of an ℓ_∞ exact penalty function leads to a derivative-free algorithm which shows a good compromise between quality of the final point and number of function evaluations required to get convergence.

Encouraged by these results we wanted to understand if the proposed derivative-free algorithm was able to efficiently solve a real world application. To this aim, we used a rough MATLAB implementation of Algorithm DeFCon to solve a problem connected with the study of an insulin-glucose model of the human body. To solve the problem we also used the constrained nonlinear minimization MATLAB routine fmincon and the freely available derivative-free MATLAB package NOMADm [2]. The latter is a recent implementation of a class of mesh-adaptive direct search (MADS) algorithms for solving nonlinear and mixed variable optimization problems with general nonlinear constraints. In particular, the employment of fmincon is useful to understand to what extent the unavailability of the derivatives can be overcome by finite difference approximations. On the other hand, the comparison with the derivative-free package NOMADm is needed to point out that the proposed algorithm is at least as efficient as an existing derivative-free code.

The study and understanding of circulatory models of glucose kinetics are of great importance in medicine and biology. Such models study the response of body tissues to an impulsive injection of a glucose bolus of known quantity. The intravenous glucose tolerance test (IVGTT) is a simple and standardized test that allows one to measure the reaction of the organism to the mentioned impulsive perturbation of the steady state. IVGTT has a documented ability to assess the functioning of the key organs involved in glucose homeostasis. Moreover, it is a powerful tool in the study of diabetes mellitus in that it is able to provide information on beta-cell function and insulin sensitivity (both peripheral and hepatic).

The IVGTT experimental protocol used prescripts the following operations [27]:

- Collection of 3 ml blood basal samples at -30, -15, and 0 min. to glucose injection.
- Injection of a 300 mg/kg glucose bolus at 0 min. immediately after the collection of the last basal sample.
- Infusion, at 20 min. from injection, of $0.03U$/kg insulin at a constant rate for 5 minutes.
- Collection of 3 ml blood samples at 2, 3, 4, 5, 6, 8, 10, 15, 20, 25, 30, 40, 60, 80, 100, 120, 140, 160, 180, 210, and 240 min. from injection of glucose, for measurements of glucose, glucose tracer, insulin, and C-peptide concentrations.

In the circulatory model of glucose kinetics studied in [24, 26, 25], the body tissues are lumped into two blocks. The heart-lungs block represents the heart chambers and the lungs, i.e., the tissues in between the right atrium and left ventricle. The periphery block represents all the remaining tissues, nourished by the entire arterial tree originating from the left ventricle (including the heart tissues nourished by the coronaries).

The dynamics of the exogenous arterial glucose concentration during the IVGTT can be modelled by the following system of ordinary differential equations:

$$\frac{dG_A(t)}{dt} = -\lambda G_A(t) + \lambda\Big[G_1(t) + G_2(t) + J/F\Big], \tag{56a}$$

$$\frac{dG_1(t)}{dt} = -\alpha_1 G_1(t) + \alpha_1\vartheta\Big[1 - E_b - \gamma Z(t)\Big]G_A(t), \tag{56b}$$

$$\frac{dG_2(t)}{dt} = -\alpha_2 G_2(t) + \alpha_2(1-\vartheta)\Big[1 - E_b - \gamma Z(t)\Big]G_A(t), \tag{56c}$$

$$\frac{dZ(t)}{dt} = -\beta Z(t) + \beta\Big[I(t) - I_b\Big], \quad Z(0) = 0. \tag{56d}$$

Here, $G_A(t)$ denotes the exogenous arterial glucose concentration, the sum between $G_1(t)$ and $G_2(t)$ denotes the mixed-venous glucose concentration, $Z(t)$ denotes the increment from the basal level of whole-body insulin fractional extraction, $I(t)$ is the insulin concentration during the exam, $J = 300\,\text{mg/kg}$ is the known intravenous glucose infusion, $F = 2688\,\text{ml}\cdot\text{min}^{-1}\cdot m^{-2}$ is the cardiac output, $\lambda = 3.84\,\text{min}^{-1}$ is the reciprocal of the mean heart-lungs transit time [17] and is patient-independent, and I_b is the basal value of insulin concentration and is patient-specific; we used $I_b = 50\,\text{pmol/l}$. The numerical solution of the system of ODE (56) requires that the insulin concentration increment $I(t) - I_b$ in (56d) is available at every time instant. For this purpose, the measured values of $I(t) - I_b$ have been smoothed and interpolated by a continuous function of time as detailed in [28]. Finally, $\alpha_1, \alpha_2, \beta, \gamma, \vartheta, E_b$ are the model parameters to be estimated in such a way that $G_A(t)$ approximates as well as possible the measurements gathered during the IVGTT. The model parameters, as suggested by biomedical engineers, can be sensibly bounded both from below and above as follows:

$$0.5 \le \alpha_1 \le 5, \qquad 0.01 \le \alpha_2 \le 0.5, \qquad 0.3 \le \vartheta \le 0.9, \qquad 0.01 \le \beta \le 1,$$
$$0.01 \le E_b \le 0.1, \qquad 5 \cdot 10^{-5} \le \gamma \le 5 \cdot 10^{-4}.$$

Let us define $t = (2, 3, 4, 5, 6, 8, 10, 15, 20, 25, 30, 40, 60, 80, 100, 120, 140, 160, 180, 210, 240)^\top$, and let g_i denote the exogenous glucose concentration measured at time t_i during the IVGTT. Then

$$f(\alpha_1, \alpha_2, \beta, \gamma, \vartheta, E_b) = \sum_{i=1}^{21} (G_A(t_i) - g_i)^2$$

is the sum of squared errors between model prediction and actual measurements. Moreover, among all the possible models, we are interested in those for which the mean periphery transit time $\vartheta/\alpha_1 + (1-\vartheta)/\alpha_2$ is greater than or equal to 2.5 minutes. This time limit is necessary to prevent models that are not representative of a human patient with medium body mass index. Thus, we end up with the following constrained problem:

$$\begin{array}{ll} \min f(\alpha_1, \alpha_2, \beta, \gamma, \vartheta, E_b) & \\ \text{s.t.} \quad g(\alpha_1, \alpha_2, \vartheta) = \vartheta/\alpha_1 + (1-\vartheta)/\alpha_2 \ge 2.5, & \\ \qquad 0.5 \le \alpha_1 \le 5, & \\ \qquad 0.01 \le \alpha_2 \le 0.5, & \\ \qquad 0.3 \le \vartheta \le 0.9, & \\ \qquad 0.01 \le \beta \le 1, & \\ \qquad 0.01 \le E_b \le 0.1, & \\ \qquad 5 \cdot 10^{-5} \le \gamma \le 5 \cdot 10^{-4}. & \end{array} \tag{57}$$

The circulatory model of glucose has been implemented in MATLAB/Simulink [29]. The solution of the system of ODE (56), which is at the basis of the model, is done numerically with a precision ξ that can be set by the user.

We started our experimentation by comparing the outcomes of Algorithm DeFCon with that of the constrained nonlinear minimization MATLAB routine fmincon and of the derivative-free package NOMADm, selecting a precision level of the ODE solver $\xi = 10^{-3}$. We ran fmincon and NOMADm by using default values for their parameters apart from the tolerances in the stopping criterion which we set to 10^{-6} as for Algorithm DeFCon. Moreover, both fmincon and NOMADm were ran by appropriately specifying the scale of the optimization variables.

Initial values for the parameters, as suggested by biomedical engineers, are

α_1	α_2	β	γ	ϑ	E_b
1.0089	0.40794	0.16481	$3.932 \cdot 10^{-4}$	0.84496	0.020991

In Figure 1 we report the actual measurements of exogenous glucose during IVGTT as crosses and plot the curve $G_A(t)$ obtained in correspondence to the initial values of the parameters as listed above.

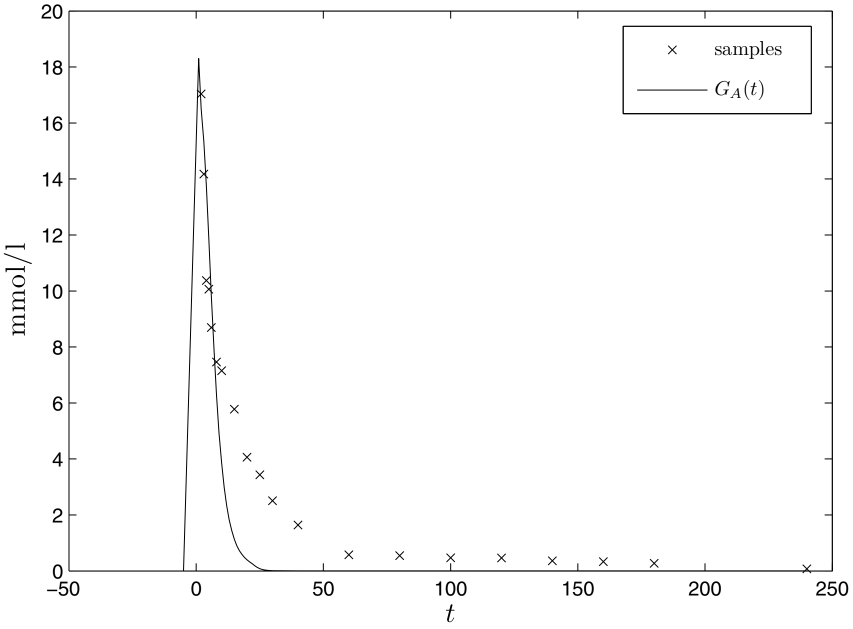


FIG. 1. *Initial configuration.*

Starting from this initial point, the proposed derivative-free algorithm (DeFCon), NOMADm, and the MATLAB routine fmincon yield the results reported in Table 1, where (n.it.) and (n.f.) denote, respectively, the number of iterations and number

TABLE 1
Results obtained by Algorithm DeFCon, NOMADm, and fmincon for $\xi = 10^{-3}$.

	DeFCon	NOMADm	fmincon
n.it.	46	144	12
n.f.	627	480	245
$\alpha_1^\star$	2.0228	1.3995	0.80479
$\alpha_2^\star$	0.11881	0.081768	0.011788
$\beta^\star$	0.055959	0.010513	1.0
$\gamma^\star$	$1.6529 \cdot 10^{-4}$	$1.4906 \cdot 10^{-4}$	$5.0 \cdot 10^{-5}$
$\vartheta^\star$	0.7409	0.84496	0.9
$E_b^\star$	0.01	0.052241	0.01
$f^\star$	2.0959	5.4409	27.6094
$g^\star$	2.547	2.5	9.6017

of functions evaluations. $\alpha_1^\star, \alpha_2^\star, \beta^\star, \gamma^\star, \vartheta^\star$, and $E_b^\star$ represent the final values of the model parameters to be estimated. Finally, $f^\star$ and $g^\star$ represent, respectively, the final values of the objective function and of the nonlinear constraint (which should be greater than or equal to 2.5). Looking at the results it can be noted that, even though all the methods manage to achieve feasibility of the final point, Algorithm DeFCon and NOMADm are able to produce a point whose objective function value is much better than that produced by fmincon. Algorithm DeFCon and NOMADm produce almost the same points. Indeed, the parameter values obtained by Algorithm DeFCon and NOMADm yield almost the same curves $G_A(t)$ even though Algorithm DeFCon reaches a better objective function value than that achieved by NOMADm, which converges in fewer function evaluations. As concerns the result computed by fmincon, it yields a curve $G_A(t)$ which is substantially different in terms of approximation of the glucose measurements (see Figure 2) from that yielded by Algorithm DeFCon and NOMADm. Comprehensibly, this better behavior of Algorithm DeFCon and NOMADm over fmincon is achieved at the expense of a higher computational burden and points out the noisy nature of the approximation problem which is at the basis of the inefficiency of fmincon.

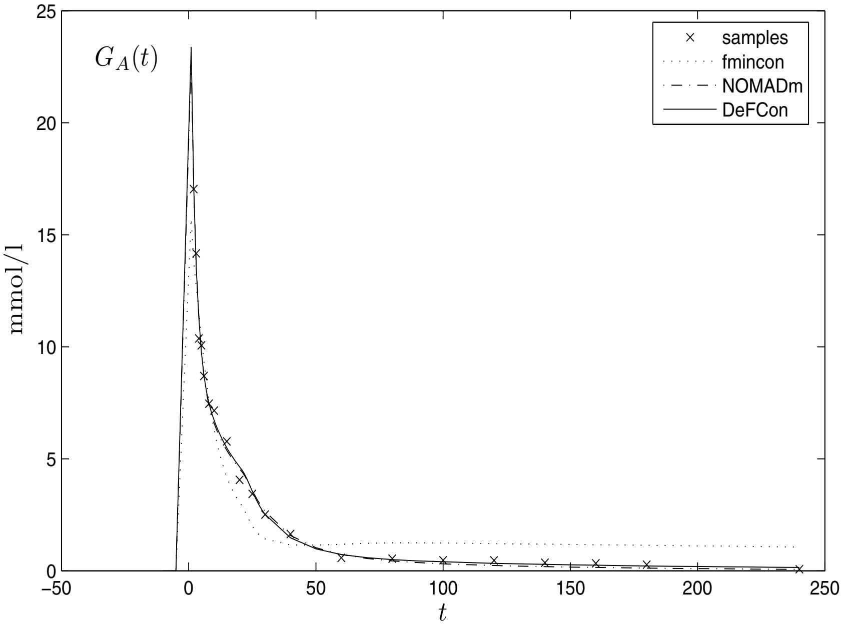


Fig. 2. *Optimal curves.*

The inefficiency of the MATLAB solver fmincon along with the modest number of iterations and function evaluations to get convergence might indicate that the ODE solver tolerance is too high for estimation of first order derivatives by finite differences to be reliable. Hence, we tried to solve the problem with increasing precision levels for the ODE solver; namely we set the precision $\xi = 10^{-4}, 10^{-5}, 10^{-6}$ and compared the results in Table 2.

As concerns the above comparison, we first note that there is only a slight change

TABLE 2

Comparison between Algorithm DeFCon, NOMADm, and fmincon.

ξ	10^{-6}			10^{-5}			10^{-4}		
	DeFCon	NOMADm	fmincon	DeFCon	NOMADm	fmincon	DeFCon	NOMADm	fmincon
n.it.	97	1211	21	68	1253	9	57	187	9
n.f.	1410	3408	316	994	3542	157	827	562	146
$\alpha_1^\star$	2.1468	1.9386	2.2779	2.1659	1.9386	0.58594	2.1261	1.5948	0.5
$\alpha_2^\star$	0.12418	0.11473	0.049484	0.12401	0.11473	0.012527	0.12318	0.091778	0.036089
$\beta^\star$	0.067333	0.046646	0.95153	0.067373	0.046646	1.0	0.06504	0.015396	0.01
$\gamma^\star$	$1.5 \cdot 10^{-4}$	$1.5 \cdot 10^{-4}$	$5.0 \cdot 10^{-5}$	$1.5 \cdot 10^{-4}$	$1.5 \cdot 10^{-4}$	$5.0 \cdot 10^{-5}$	$1.5 \cdot 10^{-4}$	$1.5 \cdot 10^{-4}$	$5.0 \cdot 10^{-4}$
$\vartheta^\star$	0.73186	0.75805	0.86005	0.73182	0.75805	0.9	0.73407	0.81762	0.89948
$E_b^\star$	0.01	0.020991	0.033761	0.01	0.020991	0.01	0.01	0.046137	0.01
$f^\star$	2.0061	2.3219	63.7352	2.0075	2.3217	69.183	2.0146	4.1823	103.9875
$g^\star$	2.5002	2.5000	3.2058	2.5005	2.5000	9.5193	2.504	2.5001	4.5842

in the points produced by Algorithm DeFCon and NOMADm. Namely, as the ODE solver precision ξ increases, Algorithm DeFCon and NOMADm, though requiring more iterations and function evaluations to converge, produce points which are very close to each other. This is confirmed by the objective and constraint function values which gain more and more accuracy as the precision ξ becomes finer. However, Algorithm DeFCon seems to be more efficient than NOMADm when the precision ξ is less than or equal to 10^{-5}. Slightly better results both for Algorithm DeFCon and NOMADm can be obtained by performing a tuning of their parameters.

On the contrary, fmincon exhibits a more unpredictable behavior converging to points that are largely different from each other in terms of parameter, objective, and constraint function values. The outcomes of fmincon seem to be unrelated to the precision level of the ODE solver apart for the fact that the computational burden increases as ξ gets finer. This inefficiency of fmincon is most probably due to the lack of derivative knowledge on the problem which fmincon tries to overcome by computing gradients by finite difference approximation. This, in turn, makes fmincon more subject to the numerical noise introduced by the ODE solver thus explaining the apparent instability of the code.

6. Conclusions. In this paper we presented a derivative-free algorithm for the solution of inequality constrained nonlinear programming problems. The method is based on the derivative-free minimization of a smooth approximation of a new (nondifferentiable) ℓ_∞ exact penalty function. We proved that the method is globally convergent towards a KKT point of the constrained problem. In order to stress the ability of our method to tackle real world problems, we reported the results obtained on a constrained problem concerning the parameter estimation of an insulin-glucose model of the human body. A comparison with another derivative-free optimization routine shows the effectiveness of the proposed method.

The convergence properties and the theoretical analysis of the proposed method has been carried out in the case where only inequality constraints are present. The method can be adapted to handle both equality and inequality constraints, preserving its convergence properties but at the expense of some nontrivial technicalities which considerably complicate the analysis. Furthermore, we remark that the realization of an efficient code was not the main aim of this paper. For this reason a fine tuning of the parameters and an efficient computation of the search directions have not been done but are the subject of continuing work.

Appendix.

Proof of Proposition 3. Since $\bar{x} \in \mathcal{F}$, then $B(\bar{x};\epsilon) = I_0(\bar{x})$. Therefore, by Proposition 2, we have

$$\left(\nabla f(\bar{x}) + \sum_{i \in I_0(\bar{x})} \frac{\lambda_i((\alpha_i - g_i(\bar{x}))^2 + \epsilon\alpha_i)}{\epsilon(\alpha_i - g_i(\bar{x}))^2} \nabla g_i(\bar{x})\right)^\top d \geq 0 \quad \forall\, d \in T(\bar{x}).$$

Then, by setting $\bar{\lambda}_i = \frac{\lambda_i((\alpha_i - g_i(\bar{x}))^2 + \epsilon\alpha_i)}{\epsilon(\alpha_i - g_i(\bar{x}))^2}$, $i \in I_0(\bar{x})$, and $\bar{\lambda}_i = 0$, $i \in \{1,\dots,m\} \setminus I_0(\bar{x})$, we have that there does not exist any direction $d \in R^n$ such that

$$\left(\nabla f(\bar{x}) + \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \nabla g_i(\bar{x})\right)^\top d < 0,$$
$$a_j^\top d \leq 0 \quad \forall j \in J(\bar{x}).$$

Hence, by using the Motzkin theorem [23], we have that $y_0 > 0$ and $\mu_j \geq 0$, $j \in J(\bar{x})$, exist such that

$$y_0 \left(\nabla f(\bar{x}) + \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \nabla g_i(\bar{x})\right) + \sum_{j \in J(\bar{x})} a_j \mu_j = 0.$$

The result follows by taking $\bar{\mu}_j = \mu_j / y_0$ for $j \in J(\bar{x})$, and $\bar{\mu}_j = 0$ for $j \notin J(\bar{x})$. □

In order to complete the proof of the exactness results of the penalty function $Z(x;\epsilon)$, we need some technical results which are reported in the following propositions.

Proposition 13. *Let $\hat{x} \in \mathcal{S}_\alpha$; then there exist numbers $\epsilon(\hat{x}) > 0$ and $\sigma(\hat{x}) > 0$ such that, for all $\epsilon \in (0, \epsilon(\hat{x})]$ and for all $x \in \mathcal{B}(\hat{x}, \sigma(\hat{x})) \cap \mathcal{S}_\alpha$ and $g(x) \not\leq 0$, there exists a direction $d \in T(\hat{x})$ satisfying $DZ(x, d; \epsilon) < 0$.*

Proof. By Assumption 2, we have that the hypotheses of Lemma 1 are satisfied at $\hat{x}$ for $I = I_\pi(\hat{x})$. Let $\mathcal{B}(\hat{x}, \rho)$ and $d \in T(\hat{x})$ be the neighborhood and the direction considered in Lemma 1. We have that $d \in T(\hat{x})$ is such that

$$\nabla \hat{g}_i(x;\epsilon)^\top d \leq -1 \tag{58}$$

for all $i \in I_\pi(\hat{x})$. By continuity, we can find a neighborhood $\mathcal{B}(\hat{x}, \sigma(\hat{x})) \subseteq \mathcal{B}(\hat{x}, \rho)$ such that, for $i \notin I_\pi(\hat{x})$ and $x \in \mathcal{B}(\hat{x}, \sigma(\hat{x})) \cap \mathcal{S}_\alpha$, we have $g_i(x) < 0$; it follows that $I_\pi(x) \subseteq I_\pi(\hat{x})$ for $x \in \mathcal{B}(\hat{x}, \sigma(\hat{x})) \cap \mathcal{S}_\alpha$.

Now let $x \in \mathcal{B}(\hat{x}, \sigma(\hat{x})) \cap \mathcal{S}_\alpha$ be an infeasible point, that is, $g(x) \not\leq 0$. Then, there must exist at least an index $i \in I_\pi(\hat{x})$ such that $g_i(x) > 0$ and $\hat{g}_i(x;\epsilon) > 0$, so that it results in $B(x;\epsilon) \subseteq I_\pi(x)$.

Therefore, recalling the expression of the directional derivative of $Z(x;\epsilon)$ and (58), we get

$$DZ(x, d; \epsilon) = \nabla f(x)^\top d + \frac{1}{\epsilon} \max_{i \in B(x;\epsilon)} \{\nabla \hat{g}_i(x;\epsilon)^\top d\} \leq \nabla f(x)^\top d - \frac{1}{\epsilon},$$

from which it follows that a value $\epsilon(\hat{x}) > 0$ exists such that, for all $\epsilon \in (0, \epsilon(\hat{x})]$ and $x \in \mathcal{B}(\hat{x}, \sigma(\hat{x})) \cap \mathcal{S}_\alpha$ with $x \notin \mathcal{F}$, it must hold that

$$DZ(x, d; \epsilon) < 0,$$

which concludes the proof. □

PROPOSITION 14. *Let $\bar{\lambda} \in R^m$ and $\bar{\mu} \in R^p$ be multipliers such that $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a KKT triple for problem* (1). *Then the following bound holds:*

$$\|\bar{\lambda}\|_q \leq \nabla f(\bar{x})^\top z,$$

where $z \in T(\bar{x})$ is a vector such that

$$\nabla g_i(\bar{x})^\top z \leq -1, \quad i \in I_0(\bar{x}). \tag{59}$$

Proof. From the fact that $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a KKT triple, it follows that, for any $z \in T(\bar{x})$ satisfying (59), we have

$$\nabla f(\bar{x})^\top z = -\sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \nabla g_i(\bar{x})^\top z - \sum_{j \in J(\bar{x})} \bar{\mu}_j a_j^\top z \geq 0.$$

Therefore, the following linear program and its dual are both feasible and bounded:

$$\begin{array}{ll} \min\limits_z & \nabla f(\bar{x})^\top z \\ & \nabla g_i(\bar{x})^\top z \leq -1, \quad i \in I_0(\bar{x}), \\ & a_j^\top z \leq 0, \qquad\qquad j \in J(\bar{x}), \end{array} \tag{60}$$

$$\begin{array}{ll} \max\limits_{u,v} & \displaystyle\sum_{i \in I_0(\bar{x})} u_i \\ & \displaystyle\sum_{i \in I_0(\bar{x})} \nabla g_i(\bar{x}) u_i + \sum_{j \in J(\bar{x})} a_j v_j = -\nabla f(\bar{x}), \\ & u, v \geq 0. \end{array} \tag{61}$$

Let $z^\star$ and $(u^\star, v^\star)$ be optimal solutions of (60) and (61), respectively. Recalling that every KKT multipliers (λ, μ) of problem (1) satisfy the constraints of problem (61), we then have

$$\|\bar{\lambda}\|_q \leq \|\bar{\lambda}\|_1 \leq \sum_{i \in I_0(\bar{x})} u_i^\star = \nabla f(\bar{x})^\top z^\star \leq \nabla f(\bar{x})^\top z$$

for any $z \in T(\bar{x})$ satisfying (59). □

PROPOSITION 15 (see [7, Proposition 8]). *A number Λ exists such that $\|\bar{\lambda}\|_\infty \leq \Lambda$ for all KKT triples $(\bar{x}, \bar{\lambda}, \bar{\mu})$ of problem* (1).

Proof. The proof follows using Proposition 14 and the same reasoning of Proposition 8 in [7]. □

Now, we can finally prove Propositions 4 and 5.

Proof of Proposition 4.

"If"-part: it follows from Proposition 10 in [9].

"Only if"-part: as $(\bar{x}, \bar{\lambda}, \bar{\mu})$ is a KKT triple for problem (1) we can write

$$\nabla f(\bar{x}) = -\left(\sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \nabla g_i(\bar{x}) + \sum_{j \in J(\bar{x})} \bar{\mu}_j a_j \right). \tag{62}$$

Recalling that $\bar{x} \in \mathcal{F}$ and that, by definition, $\hat{g}_0(x;\epsilon) = 0$, so that $B(\bar{x};\epsilon) = I_0(\bar{x}) \cup \{0\}$, the directional derivative of $Z(x;\epsilon)$ along direction d can be written as follows:

$$\begin{aligned} DZ(x,d;\epsilon) &= \nabla f(x)^\top d + \frac{1}{\epsilon} \max_{i \in B(x;\epsilon)} \{\nabla \hat{g}_i(x;\epsilon)^\top d\} \\ &= \nabla f(\bar{x})^\top d + \frac{1}{\epsilon} \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\}. \end{aligned}$$

By using (62) in the above expression we get

$$\begin{aligned} &DZ(\bar{x},d;\epsilon) = \frac{1}{\epsilon} \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} - \left(\sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \nabla g_i(\bar{x})^\top d + \sum_{j \in J(\bar{x})} \bar{\mu}_j a_j^\top d \right) \\ &\geq \frac{1}{\epsilon} \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} - \left(\sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \max\{\nabla g_i(\bar{x})^\top d, 0\} + \sum_{j \in J(\bar{x})} \bar{\mu}_j a_j^\top d \right). \end{aligned}$$

Whenever $d \in T(\bar{x})$, by definition of $T(\bar{x})$, we get

$$DZ(\bar{x},d;\epsilon) \geq \frac{1}{\epsilon} \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} - \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \max\{\nabla g_i(\bar{x})^\top d, 0\}.$$

By considering the expression of $\nabla \hat{g}_i(x;\epsilon)$, it results, for $i \in I_0(\bar{x})$,

$$\nabla \hat{g}_i(\bar{x};\epsilon) = \left(1 + \frac{\epsilon}{\alpha_i}\right) \nabla g_i(\bar{x}),$$

so that we obtain

$$DZ(\bar{x},d;\epsilon) \geq \frac{1}{\epsilon} \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} - \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \frac{\alpha_i}{\alpha_i + \epsilon} \max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}.$$

Now, recalling Proposition 15, we have that

$$\begin{aligned} \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \frac{\alpha_i}{\alpha_i + \epsilon} \max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\} &\leq \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} \sum_{i \in I_0(\bar{x})} \bar{\lambda}_i \\ &\leq \max_{i \in I_0(\bar{x})} \{\max\{\nabla \hat{g}_i(\bar{x};\epsilon)^\top d, 0\}\} m\Lambda, \end{aligned}$$

so that we can say that $\bar{x}$ is a critical point of problem (4) for all $\epsilon \in (0, \epsilon^\star]$, where $\epsilon^\star = 1/m\Lambda$.

Proof of Proposition 5. The proof follows by considering Propositions 3 and 13 and [8, 9]. □

REFERENCES

[1] C. Audet and J. E. Dennis, Jr., *A pattern search filter method for nonlinear programming without derivatives*, SIAM J. Optim., 14 (2004), pp. 980–1010.

[2] C. Audet and J. E. Dennis, Jr., *Mesh adaptive direct search algorithms for constrained optimization*, SIAM J. Optim., 17 (2006), pp. 188–217.

[3] C. Audet and J. E. Dennis, Jr., *A MADS Algorithm with a Progressive Barrier for Derivative-Free Nonlinear Programming*, Technical Report G-2007-37, Les Cachiers du GERAD, Montreal, CA, 2007.

[4] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*, Academic Press, New York, 1982.

[5] D. P. Bertsekas, *Nonlinear Programming*, 3rd ed., Athena Scientific, New York, 1999.

[6] A. R. Conn, N. I. M. Gould, and Ph. L. Toint, *A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds*, SIAM J. Numer. Anal., 28 (1991), pp. 545–572.

[7] G. Di Pillo and L. Grippo, *Globally Exact Nondifferentiable Penalty Functions*, Technical Report R.10.87, Department of Computer and Systems Science, University of Rome "La Sapienza," Rome, Italy, 1987.

[8] G. Di Pillo and L. Grippo, *On the exactness of a class of nondifferentiable penalty functions*, J. Optim. Theory Appl., 57 (1988), pp. 399–410.

[9] G. Di Pillo and L. Grippo, *Exact penalty functions in constrained optimization*, SIAM J. Control Optim., 27 (1989), pp. 1333–1360.

[10] R. Fletcher and S. Leyffer, *Nonlinear programming without a penalty function*, Math. Program., 91 (2002), pp. 239–269.

[11] N. I. M. Gould, D. Orban, and Ph. L. Toint, *Cuter and SIFDEC: A constrained and unconstrained testing environment, revisited*, ACM Trans. Math. Software, 29 (2003), pp. 373–394.

[12] J. D. Griffin and T. G. Kolda, *Nonlinearly-Constrained Optimization Using Asynchronous Parallel Generating Set Search*, Technical Report SAND2007-3257, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, 2007.

[13] S. P. Han and O. L. Mangasarian, *Exact penalty functions in nonlinear programming*, Math. Programming, 17 (1979), pp. 251–269.

[14] T. G. Kolda, R. M. Lewis, and V. Torczon, *Optimization by direct search: New perspectives on some classical and modern methods*, SIAM Rev., 45 (2003), pp. 385–482.

[15] T. G. Kolda, R. M. Lewis, and V. Torczon, *A Generating Set Direct Search Augmented Lagrangian Algorithm for Optimization with a Combination of General and Linear Constraints*, Technical Report SAND2006-5315, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, 2006.

[16] T. G. Kolda, R. M. Lewis, and V. Torczon, *Stationarity results for generating set search for linearly constrained optimization*, SIAM J. Optim., 17 (2006), pp. 943–968.

[17] N. A. Lassen and W. Perl, *Tracer Kinetic Methods in Medical Physiology*, Raven Press, New York, 1979.

[18] R. M. Lewis and V. Torczon, *Pattern search algorithms for bound constrained minimization*, SIAM J. Optim., 9 (1999), pp. 1082–1099.

[19] R. M. Lewis and V. Torczon, *Pattern search methods for linearly constrained minimization*, SIAM J. Optim., 10 (2000), pp. 917–941.

[20] R. M. Lewis and V. Torczon, *A globally convergent augmented Lagrangian pattern search algorithm for optimization with general constraints and simple bounds*, SIAM J. Optim., 12 (2002), pp. 1075–1089.

[21] G. Liuzzi, S. Lucidi, and M. Sciandrone, *A derivative-free algorithm for linearly constrained finite minimax problems*, SIAM J. Optim., 16 (2006), pp. 1054–1075.

[22] S. Lucidi, M. Sciandrone, and P. Tseng, *Objective-derivative-free methods for constrained optimization*, Math. Program., 92 (2002), pp. 37–59.

[23] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[24] A. Mari, *Circulatory models of intact-body kinetics and their relationship with compartmental and noncompartmental analysis*, J. Theor. Biol., 160 (1993), pp. 509–531.

[25] A. Mari, *Calculation of organ and whole-body uptake and production with the impulse response approach*, J. Theor. Biol., 174 (1995), pp. 341–353.

[26] A. Mari, *Determination of the single-pass impulse response of the body tissues with circulatory models*, IEEE Trans. Biom. Eng., 42 (1995), pp. 304–312.

[27] A. Mari, *Assessment of insulin sensitivity and secretion with the labelled intravenous glucose tolerance test: Improved modelling analysis*, Diabetologia, 41 (1998), pp. 1029–1039.

[28] A. Mari and A. Valerio, *A circulatory model for the estimation of insulin sensitivity*, Control Eng. Practice, 5 (1997), pp. 1747–1752.

[29] The MathWorks inc., *MATLAB, the Language of Technical Computing.*

[30] J. H. May, *Linearly Constrained Nonlinear Programming: A Solution Method That Does Not Require Analytic Derivatives*, Ph.D. thesis, Yale University, New Haven, CT, 1974.

[31] S. Xu, *Smoothing method for minimax problems*, Comput. Optim. Appl., 20 (2001), pp. 267–279.

SIAM J. OPTIM.
Vol. 20, No. 1, pp. 30–53

AN ADAPTIVE LINEAR APPROXIMATION ALGORITHM FOR COPOSITIVE PROGRAMS[*]

STEFAN BUNDFUSS[†] AND MIRJAM DÜR[‡]

Abstract. We study linear optimization problems over the cone of copositive matrices. These problems appear in nonconvex quadratic and binary optimization; for instance, the maximum clique problem and other combinatorial problems can be reformulated as such problems. We present new polyhedral inner and outer approximations of the copositive cone which we show to be exact in the limit. In contrast to previous approximation schemes, our approximation is not necessarily uniform for the whole cone but can be guided adaptively through the objective function, yielding a good approximation in those parts of the cone that are relevant for the optimization and only a coarse approximation in those parts that are not. Using these approximations, we derive an adaptive linear approximation algorithm for copositive programs. Numerical experiments show that our algorithm gives very good results for certain nonconvex quadratic problems.

Key words. copositive cone, copositive programming, quadratic programming, approximation algorithms

AMS subject classifications. 90C05, 90C20, 15A48, 15A63, 05C69

DOI. 10.1137/070711815

1. Introduction.

In this paper we are concerned with the topic of conic formulations and relaxations for binary and quadratic problems. Semidefinite relaxations have been proposed as a strong method to obtain good bounds for many combinatorial optimization problems. Quist et al. [21] suggested that one might get tighter relaxations by looking at cones other than the semidefinite one. Bomze et al. [3] were the first to observe that certain combinatorial problems like the maximum clique problem can equivalently be reformulated as a linear optimization problem over the cone of so-called completely positive matrices. A matrix A is called completely positive if it can be decomposed as $A = BB^T$ with an entrywise nonnegative matrix B. There is a large amount of papers on complete positivity in the linear algebra literature (a good survey is [1]), but the optimization community has only recently become aware of the connections between the fields.

The completely positive cone $\mathcal{C}^*$ is the dual cone of the cone $\mathcal{C}$ of copositive matrices. Formally, these cones are defined as

$$\mathcal{C} = \{A \in \mathcal{S} : x^T A x \geq 0 \text{ for all } x \in \mathbb{R}^n_+\}$$

(where $\mathcal{S}$ is the set of symmetric $n \times n$ matrices), and

$$\mathcal{C}^* = \left\{ \sum_{i=1}^{k} v_i v_i^T : v_i \in \mathbb{R}^n_+ \text{ for all } i = 1, \ldots, k \right\}.$$

Both $\mathcal{C}$ and $\mathcal{C}^*$ are closed, convex, pointed, full dimensional, nonpolyhedral cones. It can be shown that the interior of $\mathcal{C}$ is the set of strictly copositive matrices: $\operatorname{int}(\mathcal{C}) =$

[*] Received by the editors December 25, 2007; accepted for publication (in revised form) October 21, 2008; published electronically March 13, 2009.
http://www.siam.org/journals/siopt/20-1/71181.html

[†] Department of Mathematics, TU Darmstadt, Schloßgartenstr. 7, D–64289 Darmstadt, Germany (bundfuss@mathematik.tu-darmstadt.de).

[‡] Institute of Mathematics and Computer Science, University of Groningen, P.O. Box 407, 9700 AK Groningen, The Netherlands (M.E.Dur@rug.nl).

$\{A \in \mathcal{S} : x^T Ax > 0 \text{ for all } x \in \mathbb{R}^n_+ \setminus \{0\}\}$. The interior of $\mathcal{C}^*$ has recently been characterized in [10]. The extremal rays of $\mathcal{C}^*$ are known to be the rank one matrices vv^T with $v \geq 0$, while characterizing the extremal copositive matrices is an open problem. Both cones are related to the cones $\mathcal{N}$ of nonnegative symmetric matrices and $\mathcal{S}^+$ of symmetric positive semidefinite matrices, since

$$\mathcal{C} \supseteq \mathcal{S}^+ + \mathcal{N} \quad \text{and} \quad \mathcal{C}^* \subseteq \mathcal{S}^+ \cap \mathcal{N}.$$

Interestingly, for $n \times n$-matrices of order $n \leq 4$, equality holds in the above relations, whereas for $n \geq 5$, both inclusions are strict; see [1]. In contrast to $\mathcal{N}$ and $\mathcal{S}^+$, the cones $\mathcal{C}$ and $\mathcal{C}^*$ are not tractable: It is known that testing whether a given matrix is in $\mathcal{C}$ is co-NP-complete (cf. [16]). Consequently, restating a problem as an optimization problem over one of these cones does not resolve the difficulty of that problem. However, we believe that getting a good understanding of the conic formulations will help to improve the solution strategies for both binary and nonconvex quadratic problems. Moreover, in some cases copositive formulations motivate stronger semidefinite relaxations.

Up to now, the list of problems known to have representations as completely positive programs has grown to include standard quadratic problems [3], the stable set problem [15, 9], the quadratic assignment problem [20], and certain graph-partitioning problems [19]. Burer [6] showed the very general result that every quadratic problem with linear and binary constraints can be rewritten as such a problem. More precisely, he showed that a quadratic binary problem of the form

$$\begin{aligned} \min \quad & x^T Qx + 2c^T x \\ \text{s. t.} \quad & a_i^T x = b_i, \quad i = 1, \dots, m, \\ & x \geq 0, \\ & x_j \in \{0, 1\}, \quad j \in B, \end{aligned}$$

(with Q not necessarily positive semidefinite) can equivalently be written as the following linear problem over the cone of completely positive matrices:

$$\begin{aligned} \min \quad & \langle Q, X\rangle + 2c^T x \\ \text{s. t.} \quad & a_i^T x = b_i, \quad i = 1, \dots, m, \\ & \langle a_i a_i^T, X\rangle = b_i^2, \quad i = 1, \dots, m, \\ & x_j = X_{jj}, \quad j \in B, \\ & \begin{pmatrix} 1 & x \\ x & X \end{pmatrix} \in \mathcal{C}^*. \end{aligned}$$

This means that any nonconvex quadratic integer problem can equivalently be written as a linear problem over a convex cone, i.e., a convex optimization problem which has no nonglobal local optima. It is an open question whether problems with general quadratic constraints can similarly be restated as completely positive problems.

In this paper we develop an algorithm to solve the dual problem, i.e., the optimization problem over the copositive cone which can be stated in the form

$$\text{(CP)} \qquad \begin{aligned} \max \quad & \langle C, X\rangle \\ \text{s. t.} \quad & \langle A_i, X\rangle = b_i, \quad i = 1, \dots, m \\ & X \in \mathcal{C} \end{aligned}$$

with $C, A_i \in \mathbb{R}^{n \times n}, b_i \in \mathbb{R}$.

Our approach is based on new polyhedral inner and outer approximations of the copositive cone which we show to be exact in the limit. In contrast to previous approximation schemes, our approximation is not necessarily uniform for the whole cone but can be guided adaptively through the objective function, yielding a good approximation in those parts of the cone that are relevant for the optimization and only a coarse approximation in those parts that are not. Using these approximations, we derive an adaptive linear approximation algorithm for copositive programs. We show that our algorithm gives very good results for certain nonconvex quadratic problems.

Note that (CP) is related to the problem of testing whether a given matrix is in $\mathcal{C}^*$: From the fact that the cone $\mathcal{C}$ is the dual of $\mathcal{C}^*$ we have

$$\begin{aligned} A \notin \mathcal{C}^* &\Leftrightarrow \exists X \in \mathcal{C} : \langle A, X\rangle < 0 \\ &\Leftrightarrow \exists X \in \mathcal{C} : \langle I + E, X\rangle = 1, \langle A, X\rangle < 0 \\ &\Leftrightarrow \min\{\langle A, X\rangle : \langle I + E, X\rangle = 1, X \in \mathcal{C}\} < 0. \end{aligned}$$

(Here I denotes the identity and E the all ones matrix, and $\langle I + E, X\rangle = 1$ serves as a normalization constraint.) This minimization problem is of the form (CP), so an algorithm to solve (CP) can be used to decide whether or not $A \in \mathcal{C}^*$. It is an open question how a matrix A known to be in $\mathcal{C}^*$ can be factorized into $A = BB^T$, cf. [2] and [14] for attempts to answer this question.

1.1. Notation. Throughout the paper we use the following notation: The nonnegative orthant is denoted by $\mathbb{R}^n_+$, and the unit vectors are denoted by e_i. For a given vector v or matrix M, the relations $v \geq 0$ and $M \geq 0$ will be understood entrywise. We write $\mathcal{S}$ to denote the cone of symmetric matrices, $\mathcal{N} = \{A \in \mathcal{S} : A \geq 0\}$ to denote the cone of (entrywise) nonnegative matrices, and $\mathcal{S}^+ = \{A \in \mathcal{S} : A \succeq 0\}$ to denote the cone of positive semidefinite matrices. Dimensions of the cones will always be obvious from the context and therefore not stated explicitly. As usual, the inner product in $\mathcal{S}$ is defined as $\langle A, B\rangle := \operatorname{trace}(AB)$.

1.2. Relations to previous work. Since we will compare our algorithm to existing approaches, we briefly summarize previous work on copositive programming. Copositivity of a matrix is defined by positivity of a quadratic form, whence previous approaches have used various conditions which ensure positivity of polynomials.

For a given matrix $M \in \mathcal{S}$, consider the polynomial

$$P_M(x) := \sum_{i=1}^{n}\sum_{j=1}^{n} M_{ij} x_i^2 x_j^2.$$

Clearly, $M \in \mathcal{C}$ if and only if $P_M(x) \geq 0$ for all $x \in \mathbb{R}^n$. A sufficient condition for this is that $P_M(x)$ has a representation as a sum of squares (sos) of polynomials. Parrilo [17] showed that $P_M(x)$ allows a sum of squares decomposition if and only if $M \in \mathcal{S}^+ + \mathcal{N}$, yielding again the relation $\mathcal{S}^+ + \mathcal{N} \subseteq \mathcal{C}$. Using similar reasoning, Parrilo [17] defined the following hierarchy of cones (cf. also [15] and [4]) for $r \in \mathbb{N}$:

$$\mathcal{K}^r := \left\{ M \in \mathcal{S} : P_M(x) \left(\sum_{i=1}^{n} x_i^2\right)^r \text{ has an sos decomposition} \right\}.$$

Parrilo showed $\mathcal{S}^+ + \mathcal{N} = \mathcal{K}^0 \subset \mathcal{K}^1 \subset \dots$, and $\operatorname{int}(\mathcal{C}) \subseteq \bigcup_{r\in\mathbb{N}} \mathcal{K}^r$, so the cones $\mathcal{K}^r$ approximate $\mathcal{C}$ from the interior. Since the sos condition can be written as a system of linear matrix inequalities (LMIs), optimizing over $\mathcal{K}^r$ amounts to solving a semidefinite program (SDP).

Exploiting a different sufficient condition for nonnegativity of a polynomial, de Klerk and Pasechnik [15], cf. also Bomze and de Klerk [4], define

$$\mathcal{C}^r := \left\{ M \in \mathcal{S} : P_M(x) \left(\sum_{i=1}^n x_i^2 \right)^r \text{ has nonnegative coefficients} \right\}.$$

de Klerk and Pasechnik showed that $\mathcal{N} = \mathcal{C}^0 \subset \mathcal{C}^1 \subset \dots$, and $\operatorname{int}(\mathcal{C}) \subseteq \bigcup_{r \in \mathbb{N}} \mathcal{C}^r$. Each of these cones is polyhedral, so optimizing over one of them is solving an LP.

Refining these approaches, Peña et al. [18] derive yet another hierarchy of cones approximating $\mathcal{C}$. Adopting standard multiindex notation, where for a given multiindex $\beta \in \mathbb{N}^n$ we have $|\beta| := \beta_1 + \dots + \beta_n$ and $x^\beta := x_1^{\beta_1} \cdots x_n^{\beta_n}$, they define the following set of polynomials

$$\mathcal{E}^r := \left\{ \sum_{\beta \in \mathbb{N}^n, |\beta| = r} x^\beta x^T (P_\beta + N_\beta) x : P_\beta \in \mathcal{S}^+, N_\beta \in \mathcal{N} \right\}.$$

With this, they define the cones

$$\mathcal{Q}^r := \left\{ M \in \mathcal{S} : x^T M x \left(\sum_{i=1}^n x_i^2 \right)^r \in \mathcal{E}^r \right\}.$$

They show that $\mathcal{C}^r \subseteq \mathcal{Q}^r \subseteq \mathcal{K}^r$ for all $r \in \mathbb{N}$, with $\mathcal{Q}^r = \mathcal{K}^r$ for $r = 0, 1$. Similar to $\mathcal{K}^r$, the condition $M \in \mathcal{Q}^r$ can be rewritten as a system of LMIs. Optimizing over $\mathcal{Q}^r$ is therefore again an SDP.

It is a common feature of all these approximation hierarchies that they approximate $\mathcal{C}$ uniformly and do not take into account any information provided by the objective function of the optimization problem. Moreover, in all these approaches the system of LMIs (resp. linear inequalities) gets large quickly as r increases, meaning that the dimension of the SDPs increases so quickly that current SDP-solvers can only solve problems over those cones for small values of r, i.e., $r \leq 3$ at most.

In contrast to this, in our approach the approximation of $\mathcal{C}$ can be guided through the objective function in such a way that a fine approximation is reached in those regions of $\mathcal{C}$ which are relevant for the optimization, and little computational effort goes to approximating those regions of $\mathcal{C}$ which are not. The dimension (i.e., the number of variables) of the linear subproblems in our algorithm is constant, though the number of constraints grows. Moreover, solving a relaxation of a copositive program over one of the cones introduced above provides in general just a relaxation and no information on the quality of the corresponding bound (an exception is [4]). Our approach works not only with inner approximations of $\mathcal{C}$, but simultaneously with outer approximations. Therefore, it provides exact information on the approximation error and the accuracy of the solution.

We are not aware of comparable approximation schemes for the (dual) cone $\mathcal{C}^*$. A recent attempt to solve optimization problems over $\mathcal{C}^*$ is a descent algorithm by Jarre et al. [13]. We remark that another recent contribution to the field of copositive programming is a unified theory of KKT type optimality conditions and duality by Eichfelder and Jahn [11].

1.3. Outline of the paper. We start in section 2 by reviewing criteria for copositivity of a matrix. Based on these criteria, we develop inner and outer polyhedral approximations of $\mathcal{C}$ in section 3. With these cones, we state our algorithm for copositive programs and prove convergence (section 4). In section 5 we discuss how the algorithm can be fine-tuned and which details make an implementation efficient. Finally, we present numerical results in section 6.

2. Criteria for copositivity. In this section, we review some conditions for copositivity that we developed in [5]. These conditions will be the basis for approximations of the copositive cone $\mathcal{C}$ which we introduce in the next section. We start with the following:

Observation. *Let $\|\cdot\|$ denote any norm on $\mathbb{R}^n$. We have*

(a) *A is copositive $\Leftrightarrow$ $x^TAx \geq 0$ for all $x \in \mathbb{R}^n_+$ with $\|x\| = 1$,*

(b) *A is strictly copositive $\Leftrightarrow$ $x^TAx > 0$ for all $x \in \mathbb{R}^n_+$ with $\|x\| = 1$.*

If we choose the 1-norm $\|\cdot\|_1$, then the set $\Delta^S := \{x \in \mathbb{R}^n_+ : \|x\|_1 = 1\}$ is the so-called *standard simplex*. The copositivity property then translates to

$$x^TAx \geq 0 \quad \text{for all} \quad x \in \Delta^S,$$

i.e., we search for conditions which ensure that the quadratic polynomial x^TAx is nonnegative over a simplex. A convenient way to describe polynomials with respect to a simplex is to use barycentric coordinates: Let $\Delta = \operatorname{conv}\{v_1, \dots, v_n\}$ be a simplex and

$$x = \sum_{i=1}^{n} \lambda_i v_i \quad \text{with} \quad 1 = \sum_{i=1}^{n} \lambda_i.$$

Then $\lambda_1, \dots, \lambda_n \in \mathbb{R}$ are called the *barycentric coordinates* of x with respect to Δ. The representation of the quadratic form in these coordinates reads

$$x^TAx = \left(\sum_{i=1}^{n} \lambda_i v_i\right)^T A \left(\sum_{j=1}^{n} \lambda_j v_j\right) = \sum_{i,j=1}^{n} v_i^T A v_j \lambda_i \lambda_j.$$

The polynomials $\lambda_1^2, \dots, \lambda_n^2$ and $2\lambda_i\lambda_j$ $(i \neq j)$ appearing in this representation are called *Bézier–Bernstein polynomials*, and the coefficients $v_i^TAv_j$ are the corresponding *Bézier–Bernstein coefficients*. Since all λ_i are nonnegative on Δ, the next lemma is immediate:

Lemma 2.1. *Let $\Delta = \operatorname{conv}\{v_1, \dots, v_n\}$ be a simplex. If $v_i^TAv_j \geq 0$ for all $i, j \in \{1, \dots, n\}$, then $x^TAx \geq 0$ for all $x \in \Delta$.*

If Δ is the standard simplex $\Delta^S = \operatorname{conv}\{e_1, \dots, e_n\}$, then this lemma shows that A is copositive if $0 \leq e_i^TAe_j = a_{ij}$ for all i, j. This is the well-known property that any (entrywise) nonnegative matrix is copositive. This condition can be refined by looking at so-called simplicial partitions of Δ^S:

Definition 2.2. *Let Δ be a simplex in $\mathbb{R}^n$. A family $\mathcal{P} = \{\Delta^1, \dots, \Delta^m\}$ of simplices satisfying*

$$\Delta = \bigcup_{i=1}^{m} \Delta^i \qquad \textit{and} \qquad \operatorname{int} \Delta^i \cap \operatorname{int} \Delta^j = \emptyset \;\; \textit{for} \;\; i \neq j$$

is called a simplicial partition *of Δ. For convenience, we denote by $V_{\mathcal{P}}$ the set of all vertices of simplices in $\mathcal{P}$, and by $E_{\mathcal{P}}$ the set of all edges of simplices in $\mathcal{P}$.*

Simplicial partitions are a useful tool in many branches of applied mathematics. A good survey on this topic including convergence results is [12].

It is easy to see that a simplicial partition can be generated through the following "radial" subdivision of $\Delta = \operatorname{conv}\{v_1, \dots, v_n\}$: let $w \in \Delta \setminus \{v_1, \dots, v_n\}$, which is uniquely represented by

$$w = \sum_{i=1}^{n} \lambda_i v_i, \quad \text{with} \quad \lambda_i \geq 0,\ \sum_{i=1}^{n} \lambda_i = 1.$$

For each index i with $\lambda_i > 0$, form the simplex Δ^i obtained from Δ by replacing the vertex v_i by w, i.e., $\Delta^i = \operatorname{conv}\{v_1, \dots, v_{i-1}, w, v_{i+1}, \dots, v_n\}$. The collection of all those Δ^i is a simplicial partition of Δ. If w is a point on one of the longest edges of Δ, the above procedure is called *bisection* of the simplex along the longest edge. Generating a nested sequence of subsimplices of Δ through midpoint bisection along the longest edge has the nice property that this sequence converges to a singleton. This property is sometimes referred to as "exhaustiveness". It can be generalized from midpoint bisection to settings where the bisection point is an almost arbitrary point on one of the longest edges; see [12] for a detailed discussion.

Using this concept, the following theorem gives sufficient conditions for copositivity which generalize the aforementioned relation that A is copositive if all $a_{ij} \geq 0$:

THEOREM 2.3. *Let $A \in \mathcal{S}$, let $\mathcal{P}$ be a simplicial partition of Δ^S.*

(a) *If $u^T Av \geq 0$ for all $\{u, v\} \in E_{\mathcal{P}}$ and $v^T Av \geq 0$ for all $v \in V_{\mathcal{P}}$, then A is copositive.*

(b) *If $u^T Av > 0$ for all $\{u, v\} \in E_{\mathcal{P}}$ and $v^T Av > 0$ for all $v \in V_{\mathcal{P}}$, then A is strictly copositive.*

Proof. To show (a), it is sufficient to prove nonnegativity of $x^T Ax$ for $x \in \Delta^S$. So choose an arbitrary $x \in \Delta^S$. Then $x \in \Delta$ for some $\Delta \in \mathcal{P}$. By assumption, $u^T Av \geq 0$ for all combinations of vertices of this simplex Δ which, by Lemma 2.1, implies $x^T Ax \geq 0$. Part (b) is shown analogously. □

We define the diameter $\delta(\mathcal{P})$ of a partition $\mathcal{P}$ to be

$$\delta(\mathcal{P}) := \max_{\{u,v\} \in E_{\mathcal{P}}} \|u - v\|.$$

Once a partition gets finer and finer, one will eventually capture more and more strictly copositive matrices. In the limit we get a necessary condition for strict copositivity:

THEOREM 2.4. *Let $A \in \mathcal{S}$ be strictly copositive. Then there exists $\varepsilon = \varepsilon(A) > 0$ such that for all finite simplicial partitions $\mathcal{P}$ of Δ^S with $\delta(\mathcal{P}) \leq \varepsilon$ we have*

$$u^T Av > 0 \text{ for all } \{u, v\} \in E_{\mathcal{P}} \quad \text{and} \quad v^T Av > 0 \text{ for all } v \in V_{\mathcal{P}}.$$

Proof. The detailed proof can be found in [5]. It relies on strict positivity of the bilinear form $x^T Ay$ on the diagonal of the compact set $\Delta^S \times \Delta^S$, followed by a continuity argument. □

Observe that the ε in Theorem 2.4 certainly depends on the matrix A, i.e., there is not a single ε that works uniformly for all strictly copositive A. Indeed, the ε relates to how "ill-conditioned" A is.

3. Polyhedral approximations. In this section, we present polyhedral inner and outer approximations of the cone $\mathcal{C}$.

3.1. Inner approximation of $\mathcal{C}$. We use the sufficient condition of Theorem 2.3 to define inner approximations of $\mathcal{C}$. As before, consider a simplicial partition $\mathcal{P} = \{\Delta^1, \ldots, \Delta^m\}$ of Δ^S, and let $V_\mathcal{P}$ denote the set of all vertices of simplices in $\mathcal{P}$, and $E_\mathcal{P}$ the set of all edges of simplices in $\mathcal{P}$. For a given partition $\mathcal{P}$, define

$$\begin{aligned}\mathcal{I}_\mathcal{P} := \{A \in \mathcal{S} : {}& v^T Av \geq 0 \text{ for all } v \in V_\mathcal{P},\\ & u^T Av \geq 0 \text{ for all } \{u, v\} \in E_\mathcal{P}\}.\end{aligned}$$

Note that given the vertices u,v, an expression of the form $u^T Av \geq 0$ is a linear inequality for the entries of A. Therefore, $\mathcal{I}_\mathcal{P}$ is a polyhedral cone.

Obviously, $\mathcal{I}_\mathcal{P}$ depends on the partition $\mathcal{P}$. If $\mathcal{P}_1$ and $\mathcal{P}_2$ are two simplicial partitions of the same simplex, we call $\mathcal{P}_2$ a *refinement* of $\mathcal{P}_1$ if for all $\Delta \in \mathcal{P}_1$ there exists a subset $\mathcal{P}_\Delta \subseteq \mathcal{P}_2$ which is a simplicial partition of Δ.

We have the following properties:

LEMMA 3.1. *Let $\mathcal{P}, \mathcal{P}_1, \mathcal{P}_2$ denote simplicial partitions of Δ^S. Then*

(a) *$\mathcal{I}_\mathcal{P}$ is a closed convex polyhedral cone,*

(b) *$\mathcal{I}_\mathcal{P} \subseteq \mathcal{C}$, i.e., $\mathcal{I}_\mathcal{P}$ is an inner approximation of $\mathcal{C}$,*

(c) *if $\mathcal{P}_2$ is a refinement of $\mathcal{P}_1$, then $\mathcal{I}_{\mathcal{P}_1} \subseteq \mathcal{I}_{\mathcal{P}_2}$.*

Proof. (a) is obvious from the definition. (b) follows from Theorem 2.3. To prove (c), let $A \in \mathcal{I}_{\mathcal{P}_1}$, let $\Delta^2 \in \mathcal{P}_2$, and let u, v be two arbitrary vertices of Δ^2 (possibly equal). We have to show $u^T Av \geq 0$. Since $\mathcal{P}_2$ is a refinement of $\mathcal{P}_1$, there exists a simplex $\Delta^1 \in \mathcal{P}_1$ with $\Delta^2 \subseteq \Delta^1$. Therefore, u and v are convex combinations of the vertices $v_1, \ldots, v_n$ of Δ^1, i.e., $u = \sum_{i=1}^n \lambda_i v_i$ and $v = \sum_{i=1}^n \mu_i v_i$ with $\lambda_i, \mu_i \geq 0$ for all $i \in \{1, \ldots, n\}$ and $\sum_{i=1}^n \lambda_i = 1 = \sum_{i=1}^n \mu_i$. Since $v_i^T Av_j \geq 0$ for all i, j due to $A \in \mathcal{I}_{\mathcal{P}_1}$, we have

$$u^T Av = \sum_{i,j=1}^n \lambda_i \mu_j v_i^T Av_j \geq 0.$$

Therefore, $A \in \mathcal{I}_{\mathcal{P}_2}$. □

Example 3.2. If $\mathcal{P} = \{\Delta^S\}$, i.e., the partition consists only of the standard simplex, then

$$\mathcal{I}_\mathcal{P} = \{A \in \mathcal{S} : a_{ij} \geq 0 \text{ for all } i, j = 1, \ldots, n\} = \mathcal{N},$$

i.e., $\mathcal{I}_{\{\Delta^S\}}$ equals the cone $\mathcal{N}$ of nonnegative matrices.

Consider instead the partition $\mathcal{P}_2 = \{\Delta^1, \Delta^2\}$ which is derived from $\mathcal{P}$ by bisecting the edge $\{e_1, e_2\}$ at the midpoint $w := \frac{1}{2}(e_1 + e_2)$. We get $\Delta^1 = \operatorname{conv}\{w, e_2, \ldots, e_n\}$ and $\Delta^2 = \operatorname{conv}\{e_1, w, e_3, \ldots, e_n\}$. For the definition of $\mathcal{I}_{\mathcal{P}_2}$ this means that the inequality $e_1^T Ae_2 \geq 0$ (i.e., $a_{12} \geq 0$) corresponding to the bisected edge is removed and replaced by a number of new inequalities. More precisely,

$$\begin{aligned}\mathcal{I}_{\mathcal{P}_2} = \{A \in \mathcal{S} : {}& a_{ij} \geq 0 \text{ for all } \{i, j\} \neq \{1, 2\},\\ & a_{i1} + a_{i2} \geq 0 \text{ for all } i = 1, \ldots, n,\\ & a_{11} + 2a_{12} + a_{22} \geq 0\}.\end{aligned}$$

This defines a larger cone, i.e., a better approximation to $\mathcal{C}$. Observe that the system defining $\mathcal{I}_{\mathcal{P}_2}$ is redundant. This property will cause some difficulty later in the paper, cf. section 5.2. A reduced representation is

$$\begin{aligned}\mathcal{I}_{\mathcal{P}_2} = \{A \in \mathcal{S} : {}& a_{ij} \geq 0 \text{ for all } \{i, j\} \neq \{1, 2\},\\ & a_{11} + a_{12} \geq 0,\\ & a_{22} + a_{12} \geq 0\}.\end{aligned}$$

This is a reduction from n^2+n-1 to n^2 inequalities; i.e., already $O(n)$ inequalities are redundant after a single bisection step.

The next theorem shows that a sequence of simplicial partitions $\{\mathcal{P}_\ell\}$ yields a sequence of polyhedral inner approximations $\{\mathcal{I}_{\mathcal{P}_\ell}\}$ that will eventually approximate $\mathcal{C}$ with arbitrary precision, provided that the diameter $\delta(\mathcal{P})$ of the simplicial partition goes to zero.

THEOREM 3.3. *Let $\{\mathcal{P}_\ell\}$ be a sequence of simplicial partitions of Δ^S with $\delta(\mathcal{P}_\ell) \to 0$. Then we have*

$$\operatorname{int}\mathcal{C} \subseteq \bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell} \subseteq \mathcal{C}, \quad \textit{and consequently} \quad \mathcal{C} = \overline{\bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell}}.$$

Proof. Take $A \in \operatorname{int}\mathcal{C}$, i.e., A strictly copositive. Then Theorem 2.4 implies that there exists $\ell_0 \in \mathbb{N}$, such that $A \in \mathcal{I}_{\mathcal{P}_{\ell_0}}$. Therefore $A \in \bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell}$, and hence $\operatorname{int}\mathcal{C} \subseteq \bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell}$. From Lemma 3.1, we have $\mathcal{I}_{\mathcal{P}_\ell} \subseteq \mathcal{C}$ for all $\ell \in \mathbb{N}$, so $\bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell} \subseteq \mathcal{C}$. Finally, $\mathcal{C} = \overline{\bigcup_{\ell\in\mathbb{N}} \mathcal{I}_{\mathcal{P}_\ell}}$ since $\mathcal{C} = \overline{\operatorname{int}\mathcal{C}}$. ◻

This shows that our approach generates a sequence of approximating polyhedral cones $\mathcal{N} = \mathcal{I}_{\mathcal{P}_0} \subset \mathcal{I}_{\mathcal{P}_1} \subset \ldots \subset \mathcal{C}$ in a similar way as the approaches described in section 1.2. To compare our approximations to the hierarchy of polyhedral cones $\mathcal{C}^r$ by Bomze and de Klerk [4], observe that $\mathcal{C}^0 = \mathcal{N} = \mathcal{I}_{\{\Delta^S\}}$. For $\mathcal{C}^1$, it is shown in [4] that $A \in \mathcal{C}^1$ if and only if

$$\begin{aligned} a_{ii} &\geq 0, &\quad& i \in \{1,\ldots,n\},\\ a_{ii} + 2a_{ij} &\geq 0, && i \neq j,\\ a_{ij} + a_{jk} + a_{ki} &\geq 0, && i < j < k. \end{aligned}$$

To see the difference between the approaches, consider dimension $n=2$ for simplicity, in which case the above system describing $\mathcal{C}^1$ reduces to

$$a_{11} \geq 0, \quad a_{22} \geq 0, \quad a_{11} + 2a_{12} \geq 0, \quad a_{22} + 2a_{12} \geq 0. \tag{3.1}$$

Consider the partition $\mathcal{P}_1 = \{\operatorname{conv}\{e_1, v\}, \operatorname{conv}\{v, e_2\}\}$ with $v = \frac{1}{2}(e_1+e_2)$. The corresponding system of inequalities for $\mathcal{I}_{\mathcal{P}_1}$ is then

$$a_{ii} \geq 0, \qquad i \in \{1,2\}, \tag{3.2}$$

$$a_{11} + a_{12} \geq 0, \tag{3.3}$$

$$a_{22} + a_{12} \geq 0, \tag{3.4}$$

plus the redundant inequality $v^T A v \geq 0$. Obviously, system (3.2)–(3.4) is implied by (3.1), and therefore $\mathcal{I}_{\mathcal{P}_1} \supseteq \mathcal{C}^1$. As the matrix $A = \left(\begin{smallmatrix} 1 & -1 \\ -1 & 1 \end{smallmatrix}\right)$ fullfills (3.2)–(3.4) but not (3.1), we have $\mathcal{I}_{\mathcal{P}_1} \neq \mathcal{C}^1$. It is easy to see that for $v = \lambda e_1 + (1-\lambda)e_2$ with $\lambda \in [\frac{1}{3}, \frac{2}{3}]$ we have $\mathcal{I}_{\mathcal{P}_1} \supsetneq \mathcal{C}^1$, whereas for the other values of λ we get $\mathcal{I}_{\mathcal{P}_1} \not\supset \mathcal{C}^1$. These arguments extend to higher dimensions, but get much more technical there.

Comparing our approximation with $\mathcal{S}^+ + \mathcal{N}$, it is clear that there is no partition $\mathcal{P}$ such that $\mathcal{I}_\mathcal{P} \supset \mathcal{S}^+ + \mathcal{N}$ because in dimension $n=2$ we have $\mathcal{S}^+ + \mathcal{N} = \mathcal{C}$, while $\mathcal{I}_\mathcal{P}$ is a polyhedral subset of $\mathcal{C}$. However, depending on the subdivision strategy it is possible to construct partitions with $\mathcal{I}_\mathcal{P} \not\subset \mathcal{S}^+ + \mathcal{N}$.

3.2. Outer approximation of $\mathcal{C}$.

As before, consider a simplicial partition $\mathcal{P}$ of Δ^S, let $V_\mathcal{P}$ denote the set of all vertices in $\mathcal{P}$, and define

$$\mathcal{O}_\mathcal{P} := \{A \in \mathcal{S} : v^T A v \geq 0 \text{ for all } v \in V_\mathcal{P}\}.$$

It is easy to see that, similar to $\mathcal{I}_{\mathcal{P}}$, the set $\mathcal{O}_{\mathcal{P}}$ is polyhedral, as well. In analogy to Lemma 3.1, we have the following properties:

LEMMA 3.4. *Let $\mathcal{P}, \mathcal{P}_1, \mathcal{P}_2$ denote simplicial partitions of Δ^S. Then*

(a) *$\mathcal{O}_{\mathcal{P}}$ is a closed convex polyhedral cone,*

(b) *$\mathcal{O}_{\mathcal{P}} \supseteq \mathcal{C}$, i.e., $\mathcal{O}_{\mathcal{P}}$ is an outer approximation of $\mathcal{C}$,*

(c) *if $\mathcal{P}_2$ is a refinement of $\mathcal{P}_1$, then $\mathcal{O}_{\mathcal{P}_2} \subseteq \mathcal{O}_{\mathcal{P}_1}$.*

Proof. (a) is obvious from the definition. (b) If $A \in \mathcal{C}$, then $x^T A x \geq 0$ for all $x \in \Delta^S$. Since $V_{\mathcal{P}} \subset \Delta^S$, the statement follows. (c) We have $V_{\mathcal{P}_1} \subseteq V_{\mathcal{P}_2}$. Therefore, the set of inequalities describing $\mathcal{O}_{\mathcal{P}_1}$ is a subset of the set of inequalities describing $\mathcal{O}_{\mathcal{P}_2}$, and hence $\mathcal{O}_{\mathcal{P}_2} \subseteq \mathcal{O}_{\mathcal{P}_1}$. □

Example 3.5. If $\mathcal{P}$ consists only of the standard simplex, i.e., $\mathcal{P} = \{\Delta^S\}$, then

$$\mathcal{O}_{\mathcal{P}} = \{A \in \mathcal{S} : a_{ii} \geq 0 \text{ for all } i\}.$$

This corresponds to the well-known fact that a copositive matrix necessarily has nonnegative entries on the diagonal. Observe that $\mathcal{O}_{\{\Delta^S\}}$ is not pointed.

Performing a midpoint bisection of the edge $\{e_1, e_2\}$ gives the new vertex $w := \frac{1}{2}(e_1 + e_2)$ and the resulting partition $\mathcal{P}_2$ yields the set

$$\mathcal{O}_{\mathcal{P}_2} = \{A \in \mathcal{S} : a_{ii} \geq 0 \text{ for all } i, a_{11} + 2a_{21} + a_{22} \geq 0\},$$

a smaller set and better approximation to $\mathcal{C}$.

The sequence of outer approximations $\{\mathcal{O}_{\mathcal{P}_\ell}\}$ converges to the copositive cone as the partitions $\mathcal{P}_\ell$ get finer.

THEOREM 3.6. *Let $\{\mathcal{P}_\ell\}$ be a sequence of simplicial partitions of Δ^S with $\delta(\mathcal{P}_\ell) \to 0$. Then we have*

$$\mathcal{C} = \bigcap_{\ell \in \mathbb{N}} \mathcal{O}_{\mathcal{P}_\ell}.$$

Proof. Lemma 3.4(b) implies $\mathcal{C} \subseteq \bigcap_{\ell \in \mathbb{N}} \mathcal{O}_{\mathcal{P}_\ell}$. To see the reverse, take $A \notin \mathcal{C}$. Then $\bar{x}^T A \bar{x} < 0$ for some $\bar{x} \in \Delta^S$. From continuity it follows that there is an ε-neighborhood $N_\varepsilon(\bar{x})$ of $\bar{x}$ such that

$$x^T A x < 0 \quad \text{for all } x \in N_\varepsilon(\bar{x}). \tag{3.5}$$

Let $\mathcal{P} \in \{\mathcal{P}_\ell\}$ be some partition with $\delta(\mathcal{P}) < \varepsilon$. Then there is a simplex $\Delta \in \mathcal{P}$ with $\bar{x} \in \Delta$, and hence a vertex v of Δ with $\|\bar{x} - v\| < \varepsilon$, so $v \in N_\varepsilon(\bar{x})$. From (3.5), we see that $v^T A v < 0$, whence $A \notin \mathcal{O}_{\mathcal{P}}$. Therefore, $A \notin \bigcap_{\ell \in \mathbb{N}} \mathcal{O}_{\mathcal{P}_\ell}$. □

3.3. Approximations of the dual cone $\mathcal{C}^*$. Recall that the dual cone of $\mathcal{C}$ is the cone $\mathcal{C}^*$ of completely positive matrices. By duality, the dual cone of an inner (resp. outer) approximation of $\mathcal{C}$ is an outer (resp. inner) approximation of $\mathcal{C}^*$. Indeed, it is not difficult to see that for any partition $\mathcal{P}$ of Δ^S

$$\mathcal{I}_{\mathcal{P}}^* = \left\{ \sum_{\{u,v\} \in E_{\mathcal{P}}} \lambda_{uv}(uv^T + vu^T) + \sum_{v \in V_{\mathcal{P}}} \lambda_v vv^T : \lambda_{uv}, \lambda_v \in \mathbb{R}_+ \right\} \supseteq \mathcal{C}^*$$

is an outer approximation of $\mathcal{C}^*$, and

$$\mathcal{O}_{\mathcal{P}}^* = \left\{ \sum_{v \in V_{\mathcal{P}}} \lambda_v vv^T : \lambda_v \in \mathbb{R}_+ \right\} \subseteq \mathcal{C}^*$$

is an inner approximation of $\mathcal{C}^*$. From Theorems 3.3 and 3.6 we immediately get that if $\{\mathcal{P}_\ell\}$ is a sequence of simplicial partitions of Δ^S with $\delta(\mathcal{P}_\ell) \to 0$, then the approximations converge, i.e.,

$$\mathcal{C}^* = \bigcap_{\ell \in \mathbb{N}} \mathcal{I}^*_{\mathcal{P}_\ell} \qquad \text{and} \qquad \mathcal{C}^* = \overline{\bigcup_{\ell \in \mathbb{N}} \mathcal{O}^*_{\mathcal{P}_\ell}}.$$

4. An adaptive approximation algorithm for copositive programs. We now turn to the problem of solving an optimization problem over the copositive cone. The difficulty of such a problem lies in the cone condition. If the copositive cone is replaced by a linear inner or outer approximation, we get a linear program whose optimal value is a lower, respectively upper, bound of the optimal value of the original problem. We first state our algorithm and illustrate its behavior with a small example. After that, we study convergence of the algorithm.

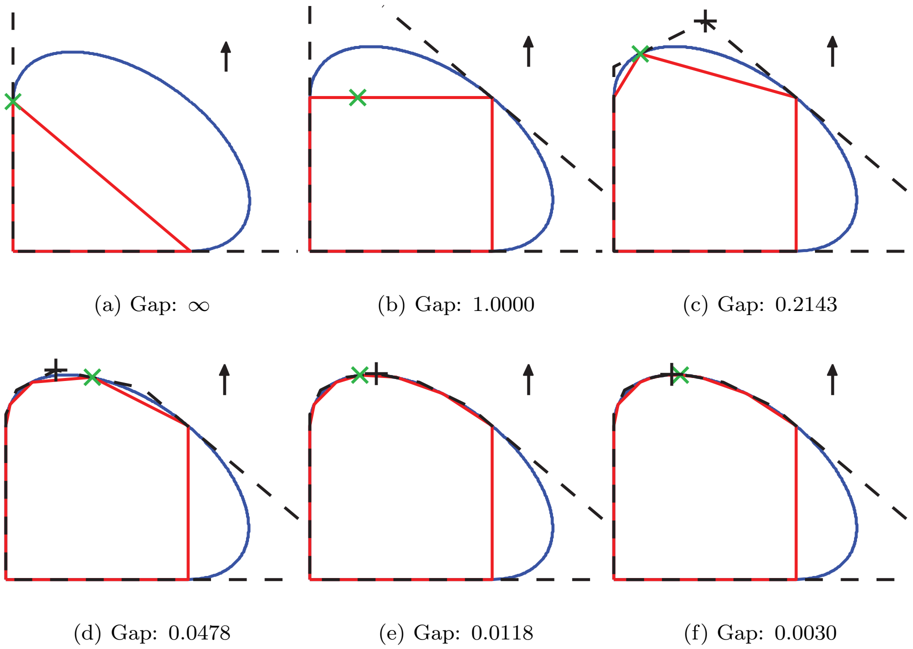
4.1. Algorithm framework. We state the algorithm for copositive programs of the form

$$\text{(CP)} \qquad \begin{array}{ll} \max & \langle C, X\rangle \\ \text{s.t.} & \langle A_i, X\rangle = b_i, \quad i = 1, \dots, m \\ & X \in \mathcal{C}. \end{array}$$

Given a solution accuracy $\varepsilon > 0$, Algorithm 1 computes an ε-optimal solution of (CP), i.e., a feasible solution X with $\frac{\langle C, X^{\text{opt}}\rangle - \langle C, X\rangle}{1 + |\langle C, X^{\text{opt}}\rangle| + |\langle C, X\rangle|} < \varepsilon$. Note that the algorithm also provides the valid lower (resp. upper) bounds $\langle C, X^{\mathcal{I}}\rangle$ (resp. $\langle C, X^{\mathcal{O}}\rangle$).

Algorithm 1 ε-approximation algorithm for (CP).

```
 1: set $\mathcal{P} = \{\Delta^S\}$
 2: solve the inner LP
        (ILP)   $\max\ \langle C, X\rangle$
                $\text{s.t.}\ \langle A_i, X\rangle = b_i, \quad i = 1, \dots, m$
                $X \in \mathcal{I}_{\mathcal{P}}$
    let $X^{\mathcal{I}}$ denote the solution of this problem
 3: solve the outer LP
        (OLP)   $\max\ \langle C, X\rangle$
                $\text{s.t.}\ \langle A_i, X\rangle = b_i, \quad i = 1, \dots, m$
                $X \in \mathcal{O}_{\mathcal{P}}$
    let $X^{\mathcal{O}}$ denote the solution of this problem
 4: if $\frac{\langle C, X^{\mathcal{O}}\rangle - \langle C, X^{\mathcal{I}}\rangle}{1 + |\langle C, X^{\mathcal{O}}\rangle| + |\langle C, X^{\mathcal{I}}\rangle|} < \varepsilon$, then
 5:     STOP: $X^{\mathcal{I}}$ is an $\varepsilon$-optimal solution of (CP)
 6: end if
 7: choose $\Delta \in \mathcal{P}$
 8: bisect $\Delta = \Delta^1 \cup \Delta^2$
 9: set $\mathcal{P} \leftarrow \mathcal{P} \setminus \{\Delta\} \cup \{\Delta^1, \Delta^2\}$
10: go to 2.
```



(a) Gap: ∞

(b) Gap: 1.0000

(c) Gap: 0.2143

(d) Gap: 0.0478

(e) Gap: 0.0118

(f) Gap: 0.0030

Fig. 4.1. *Iterations for Example* 4.1.

In this prototype algorithm it is not specified how a simplex is selected in Step 7 or how the bisection is performed in Step 8. Here lies some freedom which allows us to guide the partitioning procedure adaptively in a way that is advantageous for the optimization. The choice of the partitions also influences the convergence behavior and finiteness of the algorithm.

We will discuss these points later in more detail in section 5. First, we illustrate the behavior of this algorithm with a small example:

Example 4.1. Consider the problem

$$\begin{aligned} \max \quad & \left\langle \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}, X \right\rangle \\ \text{s. t.} \quad & \left\langle \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}, X \right\rangle = 2 \\ & X \in \mathcal{C}. \end{aligned}$$

The sequence of iterations is displayed in Figure 4.1. In this simple example, the cone $\mathcal{C}$ of symmetric copositive matrices is a cone in $\mathbb{R}^3$. The feasible set is therefore a two-dimensional set which is displayed with the curved line in the figure. The upward arrow indicates the direction of the objective function. The solid line represents the inner approximating cones $\mathcal{I}_{\mathcal{P}}$, whereas the dashed lines represent the outer approximating cones $\mathcal{O}_{\mathcal{P}}$. The symbols × and + indicate the subproblem optimal solutions (computed by an interior point solver in this example). For the starting partition, the outer approximation is unbounded, a consequence of the fact that $\mathcal{O}_{\{\Delta^S\}}$ is not pointed. “Gap” denotes the difference $\langle C, X^{\mathcal{O}} \rangle - \langle C, X^{\mathcal{I}} \rangle$.

Observe that the feasible set is approximated with high accuracy in those parts which are important for the optimization, whereas the irrelevant parts are not refined.

4.2. Convergence. We proceed to investigate convergence of Algorithm 1. Convergence of this algorithm relies on convergence of the approximating cones $\mathcal{I}_{\mathcal{P}_\ell}$ and $\mathcal{O}_{\mathcal{P}_\ell}$ as described in section 3. Therefore, we need the assumption that $\delta(\mathcal{P}_\ell) \to 0$ as $\ell \to \infty$ for the partitions generated in the algorithm. Note that by construction of Algorithm 1 we have the monotonicity $\mathcal{I}_{\mathcal{P}_\ell} \subset \mathcal{I}_{\mathcal{P}_{\ell+1}}$ and $\mathcal{O}_{\mathcal{P}_{\ell+1}} \subset \mathcal{O}_{\mathcal{P}_\ell}$.

Further, observe that feasibility of (CP) implies feasibility of (OLP), but not necessarily feasibility of (ILP). Therefore, we need assumptions which imply that (ILP) will eventually become feasible in the course of the iterations. This is done by assuming that there exists a strictly feasible point by which we mean a solution $\widehat{X}$ of the linear system $\langle A_i, \widehat{X}\rangle = b_i$ for all $i = 1, \ldots, m$ with $\widehat{X} \in \operatorname{int}(\mathcal{C})$.

Moreover, the feasible set of the outer approximation problem (OLP) may be unbounded even if the feasible set of (CP) is compact; cf. Example 4.1. The next theorem shows, however, that in this case the feasible set of the outer approximation eventually becomes bounded as the algorithm progresses.

THEOREM 4.2. *Assume the feasible set of* (CP) *is bounded and contains a strictly feasible point. Assume further that in every iteration of Algorithm* 1 *the selection of* Δ *and the bisection into* $\Delta = \Delta^1 \cup \Delta^2$ *is performed in such a way that the generated sequence* $\{\mathcal{P}_\ell\}$ *of partitions fulfills* $\delta(\mathcal{P}_\ell) \to 0$ *as* $\ell \to \infty$. *Let* ($\mathrm{ILP}_{\mathcal{P}_\ell}$) *(resp.* ($\mathrm{OLP}_{\mathcal{P}_\ell}$)*) denote the inner (resp. outer) approximation LPs corresponding to partition* $\{\mathcal{P}_\ell\}$ *in Steps* 2 *and* 3 *of Algorithm* 1, *and let* $X^{\mathcal{I}_\ell}$ *(resp.* $X^{\mathcal{O}_\ell}$*) denote the optimal solutions of* ($\mathrm{ILP}_{\mathcal{P}_\ell}$) *(resp.* ($\mathrm{OLP}_{\mathcal{P}_\ell}$)*). Then*

(a) *there exists* $\ell_0 \in \mathbb{N}$ *such that the feasible set of* ($\mathrm{ILP}_{\mathcal{P}_\ell}$) *is nonempty and bounded for any* $\ell \geq \ell_0$*; the corresponding optimal solution* $X^{\mathcal{I}_\ell}$ *is then feasible for* (CP)*;*

(b) *there exists* $\ell_1 \in \mathbb{N}$ *such that the feasible set of* ($\mathrm{OLP}_{\mathcal{P}_\ell}$) *is nonempty and bounded for any* $\ell \geq \ell_1$*;*

(c) *both sequences* $\{X^{\mathcal{I}_\ell}\}$ *and* $\{X^{\mathcal{O}_\ell}\}$ *have accumulation points, and any accumulation point of either sequence is optimal for* (CP)*.*

Proof. Let X^* denote an optimal solution of (CP), and let $\widehat{X}$ be a strictly feasible solution of (CP). Let $\mathcal{A} := \{X \in \mathcal{S} : \langle A_i, X\rangle = b_i \text{ for } i = 1, \ldots, m\}$ denote the subspace of points satisfying the linear constraints. We use the notation $\max(P)$ to denote the optimal value of a maximization problem (P).

(a) Since $\mathcal{I}_{\mathcal{P}_\ell} \subseteq \mathcal{C}$ for any $\ell \in \mathbb{N}$, the feasible sets of ($\mathrm{ILP}_{\mathcal{P}_\ell}$) are all bounded. As $\widehat{X}$ is a strictly copositive matrix, it follows from Theorem 3.3 that there exists $\ell_0 \in \mathbb{N}$ such that $\widehat{X} \in \mathcal{I}_{\mathcal{P}_{\ell_0}}$. Since also $\widehat{X} \in \mathcal{A}$, the feasible set $\mathcal{A} \cap \mathcal{I}_{\mathcal{P}_{\ell_0}}$ of ($\mathrm{ILP}_{\mathcal{P}_{\ell_0}}$) is nonempty, and so are the feasible sets of ($\mathrm{ILP}_{\mathcal{P}_\ell}$) for all $\ell \geq \ell_0$. Therefore, any such ($\mathrm{ILP}_{\mathcal{P}_\ell}$) has an optimal solution $X^{\mathcal{I}_\ell}$ which, by $\mathcal{I}_{\mathcal{P}_\ell} \subseteq \mathcal{C}$, is feasible for (CP).

(b) Since (CP) is feasible, the feasible set $\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$ of any ($\mathrm{OLP}_{\mathcal{P}_\ell}$) is nonempty, as well. To show boundedness, assume by contradiction that $\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$ is unbounded for all $\ell \in \mathbb{N}$. Take an arbitrary $X \in \mathcal{A}\cap\mathcal{C}$. Then by polyhedrality of $\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$, the set

$$\mathcal{D}_\ell := \{D \in \mathcal{S} : \|D\| = 1, X + \alpha D \in \mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell} \text{ for all } \alpha \geq 0\}$$

is nonempty for any ℓ. The monotonicity $\mathcal{O}_{\mathcal{P}_\ell} \supseteq \mathcal{O}_{\mathcal{P}_{\ell+1}}$ implies $\mathcal{D}_\ell \supseteq \mathcal{D}_{\ell+1}$ for all ℓ. Moreover, closedness of $\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$ implies closedness of $\mathcal{D}_\ell$, whence all $\mathcal{D}_\ell$ are compact. Using a theorem of Cantor, we infer that the intersection of all $\mathcal{D}_\ell$ is nonempty, i.e., there exists $\widehat{D} \in \bigcap_{\ell\in\mathbb{N}} \mathcal{D}_\ell$. But then $\{X + \alpha\widehat{D} :$

$\alpha \geq 0\} \subset \mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$ for all ℓ, and therefore

$$\{X + \alpha\widehat{D} : \alpha \geq 0\} \subset \bigcap_{\ell \in \mathbb{N}} (\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}) = \mathcal{A} \cap \bigcap_{\ell \in \mathbb{N}} \mathcal{O}_{\mathcal{P}_\ell} = \mathcal{A} \cap \mathcal{C}.$$

This contradicts the assumption that $\mathcal{A} \cap \mathcal{C}$ is bounded, and consequently there exists $\ell_1 \in \mathbb{N}$ such that $\mathcal{A} \cap \mathcal{O}_{\mathcal{P}_\ell}$ is bounded for all $\ell \geq \ell_1$.

(c) We first show the statement for the sequence $\{X^{\mathcal{I}_\ell}\}$. An accumulation point exists because the sequence $\{X^{\mathcal{I}_\ell}\}_{\ell \geq \ell_0}$ is contained in the compact feasible set of (CP). Let X^a denote an accumulation point of $\{X^{\mathcal{I}_\ell}\}$. From the inner approximation property, we have $\langle C, X^{\mathcal{I}_\ell} \rangle \leq \max(\mathrm{CP})$ for all $\ell \geq \ell_0$, so the same must hold for the accumulation point, i.e.,

$$\langle C, X^a \rangle \leq \max(\mathrm{CP}). \tag{4.1}$$

To see the converse, consider points $Z_\lambda := \lambda X^* + (1 - \lambda)\widehat{X}$ for $\lambda \in (0, 1)$. By construction, Z_λ is strictly feasible for (CP), i.e., strictly copositive. By Theorem 3.3, for each such λ there exists $\ell_\lambda \in \mathbb{N}$ such that $Z_\lambda \in \mathcal{I}_{\mathcal{P}_\ell}$ for all $\ell \geq \ell_\lambda$. Therefore,

$$\langle C, X^a \rangle = \sup_{\ell \in \mathbb{N}} \max(\mathrm{ILP}_{\mathcal{P}_\ell}) \geq \lim_{\lambda \nearrow 1} \langle C, Z_\lambda \rangle = \langle C, X^* \rangle = \max(\mathrm{CP}).$$

Combined with (4.1), this proves that X^a is optimal for (CP).

Next, we show the statement for the sequence $\{X^{\mathcal{O}_\ell}\}$. An accumulation point exists because the sequence $\{X^{\mathcal{O}_\ell}\}_{\ell \geq \ell_1}$ is contained in the compact feasible set of $(\mathrm{OLP}_{\mathcal{P}_{\ell_1}})$. Let X^A denote an accumulation point of $\{X^{\mathcal{O}_\ell}\}$. From the outer approximation property we have $\langle C, X^{\mathcal{O}_\ell} \rangle \geq \max(\mathrm{CP})$ for all ℓ, so the same must hold for the accumulation point, i.e.,

$$\langle C, X^A \rangle \geq \max(\mathrm{CP}).$$

The reverse inequality follows from $X^A \in \bigcap_{\ell \in \mathbb{N}} \mathcal{O}_{\mathcal{P}_\ell} = \mathcal{C}$, which shows that X^A is an optimal solution of (CP). □

If the assumptions of Theorem 4.2 are not fulfilled, the situation gets more involved:

If the feasible set of (CP) is empty because $\mathcal{A} = \emptyset$, then (OLP) is infeasible in the very first iteration. If the feasible set of (CP) is empty because $\mathcal{A}$ does not intersect $\mathcal{C}$, then obviously all inner approximations $(\mathrm{ILP}_{\mathcal{P}_\ell})$ are infeasible, as well, but unfortunately infeasibility of (ILP) is no certificate of infeasibility of (CP). Detection of infeasibility of (CP) is only possible if (OLP) is infeasible. We observed that in numerical examples, infeasibility of (CP) was detected through infeasibility of an outer approximation $(\mathrm{OLP}_{\mathcal{P}_\ell})$ in the course of the iterations. In exceptional cases, however, this may fail: If the set $\mathcal{A}$ is parallel to an face of $\mathcal{C}$ induced by the hyperplane $\mathcal{H} := \{X \in \mathcal{C} : v^T X v = 0\}$, then the outer approximations remain feasible unless the partitioning process eventually generates v as a vertex in $V_{\mathcal{P}}$ by pure chance.

If (CP) is feasible but has no strictly feasible point, i.e., the feasible set is contained in the boundary of $\mathcal{C}$, then clearly all outer approximations are feasible, but the inner approximations are most likely all infeasible, unless the inner approximation happens to touch the boundary of $\mathcal{C}$ in the right portion.

If (CP) is unbounded, then in most practical cases the inner approximation will also be unbounded in some finite iteration. In any case, we have the following:

Theorem 4.3. *Assume that* (CP) *has a strictly feasible solution. If* (CP) *is unbounded, then*

$$\lim_{\ell\to\infty} \max(\mathrm{ILP}_{\mathcal{P}_\ell}) = \infty.$$

Proof. If (CP) is unbounded, then there exists a sequence $\{\tilde{X}_n\}$ of feasible solutions such that $\lim_{n\to\infty}\langle C, \tilde{X}_n\rangle = \infty$. Let $\widehat{X}$ be a strictly feasible solution of (CP). Then $X_n := \frac{1}{2}\widehat{X} + \frac{1}{2}\tilde{X}_n$ is strictly feasible for all $n \in \mathbb{N}$, and

$$\lim_{n\to\infty} \langle C, X_n\rangle = \infty.$$

By Theorem 2.4, for each $n \in \mathbb{N}$ there exists an index ℓ_n such that $X_n \in \mathcal{I}_{\mathcal{P}_{\ell_n}}$. Now the assertion follows. □

5. Fine-tuning the algorithm.

As mentioned, Algorithm 1 contains freedom in Steps 7 and 8 where the partitioning process of Δ^S is guided. In this section, we discuss how the partitioning is performed in each iteration. Moreover, we consider the problem of redundancies appearing in the subproblems, and we show how the starting partition can be tuned given a known heuristic solution.

5.1. Selecting and subdividing $\boldsymbol{\Delta}$.

Generating a sequence of partitions $\{\mathcal{P}_\ell\}$ of Δ^S with $\delta(\mathcal{P}_\ell) \to 0$ results in a sequence of cones $\{\mathcal{I}_{\mathcal{P}_\ell}\}$ and $\{\mathcal{O}_{\mathcal{P}_\ell}\}$ that approximate $\mathcal{C}$ uniformly arbitrarily well. For optimization purposes, however, this is not efficient. We would rather like to obtain a high approximation accuracy in those parts of the feasible set which are relevant for the optimization, and we would like to invest as little computational effort as possible into uninteresting parts. Therefore, we use information gained through the objective function.

First note that, once an edge $\{u, v\}$ is chosen for bisection, it makes sense to partition all simplices containing this edge at the same time. Otherwise, $\{u, v\}$ would remain an edge in $E_{\mathcal{P}}$, and the corresponding cone $\mathcal{I}_{\mathcal{P}}$ would not change. We bisect all simplices at the new vertex $w := \lambda u + (1 - \lambda)v$. Experiments with various values of λ showed no big effects, whence we simply use $\lambda = \frac{1}{2}$, i.e., we perform midpoint bisection throughout.

Furthermore, observe that, when an edge $\{u, v\} \in E_{\mathcal{P}}$ is splitted, the corresponding inequality $u^T X v \geq 0$ is removed from the system describing $\mathcal{I}_{\mathcal{P}}$ and replaced by several new inequalities (cf. Example 3.2). All other inequalities present before the bisection step are also present after bisection. As the optimal value of an LP does not change if an inactive constraint is removed, it makes sense to consider for splitting only edges $\{u, v\} \in E_{\mathcal{P}}$ corresponding to active constraints, i.e., edges with $u^T X^{\mathcal{I}} v = 0$ (where $X^{\mathcal{I}}$ is the solution of (ILP) in Step 2 of the algorithm). Only in this way can we hope to improve the solution of the inner approximation.

We call an edge $\{u, v\} \in E_{\mathcal{P}}$ with $u^T X^{\mathcal{I}} v = 0$ an *active edge* and choose in Step 7 of Algorithm 1, the longest of the edges active in $X^{\mathcal{I}}$ for bisection. The next lemma states that such an edge always exists:

Lemma 5.1. *In Step* 7 *of Algorithm* 1*, there always exists* $\{u, v\} \in E_{\mathcal{P}}$ *with* $u^T X^{\mathcal{I}} v = 0$.

Proof. The proof relies on the fact that the optimal value of an LP does not change if constraints which are inactive at the solution are omitted. The solution $X^{\mathcal{I}}$ of problem (ILP) clearly fulfills $X^{\mathcal{I}} \in \mathcal{I}_{\mathcal{P}}$, i.e., $u^T X^{\mathcal{I}} v \geq 0$ for all $\{u, v\} \in E_{\mathcal{P}}$ and $v^T X^{\mathcal{I}} v \geq 0$ for all $v \in V_{\mathcal{P}}$. Assume by contradiction that all constraints $u^T X v \geq 0$ with $\{u, v\} \in E_{\mathcal{P}}$ are inactive. Then the solution of (ILP) does not change if those constraints are omitted. But this means that $X^{\mathcal{I}}$ also solves (OLP), whence the algorithm stops in Step 4 with a zero gap. □

Selecting in Step 7 of Algorithm 1 one of the longest active edges may not result in a sequence of partitions $\{\mathcal{P}_\ell\}$ with $\delta(\mathcal{P}_\ell) \to 0$. Instead of $\delta(\mathcal{P}_\ell)$, we now have to monitor the length $\alpha(\mathcal{P}_\ell)$ of the longest active edge in $\mathcal{P}_\ell$. If this quantity goes to zero, then the algorithm converges:

THEOREM 5.2. *Assume that* (CP) *has a strictly feasible point and a bounded feasible set. Let* $\{\mathcal{P}_\ell\}$ *be a sequence of simplicial partitions generated from* $\mathcal{P}_0 = \{\Delta^S\}$ *by bisecting one of the respective longest active edges* $\{u_\ell, v_\ell\}$. *Assume further that the length* $\alpha(\mathcal{P}_\ell)$ *of the respective longest edge in* $\mathcal{P}_\ell$ *goes to zero as* $\ell \to \infty$. *Then*

$$\lim_{\ell\to\infty} \max(\mathrm{ILP}_{\mathcal{P}_\ell}) = \max(\mathrm{CP}).$$

Proof. Convergence Theorem 4.2 cannot be directly applied since we do not necessarily have $\delta(\mathcal{P}_\ell) \to 0$ as $\ell \to \infty$. However, we show that there exists a sequence $\{\mathcal{R}_\ell\}$ of partitions which fulfills $\max(\mathrm{ILP}_{\mathcal{P}_\ell}) = \max(\mathrm{ILP}_{\mathcal{R}_\ell})$ for all $\ell \in \mathbb{N}$, and $\delta(\mathcal{R}_\ell) \to 0$ as $\ell \to \infty$.

Consider $\mathcal{P}_\ell$ for some $\ell \in \mathbb{N}$, and let X^ℓ be the solution of the inner approximation problem $(\mathrm{ILP}_{\mathcal{P}_\ell})$. Since $\alpha(\mathcal{P}_\ell)$ denotes the length of the longest active edge in $\mathcal{P}_\ell$, edges in $\mathcal{P}_\ell$ with length greater than $\alpha(\mathcal{P}_\ell)$ are necessarily inactive.

We construct $\mathcal{R}_\ell$ from $\mathcal{P}_\ell$ by splitting all edges in $\mathcal{P}_\ell$ which are longer than $\alpha(\mathcal{P}_\ell)$. If necessary, we repeat this process until no edge of length greater than $\alpha(\mathcal{P}_\ell)$ remains. All edges which are in $\mathcal{P}_\ell$ but not in $\mathcal{R}_\ell$ were splitted in the process of constructing $\mathcal{R}_\ell$. Therefore, they had length greater than $\alpha(\mathcal{P}_\ell)$ and thus were inactive with respect to the optimal solution of $(\mathrm{ILP}_{\mathcal{P}_\ell})$. Let (AUX) be the linear program which has the same constraints as $(\mathrm{ILP}_{\mathcal{P}_\ell})$ except for those induced by an edge from $E_{\mathcal{P}_\ell} \setminus E_{\mathcal{R}_\ell}$. Removing inactive constraints from an LP does not change the optimal value, so $\max(\mathrm{ILP}_{\mathcal{P}_\ell}) = \max(\mathrm{AUX})$. Adding constraints cannot increase the optimal value, so $\max(\mathrm{ILP}_{\mathcal{R}_\ell}) \leq \max(\mathrm{AUX}) = \max(\mathrm{ILP}_{\mathcal{P}_\ell})$. On the other hand, $\mathcal{R}_\ell$ is by construction a refinement of $\mathcal{P}_\ell$, so we immediately get $\max(\mathrm{ILP}_{\mathcal{P}_\ell}) \leq \max(\mathrm{ILP}_{\mathcal{R}_\ell})$ from Lemma 3.1. Consequently, the two values are equal.

Observe that $\delta(\mathcal{R}_\ell) \leq \alpha(\mathcal{P}_\ell)$. Now the assumption $\alpha(\mathcal{P}_\ell) \to 0$ implies $\delta(\mathcal{R}_\ell) \to 0$ as $\ell \to \infty$, so $\{\mathcal{R}_\ell\}$ fulfills the prerequisites of Theorem 4.2, and hence

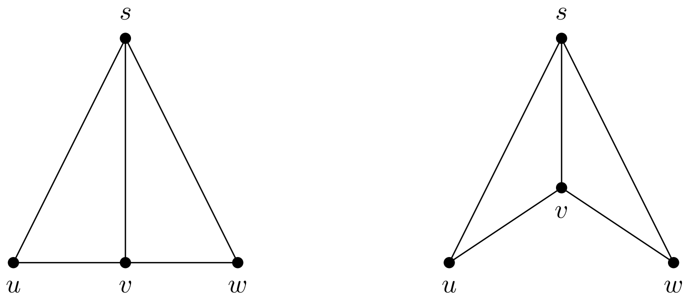
$$\lim_{\ell\to\infty} \max(\mathrm{ILP}_{\mathcal{P}_\ell}) = \lim_{\ell\to\infty} \max(\mathrm{ILP}_{\mathcal{R}_\ell}) = \max(\mathrm{CP}),$$

and the proof is complete. □

In practical implementations of our algorithm, it may happen that $\alpha(\mathcal{P}_\ell) \not\to 0$ such that convergence is not guaranteed. However, we never observed nonconvergence in our test instances (cf. section 6). If convergence does not occur, it may be necessary to alternate between bisection of the longest edge and bisection of the longest active edge to maintain convergence.

Observe that Theorem 5.2 ensures convergence of the inner approximations but not of the outer approximations. Therefore, the adaptive algorithm which splits along the longest active edges might have a positive gap. In our experiments, this seemed unproblematic. However, if the outer approximations fail to converge, a remedy is to use additional points for the outer approximation in such a way that these points eventually become dense in Δ^S.

5.2. Handling redundancies. Given a partition $\mathcal{P}$, the description $\mathcal{O}_{\mathcal{P}} := \{A \in \mathcal{S} : v^T A v \geq 0 \text{ for all } v \in V_{\mathcal{P}}\}$ does not contain any redundant inequalities.



(a) The inequalities induced by vertex v and edge $\{s, v\}$ are redundant.

(b) The inequality induced by edge $\{s, v\}$ is redundant.

FIG. 5.1. *Two situations where redundancies occur.*

This follows from the fact that for any $v \in V_{\mathcal{P}}$ and $\mathcal{H} := \{X \in \mathcal{S} : v^T X v = 0\}$ the set $\mathcal{H} \cap \mathcal{O}_{\mathcal{P}}$ is a facet of $\mathcal{O}_{\mathcal{P}}$. Indeed, assume $v^T X v = 0$ does not define a facet of $\mathcal{O}_{\mathcal{P}}$. Then there exist vertices $v_1, \ldots, v_s \in V_{\mathcal{P}}$ different from v, and $\alpha \in \mathbb{R}^s_+$ such that

$$vv^T = \sum_{i=1}^{s} \alpha_i v_i v_i^T.$$

But this contradicts the fact that vv^T is an extremal ray of $\mathcal{C}^*$.

Consequently, the description of $\mathcal{O}_{\mathcal{P}}$ contains no redundancies. Note that every bisection step generates precisely one additional vertex. Therefore, a partition $\mathcal{P}$ with m simplices has $|V_{\mathcal{P}}| = n + m$ vertices. This means that the size of the linear systems describing $\mathcal{O}_{\mathcal{P}}$ grows moderately during the iterations of our algorithm.

In contrast to this, the representation

$$\begin{aligned} \mathcal{I}_{\mathcal{P}} := \{A \in \mathcal{S} : v^T A v \geq 0 \ &\text{ for all } v \in V_{\mathcal{P}}, \\ u^T A v \geq 0 \ &\text{ for all } \{u, v\} \in E_{\mathcal{P}}\} \end{aligned}$$

contains a lot of redundancy, as has already been shown in Example 3.2. Redundancies are generated in situations as the following:

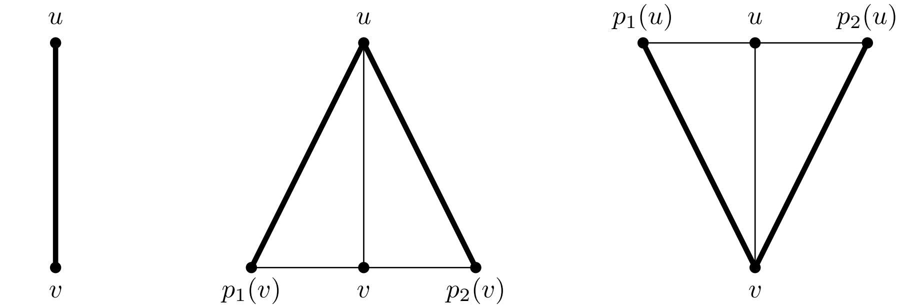
Example 5.3.

(a) For some partition $\mathcal{P}$, let $s, u, v, w \in V_{\mathcal{P}}$, and let $v = \lambda u + (1 - \lambda)w$ with some $\lambda \in (0, 1)$. Assume that $\{s, u\}$, $\{s, v\}$, $\{s, w\}$, $\{u, v\}$, $\{v, w\} \in E_{\mathcal{P}}$. See Figure 5.1(a) for a picture of this setting.
Then the inequalities $u^T A v \geq 0$ and $w^T A v \geq 0$ imply

$$(\lambda u + (1 - \lambda) w)^T A v \geq 0 \qquad \Leftrightarrow \qquad v^T A v \geq 0,$$

whence the latter inequality is redundant. Likewise, $u^T A s \geq 0$ and $w^T A s \geq 0$ imply $v^T A s \geq 0$, showing that this is a redundant inequality, too.

(b) The situation is similar if we have $v = \lambda s + \mu u + (1 - \lambda - \mu) w$ with $\lambda, \mu, (1 - \lambda - \mu) > 0$ (see Figure 5.1(b)).
As before, $s^T A v \geq 0$ is a convex combination of the inequalities $s^T A s \geq 0$, $s^T A u \geq 0$, and $s^T A w \geq 0$, and is therefore redundant.
More complicated examples can be constructed analogously.



(a) $\{u, v\} \in I$ (b) $\{u, p_1(v)\}, \{u, p_2(v)\} \in I$ (c) $\{v, p_1(u)\}, \{v, p_2(u)\} \in I$

FIG. 5.2. *The three cases where $e\{u, v\}$ is true. Edges which belong to the set I are drawn bold.*

Observe that situations like in the example occur in abundance by construction of the partition. In order to speed up our algorithm, it is therefore essential to find a way to deal with these redundancies. Note that an n-dimensional simplex has $n+1$ vertices and $\binom{n+1}{2}$ edges. Hence, for a partition $\mathcal{P}$ with m simplices, we have $|V_{\mathcal{P}}| + |E_{\mathcal{P}}| = \frac{1}{2}m(n+1)(n+2)$, so the full system describing $\mathcal{I}_{\mathcal{P}}$ would have that many constraints. This number grows too quickly, so we have to be careful to keep the system irredundant.

Unfortunately, we can not simply eliminate redundant inequalities and forget about them, since a redundant constraint may become irredundant in later iterations. This happens if a redundant inequality is a convex combination of others, and an edge corresponding to one of the "parent inequalities" is bisected in a later iteration. This phenomenon makes it necessary to keep track of the history of all vertices and edges in the partition. We do this by introducing suitable maps.

DEFINITION 5.4. *Assume that for all ℓ, partition $\mathcal{P}_{\ell+1}$ is generated from $\mathcal{P}_\ell$ through bisection of an edge in $E_{\mathcal{P}_\ell}$. We call two vertices $u, w \in V_{\mathcal{P}}$* parents *of v, if the edges $\{u, v\}$ and $\{v, w\}$ are edges of the partition $\mathcal{P}$ and there exists $\lambda \in (0, 1)$ such that $v = \lambda u + (1 - \lambda)w$. We call a map*

$$p : V_{\mathcal{P}} \to V_{\mathcal{P}} \times V_{\mathcal{P}}$$

with the property that $p(v)$ are parents of v a parent map.

For a given set $I \subset E_{\mathcal{P}}$ and for $\{u, v\} \in E_{\mathcal{P}}$, we define the boolean function e as

$$e : \{u, v\} \mapsto \begin{cases} \text{true} & \textit{if } \{u, v\} \in I \\ & \textit{or } \{u, p_1(v)\}, \{u, p_2(v)\} \in I \\ & \textit{or } \{v, p_1(u)\}, \{v, p_2(u)\} \in I \\ \text{false} & \textit{else.} \end{cases}$$

(See Figure 5.2 *for an illustration). We write e_I if it is necessary to emphasize that e depends on the set I.*

Note that for a partition $\mathcal{P}$ there may exist several parent maps. In what follows it does not matter which one is used. The most natural one (which we use in our implementation) is the "historical" parent map; i.e., if edge $\{u, v\}$ is splitted at the point w, we define $p(w) = (u, v)$.

The next lemma states that if $e\{u, v\}$ is true, then $\{u, v\}$ is a redundant edge.

Lemma 5.5. *Let p be a parent map and let $u, v \in V_{\mathcal{P}}$. If $e\{u,v\} =$ true, then there exist $\{u_1, v_1\}, \{u_2, v_2\} \in I$ and $\lambda \in (0,1)$ such that*

$$u^T A v = \lambda u_1^T A v_1 + (1-\lambda) u_2^T A v_2 \quad \textit{for all } A \in \mathcal{S}.$$

Proof. This follows immediately from the definitions. □

Definition 5.6. *Let $\{u,v\} \in I$. An edge $\{s,t\} \in E_{\mathcal{P}}$ is said to* depend on $\{u,v\}$, *if $e_I(s,t) =$ true and $e_{I \setminus \{\{u,v\}\}}(s,t) =$ false.*

We use the set I to generate a less redundant description of $\mathcal{I}_{\mathcal{P}}$. We start with $I = \{\{e_i, e_j\} : i, j = 1, \ldots, n\}$. If the partition is refined by splitting the edge $\{u,w\} \in I$ at the point $v = \lambda u + (1-\lambda) w$ with some $\lambda \in (0,1)$, then the set I is updated as follows:

- remove the edge $\{u,w\}$ from I,
- insert the edges $\{u,v\}$ and $\{v,w\}$ into I,
- for all $\{s,t\} \in E_{\mathcal{P}}$: if $\{s,t\}$ depends on $\{u,w\}$, then insert $\{s,t\}$ into I.

The next lemma shows that the set I is indeed sufficient to describe the cone $\mathcal{I}_{\mathcal{P}}$:

Lemma 5.7. *If $\mathcal{P}$ is generated from the standard simplex by bisections and the updating procedure for I described above is used, then*

$$\mathcal{I}_{\mathcal{P}} = \mathcal{I}_I := \{X \in \mathcal{S} : u^T X v \geq 0 \textit{ for all } \{u,v\} \in I\}.$$

Proof. We have $I \subseteq E_{\mathcal{P}}$ because the only edge leaving $E_{\mathcal{P}}$ also leaves I, and every edge inserted into I is an element of $E_{\mathcal{P}}$. Thus, $\mathcal{I}_I \supseteq \mathcal{I}_{\mathcal{P}}$.

Let $\{u,v\} \in E_{\mathcal{P}}$. Then $\{u,v\} \in I$ or $e\{u,v\} =$ true. Obviously the update procedure maintains this property. Using Lemma 5.5, it follows that $\mathcal{I}_I \subseteq \mathcal{I}_{\mathcal{P}}$. □

The third point of the update procedure requires knowledge of $E_{\mathcal{P}}$, whence we have to store also this information. The set $E_{\mathcal{P}}$ can also be updated efficiently:

Set $E = \{\{e_i, e_j\} : i, j = 1, \ldots, n;\ i \neq j\}$. Then obviously $E = E_{\{\Delta^S\}}$. If an edge $\{u,v\}$ is bisected at a point w, the set E is updated as follows:

- remove the edge $\{u,v\}$ from E,
- insert $\{u,w\}$ and $\{w,v\}$ into E,
- if $\{u,s\} \in E$ and $\{v,s\} \in E$, then insert $\{w,s\}$ into E.

The next lemma implies that this update procedure works, i.e., $E = E_{\mathcal{P}}$.

Lemma 5.8. *Let $\{u,v\}, \{v,w\}, \{w,u\} \in E_{\mathcal{P}}$. Then there is a simplex $\Delta \in \mathcal{P}$ such that u, v, and w are vertices of Δ.*

Proof. If $u, v, w \in \Delta^S$, then $\operatorname{conv}\{u,v,w\} \subset \Delta^S$. Since $\mathcal{P}$ is a partition of Δ^S, there exist $\Delta^1, \ldots, \Delta^m \in \mathcal{P}$ such that $\operatorname{conv}\{u,v,w\} \subseteq \bigcup_{i=1}^m \Delta^i$. Let m be minimal in the sense that $\operatorname{conv}\{u,v,w\}$ is not covered by any subset of $\{\Delta^1, \ldots, \Delta^m\}$ and assume $m > 1$. Then there exists a vertex $s \in V_{\{\Delta^1,\ldots,\Delta^m\}} \setminus \{u,v,w\}$ with $s \in \operatorname{conv}\{u,v,w\}$. Since $\mathcal{P}$ is constructed through bisections, there must be a vertex on one of the edges $\{u,v\}, \{v,w\}, \{w,u\}$. This contradicts $\{u,v\}, \{v,w\}, \{w,u\} \in E_{\mathcal{P}}$. □

5.3. Tuning the starting partition. Many interesting copositive programs arise from dualization of a completely positive program of the form

$$(\mathrm{CP}^*) \qquad \begin{array}{ll} \min & \langle C, X \rangle \\ \text{s.t.} & \langle A_i, X \rangle = b_i, \quad i \in \{1, \ldots, m\} \\ & X \in \mathcal{C}^*. \end{array}$$

This holds in particular for many combinatorial problems. For example, the stability number $\alpha(G)$ of a graph $G = (V_G, E_G)$ fulfills (cf. [15])

$$\frac{1}{\alpha(G)} = \min\{\langle Q, X \rangle : \langle E, X \rangle = 1, X \in \mathcal{C}^*\},$$

where $Q = (A_G + I)$ and A_G is the adjacency matrix of G. Often a good feasible solution X of (CP*) can be obtained through some heuristic procedure. For instance, for any stable set $S \subset V_G$ take the vector x to be a suitably scaled version of the characteristic vector of S, and take X to be xx^T.

The dual of (CP*) is a copositive program of the form

$$\begin{aligned} \max \quad & \sum_{i=1}^m b_i y_i \\ \text{s. t.} \quad & Z = C - \sum_{i=1}^m y_i A_i, \\ & Z \in \mathcal{C}, y \in \mathbb{R}^m. \end{aligned}$$

This form is equivalent to (CP); i.e., each copositive program can be transformed from one form to the other.

By weak duality, for any feasible solution X of (CP*) the value $\langle C, X\rangle$ is an upper bound for the copositive problem, so it is desirable to initialize Algorithm 1 with an outer approximation $\mathcal{O}_{\mathcal{P}_0}$ yielding a bound not worse than $\langle C, X\rangle$. The next lemma states that this is always possible.

Lemma 5.9. *Let X be feasible for* (CP*). *Then there exists a simplicial partition $\mathcal{P}$ such that the optimal value of the outer approximation* (OLP$_{\mathcal{P}}$) *is at most $\langle C, X\rangle$.*

Proof. Since $X \in \mathcal{C}^*$, it can be decomposed as $X = \sum_{k=1}^r v_k v_k^T$ with $v_1, \dots, v_r \in \mathbb{R}^n_+$. Set $w_k := \frac{v_k}{\|v_k\|_1}$. Then $w_k \in \Delta^S$, and therefore there exists a simplicial partition $\mathcal{P}$ such that $w_1, \dots, w_r \in V_{\mathcal{P}}$. Let (Z, y) be an optimal (dual) solution of the outer approximation, i.e., $(C - \sum_{i=1}^m y_i A_i) = Z \in \mathcal{O}_{\mathcal{P}}$. This implies

$$\begin{aligned} & w_k^T \left(C - \sum_{i=1}^m y_i A_i \right) w_k \geq 0 \qquad \text{for all } k \in \{1, \dots, r\} \\ \Leftrightarrow \quad & \|v_k\|_1^2 w_k^T \left(C - \sum_{i=1}^m y_i A_i \right) w_k \geq 0 \qquad \text{for all } k \in \{1, \dots, r\} \\ \Rightarrow \quad & \sum_{k=1}^r v_k^T \left(C - \sum_{i=1}^m y_i A_i \right) v_k \geq 0 \\ \Leftrightarrow \quad & \langle C, X\rangle - \sum_{i=1}^m y_i \langle A_i, X\rangle \geq 0 \\ \Leftrightarrow \quad & \langle C, X\rangle \geq \sum_{i=1}^m y_i b_i, \end{aligned}$$

which was to be shown. □

The partition $\mathcal{P}$ with $w_1, \dots, w_r \in V_{\mathcal{P}}$ can be generated iteratively by performing a radial subdivision as described in section 2 for each of the w_i at a time. Observe that in order to construct $\mathcal{P}$ it is necessary to have the decomposition $X = \sum_{k=1}^r v_k v_k^T$ (with $v_k \geq 0$ for all k) of the completely positive X. Determining this decomposition for general $X \in \mathcal{C}^*$ is a nontrivial task. However, in the combinatorial applications we have in mind (max clique, QAP, 0/1-quadratic programming), every feasible solution corresponds to a rank-one completely positive matrix X which can be utilized as described above.

6. Numerical results. We implemented our algorithm in C++ and tested our implementation on a Pentium IV, 2.8GHz Linux machine with 1GB RAM. As a solver for the linear subproblems we used COIN-OR Linear Program Solver (CLP, Version 1.3.3).

We first report results obtained for some instances of the standard quadratic optimization problem, i.e., the problem of minimizing a nonconvex quadratic form over the standard simplex:

$$\min_{x\in\Delta^S} x^T Qx. \tag{6.1}$$

This is a well-studied problem which can be restated as the copositive program

$$\begin{array}{ll} \max & \lambda \\ \text{s.t.} & Q-\lambda E\in\mathcal{C}, \\ & \lambda\in\mathbb{R}. \end{array}$$

We first discuss the behavior of our algorithm on four examples taken from [4]. These authors solve the problems by using the LP-based approximations $\mathcal{C}^r$ and the SDP-based approximations $\mathcal{K}^r$ discussed in section 1.2. As mentioned there, these approaches provide only one-sided bounds on the optimum, without any information on the solution quality. An exception is [4], where approximation estimates are given. Those bounds, however, require knowledge or a good estimate of the range (maximum minus minimum) of the quadratic form over Δ^S.

The instances

$$Q_1=\begin{pmatrix}1&0&1&1&0\\0&1&0&1&1\\1&0&1&0&1\\1&1&0&1&0\\0&1&1&0&1\end{pmatrix} \quad\text{and}\quad Q_2=\begin{pmatrix}1&0&0&0&0&0&1&1&1&1&1&1\\0&1&0&0&1&1&0&0&1&1&1&1\\0&0&1&1&0&1&0&1&0&1&1&1\\0&0&1&1&1&0&1&0&1&0&1&1\\0&1&0&1&1&0&1&1&0&1&0&1\\0&1&1&0&0&1&1&1&1&0&0&1\\1&0&0&1&1&1&1&0&0&1&1&0\\1&0&1&0&1&1&0&1&1&0&1&0\\1&1&0&1&0&1&0&1&1&1&0&0\\1&1&1&0&1&0&1&0&1&1&0&0\\1&1&1&1&0&0&1&1&0&0&1&0\\1&1&1&1&1&1&0&0&0&0&0&1\end{pmatrix}$$

are Examples 5.1 and 5.2 from [4] and correspond to the problem of determining the clique number in a pentagon and an icosahedron, respectively. The optimal values are $\frac{1}{2}$ for Q_1 and $\frac{1}{3}$ for Q_2, respectively. Our algorithm solves instance Q_1 to optimality (i.e., the gap between upper and lower bound is closed) in six iterations (0.01 sec) and instance Q_2 in 158 iterations (0.54 sec). In the latter instance we used the fact that the reciprocal of the optimal value is integer such that both lower and upper bounds could be rounded accordingly.

Using the approximating cones $\mathcal{C}^r$ and $\mathcal{K}^r$, Bomze and de Klerk [4] obtain the following results: for instance Q_1, they get the bound 0 when using $\mathcal{C}^0$ and $\frac{1}{3}$ when using $\mathcal{C}^1$. The cones $\mathcal{K}^0$ and $\mathcal{K}^1$ yield the bounds $\frac{1}{\sqrt{5}}$ and $\frac{1}{2}$, respectively. Hence, for this instance $\mathcal{K}^1$ yields the exact solution.

For instance Q_2, the respective numbers are 0 for the cone $\mathcal{C}^1$ and 0.309 for the cone $\mathcal{K}^1$. In this case, the bound obtained by using $\mathcal{K}^1$ is not exact. To use higher order approximations $\mathcal{K}^r$ with $r > 1$ is difficult, since the dimension of those problems is rapidly increasing. Bomze and de Klerk do not report a bound obtained by using $\mathcal{K}^2$.

The next instance

$$Q_3 = \begin{pmatrix} -14 & -15 & -16 & 0 & 0 \\ -15 & -14 & -12.5 & -22.5 & -15 \\ -16 & -12.5 & -10 & -26.5 & -16 \\ 0 & -22.5 & -26.5 & 0 & 0 \\ 0 & -15 & -16 & 0 & -14 \end{pmatrix}$$

is Example 5.3 from [4] and arises in a model in population genetics. Its optimal value is $-16\frac{1}{3}$. Bomze and de Klerk report the bounds -21 for $\mathcal{C}^1$, while cone $\mathcal{K}^1$ gives the exact result. Our algorithm takes 44 iterations (0.03 sec) to solve the problem with an accuracy of 10^{-6}.

Finally,

$$Q_4 = \begin{pmatrix} 0.9044 & 0.1054 & 0.5140 & 0.3322 & 0 \\ 0.1054 & 0.8715 & 0.7385 & 0.5866 & 0.9751 \\ 0.5140 & 0.7385 & 0.6936 & 0.5368 & 0.8086 \\ 0.3322 & 0.5866 & 0.5368 & 0.5633 & 0.7478 \\ 0 & 0.9751 & 0.8086 & 0.7478 & 1.2932 \end{pmatrix}$$

corresponds to Example 5.3 from [4] after homogenization. This is an example coming from portfolio optimization. The results reported in [4] are 0.3015 for $\mathcal{C}^1$ and 0.4839 for $\mathcal{K}^1$, which is optimal. Our algorithm takes 27 iterations (0.01 sec) to obtain an accuracy of 10^{-6}.

Next, we consider an example taken from Peña et al. [18]. For a graph with 17 vertices, they propose bounds on the clique number obtained by solving SDPs over the cones $\mathcal{Q}^r$ (cf. section 1.2). They state that using $\mathcal{Q}^4$ is beyond current computational capabilities. This is indeed a hard instance: our algorithm solves this problem to optimality in 14,411 iterations (20 hours, 18 min, and 5 sec). In this instance we again used the fact that the reciprocal is integer to round the bounds appropriately.

We also tested some of the max-clique instances from the Second DIMACS Challenge ([8]). The smallest instance, Johnson 8-2-4, a graph with 28 vertices, was solved to optimality in 946 iterations (1 min and 33 sec). We also solved the Hamming 6-4 instance, a graph with 64 vertices. This instance took 2,385 iterations (57 min and 52 sec). For all other instances from this library, our algorithm produced only poor bounds within two hours of computation time (the best lower bound was usually 3, and the upper bound stayed at $+\infty$).

We also tried to solve other combinatorial problems like the quadratic assignment problem using a formulation of Povh and Rendl [20]. However, for most of these instances, our algorithm gave only trivial or weak bounds.

Finally, we generated random instances of the standard quadratic optimization problem (6.1), where the entries of the symmetric matrix $Q \in \mathbb{R}^{n\times n}$ were uniformly distributed in $[-n, n]$. For each size, 100 instances where generated. The algorithm was stopped when the relative gap between upper and lower bound was smaller than 10^{-6}. The results are listed in Tables 6.1 and 6.2.

The first column in the tables denotes the problem dimension, i.e., the number of variables. The 2nd and 3rd columns describe the average and maximal number of iterations. Finally, the last four columns give information about the cpu-time.

Table 6.1

Numerical results for randomly generated instances of the standard quadratic optimization problem, obtained on a Pentium IV, 2.8*GHz Linux machine with* 1*GB RAM. All problems were solved up to a relative tolerance of* 10^{-6}.

	Iterations		cpu-time (sec.)			
n	avg	max	init	avg	min	max
10	4.25	38	0.0001	0.0034	0	0.04
30	3.26	26	0.0019	0.0056	0	0.05
50	3.78	40	0.0046	0.0124	0	0.11
100	3.32	34	0.0269	0.0557	0.01	0.68
200	2.97	35	0.1154	0.2202	0.04	3.02
500	3.17	27	0.5451	1.5483	0.38	14.26
750	2.92	23	1.9535	3.4373	0.88	30.24
1,000	3.14	29	2.5706	5.9362	1.48	59.89
1,500	4.33	75	5.9710	19.4610	3.51	366.35
2,000	2.85	24	11.4993	23.9875	6.26	225.21

Table 6.2

Numerical results for randomly generated instances of the standard quadratic optimization problem, obtained on a 16 *Dual-Core AMD Opteron*$^{\text{TM}}$ 8220 *machine with* 2.8*GHz frequency and* 130*GB RAM. Only one core was used in our computations. All problems were solved up to a relative tolerance of* 10^{-6}.

	Iterations		cpu-time (sec.)			
n	avg	max	init	avg	min	max
2,500	3.13	53	8.9037	30.3367	7.22	571.38
3,000	2.56	22	14.3911	34.5022	10.23	338.79
4,000	2.85	25	26.6361	70.8114	18.48	698.08
5,000	2.45	18	44.2364	101.155	31.18	872.96
7,000	2.45	23	91.2996	203.620	59.89	2,187.65
10,000	2.97	27	192.3010	477.258	116.08	5,184.74

The cpu-time was measured in two parts: The first part is the initialization time, which is the time needed to set up the starting LP and feed it to the solver. The initialization time is the same for all instances of the same size and is listed in the column init. The second part is the actual solution time, which is the elapsed time from solving the starting LP to termination of the algorithm. This time differs not only with the size but also with the data of the instance. Therefore, the average, minimum, and maximum solution times are stated in the respective columns.

As can be seen from Table 6.1, the solution times for these problems are not bad. However, our algorithm requires a lot of memory, and for this reason higher dimensional problems took more time on this computer due to memory swapping. Therefore, we did some further experiments on a computer with larger memory: We used a 16 Dual-Core AMD Opteron$^{\text{TM}}$ 8220 machine with 2.8GHz frequency and 130GB RAM. Our algorithm used only one of the CPUs. On this machine, we were able to solve even higher dimensional problems in very reasonable time, as can be seen in Table 6.2.

Observe that in all instances the number of iterations of our algorithm is very low and comparable to interior point methods.

To provide some intuition on how difficult the problems in Tables 6.1 and 6.2 are to solve to global optimality, we tried solving these problems with BARON [22] which is available via the NEOS server. Obviously, it was impossible to solve 100 instances for each dimension through the NEOS server. Therefore, we were only able to try a few random instances, which admittedly only give a rough picture. Nonetheless, we believe that running 100 instances per dimension would not give an entirely different pattern. Observe that on the NEOS server, each job is allotted a run time of 1,000 seconds only. NEOS currently uses version BARON 8.1.4.

We observed that for instances of size 10 the solution times of BARON were similar to ours. With instances of size 30 and bigger, BARON did not succeed to solve the problems to optimality within the given 1,000 seconds. For instances of size 250 we observed that BARON ran into memory problems and accordingly returned an error message. So it seems that BARON cannot compete with our method for this specific type of problems in large dimensions.

Bomze and de Klerk [4] also state some numerical results for randomly generated instances of the standard quadratic optimization problem. They did calculations for the linear and semidefinite approximations resulting from the cones $\mathcal{C}^1$ and $\mathcal{K}^1$, respectively. Compared with these results, our algorithm is much faster even in consideration of the faster hardware, and is able to solve much bigger instances. Moreover, we get a guaranteed solution accuracy of at least 10^{-6} for all instances.

7. Conclusions. We introduced new polyhedral inner and outer approximations of the copositive cone and presented a solution algorithm for copositive programs which uses this approximation scheme. The advantage of our algorithm is that it does not approximate the copositive cone uniformly, but can be guided by the objective function. Numerical experiments show that the algorithm works very well for quadratic programs over the simplex.

Open points of interest are:

- Can we use our method to solve other types of quadratic optimization problems? We tried to solve some box-constrained problems, but were unable to solve even medium size instances.
- Can we tailor our method towards combinatorial problems like the quadratic assignment problem, or can we find better copositive formulations of those problems?

Our approach can easily be extended to optimization problems involving more general notions of copositivity in the sense of [11]. Here one is concerned with matrices which are copositive with respect to some general cone $\mathcal{D}$, i.e., matrices that induce a quadratic form nonnegative not over $\mathbb{R}^n_+$ but over $\mathcal{D}$. If $\mathcal{D}$ is polyhedral and pointed, then it is easy to find a base $\mathcal{B}$ such that $\mathbb{R}_+\mathcal{B} = \mathcal{D}$. Instead of working with simplicial partition of Δ^S, one then has to work with partitions of $\mathcal{B}$. The computational effort of course increases if the structure of $\mathcal{B}$ is more complex. Also, we are not aware of applications that necessitate optimization over $\mathcal{D}$-copositive matrices, so we believe the canonical setting is the most interesting, but see [7] for an application of the problem of deciding $\mathcal{D}$-copositivity of a matrix.

Acknowledgments. We are grateful to the referees for taking the time to provide detailed and highly valuable comments on this paper.

REFERENCES

[1] A. Berman and N. Shaked-Monderer, *Completely Positive Matrices*, World Scientific, River Edge, NJ, 2003.

[2] A. Berman and U. Rothblum, *A Note on the Computation of the CP-Rank*, Linear Algebra Appl., 419 (2006), pp. 1–7.

[3] I.M. Bomze, M. Dür, E. de Klerk, C. Roos, A.J. Quist, and T. Terlaky, *On copositive programming and standard quadratic optimization problems*, J. Global Optim., 18 (2000), pp. 301–320.

[4] I.M. Bomze and E. de Klerk, *Solving standard quadratic optimization problems via linear, semidefinite and copositive programming*, J. Global Optim., 24 (2002), pp. 163–185.

[5] S. Bundfuss and M. Dür, *Algorithmic copositivity detection by simplicial partition*, Linear Algebra Appl., 428 (2008), pp. 1511–1523.

[6] S. Burer, *On the copositive representation of binary and continuous nonconvex quadratic programs*, Math. Program., in print. Available at http://dx.doi.org/10.1007/s10107-008-0223-z

[7] G. Danninger, *A recursive algorithm for determining (strict) copositivity of a symmteric matrix*, Methods Oper. Res., 62 (1990), pp. 45–52.

[8] DIMACS, *Second DIMACS Challenge* (1992–1993). Test instances available at: http://dimacs.rutgers.edu/Challenges/.

[9] I. Dukanovic and F. Rendl, *Copositive programming motivated bounds on the stability and the chromatic numbers*, Math. Program., in print. Available at http://dx.doi.org/10.1007/s10107-008-0233-x

[10] M. Dür and G. Still, *Interior points of the completely positive cone*, Electronic J. Linear Algebra, 17 (2008), pp. 48–53.

[11] G. Eichfelder and J. Jahn, *Set-semidefinite optimization*, J. Convex Analysis, 15 (2008), pp. 767–801.

[12] R. Horst, *On generalized bisection of n-simplices*, Math. Comput., 218 (1997), pp. 691–698.

[13] F. Jarre, I.M. Bomze, and F. Rendl, *Quadratic Factorization Heuristics for Copositive Programming*, working paper presented at the Czech-French-German Conference on Optimization, September 2007, University of Heidelberg.

[14] F. Jarre and K. Schmallowsky, *On the computation of C^* certificates*, Report, Mathematisches Institut, Universität Düsseldorf, 2008. Available at: http://www.optimization-online.org/DB_HTML/2008/05/1969.html.

[15] E. de Klerk and D.V. Pasechnik, *Approximation of the stability number of a graph via copositive programming*, SIAM J. Optim., 12 (2002), pp. 875–892.

[16] K. G. Murty and S. N. Kabadi, *Some NP-complete problems in quadratic and nonlinear programming*, Math. Program., 39 (1987), pp. 117–129.

[17] P. Parrilo, *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*, Ph.D. Dissertation, California Institute of Technology, 2000. Available at: http://etd.caltech.edu/etd/available/etd-05062004-055516/.

[18] J. Peña, J. Vera, and L. Zuluaga, *Computing the stability number of a graph via linear and semidefinite programming*, SIAM J. Optim., 18 (2007), pp. 87–105.

[19] J. Povh and F. Rendl, *A copositive programming approach to graph partitioning*, SIAM J. Optim., 18 (2007), pp. 223–241.

[20] J. Povh and F. Rendl, *Copositive and semidefinite relaxations of the quadratic assignment problem*, to appear. Preprint available at: http://www.optimization-online.org/DB_HTML/2006/10/1502.html.

[21] A.J. Quist, E. de Klerk, C. Roos, and T. Terlaky, *Copositive relaxation for general quadratic programming*, Optim. Methods Software, 9 (1998), pp. 185–208.

[22] M. Tawarmalani and N. V. Sahinidis, *Global optimization of mixed-integer nonlinear programs: A theoretical and computational study*, Math. Program., 99 (2004), pp. 563–591.

SIAM J. Optim.
Vol. 20, No. 1, pp. 54–77

CONVERGENT NETWORK APPROXIMATION FOR THE CONTINUOUS EUCLIDEAN LENGTH CONSTRAINED MINIMUM COST PATH PROBLEM*

RANGA MUHANDIRAMGE†, NATASHIA BOLAND‡, AND SONG WANG§

Abstract. In many path-planning situations we would like to find a path of constrained Euclidean length in $\mathbb{R}^2$ that minimizes a line integral. We call this the Continuous Length-Constrained Minimum Cost Path Problem (C-LCMCPP). Generally, this will be a nonconvex optimization problem, for which continuous approaches ensure only locally optimal solutions. However, network discretizations yield weight constrained network shortest path problems (WCSPPs), which can in practice be solved to global optimality, even for large networks; we can readily find a *globally* optimal solution to an *approximation* of the C-LCMCPP. Solutions to these WCSPPs yield feasible solutions and hence *upper bounds*. We show how networks can be constructed, and a WCSPP in these networks formulated, so that the solutions provide *lower bounds* on the global optimum of the continuous problem. We give a general convergence scheme for our network discretizations and use it to prove that both the upper and lower bounds so generated converge to the global optimum of the C-LCMCPP, as the network discretization is refined. Our approach provides a computable lower bound formula (of course, the upper bounds are readily computable). We give computational results showing the lower bound formula in practice, and compare the effectiveness of our network construction technique with that of standard grid-based approaches in generating good quality solutions. We find that for the same computational effort, we are able to find better quality solutions, particularly when the length constraint is tighter.

Key words. constrained shortest paths, Eikonal equations, optimal trajectories, network optimization, global optimization

AMS subject classifications. 65K10, 90B10, 90C35

DOI. 10.1137/070695356

1. Introduction.

Path-planning problems in networks have been widely studied, with numerous applications in diverse fields such as telecommunications routing (see, for example, [10]) and airline scheduling (see, for example, [1]). Continuous path-planning problems also arise in varied contexts, such as robotics [14], highway construction [9, 4], and military path planning [8, 12, 19, 18, 2, 21]. Our interest was motivated by the problem of planning a path through a naval minefield, which led us to formulate a path-planning problem in 2D Euclidean space, having the following form.

Let $F : \mathbb{R}^2 \mapsto [0, \infty)$ be a nonnegative, Hölder continuous function defined on a compact, convex domain $\Omega \subset \mathbb{R}^2$. We refer to F as the *cost function.* In a military application F may, for example, represent the risk distribution in a spatial domain of detecting an aircraft by radar, or of a ship detonating a mine in a naval minefield. Such applications are likely to yield functions F that are nonconvex, and indeed multimodal (see, for example, [19, 2, 21]). Note that the class of Hölder continu-

*Received by the editors June 25, 2007; accepted for publication (in revised form) October 29, 2008; published electronically March 13, 2009. Funding for this work was provided by the Hackett Foundation and the Defence Science and Technology Organisation.
http://www.siam.org/journals/siopt/20-1/69535.html

†School of Information Technology, Monash University, Caufield East, 3145, Melbourne, Australia (Ranga.Mulhandiramge@infotech.monash.edu.au).

‡Department of Mathematics and Statistics, University of Melbourne, Parkville, 3010, Melbourne, Australia (natashia@unimelb.edu.au).

§Department of Mathematics and Statistics, University of Western Australia, 35 Stirling Highway, Crawley, 6009, Perth, Australia (swang@maths.uwa.edu.au).

ous functions exclude functions that have discontinuities or singularities. However, many cost functions used in the literature satisfy the Hölder condition, for example, probability of detection for submarines used by Caccetta et al. [2] and the cost functions relating to weather disruptions and the reliability of the weather forecast used by Mitchell and Sastry [16]. The total cost for a path is obtained by integrating F along that path. While minimizing the total cost, we also wish to limit the Euclidean length of our path: This can be used to model a practical time or fuel constraint. The Continuous Length-Constrained Minimum Cost Path Problem (C-LCMCPP) can be stated as follows: Find a piecewise differentiable curve in Ω between a given start point a and end point b that minimizes the line integral of F subject to the constraint that the Euclidean length of the curve is less than or equal a prescribed value $\bar{L}$.

To be more precise, let $\mathcal{C}([0,1],\Omega)$ denote all piecewise differentiable curves parameterized by $s \in [0,1]$ such that for any $p \in \mathcal{C}([0,1],\Omega)$ we have $p(s) \in \Omega$ for all $s \in [0,1]$. Define

$$\Gamma = \{p \in \mathcal{C}([0,1],\Omega) : p(0) = a, p(1) = b\}.$$

Then the C-LCMCPP can be stated mathematically as:

$$\begin{aligned} \min_{p\in\Gamma} J[p] &= \int_0^1 F(p(s))||p'(s)||ds \\ s.t.\ Eu[p] &= \int_0^1 ||p'(s)||ds \leq \bar{L}, \end{aligned} \tag{1.1}$$

where $||.||$ denotes the Euclidean norm. The piecewise differentiability of the paths in Γ make the path integrals in (1.1) well defined. An instance of the C-LCMCPP takes the form $(\Omega, F, \bar{L}, a, b)$. We are interested in the general case, with no further assumptions on F.

The C-LCMCPP could be approached directly as a continuous problem, but has more commonly been tackled via network discretization. We discuss the former approach first. The two principal continuous approaches that are applicable are (i) variational techniques, such as solution of the Euler–Lagrange equation, or (ii) solution of the Hamilton–Jacobi–Bellman equation. The former are discussed, for example, by Zabarankin et al. [21] and Caccetta et al. [2]. However, variational techniques can only ensure locally optimal solutions (see, for example, introductory remarks in Tsitsiklis [20], and references therein). Globally optimal solutions can, in principle, be obtained via the Hamilton–Jacobi–Bellman (HJB) equation (see [20] for an excellent exposition). These have been extensively explored in the case without the Euclidean length constraint, which we refer to as the C-MCPP. In this case, the problem is equivalent to solving the Eikonal equation,

$$||\nabla\tau|| = F(x, y), \tag{1.2}$$

where τ, the value function, is the time of arrival of a disturbance propagating from an initial set on which $\tau = 0$, travelling at a given “slowness” (the inverse of the speed of propagation), F, at each point. Numerical approaches to solution of this problem all involve discretization, and there have been several schemes proposed for which convergence to a global optimum has been proved; we believe Cristiani and Falcone [5] provide the most recent instance, and give a comprehensive review of previous approaches. The method in [5] is shown to converge under the relatively mild assumption that the speed function (the pointwise reciprocal of our F) is Lipschitz continuous.

For the problem of interest to us, the C-MCPP *with* the length constraint, we believe no similar methods are known. Indeed, the only approach to constrained problems via the Eikonal equation that we are aware of is that of Mitchell and Sastry [16]. Their interest is finding paths for aircraft that minimize fuel consumption (i.e., path length) subject to constraints on the probability of encountering bad weather with a penalty relating to the reliability of the weather forecast in different regions. To handle constraints, they recast them as objectives, incorporating them in the objective function with a multiplier; this returns the problem to one of solving an Eikonal equation, where now the cost function, or speed, incorporates terms related to the constraints. Their method samples from possible multipliers, and so samples from the set of Pareto-optimal solutions for the multiobjective problem. As is noted in [16], this will not necessarily yield an optimal solution to the constrained problem.

We now discuss approaches based on network discretization. In such approaches, the spatial domain Ω is discretized, and represented by a set of points, including a and b, which are used as vertices in a network. The arcs in the network restrict the path to travel only between pairs of vertices connected by an arc, and the cost of each arc is taken to be the integral of F along the straight line between the two endpoints of the arc. The problem of finding a minimum cost path in the network from a to b is now a standard network shortest path problem, which is easily solvable, with techniques such as Dijkstra's algorithm [6] or the A* algorithm [11], to give globally optimal solutions. Problems with the additional Euclidean length constraint take the form of a Weight-Constrained Shortest Path Problem (WCSPP) in a network, which is also now very well solved for practical purposes, for example, using the recent approaches of Dumitrescu and Boland [7], Carlyle and Wood [3], or Muhandiramge and Boland [17]. In either case, solving the network shortest path problem provides a feasible solution to the continuous problem, and so yields an upper bound on its value.

For further detail of how continuous problems, particularly those arising from path planning in a threat environment, can be modeled as network path problems, we refer the reader to the paper of Zabarankin et al. [21], which gives an excellent exposition. Zabarankin et al. [21] also derive analytic solutions for the case of a single point threat, and so can demonstrate computationally that in such cases the upper bounds generated from network discretizations are very close to the exact global optimum. Most work along these lines rests with constructing the network discretization and solving the corresponding network path problem: The focus is on modeling other practical complications, such as curvature constraints. For example, Piatko et al. [19, 18] discretize with points on a square grid, with arcs from each point to either its four, or eight, nearest neighbors. Similar networks were used by Fahlen [8], Caccetta et al. [2], and Zabarankin et al. [21], although [8] and [21] both describe using sixteen neighbors in the 2D case, and [21] further considers the 3D case. Curvature (turning angle) constraints are considered in [8, 21], while [2] permits variable speeds for path traversal, selected from a finite set of possible speeds.

By contrast, Kim and Hespanha [12], in work on an anisotropic form of the problem (cost depends on direction) without the length constraint, focus on finding network constructions that provide better approximations. They develop a novel network construction method that they call "honeycomb" sampling. This method selects points at random from the spatial domain, according to a probability distribution that ensures either the points are close together, or that a term related to gradients of the cost function is small. The Voronoi diagram for these points is then constructed, and nodes on the network are sampled from the edges of the Voronoi diagram. The honeycomb sampling is compared computationally with (i) a network with nodes selected uniformly at random from the spatial domain, and (ii) a network with nodes selected at random according to a probability distribution based on cost function gradients. Kim and Hespanha [12] report average reductions in the cost of the network paths found using honeycomb sampling of around 7.5% over the uniform sampling networks and around 11% over the gradient-based sampling method. Unfortunately, [12] does not say how their network nodes were connected (they don't define the arcs), so it is difficult to assess the relative computational effort for these approaches.

Network discretizations have also been explored by authors in contexts other than that of the C-LCMCPP or C-MCPP. Kimmel and Kiryati [13] used a grid network and local refinement procedure to find the minimum length path on a underlying 3D surface given a digitization of the surface. First, the surface was represented by a graph with a node for each surface voxel and an edge from each node to all the surface voxel nodes up to one unit away in each direction (a total of 26 possible different directions). Rather than weight these links by their length, they weighted them using a path length estimator which gives an unbiased estimate of the path length on the actual underlying surface. They then found the shortest path in this network using a standard network shortest path algorithm. Since grid networks suffer from discretization bias, [13] also used a curve shortening flow method [15] to shorten the path to a local optimum. Caccetta et al. [2] similarly combine an initial discretization step with a subsequent local optimization step based on variational techniques. They use a standard grid network, solve the corresponding network problem approximately, but then take the resulting path as an initial point for an optimal control solver, to derive a locally optimal solution.

As far as we are aware, the only work that considers the issue of how far the solution to the network discretization problem is from that of the original continuous problem, or considers the possibility of convergence to the globally optimal solution as the network is refined, is that of Kim and Hespanha [12]. As mentioned earlier, they tackle an anisotropic problem, and do not apply a length constraint. For this case, they provide a lower bound formula. Unfortunately, their formula involves a set of points in the spatial domain that they prove to exist, but which they don't explicitly show how to construct. They simply require the set to be "sufficiently dense". Thus they cannot readily use their formula to compute a lower bound from a path found in a given network. Furthermore, although their network construction is motivated by the theory they provide, it is not explicitly proved to converge to the globally optimal solution. Indeed, since their construction relies on randomized sampling, such a proof would have to include some kind of "almost surely" condition.

In this paper, we give a general scheme for convergence of network discretizations. With this scheme, we show that if we solve the corresponding WCSPP with path lengths constrained to $\bar{L}(1+\gamma)$, where γ depends on the network construction, then we can compute a lower bound on the global optimum of the C-LCMCPP. We prove furthermore that this bound converges to the global optimum as the network is refined in a way described later. We also prove that the solution to the WCSPP with path length constrained to $\bar{L}$ (an upper bound on the global optimum of the C-LCMCPP) also converges to the global optimum as the network is refined.

This is, of course, of theoretical interest, but from a practical point of view, we still need to construct good network discretizations. An advantage of network solution methods that make them useful for nonconvex problems is they find global optima within the network. However how accurately the network solution reflects the continuous solution depends greatly on the structure of the network used. One point that is not hard to see is that the standard grid networks, with arcs only connecting points to a handful of their nearest neighbors, cannot, in general, converge to globally optimal solutions of the C-LCMCPP. With such networks, the set of gradients available to the network path is simply not rich enough to ensure it can well approximate the optimal continuous path; grid networks suffer from significant discretization bias. A complete network on a set of grid points would suffice, but a complete network on a fine grid has an enormous number of arcs, and even efficient shortest path algorithms are unlikely to be practical if we attempt to use complete networks. (We note that in the unconstrained case of the C-MCPP, the method of Cristiani and Falcone [5] implicitly considers a much larger set of tangent directions by updating node values using the multiple neighboring node values simultaneously. This allows them to prove convergence.)

Thus the challenge is to structure a network that is "just right". It needs to be rich enough to well approximate any optimal path, but not so dense as to make solution of the network problems impractical. By structuring our network carefully, we can overcome the discretization effects with a purely network method, avoiding the need for a local refinement procedure. We can also guarantee that our solution will converge to the true optimum as we refine our network. We have met this challenge with what we call a "cellular" network construction, based on triangular tessellation of the spatial domain, and hexagonal cells. This network is sparse, while still meeting the conditions for convergence.

We give the results of numerical computations, showing the effects of refining the network discretization on the lower and upper bounds computed. We also compare the upper bounds found with those found using the standard grid approach, using computational effort. This shows that the cellular network gives better solutions, particularly when the length constraint is tighter.

Thus our contribution in this work is what we believe is the first approach to a constrained continuous minimum cost path problem that is proved to converge to the globally optimal solution, under mild assumptions on the cost function. We also provide computable lower bounds, and a network construction that is sparse, while still providing better approximations to the continuous solution than standard approaches.

The paper is structured as follows: First we formalize the concept of using a network to approximate the C-LCMCPP; next we outline the properties of network that produce a convergent solution; and lastly we create a method of constructing networks with these properties and give numerical results.

2. Network formulation. To solve the C-LCMCPP using a network formulation we create a network $G = (V, A)$ consisting of nodes and directed edges in Ω such that nodes are located at the start point a and end point b and at least one *network path* exists that connects a and b. A network path p from node v_0 to node v_m in G is a sequence of arcs $p = ((v_0, v_1), (v_1, v_2), \dots, (v_{m-1}, v_m))$ such that $(v_{k-1}, v_k) \in A$ for all $k \in \{1, \dots, m\}$. For convenience, we will assume that the graph has a unique directed edge between any ordered pair of nodes so p can be equivalently written $p = (v_0, v_1, \dots, v_{m-1}, v_m)$.

As the nodes in our network are also points in the Euclidean plane, we treat them both as abstract network nodes, e.g., $v \in V$ and also directly as coordinate in 2-space, e.g., $||v_i - v_{i+1}||$ and $v \in \Omega$. The context in which a node is mentioned indicates in what capacity it is to be treated.

We assign each edge a *cost* which is the line integral of F along the edge and a *weight* which is the Euclidean length of the edge. We then solve the corresponding weight constrained shortest path problem (WCSPP): Finding a network path from a to b that minimizes the sum of the costs of the edges in the path while keeping the total length of the path less than or equal to a given weight limit.

The quality of our network approximation depends a great deal on the structure of the network. In particular, we would like the difference between the optimal objective function value for the C-LCMCPP and the corresponding WCSPP to be as small as possible. We would also like this difference to shrink to zero as we refine our network. We formalize the network design into the concept of a *network construction* as follows.

DEFINITION 2.1. *A* network construction $\mathcal{G}$ *is a method that, given an instance* I *of the C-LCMCPP and a finite vector of real parameters* P *from a parameter domain* S*, will produce a finite directed network* $\mathcal{G}(I, P)$ *which includes* a *and* b *as nodes and in which a network path from* a *to* b *exists.*

The important point is that a network construction can take many different vectors of parameter values and thus produce many related networks for a given instance, e.g., many different grid spacings for a grid network. We would like to have a network construction that, given the right series of parameters, produces a series of networks whose WCSPP optimal objective function values converge to the objective function value of the C-LCMCPP. This is the focus of the next definition.

DEFINITION 2.2. *A* convergent network construction $\mathcal{G}$ *is a network construction for which for any instance* I *of the C-LCMCPP we can find a sequence of parameter sets* $(P_1, P_2, \dots)$ *with* $P_k \in S$ *for* $k \in \mathbb{Z}^+$ *such that the difference between the objective function value of the solution to* I *and the approximate WCSPP solution using the network* $\mathcal{G}(I, P_n)$ *goes to zero as* n *goes to infinity. The WCSPPs may use a different weight limit to the C-LCMCPP.*

In the following section, we formulate a convergent network construction and in the process obtain a calculable lower bound on the cost of the optimal solution of the C-LCMCPP.

3. $(\boldsymbol{\delta}, \boldsymbol{\epsilon}, \boldsymbol{\kappa})$-approximation networks. In this section, we outline the properties of a network that allow us to relate the solution of the WCSPP over the network to the corresponding C-LCMCPP. Such properties are given by Definition 3.1, and we call a network that satisfies these properties a $(\delta, \epsilon, \kappa)$-approximation network. We will give an example later of how such a network is constructed, but for now we concentrate on proving convergence using the abstract properties of the network without the distraction of outlining the full network construction method.

The motivation behind the definition is that to approximate a continuous path integral using a network path, we would ideally like the end points of the edges to be on the path, as standard for the Riemann sum definition of a path integral. In our case, however, we want to be able to approximate the integral for any reasonable path in our space using a finite network; thus we cannot guarantee that the approximating points will be directly on the path. Therefore, the best we can do is guarantee that the approximating points are within some distance of the path.

To make this guarantee, for any path $p \in \Gamma$, we have a sequence $p_G = (v_0, \dots, v_N)$, with $v_k \in V$ for $k \in \{0, 1, \dots, N\}$, which forms the network approximation to the path. To relate the network path to the continuous path it approximates, we have the corresponding sequence $(p(s_0), p(s_1), p(s_2), \dots, p(s_N))$ of points on the path p, in which each point, $p(s_k)$, is at most ϵ away from the nearest corresponding node, v_k, for each $k = 0, \dots, N$. The distance between these points on the path is bounded below by δ and above by $\kappa\delta$. The distance $\kappa\delta$ corresponds to the maximum on the distance between points in a Riemann sum.

Definition 3.1. *A $(\delta, \epsilon, \kappa)$-approximation network $G = (V, A)$ with $\delta > 0, \epsilon > 0, \kappa > 1 \in \mathbb{R}$ for an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP has the following properties:*

1. *the set V contains a and b;*
2. *for each node $v \in V \setminus \{b\}$ there is a closed, connected region $R_v \subseteq \Omega$ such that each R_v can be enclosed by a circle of radius $\kappa\delta$; and*
3. *for each $p \in \Gamma$ with $Eu[p] \leq \bar{L}$ there is an ordered sequence of points on the path p given by $(a = p(s_0), p(s_1), \ldots, p(s_{N-1}), p(s_N) = b)$ with $0 = s_0 < s_1 < \ldots < s_{N-1} < s_N = 1$, and a path $(a = v_0, v_1, \ldots, v_{N-1}, v_N = b)$ in the network G, i.e., a sequence with $v_k \in V$ and $(v_{k-1}, v_k) \in A$ for all $k \in \{1, \ldots, N\}$, such that:*
   (a) *$||p(s_k) - p(s_{k-1})|| \geq \delta$, for all $k \in \{1, \ldots, N\}$,*
   (b) *$||v_k - p(s_k)|| \leq \epsilon$, for all $k \in \{0, \ldots, N\}$, and*
   (c) *$p(s) \in R_{v_k}$ for all $s \in [s_k, s_{k+1}]$ and $\alpha v_k + (1-\alpha)v_{k+1} \in R_{v_k}$ for all $\alpha \in [0, 1]$, for each $k \in \{0, \ldots, N-1\}$.*

3.1. Length relationship. Consider a $(\delta, \epsilon, \kappa)$-approximation network for an instance I of the C-LCMCPP. For an optimal solution p^* of I, we have the sequence of points $(p^*(s_0), \ldots, p^*(s_N))$ on the path guaranteed by Definition 3.1 and clearly

$$Eu[p^*] \geq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||. \tag{3.1}$$

We would now like to find, for any path $p \in \Gamma$, the relationship between the Euclidean length of the piecewise linear path formed by $(p(s_0), p(s_1), p(s_2), \ldots, p(s_N))$, and that formed by the corresponding network path $(v_0, v_1, v_2, \ldots, v_N)$ satisfying the conditions of Definition 3.1.

Lemma 3.2. *Any $(\delta, \epsilon, \kappa)$-approximation network $G = (V, A)$ for an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP will have the property that for any path $p \in \Gamma$, there is a sequence of points on the path $(p(s_0), p(s_1), p(s_2), \ldots, p(s_N))$ and a sequence of nodes $(v_0, v_1, v_2, \ldots, v_N)$ with $0 = s_0 < s_1 < \ldots < s_{N-1} < s_N = 1$ and $v_k \in V$ for all $k \in \{0, \ldots, N\}$, such that $||v_k - v_{k-1}|| \leq ||p(s_k) - p(s_{k-1})||(1 + \gamma)$ for all $k \in \{1, \ldots, N\}$ and $\gamma \in \Phi_G$ where*

$$\Phi_G = \left[c_1 \frac{\epsilon}{\delta} + c_2 \frac{\epsilon^2}{\delta^2}, \infty\right) \tag{3.2}$$

for some $c_1, c_2 \in [0, 2]$ independent of p.

Proof. Let $G = (V, A)$ be a $(\delta, \epsilon, \kappa)$-approximation network for an instance I of the C-LCMCPP. For any path $p \in \Gamma$, let $(p(s_0)$, $p(s_1)$, $p(s_2)$, $\ldots, p(s_N))$ and $(v_0, v_1, v_2, \ldots, v_N)$ be the sequences guaranteed to exist by Definition 3.1. Let $L_k = ||v_k - v_{k-1}||$, $\delta_k = ||p(s_k) - p(s_{k-1})||$ for $k \in \{1, \ldots, N\}$, and $\epsilon_k = ||p(s_k) - v_k||$ for $k \in \{0, \ldots, N\}$.

For $k \in \{1, \ldots, N\}$, if $p(s_{k-1}) \neq v_{k-1}$, let α_k be the angle defined by $p(s_k)$, $p(s_{k-1})$ and v_{k-1} measured anticlockwise from the segment $\{p(s_{k-1}), p(s_k)\}$, and if $p(s_k) \neq v_k$, let β_k be the angle defined by $p(s_{k-1})$, $p(s_k)$ and v_k also measured anticlockwise from the segment $\{p(s_{k-1}), p(s_k)\}$. If $p(s_{k-1}) = v_{k-1}$, i.e., $\epsilon_{k-1} = 0$, set α_k to 0 and similarly if $p(s_k) = v_k$, i.e., $\epsilon_k = 0$, set β_k to 0.

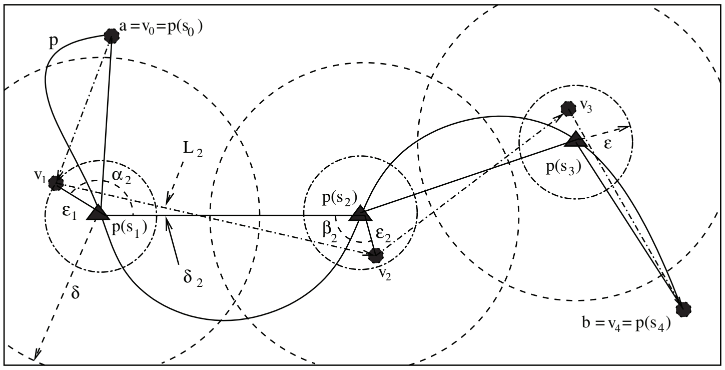


Fig. 3.1. *Representation of a path p and its network approximation. Parts of the diagram applicable to Lemma* 3.2 *for $k = 2$ are labelled. The small circles have radius ϵ and the large circles have radius δ.*

Then from Figure 3.1, we can deduce, by decomposing the segments $\{p(s_{k-1}), v_{k-1}\}$ and $\{p(s_k), v_k\}$ into components parallel and perpendicular to $\{p(s_{k-1}), p(s_k)\}$ and using Pythagoras, that

$$L_k^2 = (\delta_k - \epsilon_{k-1}\cos(\alpha_k) - \epsilon_k\cos(\beta_k))^2 + (\epsilon_{k-1}\sin(\alpha_k) + \epsilon_k\sin(\beta_k))^2.$$

To get an upper bound on how much longer L_k could be compared to δ_k, we take the absolute value of the component contributions. We also note that $0 \le \epsilon_{k-1}, \epsilon_k \le \epsilon$ due to Condition 3(b) of Definition 3.1. Using this, we then get

$$L_k^2 \le (\delta_k + \epsilon|\cos(\alpha_k)| + \epsilon|\cos(\beta_k)|)^2 + (\epsilon|\sin(\alpha_k)| + \epsilon|\sin(\beta_k)|)^2.$$

The effect of the absolute value signs on the sine and cosine function can be replicated if we make the following transformation which keeps the angles in the range $[0, \frac{\pi}{2}]$:

$$\alpha^k = \begin{cases} \alpha_k & \alpha_k \in [0, \frac{\pi}{2}], \\ \pi - \alpha_k & \alpha_k \in [\frac{\pi}{2}, \pi], \\ \alpha_k - \pi & \alpha_k \in [\pi, \frac{3\pi}{2}], \\ 2\pi - \alpha_k & \alpha_k \in [\frac{3\pi}{2}, 2\pi]. \end{cases}$$

Using the same transformation function for β_k, we then expand and simplify using trigonometric identities and then estimate L_k as follows:

$$\begin{aligned} L_k^2 &\le \delta_k^2 + 2\epsilon^2 + 2\delta_k\epsilon(\cos(\alpha^k) + \cos(\beta^k)) + 2\epsilon^2\cos(\alpha^k - \beta^k) \\ \implies L_k &\le \sqrt{\delta_k^2 + 2\epsilon^2 + 2\delta_k\epsilon(\cos(\alpha^k) + \cos(\beta^k)) + 2\epsilon^2\cos(\alpha^k - \beta^k)} \\ &= \delta_k\sqrt{1 + 2\left(\frac{\epsilon}{\delta_k}(\cos(\alpha^k) + \cos(\beta^k)) + \frac{\epsilon^2}{\delta_k^2}(1 + \cos(\alpha^k - \beta^k))\right)} \\ &\le \delta_k\sqrt{1 + 2\left(\frac{\epsilon}{\delta}(\cos(\alpha^k) + \cos(\beta^k)) + \frac{\epsilon^2}{\delta^2}(1 + \cos(\alpha^k - \beta^k))\right)} \\ &\le \delta_k\left(1 + \frac{\epsilon}{\delta}(\cos(\alpha^k) + \cos(\beta^k)) + \frac{\epsilon^2}{\delta^2}(1 + \cos(\alpha^k - \beta^k))\right) \\ &= \delta_k\left(1 + c_1^k\frac{\epsilon}{\delta} + c_2^k\frac{\epsilon^2}{\delta^2}\right), \end{aligned}$$

where we have used $\delta_k \geq \delta$ from Condition 3(a) of Definition 3.1 and the inequality $\sqrt{1+x} \leq 1 + \frac{x}{2}$, for any $x \geq 0$.

In the above, $c_1^k = \cos(\alpha^k) + \cos(\beta^k)$ and $c_2^k = 1 + \cos(\alpha^k - \beta^k)$. As $\alpha^k, \beta^k \in [0, \frac{\pi}{2}]$, this implies both $\cos(\alpha^k), \cos(\beta^k) \in [0,1]$. Also $(\alpha^k - \beta^k) \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ so $\cos(\alpha^k - \beta^k) \in [0,1]$. Thus it is clear that $c_1^k, c_2^k \in [0,2]$. We define $c_1(p)$ as $\max_{k \in \{1,\ldots,N\}} c_1^k$ and $c_2(p)$ as $\max_{k \in \{1,\ldots,N\}} c_2^k$ and note that $c_1(p), c_2(p) \in [0,2]$.

Now, let c_1 and c_2 be the supremum of $c_1(p)$ and $c_2(p)$, respectively, over all paths $p \in \Gamma$. We see that $c_1, c_2 \in [0,2]$. Thus

$$L_k \leq \delta_k \left(1 + c_1 \frac{\epsilon}{\delta} + c_2 \frac{\epsilon^2}{\delta^2}\right)$$
$$L_k \leq \delta_k (1+\gamma),$$

where $\gamma \in [c_1 \frac{\epsilon}{\delta} + c_2 \frac{\epsilon^2}{\delta^2}, \infty)$.

If we return to our original definition of L_k and δ_k we get $L_k = ||v_k - v_{k-1}|| \leq ||p(s_k) - p(s_{k-1})||(1+\gamma) = \delta_k(1+\gamma)$ for all $k \in \{1,\ldots,N\}$, where $\gamma \in [c_1 \frac{\epsilon}{\delta} + c_2 \frac{\epsilon^2}{\delta^2}, \infty) = \Phi_G$ for some $c_1, c_2 \in [0,2]$ independent of p. □

Corollary 3.3. *Let p^* be an optimal path of an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP and G be a $(\delta, \epsilon, \kappa)$-approximation network for I. Then the network approximation $p_G^* = (v_0, v_1, v_2, \ldots, v_N)$ to p^*, guaranteed to exist by Definition* 3.1*, satisfies $Eu[p_G^*] \leq \bar{L}(1+\gamma)$ for $\gamma \in \Phi_G$ where Φ_G is defined in* (3.2)*.*

Proof. Let $(s_0, \ldots, s_N)$ be defined for p^* as per Definition 3.1. Then

$$\begin{aligned} Eu[p_G^*] &= \sum_{k=1}^{N} ||v_k - v_{k-1}|| \\ &\leq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||(1+\gamma) && \text{by Lemma 3.2} \\ &\leq Eu[p^*](1+\gamma) && \text{by (3.1).} \end{aligned}$$

Now as p^* is feasible for the instance I of the C-LCMCPP, we have $Eu[p^*] \leq \bar{L}$ so

$$Eu[p_G^*] \leq \bar{L}(1+\gamma). \quad \square$$

Corollary 3.3 is important because it tells us that by relaxing the weight constraint in our network by the factor $1+\gamma$, the network path p_G^* that approximates the continuous optimal solution will be a feasible path in our WCSPP approximation.

3.2. Lower bounds. Let G be a $(\delta, \epsilon, \kappa)$-approximation network for an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP. Then for p^* an optimal solution to I, the sequences with properties given by Definition 3.1, that is $(p^*(s_0), p^*(s_1), \ldots, p^*(s_N))$ with $0 = s_0 < s_1 < \ldots < s_N = 1$ and the network approximation $p_G^* = (v_0, v_1, \ldots, v_N)$ to p^*, are guaranteed to exist. Using the sequence $(p^*(s_0), p^*(s_1), \ldots, p^*(s_N))$ to put a lower bound on the optimal solution to I we get

$$J[p^*] \geq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})|| M^{\downarrow}(v_{k-1}), \tag{3.3}$$

where $M^{\downarrow}(v_{k-1})$ is the minimum value of F on the region $R_{v_{k-1}}$. Here we have used Condition 3(c) from Definition 3.1 that the path segment between p_{k-1}^* and p_k^* is entirely in the region $R_{v_{k-1}}$. Remember $\kappa > 1$ and for each $v \in V$, R_v is a closed connected region around v that is contained in a circle of radius $\kappa\delta$.

Using the sequence $(v_0, v_1, \ldots, v_N)$ to put an upper bound on the cost of the network approximation to p^* we get

$$J[p_G^*] \leq \sum_{k=1}^{N} ||v_k - v_{k-1}||M^{\uparrow}(v_{k-1}), \tag{3.4}$$

where $M^{\uparrow}(v_{k-1})$ is the maximum value of F on the region $R_{v_{k-1}}$. Here we again use Condition 3(c) from Definition 3.1 to guarantee that the arc between v_{k-1} and v_k is entirely in the region $R_{v_{k-1}}$.

Using inequalities (3.3) and (3.4) we get

(3.5)

$$\begin{aligned}
&J[p^*] - \frac{J[p_G^*]}{1+\gamma} \\
&\geq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||M^{\downarrow}(v_{k-1}) - \sum_{k=1}^{N} \frac{||v_k - v_{k-1}||}{1+\gamma} M^{\uparrow}(v_{k-1}) \\
&\geq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||M^{\downarrow}(v_{k-1}) - \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||M^{\uparrow}(v_{k-1}) \\
&\geq \sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||(M^{\downarrow}(v_{k-1}) - M^{\uparrow}(v_{k-1})),
\end{aligned}$$

where $\gamma \in \Phi_G$. At this stage we make the following definition.

Definition 3.4. $\Delta_G = \max_{v \in V \setminus \{b\}}(M^{\uparrow}(v) - M^{\downarrow}(v))$ *or equivalently using the definition of* $M^{\uparrow}(v)$ *and* $M^{\downarrow}(v)$, $\Delta_G = \max_{v \in V \setminus \{b\}}(\max_{x \in R_v} F(x) - \min_{y \in R_v} F(y))$.

The parameter Δ_G represents the maximum over all $v \in V$ of the variation of the underlying F function over the regions R_v. Using this definition we proceed as follows:

(3.6)

$$\begin{aligned}
J[p^*] - \frac{J[p_G^*]}{1+\gamma} &\geq -\sum_{k=1}^{N} ||p^*(s_k) - p^*(s_{k-1})||\Delta_G && \text{using Definition 3.4} \\
&\geq -Eu[p^*]\Delta_G && \text{by (3.1)} \\
&\geq -\bar{L}\Delta_G && \text{as } Eu[p^*] \leq \bar{L}.
\end{aligned}$$

Rearranging (3.6) gives us the relation

$$J[p^*] \geq \frac{J[p_G^*]}{1+\gamma} - \bar{L}\Delta_G. \tag{3.7}$$

If we consider the WCSPP for network G with a relaxed weight constraint, i.e., finding the network path in G between nodes a and b with length less than or equal to $\bar{L}(1+\gamma)$, we know that any network approximation p_G^* to the optimal path p^* of instance I is feasible for the relaxed WCSPP by Corollary 3.3. Thus if q_G^* is an optimal solution to the relaxed WCSPP, then we know $J[q^*_G] \leq J[p^*_G]$ by the definition of the optimality of q^*_G. Then

$$J[p^*] \geq \frac{J[q^*_G]}{1+\gamma} - \bar{L}\Delta_G. \tag{3.8}$$

Letting $J^*(\bar{L}) = J[p^*]$ be the optimal objective function value of an instance I of the C-LCMCPP and $J^*_G(\bar{L}(1+\gamma)) = J[q^*_G]$ the optimal objective function value of the relaxed WCSPP using the network G, we get

$$J^*(\bar{L}) \geq \frac{J^*_G(\bar{L}(1+\gamma))}{1+\gamma} - \bar{L}\Delta_G. \tag{3.9}$$

Using this relation, we can calculate concrete lower bounds on the solution of the C-LCMCPP as we will demonstrate in section 5. Note that if we cannot find the optimal objective function value of the relaxed WCSPP, any lower bound on optimal value can replace $J^*_G(\bar{L}(1+\gamma))$ in the formula and produce a valid, if worse, lower bound on $J^*(\bar{L})$.

3.3. κ-regular network constructions and convergence. To create a convergent network construction we define κ-regular network constructions in Definition 3.5. When given the right series of parameters a κ-regular network construction produces a series of $(\delta, \epsilon, \kappa)$-approximation networks for which δ and $\frac{\epsilon}{\delta}$ approach zero. This property will help us show that κ-regular network constructions are convergent network constructions.

Definition 3.5. *A κ-regular network construction is a network construction $\mathcal{G}$ with parameter domain S for which for any instance I of the C-LCMCPP there exists:*

1. *A sequence of parameter vectors $(P_1, P_2, \dots)$, $P_k \in S, \forall k \in \mathbb{Z}^+$,*
2. *A sequence $(\delta_1, \delta_2, \dots)$ with $\delta_k > 0, \forall k \in \mathbb{Z}^+$ such that $\lim_{k\to\infty} \delta_k = 0$ and,*
3. *A sequence $(\epsilon_1, \epsilon_2, \dots)$ with $\epsilon_k > 0, \forall k \in \mathbb{Z}^+$ such that $\lim_{k\to\infty} \frac{\epsilon_k}{\delta_k} = 0$,*

such that $\mathcal{G}(I, P_k)$ is a $(\delta_k, \epsilon_k, \kappa)$-approximation for all $k \in \mathbb{Z}^+$.

Before we prove convergence, we need to introduce the following theorem.

Theorem 3.6. *Let $J^*(L)$ with $L \in [L_{min}, \infty)$ be the optimal objective function value of instance $I = (\Omega, F, L, a, b)$ of the C-LCMCPP, where $L_{min} = ||b - a||$ is the minimum distance between a and b. Then $J^*(L)$ is monotonically decreasing and continuous for $L \in [L_{min}, \infty)$ if F is Hölder continuous and Ω is convex.*

Proof. To show $J^*(L)$ is monotonically decreasing, we note that if $p^*(L_1)$ is an optimal solution to the C-LCMCPP for weight limit L_1, then for $L_2 > L_1$, $p^*(L_1)$ is a feasible solution to the C-LCMCPP with weight limit L_2. Thus if $L_1 < L_2$, $J[p^*(L_1)] = J^*(L_1) \geq J^*(L_2) = J[p^*(L_2)]$ so $J^*(L)$ must be monotonically decreasing. Due to space limitations, the proof that $J^*(L)$ is continuous is omitted. □

Note, however, that $J^*(L)$ may not be continuous if we allow obstacles as these would result in either a discontinuity in F or nonconvexity of Ω. In fact, it is easy to construct an example with obstacles in which the function $J^*(L)$ is discontinuous. In this paper we do not consider obstacles; recall our initial assumption that F is continuous and Ω is convex.

Theorem 3.7. *A κ-regular network construction is a convergent network construction.*

Proof. Consider a κ-regular network construction $\mathcal{G}$. For any instance $I = (\Omega, F, \bar{L}, a, b)$, we know by Definition 3.5 that there exists a sequence of parameter vectors, $(P_1, P_2, \dots)$, such that $G(n) = (V(n), A(n)) = \mathcal{G}(I, P_n)$ is a $(\delta(n), \epsilon(n), \kappa)$-approximation network and $\lim_{n\to\infty} \delta(n) = 0$ with $\lim_{n\to\infty} \frac{\epsilon(n)}{\delta(n)} = 0$.

We see by rearranging (3.9) that

$$J^*_{G(n)}(\bar{L}(1+\gamma(n))) \leq (J^*(\bar{L}) + \bar{L}\Delta_{G(n)})(1+\gamma(n)), \tag{3.10}$$

where $\gamma(n) = 2\frac{\epsilon(n)}{\delta(n)} + 2\frac{\epsilon(n)^2}{\delta(n)^2} \in \Phi_{G(n)}$.

To show that the right-hand side of (3.10) converges to $J^*(\bar{L})$, we need to show that we can refine the network in such a way that $\lim_{n\to\infty} \Delta_{G(n)} = 0$ and $\lim_{n\to\infty} \gamma(n) = 0$.

Letting $R_v(n)$ be the region around the node $v \in V(n) \setminus \{b\}$ guaranteed to exist by Definition 3.1, we see clearly from Definition 3.4,

$$\Delta_{G(n)} = \max_{v\in V(n)\setminus\{b\}} (\max_{x\in R_v(n)} F(x) - \min_{y\in R_v(n)} F(y)),$$

that zero is a lower bound on $\Delta_{G(n)}$. We know from Definition 3.1 that each region $R_v(n)$ is contained in a disk of radius $\kappa\delta(n)$. Thus the maximum distance between points in the set $R_v(n)$ is $2\kappa\delta(n)$. Now as F is Hölder continuous, for any points x and y in Ω we have $|F(x) - F(y)| \leq K||x-y||^\sigma$ for some positive constant K and $0 < \sigma \leq 1$. Hence we have

$$\begin{aligned}\lim_{n\to\infty} \Delta_{G(n)} &= \lim_{n\to\infty} \max_{v\in V(n)\setminus\{b\}} (\max_{x\in R_v(n)} F(x) - \min_{y\in R_v(n)} F(y)) \\ &\leq \lim_{n\to\infty} \max_{v\in V(n)\setminus\{b\}} K||\operatorname*{argmax}_{x\in R_v(n)} F(x) - \operatorname*{argmin}_{y\in R_v(n)} F(y)||^\sigma && \text{by Hölder condition} \\ &\leq \lim_{n\to\infty} K(2\kappa\delta(n))^\sigma \\ &\leq 0 && \text{as } \lim_{n\to\infty}\delta(n) = 0.\end{aligned}$$

Thus $\lim_{n\to\infty} \Delta_{G(n)} = 0$. Now

$$\lim_{n\to\infty}\gamma(n) = \lim_{n\to\infty}\left(c_1\frac{\epsilon(n)}{\delta(n)} + c_2\frac{\epsilon(n)^2}{\delta(n)^2}\right) = 0 \qquad \text{as } \lim_{n\to\infty}\frac{\epsilon(n)}{\delta(n)} = 0.$$

So finally, using $\lim_{n\to\infty} \Delta_{G(n)} = 0$ and $\lim_{n\to\infty}\gamma(n) = 0$, we get

$$\begin{aligned}&\lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n))) \leq \lim_{n\to\infty}(J^*(\bar{L}) + \bar{L}\Delta_{G(n)})(1+\gamma(n)) \\ &\lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n))) \leq J^*(\bar{L}).\end{aligned}$$

To show convergence we need a corresponding lower bound on $\lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n))$. We know $J^*(L)$ is continuous on $L \in [||b-a||, \infty)$ by Theorem 3.6. A continuous map of a convergent sequence is convergent, and thus

$$\begin{aligned}\bar{L} &= \lim_{n\to\infty} \bar{L}(1+\gamma(n)) && \text{as } \lim_{n\to\infty}\gamma(n) = 0 \\ J^*(\bar{L}) &= \lim_{n\to\infty} J^*(\bar{L}(1+\gamma(n))) && \text{as } J^*(L) \text{ is continuous} \\ J^*(\bar{L}) &\leq \lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n))),\end{aligned}$$

where we have used $J^*_{G(n)}(\bar{L}(1+\gamma(n))) \geq J^*(\bar{L}(1+\gamma(n)))$ by the optimality condition of the continuous optimal solution. Thus

$$J^*(\bar{L}) \leq \lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n)) \leq J^*(\bar{L}).$$

So

$$\lim_{n\to\infty} J^*_{G(n)}(\bar{L}(1+\gamma(n)) = J^*(\bar{L}).$$

This shows that if we choose the sequence of parameters $(P_1, P_2, \dots)$ guaranteed to exist by Condition 1 of Definition 3.5, then the optimal objective function values of the WCSPP's for the networks $G(n) = \mathcal{G}(I, P_n)$ will converge to the objective function value of the C-LCMCPP as $n \to \infty$. Thus we have shown a κ-regular network construction is convergent. □

Theorem 3.7 shows us that the WCSPP solutions for a κ-regular network construction using the relaxed weight constraint converge to the solution of the C-LCMCPP. However, we can also show that the solutions to the WCSPP using the same weight constraint as the C-LCMCPP also converge.

THEOREM 3.8. *Given an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP with $\bar{L} > ||a-b||$ and a κ-regular network construction $\mathcal{G}$, the solutions to the WCSPP using the network $\mathcal{G}(I, P_n)$ and the weight limit $\bar{L}$ will converge to the solution of I for some sequence of parameter sets $(P_1, P_2, \dots)$.*

Proof. Choose the parameter set $(P_1, P_2, \dots)$ guaranteed to exist by Definition 3.5 such that $G(n) = (V(n), A(n)) = \mathcal{G}(I, P_n)$ is a $(\delta(n), \epsilon(n), \kappa)$-approximation network with $\lim_{n\to\infty} \delta(n) = 0$ and $\lim_{n\to\infty} \frac{\epsilon(n)}{\delta(n)} = 0$. Let $\gamma(n) = 2\frac{\epsilon(n)}{\delta(n)} + 2\frac{\epsilon(n)^2}{\delta(n)^2} \in \Phi_{G(n)}$.

Choose a sequence L_n and integer N such that $\bar{L} = L_n(1+\gamma(n))$ and $L_n > ||a-b||$ for all $n \in \{N, N+1, \dots\}$. Note that $\lim_{n\to\infty} L_n = \bar{L}$ and $L_n < \bar{L}$ for all $n \in \mathbb{Z}^+$. As $G(n)$ is a $(\delta(n), \epsilon(n), \kappa)$-approximation network to problem instance $(\Omega, F, \bar{L}, a, b)$, then $G(n)$ is also a $(\delta(n), \epsilon(n), \kappa)$-approximation network for the problem instance (Ω, F, L_n, a, b); this is readily seen from Definition 3.1 noting $L_n < \bar{L}$. Hence, applying (3.9) for $n \in \{N, N+1, \dots\}$, we get

$$\begin{aligned} J^*(L_n) &\geq \frac{J^*_{G(n)}(L_n(1+\gamma(n)))}{1+\gamma(n)} - L_n\Delta_{G(n)} \\ &\geq \frac{J^*_{G(n)}(\bar{L})}{1+\gamma(n)} - L_n\Delta_{G(n)} \qquad\qquad \text{as } L_n(1+\gamma(n)) = \bar{L}. \end{aligned}$$

Rearranging, we get

$$J^*_{G(n)}(\bar{L}) \leq (J^*(L_n) + L_n\Delta_{G(n)})(1+\gamma(n)).$$

Taking limits and using reasoning similar to that used in the proof of Theorem 3.7, we obtain

$$\lim_{n\to\infty} J^*_{G(n)}(\bar{L}) \leq (J^*(L_n) + L_n\Delta_{G(n)})(1+\gamma(n)), \tag{3.11}$$

which implies

$$\lim_{n\to\infty} J^*_{G(n)}(\bar{L}) \leq J^*(\bar{L}) \qquad\qquad \text{as } J^*(.) \text{ continuous.} \tag{3.12}$$

Also, the optimization problem that defines $J^*_{G(n)}$ has a domain that is a subset of the domain of the optimization problem that defines J^* so $J^*(\bar{L}) \leq J^*_{G(n)}(\bar{L})$ for all $n \in \mathbb{Z}^+$. Thus $\lim_{n\to\infty} J^*_{G(n)}(\bar{L}) \geq J^*(\bar{L})$. So clearly $\lim_{n\to\infty} J^*_{G(n)}(\bar{L}) = J^*(\bar{L})$. This completes the proof. □

By solving the WCSPP corresponding to a given instance of the C-LCMCPP with the original rather than relaxed weight constraint we obtain feasible solutions to the optimization problem and an upper bound. Theorem 3.8 tells us that the upper bounds will converge to the true continuous optimal solution as we refine our framework.

Note that the convergence proof does not work if the length constraint is equal to $\|a-b\|$. In this case, there is only one possible path, being the straight line from a to b. However, in our successive network approximations to this problem this path may not appear in our network as all paths from a to b in our network may be slightly longer that $\|b-a\|$. The WCSPP would then have no feasible solution for $\bar{L} = \|b-a\|$ and thus $J_G^*(\|a-b\|)$ would be undefined. In this case, the successive approximations could not be said to converge.

4. A specific κ-regular network construction technique. In this section we create a κ-regular network construction $\mathcal{G}$ that satisfies Definition 3.5. We will do this using one network to create a scaffolding with cells of size of order δ and then placing a second network used to solve the WCSPP on this scaffolding. The nodes of the second network are placed on the boundaries of the cells with the nodes spaced at most 2ϵ apart. For reasons that will become clear later, we will call this a *cellular* network construction.

Our parameter space will be the set $S = \{(i,j,M) \in \mathbb{Z}^{\oplus} \times \mathbb{Z}^{\oplus} \times \mathbb{Z}^{+} : (i,j) \neq (0,0)\}$ where $\mathbb{Z}^{\oplus}$ is the set of nonnegative integers. We will first show that each network constructed is a (δ,ϵ,κ)-approximation network where $\delta = \frac{\sqrt{3}}{2}\frac{\|b-a\|}{\sqrt{i^2+j^2+ij}}$, $\epsilon = \frac{1}{\sqrt{3}M}\delta$ and $\kappa = \frac{4}{\sqrt{3}}$.

For an instance $I = (\Omega, F, \bar{L}, a, b)$ of the C-LCMCPP, we construct our network by first creating a tessellation of equilateral triangles covering all of $\mathbb{R}^2$ such that a and b (the start and end points) are located at triangle corners. The triangle size and orientation is specified by the parameters i and j. Specifically, we find the side length l_{ij} and unit vectors d_1 and d_2 such that d_2 points $\frac{\pi}{3}$ radians anticlockwise to the direction of d_1 and $a + il_{ij}d_1 + jl_{ij}d_2 = b$. Using the cosine rule and referring to Figure 4.1(a) we see that

$$\|b-a\|^2 = i^2 l_{ij}^2 + j^2 l_{ij}^2 - 2ij l_{ij}^2 \cos\left(\frac{2\pi}{3}\right),$$

which we simplify and rearrange to give

$$l_{ij} = \frac{\|b-a\|}{\sqrt{i^2+j^2+ij}}.$$

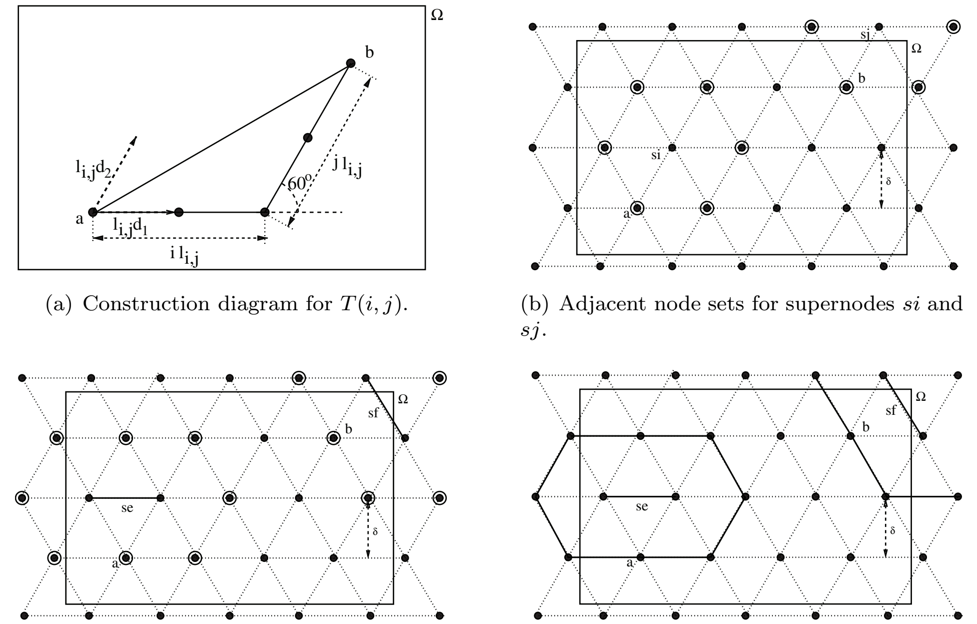
We can then find d_1 and d_2 using the sine rule. We construct our tessellation to align the vectors d_1 and d_2 with the side length of the triangles given by l_{ij}. Figure 4.1(b) shows the resulting tessellation for parameter vector $(i,j) = (2,2)$.

We define

$$\delta = \frac{\sqrt{3}}{2} l_{ij} = \frac{\sqrt{3}}{2}\frac{\|b-a\|}{\sqrt{i^2+j^2+ij}}.$$

This definition makes δ the perpendicular height of the equilateral triangles that form our tessellation.

We will view this tessellation as a network which we call $T(i,j) = (SN, SE)$. To distinguish this network from the one over which the WCSPP is solved, we call the



(a) Construction diagram for $T(i,j)$.

(b) Adjacent node sets for supernodes si and sj.

(c) Boundary supernode sets for superedges se and sf.

(d) Boundary superedge sets for superedges se and sf.

FIG. 4.1. *These diagrams show the construction of and examples of the various definitions for our scaffolding graph. The network was created using $(i,j) = (2,2)$. Knowing (i,j) and the position of a and b, we can easily find the length l_{ij} and the vectors d_1, d_2. These are used to construct our scaffolding graph (SN_Ω, SE_Ω). The supernode set SN_Ω are the black dots shown and the superedge set SE_Ω are dotted lines shown on the diagram.*

elements of SN *supernodes* and the elements of SE *superedges*. We place a supernode at every triangle vertex to form the set $SN = \{a + ml_{ij}d_1 + nl_{ij}d_2 : m, n \in \mathbb{Z}\}$. Then the set of superedges are defined by $SE = \{se = \{si, sj\} : si, sj \in SN, \|si - sj\| = l_{ij}\}$. Note that superedges are undirected.

DEFINITION 4.1. *We say the superedge $se = \{si, sj\}$* contains *a point $x \in \Omega$ if there exists $\lambda \in [0,1] \subset \mathbb{R}$ such that $x = \lambda si + (1-\lambda)sj$.*

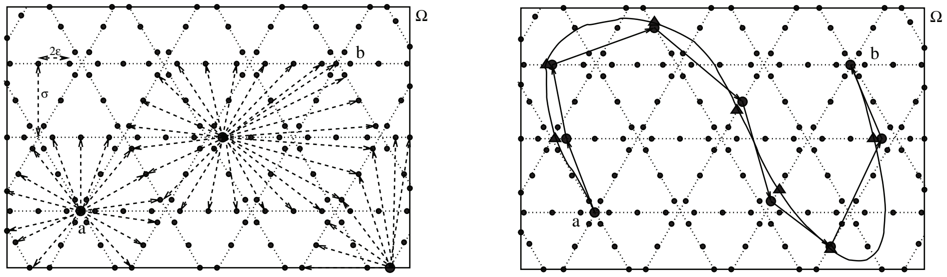
Naturally, we will only be interested in the part of the tessellation that covers Ω. Let the set of triangles in $T(i,j)$ that cover Ω, i.e., all triangles that intersect with Ω, be denoted by $Tri_\Omega = \{\triangle = \{\{si, sj\}, \{si, sk\}, \{sj, sk\}\} \subset SE : \exists se \in \triangle$ and $\exists x \in \Omega$ s.t. se contains $x\}$. We define a new set of supernodes $SN_\Omega = \{sn \in SN : \exists \triangle \in Tri_\Omega$ with $se \in \triangle$ s.t. $sn \in se\}$. We define $SE_\Omega = \{\{si, sj\} \in SE : si, sj \in SN_\Omega\}$. This gives us our scaffolding graph (SN_Ω, SE_Ω). We will also make the following definitions to ease the rest of the discussion.

DEFINITION 4.2. *The adjacent node set to a supernode $si \in SN_\Omega$ is the set $Adj(si) = \{sj \in SN_\Omega : \|sj - si\| = l_{ij}\}$.*

DEFINITION 4.3. *The* boundary supernode set *of a superedge $se = \{si, sj\} \in SE_\Omega$ is the set $BndyNodes(se) = (Adj(si) \cup Adj(sj)) \setminus \{si, sj\}$.*

DEFINITION 4.4. *The* boundary superedge set *of a superedge $se \in SE_\Omega$ is the set $Bndy(se) = \{\{si, sj\} \in SE_\Omega : si, sj \in BndyNodes(se)\}$.*

We will place the nodes of our network on the sections of the superedges inside Ω, and space the nodes such that each point in Ω contained by a superedge is at most ϵ from a node on that same superedge. We first choose an integer $M \geq 1$ and let



(a) Diagram of network showing how nodes (black dots) are placed and the leaving edges (dashed arrows) for selected nodes.

(b) Diagram of network approximation to a continuous path. The triangles are the $p(s_k)$'s and the larger circles are the corresponding v_k's.

FIG. 4.2. *These diagrams illustrate node placement, arc choice and path approximation in κ-regular networks.*

$\epsilon = \frac{1}{\sqrt{3}M}\delta$. This makes ϵ the length of the side of a tessellation triangle divided by $2M$. We then use the following procedure to place nodes on the superedges, producing our node set V:

For each superedge $se = \{si, sj\} \in SE_\Omega$,

1. **If** $si \in \Omega$, place a node at a distance ϵ along the superedge from si and place subsequent nodes at a distance 2ϵ as long as each node is placed inside Ω. If the next node to be placed is outside Ω and if the intersection of the superedge and boundary of Ω is a distance greater than ϵ from the last node, or it is the first node to be placed, we place a node on the intersection of the boundary of Ω and se; otherwise we do not place a node.
2. **Else If** $sj \in \Omega$, follow rule 1 but start at sj instead of si.
3. **Else If** $si, sj \notin \Omega$, and there exists $x \in \Omega$ such that se contains x, we start by placing a node at one of the intersections between the superedge and the boundary of Ω. We then place nodes at intervals of 2ϵ until the next node to be placed would be outside Ω. If the other intersection se and the boundary of Ω is a distance greater than ϵ from the last node, we place a node at the other intersection of the superedge and the boundary.
4. **Else If** there does not exist $x \in \Omega$ such that se contains x, then we do not place nodes on that super edge.

The above procedure defines our node set V. An example of the placement of nodes can be found in Figure 4.2(a). Next, we define the edge connectivity in our network but first we make the following definitions.

DEFINITION 4.5. *For node $i \in V \setminus \{a, b\}$ its* boundary superedge set, *$Bndy(i)$, is the set $Bndy(se)$ where i is contained by se. This is well defined as each node in $V \setminus \{a, b\}$ is contained by one and only one superedge. The boundary superedge set of a is $Bndy(a) = \{se = \{si, sj\} \in SE_\Omega : si, sj \in Adj(a)\}$. $Bndy(b)$ is not defined.*

DEFINITION 4.6. *A point $x \in \Omega$ is* contained *in $Bndy(j)$ for $j \in V \setminus \{b\}$ if there exists $se \in Bndy(j)$ such that x is contained by se.*

To create our edge set A we will place an edge from each node $i \in V \setminus \{b\}$ to every node $v \in V \setminus \{a\}$ contained by $Bndy(i)$. Note that no edges terminate at a and that there is an edge ending at b from any node which contains b in its boundary superedge set. Note, however, that no edges originate at b. Examples of edges are shown in Figure 4.2(a).

At this stage we have a network construction $\mathcal{G}$ that produces a network $G = (V, A)$ for a given instance I of the C-LCMCPP and parameter vector $P \in S$. We now wish to check that each network produced is a $(\delta, \epsilon, \kappa)$-approximation network.

For any path $p \in \Gamma$ from a to b, we will construct its network approximation in $G = (V, A)$ in the following manner and show it satisfies the properties of Definition 3.1.

1. Let $k = 0$, $v_0 = a$, and $s_0 = 0$ (meaning $p(s_0) = a$).
2. Let $k = k + 1$. Let $s_k \in (s_{k-1}, 1]$ be the smallest value such that $p(s_k)$ is contained in $Bndy(v_{k-1})$.
3. Let v_k be the closest node in $V \setminus \{a, b\}$ on the superedge containing $p(s_k)$. If $p(s_k)$ is located on a supernode, it will be contained by many superedges. In this case we choose v_k to be the closest node on any of the superedges in $Bndy(v_{k-1})$ containing $p(s_k)$, breaking ties arbitrarily.
4. If b is contained in $Bndy(v_{k-1})$ and if there is no $s \in [s_k, 1]$ such that $p(s)$ is contained in $Bndy(v_k)$, then let $v_k = v_N = b$ (replacing the last choice of v_k made in step 3) and $s_k = s_N = 1$ and stop. Otherwise go to step 2.

For any path $p \in \Gamma$ we have thus produced two sequences $(v_0, v_1, \ldots, v_N)$ and $(p(s_0), p(s_1), \ldots, p(s_N))$ with $v_k \in V$ for $k \in \{0, \ldots, N\}$ and $0 = s_0 < s_1 < \ldots < s_{N-1} < s_N = 1$. An example of the sequences $(v_0, v_1, \ldots, v_N)$ and $(p(s_0), p(s_1), \ldots, p(s_N))$ for a particular path and network is shown in Figure 4.2(b).

The Euclidean distance from any point on a superedge to any point on the boundary superedge set of that superedge is greater than or equal to δ; see Figure 4.1. We can see that as $p(s_k)$ lies on the boundary superedge set of the superedge which contains both v_{k-1} and $p(s_{k-1})$, we have $\|p(s_k) - p(s_{k-1})\| \geq \delta$ for $k \in \{2, N\}$. Also, all points contained by the boundary superedge set of node a are a distance greater than δ from a so $\|p(s_1) - p(s_0)\| \geq \delta$ as $p(s_1)$ is on the boundary superedge set on node a. Thus Condition 3(a) of Definition 3.1 is satisfied.

The point $p(s_k)$, $k \in \{1, \ldots, N-1\}$ will be approximated by the nearest node on the same superedge. Nodes are placed on superedges such that any point on the super edge is at most ϵ away from a node, and thus the spacing of nodes will satisfy the condition $\|v_k - p(s_k)\| \leq \epsilon$, $\forall k \in \{1, \ldots, N-1\}$. As $a = p(s_0) = v_0$ and $b = p(s_N) = v_N$, we satisfy Condition 3(b) of Definition 3.1.

By having edges run from each node $i \in V \setminus \{b\}$ to all the nodes contained by $Bndy(i)$, we can see that the edges required by a network approximation $(v_0, v_1, v_2, \ldots, v_{N-1}, v_N)$ to any path $p \in \Gamma$, that is the edges (v_0, v_1), (v_1, v_2), $\ldots$, (v_{N-1}, v_N), are in A. This satisfies Condition 3 of Definition 3.1.

To define the regions R_v we make the following definition.

Definition 4.7. *For supernode $si \in SN_\Omega$ the closed region $Reg(si) = \{\lambda si + (1-\lambda)(\mu sj + (1-\mu)sk) : \lambda, \mu \in [0, 1], \{sj, sk\} \in SE_\Omega, \{sj, sk\} \subseteq Adj(si)\}$. For a set of supernodes, we will extend the definition of $Reg(.)$ to be the union of the set of regions for each supernodes, i.e., $Reg(\{si_1, si_2, \ldots, si_n\}) = \bigcup_{k=1}^{n} Reg(si_k)$.*

The region $Reg(si)$ will generally be a regular hexagon around supernode si, except where the network is truncated near the boundary of Ω.

The regions R_v, $v \in V \setminus \{a, b\}$ in Definition 3.1 are satisfied in our construction by the regions $Reg(se) \cap \Omega$ where se is the superedge containing node v with the exception of the case where b is contained by the boundary superedge set of se, in which case R_v is given by $Reg(se \cup sb_1 \cup sb_2) \cap \Omega$ where sb_1 and sb_2 are the superedges in the boundary superedge set of v that contain b. Examples of the regions near node b are given in Figure 4.3. For node a, $R_a = Reg(a) \cap \Omega$ except in the unlikely case that the boundary of a contains b. In this case $R_a = Reg(\{a\} \cup sb_1 \cup sb_2) \cap \Omega$ where

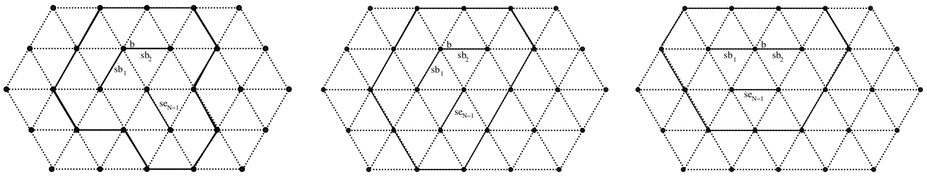


FIG. 4.3. *Diagrams of the shapes of the regions around end node b.*

sb_1 and sb_2 are the superedges in the boundary superedge set of a that contain b. Each of the regions can be enclosed in a circle of radius at most $\frac{4}{\sqrt{3}}\delta$. This means κ for our construction is $\frac{4}{\sqrt{3}}$.

For path p, the section of the path from $p(s_{k-1})$ to $p(s_k)$ and the points on the edge from v_{k-1} to v_k given by $\lambda v_{k-1} + (1-\lambda)v_k$ for $\lambda \in [0,1]$ are entirely contained in the region $R_{v_{k-1}}$ for $k \in \{1,\dots,N\}$. This satisfies the need for the regions R_v for $v \in V \setminus \{b\}$ and Condition 3(c) of Definition 3.1.

Thus for any instance $I = (\Omega, F, L, a, b)$ of the C-LCMCPP, the network $\mathcal{G}(I,P)$ for $P \in S$ produced by our cellular network construction will be a (δ,ϵ,κ)-approximation network.

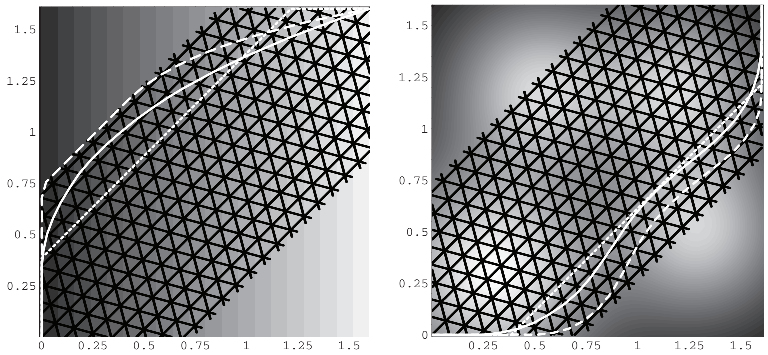
We now wish to show that our cellular network construction is a κ-regular network construction. For any instance $I = (\Omega, F, L, a, b)$ of the C-LCMCPP, consider the network $\mathcal{G}(I,(k,0,k))$. This network will be a $(\delta_k,\epsilon_k,\kappa)$-approximation network for $\delta_k = \frac{\|a-b\|\sqrt{3}}{2k}$, $\epsilon_k = \frac{\|a-b\|}{2k^2}$, and $\kappa = \frac{4}{\sqrt{3}}$.

Noting the requirements of Definition 3.5, we see that for any instance I of the C-LCMCPP there is a sequence of parameter vectors $(P_1, P_2, P_3, \dots) = ((1,0,1), (2,0,2), (3,0,3), \dots)$, a sequence of δ values $(\delta_1,\delta_2,\delta_3,\dots) = \big(\frac{\|a-b\|\sqrt{3}}{2}, \frac{\|a-b\|\sqrt{3}}{4}, \frac{\|a-b\|\sqrt{3}}{6}, \dots\big)$ with $\lim_{k\to\infty}\delta_k = 0$, and a sequence of ϵ values $(\epsilon_1,\epsilon_2,\epsilon_3,\dots) = \big(\frac{\|a-b\|}{2}, \frac{\|a-b\|}{8} \frac{\|a-b\|}{18}, \dots\big)$ with $\lim_{k\to\infty}\frac{\epsilon_k}{\delta_k} = 0$, such that $\mathcal{G}(I,P_k)$ is a $(\delta_k,\epsilon_k,\kappa)$-approximation network. Thus the cellular network construction outlined in this section is a κ-regular network construction with $\kappa = \frac{4}{\sqrt{3}}$.

5. Numerical experiments.

In this section we will test our κ-regular network construction method numerically. We implemented the lower bounds scheme in two different ways. Both schemes use Mathematica to calculate the node positions but differ in the method of calculating edge costs. In the Gaussian scheme, we used Mathematica to explicitly calculate the edge costs in the network using three-point Gaussian quadrature. These edges are then exported to a WCSPP solver written in C++. The trapezoidal scheme instead uses Mathematica to calculate function values at each node which are then exported to the WCSPP solver which calculates edge costs on the fly using the trapezoidal rule. The Gaussian scheme is more accurate, especially for smaller networks, whereas the trapezoidal scheme is faster as it utilizes the fact the edges share start and end positions and thus needs fewer function evaluations. The number of function evaluations is equal to the number of nodes for the trapezoidal scheme, whereas it is a multiple of the number of edges for the Gaussian scheme. The trapezoidal scheme also allows larger networks as the edges are not stored explicitly. Note that while it would be possible to calculate edge costs on the fly using Gaussian quadrature, this was not implemented.

For both schemes, Δ_G was found by numerically finding the maximum and minimum value of $F(x)$ for each region R_v, $v \in V \setminus \{b\}$ using Mathematica. Note that



(a) $F_1(x, y)$

(b) $F_2(x, y)$

Fig. 5.1. *Contour plot of functions $F_1(x,y)$ and $F_2(x,y)$ overlaid with the nodes of cellular network with parameters $(i,j,M) = (24,0,24)$. The lighter regions have higher function values. The shading scale on the two plots is not the same. Some parts of the region which are not length feasible have not been meshed. The edges are not shown to avoid cluttering the diagram. The paths corresponding to the upper bound in the cellular network are the solid white lines. The paths corresponding to the relaxed weight constraint $\bar{L}(1+\gamma)$ are the long dashed lines. Both problems were solved using the Gaussian scheme. We also calculated the upper bound paths given by a grid network of* 721 *by* 721 *nodes which are shown as the short dashed lines. We can see that the cellular network produces a smoother path than the grid network. Note that the paths are all piecewise linear and have not been smoothed in anyway.*

all nodes on the same superedge share the same region R_v; thus only one calculation per superedge is required. We used a γ calculated by the formula

$$\gamma = \frac{2}{\sqrt{3}M} + \frac{2}{3M^2}, \tag{5.1}$$

where M is the number of nodes per side length. Note that we have used the pessimistic choice of $c_1 = c_2 = 2$ in (3.2) to calculate γ. The calculations were performed on a Pentium 4 2.4GHz with 512Mb RAM running under Linux.

We use two test functions. The first is $F_1(x,y) = x$. For the second we define a constituent function:

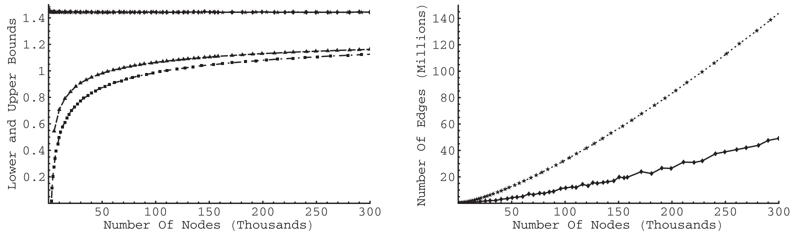
$$G_{\phi_1,\phi_2,\sigma}(x) = \frac{1}{\pi\sigma^2} e^{-\frac{(x_1-\phi_1)^2+(x_2-\phi_2)^2}{\sigma^2}},$$

and use the following as our test function,

$$F_2(x) = G_{.3,.3,.5}(x) + 0.5(G_{1.3,.4,.4}(x) + G_{.5,1.2,.4}(x) + G_{1.2,1.2,.4}(x)).$$

For both F_1 and F_2 our region Ω is the closed square with corners at $(0,0)$ and $(1.6,1.6)$. The start point a is $(0,0)$ and the end point b is $(1.6,1.6)$. The weight limit $\bar{L}$ is 1.1 times the distance between the start and end nodes or approximately 2.489 units. Both functions are plotted in Figure 5.1.

Figure 5.1 shows an example of a network and path that results from our cellular network construction for both $F_1(x,y)$ and $F_2(x,y)$. The solid white lines are the best upper bound paths, found by solving the WCSPP calculation using $\bar{L}$ as the length constraint. This is our approximate solution to the C-LCMCPP for this network. To obtain a lower bound, we use the relaxed weight constraint $\bar{L}(1+\gamma)$ and solve the WCSPP to get a lower bound path, which are the long dashed lines. The objective



(a) Upper and lower bound.

(b) Number of nodes vs number of edges.

FIG. 5.2. *The upper and lower bounds for the C-LCMCPP for the function $F_1(x, y)$ vs the number of nodes. The squares are the lower bounds for $i = M$ and the triangles are the results when i and M are optimized for an approximately constant number of nodes. The stars are the upper bounds for the $i = M$ case and the diamonds are the upper bounds for the optimized i and M case. We can see that the optimization offers an improvement on the lower bound. We can also see a diminishing return on the improvement to the lower bound as we use more nodes. The number of edges vs the number of nodes is also given in* (b) *in which the stars indicate the $i = M$ case and the diamonds indicate the i and M optimized case.*

function values corresponding to these paths are $J^*_G(\bar{L}(1+\gamma))$ which are used in the lower bounds formula given by (3.9).

For comparison we have shown the paths that result from using a grid network with edges to the 8 nearest neighbors, shown as the short dashed line. We can see that the paths are not smooth compared to the ones obtained via the cellular construction. The number of nodes in the grid network was chosen to approximately equal the average node density of the cellular network. The objective function values of the grid network are also higher than that of the cellular network: 1.58602 vs 1.44257 or 9.9% higher for $F_1(x, y)$ 1.783 vs 1.734 or 2.8% higher for $F_2(x, y)$.

Figures 5.2(a) and 5.2(b) give the results of C-LCMCPP using successively larger networks to improve the lower bound. The trials were aborted at approximately 300,000 nodes and 150,000,000 edges when the WCSPP's became too big to solve effectively due to computational memory limitations.

Two methods of choosing i and M were tested. In the first, we set $i = M$ and in the second we chose i and M so that they provided the best lower bound for an approximately constant number of nodes. To do this we note that for our cellular network

$$|V| \approx K_v i^2 M \text{ and}$$

$$i \approx \sqrt{\frac{|V|}{K_v M}}. \tag{5.2}$$

We approximate Δ_G by

$$\Delta_G \approx \frac{K_\Delta}{i}. \tag{5.3}$$

Substituting (5.2), (5.3), and the formula for γ, (5.1), into the lower bounds formula, (3.9), gives

$$LB \approx \frac{J^*_G(\bar{L}(1+\gamma))}{1 + \frac{2}{\sqrt{3}M} + \frac{2}{3M^2}} - \bar{L}K_\Delta\sqrt{\frac{K_v M}{|V|}}. \tag{5.4}$$

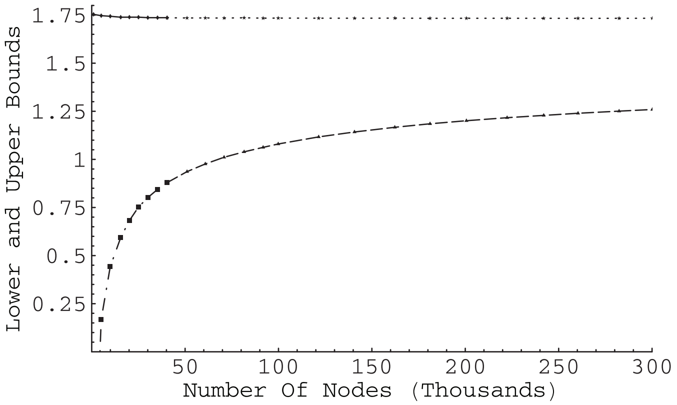


FIG. 5.3. *The upper and optimum lower bounds for the C-LCMCPP for the function $F_2(x,y)$ vs the number of nodes. The squares are the lower bounds calculated using the Gaussian scheme and the triangles are the lower bounds calculated using the trapezoidal scheme. The diamonds and the stars are the upper bounds calculated using Gaussian and trapezoidal scheme, respectively.*

Given previous lower bounds calculations we can estimate the values of $J_G^*(\bar{L}(1+\gamma))$, K_Δ and K_v, and then given a number of nodes $|V|$ we can calculate an approximately optimal value of M and use (5.2) to find i. We then vary i around this approximate optimal value to find a local optimum and report this value as the optimal i and M combination for a particular number of node in Figure 5.2.

The lower bound calculated by (3.9) steadily improves from being negative to becoming positive at 2,717 nodes and 227,909 edges when $i = M = 13$ and increases to within 21.5% of the upper bound at 318,946 nodes and 156,815,926 nodes when $i = M = 64$. Optimizing the choice of i and M results in a slight increase of the best lower bound to within 19.5% of the least upper bound using a network with $(i, M) = (108, 21)$ having 299,910 nodes and 49,165,143 edges.

Even though the smaller networks produce useless negative lower bounds, they produce competitive upper bounds. The upper bound for $i = M = 7$ calculated using a network of 404 nodes and 14,383 edges was 1.452 which was within 1% of the best upper bound of 1.442 produced for $i = M = 64$ with 318,946 nodes and 156,815,926 edges. We can see that the upper bound converges at a much faster rate than the lower bound. In light of the results of Zabarankin et al. [21], who compare grid network solutions to analytic solutions available in specific cases, we believe the upper bounds we compute to be very close to the corresponding global optima.

In Figure 5.3 we find the upper and lower bounds for a given number of nodes using the optimal choice of i and M for the function $F_2(x,y)$. The accuracy of the numerical integration is important in the smaller networks, which have longer edges; thus the Gaussian scheme was used for small numbers of nodes. When the size of the networks became prohibitively large for the Gaussian scheme we switched to the trapezoidal scheme. The accuracy of the trapezoidal scheme improves noticeably when the length of the edges is decreased; for example, for $(i, j, M) = (74, 0, 6)$, which produces a network with 40,358 nodes and 1,869,186 edges, the Gaussian scheme produces an upper bound of 1.73661 and the trapezoidal scheme produces an upper bound of 1.73614, a difference of less than 0.03%. The best lower bound found in this case was within 27.4% of the best upper bound. The trapezoidal scheme was also much faster with the above instance running in 1h51m using the trapezoidal scheme as opposed to 7h27m using the Guassian scheme.

Table 5.1

Percentage improvement in the upper bound when changing from a 8-nearest neighbor grid network to a cellular with similar number of edges. The results are averaged over 15 different functions. The standard deviation of the percentage improvement in the upper bound is given as std dev and N^o indicates the number of instances in which the cellular network produced a better result than the grid network out of the 15 instances.

	(i,j,M)	(6,0,3)	(12,0,6)	(24,0,12)	(30,15)
	grid width	21	88	364	572
	$\|A\|$ cell	3279	60508	1053426	2611897
	$\|A\|$ grid	3280	60900	1055604	2610612
$L = 2.489$	% mean UB gain	9.3	12.8	11.5	11.5
	std dev.	17.8	16.2	16.4	16.4
	N^o	9	15	15	15
$L = 2.715$	% mean UB gain	0.0	7.4	9.0	10.1
	std dev.	13.4	8.0	8.4	11.3
	N^o	8	15	15	15
$L = 2.942$	% mean UB gain	-6.1	0.5	2.2	1.9
	std dev.	8.3	1.9	1.7	1.9
	N^o	4	11	13	12

Surprisingly, the majority of the computational time was spent on evaluating $F(x,y)$ and/or evaluating line integrals, exporting data to the WCSPP solver, and calculating Δ_G rather than solving the resulting NP-hard WCSPP. As we were focused on pushing the lower bound as high as possible, we tested some problems with extremely long run times. For example, the time for a complete run, that is, calculating the node positions, calculating Δ_G, calculating the edges weights, exporting the edge data to the WCSPP solver, and solving two WCSPPs (for the upper and lower bound) for $F_2(x,y)$ for $(i,j,M) = (143,0,12)$ using 300,170 nodes and 28,302,752 edges was close to 10 hours giving an upper bound of 1.734 and a lower bound of 1.259. However, we were able to obtain reasonable upper bounds in much shorter times; for example, it took only 6min 7sec to do the same calculation with $(i,j,M) = (14,0,4)$ to get an upper bound of 1.754, though the network was not large enough to provide a positive lower bound.

Though calculating lower bounds can involve extreme computational effort, if we are looking only for feasible solutions, then we can use much smaller networks and get reasonable results. Given that the upper bound for cellular networks seems to converge quite rapidly, we compared the values of the upper bound, thus the best feasible solution found for the problem, to the upper bounds produced by 8 nearest neighbor grid network with a similar number of edges. We again set Ω to the closed square with corners at $(0,0)$ and $(1.6,1.6)$ and the start point to $a = (0,0)$ and the end point to $b = (1.6,1.6)$. We used three different weight limits for each function: 1.1, 1.2, and 1.3 times the distance from the start to end point, respectively. Besides using the functions F_1 and F_2, all the other functions we used were a sum of 15 Gaussians of the form $G_{\phi_1,\phi_2,\sigma}(x)$ with uniformly random centers, $(\phi_1,\phi_2) \in \Omega$, and σ uniformly random in the range $[.1,.3]$.

Table 5.1 shows us that, as we would expect, in the majority of cases, the upper bound produced by the cellular network is lower than that produced by a grid network with a similar number of edges. The clearest trend is the mean percentage improvement of the cellular network over the grid network improves as the weight constraint is made tighter. The cellular networks also tend to do better than the corresponding grid network when the networks are made larger.

We can also see that the variability of the improvement increases with the tighter weight constraint. This may be due to the weight constraint forcing the choice of high cost arcs in the grid network as paths in the grid network are longer than they need be due to the restrictions in the number of directions available.

6. Conclusion. We have produced a network approximation method to the continuous length constrained minimum cost path problem (C-LCMCPP) for which we can show the network approximation converges to the continuous solution as the network is enlarged in an appropriate manner. We then defined $(\delta, \epsilon, \kappa)$-approximation networks and showed that for such a network, a lower bound can be calculated. We went on to define a κ-regular network construction, which can produce a sequence of $(\delta, \epsilon, \kappa)$-approximation networks such that the lower bound (and upper bound) converges to the continuous optimum.

Having developed the theory, we then created a specific example of a κ-regular network construction, which we dubbed a cellular network construction, and tested it computationally. We were able to calculate lower bounds that came within 19.5% of the best upper bound. We also found that the cellular networks produced upper bounds that converged rapidly and that corresponded to smoother paths with lower objective function values than the solutions produced by grid networks.

In the future, we wish to improve the lower bounds further. One possible method is a nonuniform triangulation of space so that node placement better reflects the contours of the underlying function. We also wish to explore the potential of iteratively eliminating regions of space using lower bounds; this would allow better lower bounds to be obtained for the same computational effort.

We may also look at restricting the underlying function to be a triangulated surface to simplify function evaluations, given that this is the way many surfaces in practical situations are represented. The function triangulation could be made to coincide with the triangulation of the cellular network. Implementing this method of function evaluation in C++ would offer a significant speed up to the algorithm over using analytical functions in Mathematica.

The cellular network concept may also be applied to higher dimensional spaces. One could imagine the cells becoming interlocking polytopes with nodes placed on the facets of these polytopes. While the theory underlying such a construction may be relatively straightforward, the number of nodes required by the discretization would grow enormously. However, given that the upper bounds for two dimensional networks converge rapidly, it may be possible that useful upper bounds for higher dimensional problems are attainable.

Lastly, we may wish to change the length constraint to a constraint with the same form of the objective function. This would allow more versatility in the application of the theory to practical situations. The main challenge here would be proving that suitably relaxing the constraint guarantees that a network approximation to an optimal path exists.

Acknowledgment. We would like to thank Tim Robinson for setting us on this path.

REFERENCES

[1] C. Barnhart, N. L. Boland, L. W. Clarke, E. L. Johnson, G. L. Nemhauser, and R. G. Shenoi, *Flight string models for aircraft fleeting and routing*, Transport. Sci., 32 (1998), pp. 208–220.

[2] L. Caccetta, I. Loosen, and V. Rehbock, *Modelling transit paths for military vehicles*, in Proceedings of the 16th Congress of the Modelling and Simulation Society of Australia and New Zealand, 2005, pp. 1751–1757.

[3] W. M. Carlyle and R. K. Wood, *Lagrangian relaxation and enumeration for solving constrained shortest-path problems*, in 38th Annual ORSNZ Conference, University of Waikato, Hamilton, New Zealand, November 2003.

[4] E. P. Chew, C. J. Goh, and T. F. Fwa, *Simultaneous optimization of horizontal and vertical alignments for highways*, Transport. Res. B-Meth., 23B (1989), pp. 315–329.

[5] E. Cristiani and M. Falcone, *Fast semi-Lagrangian schemes for the Eikonal equation and applications*, SIAM J. Numer. Anal., 45 (2007), pp. 1979–2011.

[6] E. W. Dijkstra, *A note on two problems in connexion with graphs*, Numer. Math., 1 (1959), pp. 269–271.

[7] I. Dumitrescu and N. L. Boland, *Improved preprocessing, labelling and scaling algorithms for the weight-constrained shortest path problem*, Networks, 43 (2003), pp. 135–153.

[8] A. Fahlen, *Missile routing for a stand-off missile*, Master's Thesis, Optimeringslara, Matematiska institutionen, November 2000.

[9] C. J. Goh, E. P. Chew, and T. F. Fwa, *Discrete and continuous models for computation of optimal vertical highway alignment*, Transport Res. B-Meth., 22B (1988), pp. 399–409.

[10] L. Guo and I. Matta, *Search space reduction in QoS routing*, Comput. Network, 41 (2003), pp. 73–88.

[11] P. E. Hart, N. J. Nilsson, and B. Raphael, *A formal basis for the heuristic determination of minimum cost paths*, IEEE Trans. Syst. Sci. Cyb. SSC4, 2 (1968), pp. 100–107.

[12] J. Kim and J. P. Hespanha, *Discrete approximations to continuous shortest-path problems: Application to minimum-risk path planning for groups of UAVs*, in the 42nd Conference on Decision & Control, Hawaii, December 2003.

[13] R. Kimmel and N. Kiryati, *Finding shortest paths on surfaces by fast global approximation and precise local refinement*, Int. J. Pattern Recogn., 10 (1996), pp. 643–656.

[14] R. Kimmel and J. A. Sethian, *Optimal algorithm for shape from shading and path planning*, J. Math. Imaging Vision, 14 (2001), pp. 237–244.

[15] R. Kimmel and G. Shapiro, *Shortening three-dimensional curves via two-dimensional flows*, Comput. Math. Appl., 29 (1995), pp. 49–62.

[16] I. M. Mitchell and S. Sastry, *Continuous path planning with multiple constraints*, in the 42nd Conference on Decision and Control, Hawaii, December 2003.

[17] R. D. Muhandiramge and N. L. Boland, *Simultaneous solution of related Lagrangean dual problems with iterated preprocessing for solving the weight constrained shortest path problem*, Networks 2008, to appear.

[18] C. D. Piatko, C. P. Diehl, P. McNamee, C. Resch, and I. Wang, *Stochastic search and graph techniques for MCM path planning*, in Detection and Remediation Technologies for Mines and Minelike Targets VII, J. T. Broach, R. S. Harmon, and G. J. Dobeck, eds., SPIE, 2002.

[19] C. D. Piatko, C. Priebe, L. Cowen, I. Wang, and P. McNamee, *Path planning and mine countermeasures command and control*, in Detection and Remediation Technologies for Mines and Minelike Targets VI, SPIE, 2001.

[20] J. N. Tsitsiklis, *Efficient algorithms for globally optimal trajectories*, IEEE Trans. Automat. Contr., 40 (1995), pp. 1528–1538.

[21] M. Zabarankin, S. Uryasev, and R. Murphey, *Aircraft routing under the risk of detection*, Naval Res. Logist., 53 (2006), pp. 728–747.

SIAM J. OPTIM.
Vol. 20, No. 1, pp. 78–103

A NEW REGULARIZATION SCHEME FOR MATHEMATICAL PROGRAMS WITH COMPLEMENTARITY CONSTRAINTS*

ABDESLAM KADRANI†, JEAN-PIERRE DUSSAULT†, AND ABDELHAMID BENCHAKROUN†

Abstract. We propose a new regularization scheme for mathematical programs with complementarity constraints (MPCC) by relaxing all the constraints of the complementarity system. We show that, under the MPCC-linear independence constraint qualifications (MPCC-LICQ), the Lagrange multipliers exist for this regularization. Our method has strong convergence properties under MPCC-linear independence constraint qualifications and some weak conditions of the strict complementarity. In particular, under MPCC-LICQ, it is shown that any accumulation point of the regularized stationary points is M-stationary for the MPCC problem, and if the asymptotically weak nondegeneracy condition holds at a stationary point of the regularized problem, then it is strongly stationary. An algorithm for solving the proposed regularization is presented and numerical experiments are reported. Some comparisons with other methods are discussed with illustrative examples.

Key words. complementarity constraints, MPCC, regularization, nonlinear programming

AMS subject classifications. 65K10, 49M37, 90C30

DOI. 10.1137/070705490

1. Introduction.

We consider the following mathematical program with complementarity constraints (MPCC):

$$
\begin{array}{ll}
\min & f(x) \\
\text{s.t.} & \\
& g(x) \le 0, \qquad h(x) = 0, \\
& G_l(x) \ge 0, \quad H_l(x) \ge 0, \\
& G_l(x) H_l(x) \le 0, \\
& l = 1, 2, \ldots, n_c,
\end{array}
\tag{1.1}
$$

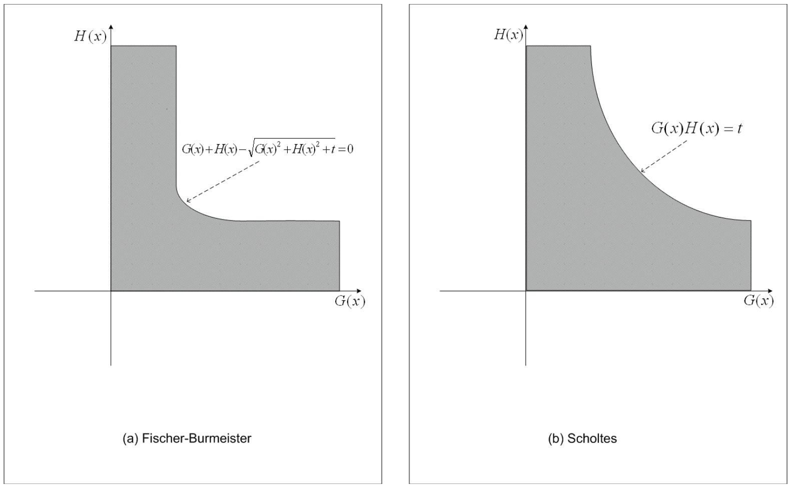
where $f : \boldsymbol{R}^n \to \boldsymbol{R}$, $g : \boldsymbol{R}^n \to \boldsymbol{R}^{n_i}$, $h : \boldsymbol{R}^n \to \boldsymbol{R}^{n_e}$, and $G, H : \boldsymbol{R}^n \to \boldsymbol{R}^{n_c}$ are continuously differentiable. This problem plays an important role in many fields such as engineering design, economic equilibrium, and multilevel games, and has attracted much attention in the recent literature. The major difficulty in solving (1.1) is that its constraints fail to satisfy the Mangasarian Fromovitz constraint qualifications (MFCQ) at any feasible point [5]. This implies an empty or an unbounded KKT multipliers set, so that the standard methods may encounter difficulties, or even fail for solving this problem.

Over the past years, many papers that dealt with MPCC appeared in the literature. Some of them have gone into the search for the existence and the stationarity characterizations of its solution, while others proposed a number of approaches to solve MPCC, for example, the sequential quadratic programming method [1, 2, 10, 11, 17, 23], the branch and bound approach [3], the smoothing implicit

*Received by the editors October 16, 2007; accepted for publication (in revised form) November 5, 2008; published electronically March 13, 2009. This work was partially supported by NSERC grant OGP0005491 and OGP0036512.

http://www.siam.org/journals/siopt/20-1/70549.html

†Département d'Informatique, Université de Sherbrooke, Sherbrooke, Québec J1K 2R1, Canada (Abdeslam.Kadrani@Usherbrooke.ca, Jean-Pierre.Dussault@Usherbrooke.ca, A.Benchakroun@Usherbrooke.ca).



(a) Fischer-Burmeister

(b) Scholtes

Fig. 1.1. *The smooth regularizations* [12, 28].

programming approaches [6, 23], and interior point methods [4, 8, 20, 22, 23]. In parallel, considerable efforts have investigated the search of the regular approximations of MPCC to apply usual nonlinear programming algorithms. Fukushima and Pang [12] and Facchinei et al. [9] suggested a smoothing family, by replacing the complementarity system with the perturbed Fischer–Burmeister function [18]; see Figure 1.1(a):

$$\phi_l(G_l(x), H_l(x), t) = G_l(x) + H_l(x) - \sqrt{G_l^2(x) + H_l^2(x) + t} = 0.$$

Hu and Ralph [15] presented a penalty method where the complementarity term $G_l(x)H_l(x) \leq 0$ is moved to the objective in the form of an l_1-penalty function:

$$\min f(x) + \rho \sum_{l=1}^{n_c} G_l(x)H_l(x).$$

Scholtes [28], see Figure 1.1(b), presented a regularization scheme where the complementarity term $G_l(x)H_l(x) \leq 0$ is changed to

$$G_l(x)H_l(x) \leq t, \quad l = 1, \dots, n_c,$$

and the relaxation parameter t is driven to zero.

Subsequently, Demiguel et al. [8], see Figure 1.2(a), proposed the following regularization:

$$G_l(x) \geq -t_1, \quad H_l(x) \geq -t_1, \quad G_l(x)H_l(x) \leq t, \quad l = 1, \dots, n_c.$$

Lin and Fukushima [21], see Figure 1.2(b), considered the following regularization scheme,

$$G_l(x)H_l(x) \leq t^2, \quad (G_l(x) + t)(H_l(x) + t) \geq t^2, \quad l = 1, 2, \dots, n_c.$$

All these methods use regularizations that transform the thin and nonsmooth, nonconvex feasible region into a thick and smooth one, and because of the smoothing

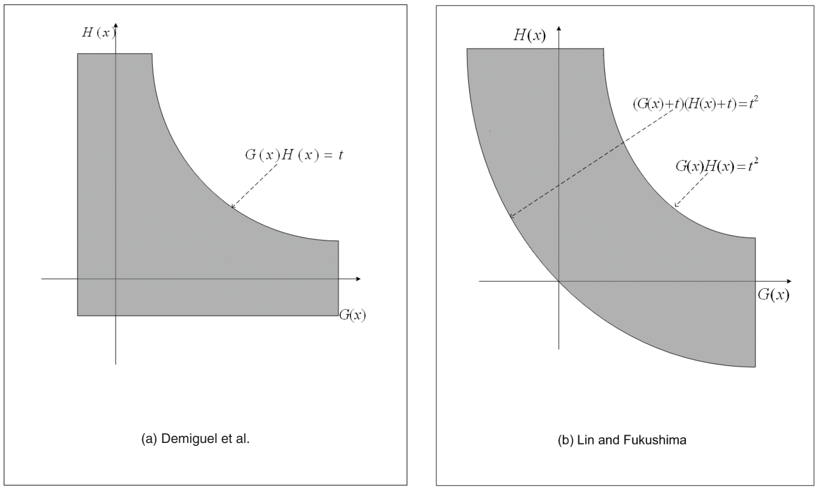


FIG. 1.2. *The smooth regularizations* [8, 21].

TABLE 1.1
The stationary points of the smooth regularizations for the problem (1.2).

Ref.	case 1	case 2
[9, 12]	$(x,\ y) = \left(1,\ \frac{t}{2}\right),\ \eta = (1 - \frac{t}{2})$ $(x,\ y) = \left(\frac{t}{2},\ 1\right),\ \eta = 1 - \frac{t}{2}$	$(x,\ y,\ \eta) = \sqrt{t}\left(\frac{1}{2},\ 1,\ \frac{2}{\sqrt{t}} - \frac{3}{2}\right)$ $(x,\ y,\ \eta) = \sqrt{t}\left(1,\ \frac{1}{2},\ \frac{2}{\sqrt{t}} - \frac{3}{2}\right)$
[8, 28]	$(x,\ y) = \frac{1}{2}(1,\ 1) \pm \left(\sqrt{\frac{1}{4} - t}\right)(1,\ -1)$ for $t \leq \frac{1}{4}$; $(\mu_1,\ \mu_2,\ \eta) = (0,\ 0,\ 1)$	$(x,\ y) = \sqrt{t}\,(1,\ 1)$ $(\mu_1,\ \mu_2,\ \eta) = \left(0,\ 0,\ \frac{1-\sqrt{t}}{\sqrt{t}}\right)$

[21]	$(x,\ y) = \frac{1}{2}(1,\ 1) \pm \left(\sqrt{\frac{1}{4} - t^2}\right)(1,\ -1)$
	for $t \leq \frac{1}{2}$, $(\eta_1, \eta_2\ ) = (1,\ 0)$
	$(x,\ y) = \left(1 - t - y,\ \frac{(1-t)\pm\sqrt{(1-t)(1+3t)}}{2}\right)$
	for $t \leq 1$, $(\eta_1, \eta_2\ ) = (0,\ -1)$
		$(x,\ y) = (t,\ t)$
		$(\eta_1, \eta_2\ ) = \left(\frac{1-t}{t},\ 0\right)$
		$(x,\ y) = (-2t,\ -2t)$
		$(\eta_1, \eta_2\ ) = \left(0,\ \frac{1+2t}{t}\right)$

aspect, the sequence of stationary point generated by these methods may converge to a stationary point where the associated Lagrange multipliers are of the wrong signs (C-stationary), as illustrated by the following example.

Example 1. Consider the following problem:

$$\begin{array}{ll} \min & \frac{1}{2}(x-1)^2 + \frac{1}{2}(y-1)^2 \\ \text{s.t.} & x \geq 0, \quad y \geq 0, \quad xy \leq 0. \end{array} \tag{1.2}$$

Let $\mu_1, \mu_2, \eta, \eta_1$, and η_2 be the Lagrange multipliers corresponding to the constraints of the different regularization schemes described above. Table 1.1 summarizes the stationary points of each regularized problem:

It is clear that all stationary points in case 1 converge to $(1,\ 0)$ and $(0,\ 1)$ which are the good stationary points of (1.2) with good signs for the Lagrange multipliers (S-stationary), but the solutions in case 2 converge to a point $(x,\ y) = (0,\ 0)$ which is only C-stationary for the problem (1.2), that is, a stationary point where its Lagrange multipliers are of wrong signs.

Another difficulty in dealing with the smooth regularization is that the regularized problem may fail to have a stationary point even if the MPCC problem has one, as the following example illustrates:

Example 2.

$$\begin{array}{ll} \min & -y \\ \text{s.t.} & x \geq 0, \quad y \geq 0, \quad xy \leq 0. \end{array} \tag{1.3}$$

Every point on the positive x-axis is a local minimizer of (1.3), but the smooth regularizations have no stationary point. It is easy to verify that the second-order necessary optimality condition does not hold at solutions in case 2 for the smooth regularized problems in Example 1 and at any point of the x-axis for the original MPCC problem (1.3). Consequently, the previous smooth regularizations need some second-order condition to ensure that stationary points exist for the regularized problem and also that such points are not spurious C-stationary points of the MPCC. This necessity to resort to second-order conditions in order to obtain first-order stationary results is undesirable.

In this paper, we present a new regularization scheme NLP(t) where the complementarity system is relaxed to inequalities with a relaxation parameter t, see Figure 1.3:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & \\ & g(x) \leq 0, \qquad h(x) = 0, \\ & G_l(x) \geq -t, \quad H_l(x) \geq -t, \\ & (G_l(x) - t)(H_l(x) - t) \leq 0, \\ & l = 1, 2, \ldots, n_c. \end{array} \tag{1.4}$$

With the proposed regularization (1.4) we do not have to assume any second-order condition to ensure the existence of the regularized stationary points or that such points are not attracted by unwanted C-stationary points of the MPCC. We will show that, under the MPCC-linear independence constraint qualifications (MPCC-LICQ), the standard linear independence constraint qualifications (LICQ) hold for each feasible point of the problem (1.4) except the points x which satisfies $G_{l_0}(x) = H_{l_0}(x) = t$ for some l_0. However, we show that the Lagrange multipliers exist at these points. Global convergence results will be derived under fairly general conditions. It is shown that a cluster point of the stationary points of (NLP(t)) is M-stationary under the MPCC-linear independence constraint qualification (MPCC-LICQ). The convergence to the strong stationary point is deduced if we suppose some nondegeneracy conditions. We further provide some conditions which guarantee that a local minimizer of the MPCC is a limit point of the local minimizers of (NLP(t^k)) as t^k decreases to zero.

The existence of the Lagrange multipliers is the main motivation to consider an active set method for solving this regularization scheme. An interesting idea, for reducing the relaxation parameter t, similar to the so-called elastic mode, is to use an explicit penalization of the parameter t. For a given penalty parameter ρ, this penalized problem can be written as follows:

$$\begin{array}{ll} \min & f(x) + \rho t \\ \text{s.t.} & \\ & g(x) \leq 0, \qquad h(x) = 0, \\ & G_l(x) \geq -t, \quad H_l(x) \geq -t, \\ & (G_l(x) - t)(H_l(x) - t) \leq 0, \\ & l = 1, 2, \ldots, n_c. \end{array} \tag{1.5}$$

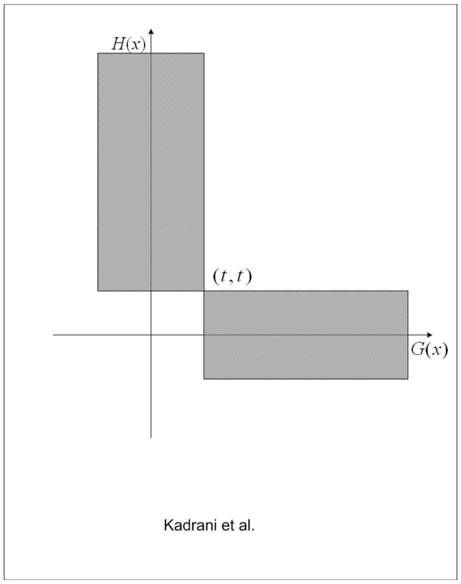


FIG. 1.3. *Our nonsmooth regularization* (1.4).

A similar analysis for the regularization scheme (1.4) may also be extended to this variant. The main difference with the elastic mode as proposed in [2] is that our approach relaxes the complementarity constraints and keeps them explicit as constraints, but the elastic mode [2] removes the complementarity term from the formulation (1.1) by adding an l_1-penalty $c\ (G(x))^T H(x)$ to the objective function, where c is a positive parameter. Moreover, the sequence of first-order points (x^k, t^k) of the formulation (1.4) has an accumulation point that is M-stationary under MPCC-LICQ and strongly stationary if the penalty parameter ρ^k is bounded, while the elastic mode in [2] has similar convergence results but with a sequence of second-order points.

The paper is developed as follows. In section 2, we review various stationarity concepts of the MPCC (1.1) and we show that, under the linear independence constraint qualifications of MPCC (MPCC-LICQ), the standard linear independence constraint qualifications hold for this regularization except some particular points. In section 3, the existence of the Lagrange multipliers is shown for every feasible point of the regularized problem. In section 4, we define a weak stationarity of MPCC used in our convergence analysis. It is shown that, under some conditions, the KKT points of the problem NLP(t) converge to the strongly stationary point of the MPCC as the relaxation parameter decreases to zero. In section 5, we investigate the properties of the attractors. In section 6, the extension of the proposed regularization is discussed. In section 7, we present a practical algorithm for MPCC problem (1.1) and the convergence results of the penalized problem. The active set method applied to original MPCC (1.1) and the proposed regularization is discussed. We conclude the paper in section 8.

We will use the following notations. A letter with superscript or subscript k is related to the kth iteration. We denote by e_q the vector of length q whose entries are all 1, that is, $e_q = (1, 1, \ldots, 1)^T$. The notation $O(\cdot)$ is used in the usual sense. We often have to deal with different index sets for the active constraints of the MPCC or regularization problems. Here is a list of them:

$$\begin{aligned}
&\mathcal{I}_F = \mathcal{I}_F(x) = \{i:\ F_i(x)=0\}\,, && \mathcal{I}_F^{\pm} = \mathcal{I}_F^{\pm}(x,t) = \{i:\ F_i(x)\pm t=0\}\,,\\
&\mathcal{I} = \mathcal{I}_G \cap \mathcal{I}_H,\ \mathcal{I}_{GH} = \mathcal{I}_G \cup \mathcal{I}_H, && \mathcal{I}^{-} = \mathcal{I}_G^{-} \cap \mathcal{I}_H^{-},\ \mathcal{I}_{GH}^{\pm} = \mathcal{I}_G^{\pm} \cup \mathcal{I}_H^{\pm},\\
&\mathcal{I}^* = \mathcal{I}_{G^*} \cap \mathcal{I}_{H^*},\ \mathcal{I}_{G^*H^*} = \mathcal{I}_{G^*} \cup \mathcal{I}_{H^*}, && \mathcal{I}_k^{-} = \mathcal{I}_{G^k}^{-} \cap \mathcal{I}_{H^k}^{-},\ \mathcal{I}_{G^kH^k}^{\pm} = \mathcal{I}_{G^k}^{\pm} \cup \mathcal{I}_{H^k}^{\pm},
\end{aligned}$$

with $F^k = F(x^k), F^* = F(x^*)$, and $F \in \{g, G, H\}$.

2. Stationarity concepts, constraints qualifications, and strict complementarity.

In this section, we recall some notions from the MPCC theory and its regularization which we will use in the sequel. We give the different types of stationarity and the constraint qualifications for MPCC (1.1) in subsection 2.1 and the first-order stationarity conditions for the NLP(t) in subsection 2.1.

2.1. Stationarity, linear independence constraint qualification, and strict complementarity for MPCC.

A feasible point x^* of (1.1) is called *critical* or *weakly stationary* [26], if there exist MPCC multipliers ν^*, π^*, μ_1^*, μ_2^* satisfying:

$$\begin{aligned}
\nabla_x \mathcal{L}(x^*, \nu^*, \pi^*, \mu_1^*, \mu_2^*) &= 0,\\
0 \le \nu^* \perp g(x^*) &\le 0,\\
\mu_1^* \perp G(x^*) &\ge 0,\\
\mu_2^* \perp H(x^*) &\ge 0,
\end{aligned} \tag{2.1}$$

where $\mathcal{L}(x, \nu, \pi, \mu_1, \mu_2)$ is the MPCC Lagrangian of (1.1) at $(x^*, \nu^*, \pi^*, \mu_1^*, \mu_2^*)$:

$$\mathcal{L}(x, \nu, \pi, \mu_1, \mu_2) = f(x) + \nu^T g(x) + \pi^T h(x) - \mu_1^T G(x) - \mu_2^T H(x).$$

Scheel and Scholtes [26] give the following types of stationary point x^*:

C-stationarity:

$$\nu_i^* \ge 0 \text{ and } \mu_{1,l}^* \mu_{2,l}^* \ge 0 \text{ for all } l \in \mathcal{I}^*; \tag{2.2}$$

M-stationarity:

$$\nu_i^* \ge 0 \text{ and for all } l \in \mathcal{I}^* \text{ either } \mu_{1,l}^* > 0 \text{ and } \mu_{2,l}^* > 0, \text{ or } \mu_{1,l}^* \mu_{2,l}^* = 0, \tag{2.3}$$

B-stationarity: the points for which $d = 0$ is a solution of the problem obtained by linearizing all the data of (NLP) with the exception of the complementarity constraints $G(x) \circ H(x) \le 0$,

S-stationarity:

$$\nu_i^* \ge 0 \text{ and } \mu_{1,l}^* \ge 0, \mu_{2,l}^* \ge 0 \text{ for all } l \in \mathcal{I}^*. \tag{2.4}$$

If a point is a strong stationary point, then it is also a stationary point of any other type [26]. Also, a stationary point of any type is a weak stationary point [26].

Remark. C-stationarity and M-stationarity actually are Fritz–John points for MPCC with a vanishing objective multiplier. The reason for considering these various weaker stationarity concepts is that such points are potential attractors of methods based on the penalization or the regularization [8, 9, 12, 15, 16, 21, 28]. S-stationary is what we look for. As developed in the sequel, C-stationary may be avoided using the MPCC structure.

Definition 2.1. *The MPCC-LICQ is satisfied at the x^* if the following set of vectors is linearly independent:*

$$\{\nabla g_i(x^*)|\ i \in \mathcal{I}_{g^*}\} \cup \{\nabla h_j(x^*)\} \cup \{\nabla G_k(x^*)|\ l \in \mathcal{I}_{G^*}\} \cup \{\nabla H_l(x^*)|\ l \in \mathcal{I}_{H^*}\}\,.$$

We now give two varieties of strict complementarity at a weak stationary point [25], which we will use in the sequel.

DEFINITION 2.2. *Let x^* be a weak stationary point for MPCC.*

- *The upper level strict complementarity (ULSC) holds at x^*, if there exists MPCC multipliers satisfying* (2.1) *with $\mu^*_{m,1}\mu^*_{m,2} \neq 0$ for all $m \in \mathcal{I}^*$.*
- *The lower level strict complementarity (LLSC) holds at x^*, if there exists MPCC multipliers satisfying* (2.1) *and $\mathcal{I}^* = \emptyset$.*

2.2. Stationarity conditions for NLP(t). In this subsection, we discuss the exact and inexact first-order stationarity conditions for NLP(t). We define the Lagrangian function for this problem as follows:

$$\begin{aligned}\mathcal{L}_t(x,\nu,\pi,\mu_1,\mu_2,\eta) = \ & f(x) + \sum_{i=1}^{n_i} \nu_i g_i(x) + \sum_{j=1}^{n_e} \pi_j h_j(x) - \sum_{l=1}^{n_c} \mu_{1,l}(G_l(x)+t) \\ & - \sum_{l=1}^{n_c} \mu_{2,l}(H_l(x)+t) + \sum_{m=1}^{n_c} \eta_m (G_m(x)-t)(H_m(x)-t).\end{aligned} \tag{2.5}$$

The first-order necessary optimality conditions for the problem NLP(t) are then if x is a local minimum of NLP(t), then there exists ν, π, μ_1, μ_2, and η such that

$$\begin{aligned} \nabla_x \mathcal{L}_t(x,\nu,\pi,\mu_1,\mu_2,\eta) = 0, \\ 0 \le \nu \perp g(x) \le 0, \\ 0 \le \mu_1 \perp (G(x) + t\cdot e_c) \ge 0, \\ 0 \le \mu_2 \perp (H(x) + t\cdot e_c) \ge 0, \\ 0 \le \eta \perp [(G(x) - t\cdot e_c)(H(x) - t\cdot e_c)] \le 0, \end{aligned} \tag{2.6}$$

within an algorithmic framework, we will need the following notion of approximate stationarity.

DEFINITION 2.3. *We say that x is an ϵ-stationary point of NLP(t) if there exist multipliers $\nu, \pi, \mu_1, \mu_2, \eta$ satisfying*

$$\begin{aligned} & \|\nabla_x \mathcal{L}_t(x,\nu,\pi,\mu_1,\mu_2,\eta)\|_\infty \le \epsilon, \\ & 0 \le \nu, g(x) \le 0, \nu^T g(x) \ge -\epsilon \\ & |h(x)| \le \epsilon, \left|\pi^T h(x)\right| \le \epsilon \\ & 0 \le \mu_1, (G(x) + t\cdot e_c) \ge 0, \mu_1^T(G(x)+t\cdot e_c) \le \epsilon, \\ & 0 \le \mu_2, (H(x) + t\cdot e_c) \ge 0, \mu_2^T(H(x)+t\cdot e_c) \le \epsilon, \\ & 0 \le \eta, (G(x) - t\cdot e_c)(H(x) - t\cdot e_c) \le 0, \\ & \eta^T [(G(x) - t\cdot e_c)(H(x) - t\cdot e_c)] \ge -\epsilon. \end{aligned} \tag{2.7}$$

We next discuss constraint qualifications for problem (NLP(t)). We denote

$$\mathcal{D} = \{x : \exists l \text{ such that } G_l(x) = t, H_l(x) = t\}. \tag{2.8}$$

The following result shows that the MPCC-LICQ implies the LICQ of NLP(t) for sufficiently small t.

THEOREM 2.4. *Suppose that MPCC-LICQ holds at a feasible point x^* of the NLP* (1.1)*, then there exists a neighborhood $\mathcal{U}$ of x^* and a scalar $t^* > 0$ such that for every $t \in (0, t^*)$ the LICQ holds at every feasible point $x \in \mathcal{U} \setminus \mathcal{D}$ of NLP(t).*

Proof. We have the following relations:

$$\begin{aligned} \mathcal{I}_g \subseteq \mathcal{I}_{g^*}, \\ \mathcal{I}^+_{GH} \cup \mathcal{I}^-_{GH} \subseteq \mathcal{I}_{G^*H^*}, \\ \mathcal{I}^+ \cap \mathcal{I}^-_{GH} = \emptyset, \end{aligned} \tag{2.9}$$

which hold for all x in $\mathcal{U}$ and t in $(0, t^*)$ for sufficiently small $t^* > 0$. Indeed, for such t, the active gradients of NLP(t) at a feasible point $x \in \mathcal{U}$ are

$$\begin{array}{rl} \nabla g_i(x), & i \in \mathcal{I}_g, \\ \nabla h_j(x), & j = 1, \dots, n_e, \\ \nabla G_l(x), & l \in \mathcal{I}_G^+, \\ \nabla H_r(x), & r \in \mathcal{I}_H^+, \\ (H_m(x) - t)\nabla G_m(x) + (G_m(x) - t)\nabla H_m(x), & m \in \mathcal{I}_{GH}^-, \end{array}$$

We then note that if $m \in \mathcal{I}_{GH} \cap (\mathcal{I}_G^+ \cup \mathcal{I}_H^+)$ we have

$$\nabla((G_m(x) - t)(H_m(x) - t)) = \begin{cases} -2t\nabla H_m(x), & \text{if } m \in \mathcal{I}_G^+, \\ \\ -2t\nabla G_m(x), & \text{if } m \in \mathcal{I}_H^+. \end{cases}$$

In view of the MPCC-LICQ assumption and (2.9), the equation

$$\begin{aligned} &\sum_{i \in \mathcal{I}_g} \nu_i \nabla g_i(x) + \sum_{j=1}^{n_e} \pi_j \nabla h_j(x) - \sum_{l \in \mathcal{I}_G^+} \mu_{l,1} \nabla G_l(x) - \sum_{r \in \mathcal{I}_H^+} \mu_{r,2} \nabla H_l(x) \\ &\quad + \sum_{m \in \mathcal{I}_{GH}^-} [\eta_m (H_m(x) - t)\nabla G_m(x) + \eta_m (G_m(x) - t)\nabla H_m(x)] = 0 \end{aligned} \tag{2.10}$$

implies that $\nu_i = \pi_j = \mu_{l,1} = \mu_{r,2} = 2\eta_m t = \eta_m(H_m(x) - t) = \eta_m(G_m(x) - t) = 0$. Finally, if $m \in \mathcal{I}_{GH}^- \setminus \mathcal{I}_{GH}^+$ we have either $G_m(x) = t$ or $H_m(x) = t$ but not both, because $x \notin \mathcal{D}$ by assumption. Hence $\eta_m = 0$. □

3. Existence of Lagrange multipliers for the NLP(t). This section deals with the existence of Lagrange multipliers for a stationary point x of the regularized problem NLP(t) under the MPCC-LICQ. Let $\mathcal{P}(\mathcal{I}^-)$ be a set of parts of $\mathcal{I}^-$. We associate with an index set $I \in \mathcal{P}(\mathcal{I}^-)$ the ordinary nonlinear program $NLP_I(t)$:

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & \\ & g(x) \le 0, \quad h(x) = 0, \\ & G_l(x) \ge -t, \quad H_l(x) \ge -t, \qquad l \notin \mathcal{I}^-, \\ & (G_l(x) - t)(H_l(x) - t) \le 0, \qquad l \notin \mathcal{I}^-, \\ & G_r(x) \ge t, \quad -t \le H_r(x) \le t, \quad r \in I, \\ & H_m(x) \ge t, \quad -t \le G_m(x) \le t, \quad m \in I^c, \end{array} \tag{3.1}$$

where I^c denotes the complement of I in $\mathcal{I}^-$. We denote by $\mathcal{F}_{NLP_{I(t)}}$ and $\mathcal{F}_{NLP(t)}$ the feasible sets of the programs (3.1) and NLP(t) respectively, and obtain the relations

$$\mathcal{F}_{NLP_I}(t) \subseteq \mathcal{F}_{NLP(t)} = \bigcup_{I \in \mathcal{P}(\mathcal{I}^-)} \mathcal{F}_{NLP_I}(t)$$

locally around the point x. In particular, x is a local minimizer of program NLP(t) if and only if it is a local minimizer of the problem (3.1) for every I. Since the stationary points of the problem (3.1) do not belong to the set $\mathcal{D}$, the following lemma is an immediate consequence of the Theorem 2.4.

Lemma 3.1. *Suppose that x^* satisfies MPCC-LICQ, then there exists a neighborhood $\mathcal{U}$ of x^* and a scalar $t^* > 0$ such that for every $t \in (0, t^*)$ LICQ holds at every feasible point $x \in \mathcal{U}$ of problem* (3.1).

Lemma 3.2. *Let x be a solution of $NLP_I(t)$ for every $I \in \mathcal{P}(\mathcal{I}^-)$. Suppose that the LICQ holds at x for $NLP_I(t)$, then for $m \in \mathcal{I}^-$ the Lagrange multipliers $\mu_{1,m}$ and $\mu_{2,m}$ associated, respectively, to the constraints $G_m(x)$ and $H_m(x)$ vanish.*

Proof. x is a local minimizer of the $NLP_I(t)$ which satisfies the LICQ, implies the existence and uniqueness of the Lagrange multipliers $(\nu, \pi, \mu_1, \mu_2, \eta)$ such that

$$\begin{aligned}\nabla f(x) = &- \sum_{i \in \mathcal{I}_g} \nu_i \nabla g_i(x) - \sum_{j=1}^{n_e} \pi_j \nabla h_j(x) + \sum_{l \in \mathcal{I}_G^-} \mu_{1,l} \nabla G_l(x) + \sum_{l \in \mathcal{I}_H^-} \mu_{2,l} \nabla H_l(x) \\ &- \sum_{m \in \mathcal{I}_G^+ \backslash \mathcal{I}^-} \eta_m (H_m - t) \nabla G_m(x) - \sum_{m \in \mathcal{I}_H^+ \backslash \mathcal{I}^-} \eta_m (G_m - t) \nabla H_m(x) \\ &- \sum_{m \in \mathcal{I}^-} (\mu_{1,m} \nabla G_m(x) + \mu_{2,m} \nabla H_m(x)),\end{aligned} \tag{3.2}$$

$$\begin{aligned}&\nu_{\mathcal{I}_g} \geq 0, \quad \mu_{1,\mathcal{I}_G^-} \geq 0, \quad \mu_{2,\mathcal{I}_H^-} \geq 0, \quad \eta_{1,\mathcal{I}_G^+} \geq 0, \quad \eta_{2,\mathcal{I}_H^+} \geq 0, \\ &\mu_{1,I} \leq 0, \quad \mu_{2,I} \geq 0, \quad \mu_{1,I^c} \geq 0, \quad \mu_{2,I^c} \leq 0,\end{aligned}$$

x is also a local minimizer of the $NLP_{I^c}(t)$ which satisfies the LICQ, then there exists the only Lagrange multipliers $(\bar{\nu}, \bar{\pi}, \bar{\mu_1}, \bar{\mu_2}, \bar{\eta})$ such that we have the same conditions as (3.2) with this small modification in the two last lines

$$\begin{aligned}\bar{\mu}_{1,I} \geq 0, &\quad \bar{\mu}_{2,I} \leq 0, \\ \bar{\mu}_{1,I^c} \leq 0, &\quad \bar{\mu}_{2,I^c} \geq 0.\end{aligned}$$

Hence, under the LICQ we have

$$\begin{aligned}\nu_{\mathcal{I}_g} = \bar{\nu}_{\mathcal{I}_g}, &\quad \pi_j = \bar{\pi}_j, j = 1, \ldots, n_e, \\ \mu_{1,\mathcal{I}_G^-} = \bar{\mu}_{1,\mathcal{I}_G^-}, &\quad \mu_{2,\mathcal{I}_H^-} = \bar{\mu}_{2,\mathcal{I}_H^-}, \\ \eta_{1,\mathcal{I}_G^+} = \eta_{1,\mathcal{I}_G^+}, &\quad \eta_{2,\mathcal{I}_H^+} = \eta_{2,\mathcal{I}_H^+}, \\ 0 \geq \mu_{1,I} = \bar{\mu}_{1,I} \geq 0, &\quad 0 \leq \mu_{2,I} = \bar{\mu}_{2,I} \leq 0, \\ 0 \geq \mu_{1,I^c} = \bar{\mu}_{1,I^c} \geq 0, &\quad 0 \leq \mu_{2,I^c} = \bar{\mu}_{2,I^c} \leq 0.\end{aligned} \tag{3.3}$$

From the two last lines in (3.3), we deduce that

$$\mu_{1,\mathcal{I}^-} = \bar{\mu}_{1,\mathcal{I}^-} = 0, \quad \mu_{2,\mathcal{I}^-} = \bar{\mu}_{2,\mathcal{I}^-} = 0. \qquad \square$$

Theorem 3.3. *Let $\{t^k\}$ be a sequence of positive scalars tending to zero, let $\{x^k\}$ be a stationary point of NLP(t) tending to x^*, and suppose MPCC-LICQ holds at x^*. Then, for every sufficiently large k there exists Lagrange multipliers*

$$\begin{aligned}\nu_i^k, &\quad i \in \mathcal{I}_{g^k} &\quad \pi_j^k, &\quad j = 1, \ldots, n_e, \\ \mu_{1,r}^k, &\quad l \in \mathcal{I}_{G^k}^+, &\quad \mu_{2,r}^k, &\quad r \in \mathcal{I}_{H^k}^+, \\ \eta_m^k, &\quad m \in \mathcal{I}_{G^k H^k}^- & & \end{aligned}$$

for NLP(t^k) at x^k, and they are unique if $m \in \mathcal{I}_{G^k H^k}^- \setminus \mathcal{I}_k^-$.

Proof. Let x^k be a stationary point of the regularized problem NLP(t^k). Then there exists a nonvanishing vector of the Lagrange multipliers $(\alpha^k, \nu^k, \pi^k, \mu_1^k, \mu_2^k, \eta^k)$ such that the Fritz–John conditions are satisfied:

$$\begin{aligned}&\nabla L(\alpha^k, \nu^k, \pi^k, \mu_1^k, \mu_2^k, \eta^k) = 0, \\ &\nu_{\mathcal{I}_g^k}^k \geq 0, \quad \mu_{1,\mathcal{I}_{G^k}^-}^k \geq 0, \quad \mu_{2,\mathcal{I}_{H^k}^-}^k \geq 0, \\ &\eta_{1,\mathcal{I}_{G^k}^+}^k \geq 0, \quad \eta_{2,\mathcal{I}_{H^k}^+}^k \geq 0,\end{aligned} \tag{3.4}$$

where

$$\begin{aligned}\nabla L(\alpha,\nu,\pi,\mu_1,\mu_2,\eta) = \quad & \alpha\nabla f(x) + \sum_{i\in\mathcal{I}_g}\nu_i\nabla g_i(x) + \sum_{j=1}^{n_e}\pi_j\nabla h_j(x)\\ & - \sum_{l\in\mathcal{I}_G^-}\mu_{1,l}\nabla G_l(x) + \sum_{m\in\mathcal{I}_G^+}\eta_m(H_m - t)\nabla G_m(x)\\ & - \sum_{l\in\mathcal{I}_H^-}\mu_{2,l}\nabla H_l(x) + \sum_{m\in\mathcal{I}_H^+}\eta_m(G_m - t)\nabla H_m(x).\end{aligned}$$

The constraint $\left(G_m(x^k) - t^k\right)\left(H_m(x^k) - t^k\right) \leq 0$ is active, if one or both of the functions G_m and H_m is equal to t^k at x^k. If there is no index such that both functions G_m and H_m are equal to t^k at x^k, i.e, $\mathcal{I}_k^- = \emptyset$, then the result is a consequence of the Theorem 2.4, since the LICQ holds at x^k for sufficiently large k. Consequently, α^k and the Lagrange multipliers $\left(\nu^k, \pi^k, \mu_1^k, \mu_2^k, \eta^k\right)$ are unique. We now consider the case when there exists at least one index m such that $G_m(x^k) = G_m(x^k) = t^k$, i.e, $\mathcal{I}_k^- \neq \emptyset$. The point x^k is a stationary point of the NLP(t^k), then x^k is a stationary point of $NLP_I(t^k)$ for every $I \in \mathcal{P}(\mathcal{I}_k^-)$. Since x^* satisfies MPCC-LICQ, then from Lemma 3.1, the LICQ holds at x^k for the problem $NLP_I(t^k)$ for every $I \in \mathcal{P}(\mathcal{I}_k^-)$. Lemma 3.2 implies that the Lagrange multipliers $\mu_{1,\mathcal{I}_k^-}$ and $\mu_{2,\mathcal{I}_k^-}$ vanish and $(\nu^k_{\mathcal{I}_g^k}, \mu^k_{1,\mathcal{I}^-_{G^k}}, \mu^k_{2,\mathcal{I}^-_{H^k}}, \eta^k_{1,\mathcal{I}^+_{G^k}}, \eta^k_{2,\mathcal{I}^+_{H^k}})$ are unique. Hence, the stationarity conditions of the $NLP_I(t^k)$ at x^k are as follows:

$$\begin{aligned}&\nabla L(1,\nu^k,\pi^k,\mu_1^k,\mu_2^k,\eta^k) = 0,\\ &\nu^k_{\mathcal{I}_{g^k}} \geq 0, \quad \mu^k_{1,\mathcal{I}^-_{G^k}} \geq 0, \quad \mu^k_{2,\mathcal{I}^-_{H^k}} \geq 0,\\ &\eta^k_{1,\mathcal{I}^+_{G^k}} \geq 0, \quad \eta^k_{2,\mathcal{I}^+_{H^k}} \geq 0,\\ &\mu^k_{1,\mathcal{I}_k^-} = 0, \quad \mu^k_{2,\mathcal{I}_k^-} = 0,\end{aligned} \tag{3.5}$$

which implies that x^k is a KKT point of the NLP(t^k). □

We give in the following a limiting behavior of an approaching sequence of the stationary points sequence for the regularized problem (NLP(t)).

4. Convergence results. In this section, we consider the limiting behavior of problem (NLP(t^k)) as $t^k \to 0$. We denote by $\mathcal{F}$ the feasible set of problem (1.1). The next theorem establishes the relations between the solutions of the original problem MPCC and those of the regularization NLP(t), under some classical MPCC conditions. We start by giving a sufficient condition which guarantee the convergence to B-stationary point of MPCC.

Definition 4.1. *A sequence $\{x^k\}$ is asymptotically weakly nondegenerate, if $\{x^k\} \longrightarrow x^*$ as $\{t^k\} \searrow 0$, and there is a $t^* > 0$ such that for $t \in (0, t^*)$ one has*

$$-1 \leq \frac{G_i(x^k)}{H_i(x^k)} \leq 1, \quad i \in (\mathcal{I}^-_{H^k} \setminus \mathcal{I}^+_{G^k}) \cap \mathcal{I}^*,$$

and

$$-1 \leq \frac{H_i(x^k)}{G_i(x^k)} \leq 1, \quad i \in (\mathcal{I}^-_{G^k} \setminus \mathcal{I}^+_{H^k}) \cap \mathcal{I}^*.$$

This definition is similar to the one given by Liu and Sun [22] as well as Fukushima, Liu, and Pang [11]. It means that for all $i \in ((\mathcal{I}^-_{H^k} \setminus \mathcal{I}^+_{G^k}) \cup (\mathcal{I}^-_{G^k} \setminus \mathcal{I}^+_{H^k})) \cap \mathcal{I}^*$ the functions G_i and H_i tend to zero with the same order. We note that the asymptotically weak nondegeneracy condition is a key assumption for our convergence result. This condition is not excessively stringent because it is implied by the lower level strict complementarity, ($\mathcal{I}^* = \emptyset$), and the upper level strict complementarity, ($\mu^*_{i,1}\mu^*_{i,2} \neq 0, i \in \mathcal{I}^*$), and is weaker than both; see example 4 below.

THEOREM 4.2. *Let $\{t^k\} \subseteq (0, +\infty)$ be convergent to 0, $\{\epsilon^k\}$ be a nonnegative convergent sequence with $\epsilon^k \to 0$, and x^k be ϵ^k-stationary point of NLP(t^k) for each k. Let x^* be any accumulation point of the sequence $\{x^k\}$. We suppose that the MPCC-LICQ holds at x^*. Then the following statements hold:*

1. *x^* is M-stationary for the problem* (1.1) *with unique multipliers $\nu^*, \pi^*, \mu^*_1, \mu^*_2$.*
2. *If $\{x^k\}$ is asymptotically weakly nondegenerate, then x^* is strongly stationary for* (1.1).

Proof. Suppose without loss of generality that $\{x^k\} \to x^*$. Since all the functions involved in the problem are continuous, and $\mathcal{F}$ is closed, hence $x^* \in \mathcal{F}$. Let $(\nu^k, \pi^k, \mu^k_1, \mu^k_2, \eta^k)$ be multipliers associated with x^k. From ϵ^k-stationarity conditions (2.7) at point x^k, we have

$$\begin{array}{l}
\nu^k_i g_i(x^k) \geq (g(x^k))^T \nu^k \geq -\epsilon^k,\\
\mu^k_{1,l}(G_l(x^k) + t^k) \leq (G(x^k) + t^k \cdot e_c)^T \mu^k_1 \leq \epsilon^k,\\
\mu^k_{2,r}(H_r(x^k) + t^k) \leq (H(x^k) + t^k \cdot e_c)^T \mu^k_2 \leq \epsilon^k,\\
\eta^k_m(G_m(x^k) - t^k)(H_m(x^k) - t^k) \geq \left[(G^k - t^k \cdot e_c) \circ (H^k - t^k \cdot e_c)\right]^T \eta^k \geq -\epsilon^k.
\end{array} \tag{4.1}$$

It follows that the multipliers associated with nonactive constraints are

$$\begin{array}{llllll}
\nu^k_i & = O(\epsilon^k), & i \notin \mathcal{I}_{g^k}; & \eta^k_m & = O(\epsilon^k), & m \notin \mathcal{I}^-_{G^k H^k},\\
\mu^k_{1,l} & = O(\epsilon^k), & l \notin \mathcal{I}^+_{G^k}; & \mu^k_{2,r} & = O(\epsilon^k), & r \notin \mathcal{I}^+_{H^k}.
\end{array}$$

For sufficiently large k, we construct a matrix $M(x^k)$ whose columns consist of the vectors

$$\begin{array}{llll}
\nabla g_i(x^k), & i \in \mathcal{I}_{g^*}; & \nabla h_j(x^k), & j = 1, \ldots, n_e,\\
\nabla G_l(x^k), & l \in \mathcal{I}_{G^*}; & \nabla H_r(x^k), & r \in \mathcal{I}_{H^*},
\end{array}$$

this matrix converges to, as $k \to +\infty$, $M(x^*)$ with columns

$$\begin{array}{llll}
\nabla g_i(x^*), & i \in \mathcal{I}_{g^*}; & \nabla h_j(x^*), & j = 1, \ldots, n_e,\\
\nabla G_l(x^*), & l \in \mathcal{I}_{G^*}; & \nabla H_r(x^*), & r \in \mathcal{I}_{H^*}.
\end{array}$$

By (2.7), we have

$$\begin{array}{l}
(\nabla g^k)^T \nu^k = \displaystyle\sum_{i \in \mathcal{I}_{g^k}} \nu^k_i \nabla g^k_i + O(\epsilon^k),\\
(\nabla G^k)^T \mu^k_1 = \displaystyle\sum_{l \in \mathcal{I}^+_{G^k}} \mu^k_{1,l} \nabla G^k_l + O(\epsilon^k), \quad (\nabla H^k)^T \mu^k_2 = \displaystyle\sum_{r \in \mathcal{I}^+_{H^k}} \mu^n_{2,r} \nabla H^k_r + O(\epsilon^k),\\
(\nabla \Phi^k)^T \eta^k = \displaystyle\sum_{m \in \mathcal{I}^-_{G^k}} \eta^k_m (H^k_m - t^k) \nabla G^k_m + \sum_{m \in \mathcal{I}^-_{H^k}} \eta^k_m (G^k_m - t^k) \nabla H^k_m + O(\epsilon^k),
\end{array} \tag{4.2}$$

where $\Phi^k = (G(x^k) - t^k \cdot e_c) \circ (H(x^k) - t^k \cdot e_c)$. Thus, taking into account these facts, we can write the first row of (2.7) as follows

$$
\begin{aligned}
-\nabla f(x^k) = {} & \sum_{i\in\mathcal{I}_{g^k}} \nu_i^k \nabla g_i^k + \sum_{j=1}^{n_e} \pi_j^k \nabla h_j^k - \sum_{l\in\mathcal{I}_{G^k}^+} \mu_{1,l}^k \nabla G_l^k - \sum_{r\in I_H^+} \mu_{2,r}^k \nabla H_r^k \\
& + \sum_{m\in\mathcal{I}_{G^k}^-} \eta_m^k (H_m^k - t^k)\nabla G_m^k + \sum_{m\in\mathcal{I}_{H^k}^-} \eta_m^k ({G_m}^k - t^k)\nabla H_m^k + O(\epsilon^k),
\end{aligned} \tag{4.3}
$$

we can restate the relation above as follows:

$$
\begin{aligned}
-\nabla f(x^k) = {} & \sum_{i\in\mathcal{I}_{g^k}} \nu_i^k \nabla g_i(x^k) + \sum_{j=1}^{n_e} \pi_j^k \nabla h_j(x^k) \\
& - \sum_{l\in(\mathcal{I}^+_{G^k H^k}\setminus\mathcal{I}^-_{G^k H^k})\cap\mathcal{I}^*} (\mu_{1,l}^k \nabla G_l(x^k) + \mu_{2,l}^k \nabla H_l(x^k)) \\
& - \sum_{l\in\mathcal{I}^+_{G^k}\cap\mathcal{I}^-_{H^k}\cap\mathcal{I}^*} \left[\mu_{1,l}^k \nabla G_l(x^k) - \eta_l^k (G_l(x^k) - t^k)\nabla H_l(x^k)\right] \\
& - \sum_{l\in\mathcal{I}^+_{H^k}\cap\mathcal{I}^-_{G^k}\cap\mathcal{I}^*} \left[\mu_{2,l}^k \nabla H_l(x^k) - \eta_l^k (H_l(x^k) - t^k)\nabla G_l(x^k)\right] \\
& - \sum_{l\in(\mathcal{I}^-_{G^k}\setminus\mathcal{I}^+_{H^k})\cap\mathcal{I}^*} \left[-\eta_l^k (H_l(x^k) - t^k)\nabla G_l(x^k) - 0\nabla H_l(x^k))\right] \\
& - \sum_{l\in(\mathcal{I}^-_{H^k}\setminus\mathcal{I}^+_{G^k})\cap\mathcal{I}^*} \left[0\nabla G_l(x^k) - \eta_l^k (G_l(x^k) - t^k)\nabla H_l(x^k)\right] \\
& - \sum_{l\in\mathcal{I}^+_{G^k}\setminus\mathcal{I}_{H^*}} \mu_{1,l}^k \nabla G_l(x^k) - \sum_{l\in\mathcal{I}^-_{G^k}\setminus\mathcal{I}_{H^*}} -\eta_l^k (H_l(x^k) - t^k)\nabla G_l(x^k)) \\
& - \sum_{l\in\mathcal{I}^+_{H^k}\setminus\mathcal{I}_{G^*}} \mu_{2,l}^k \nabla H_l(x^k) - \sum_{l\in\mathcal{I}^-_{H^k}\setminus\mathcal{I}_{G^*}} -\eta_l^k (G_l(x^k) - t^k)\nabla H_l(x^k) \\
& - \sum_{l\in\mathcal{I}_{G^*}\setminus(\mathcal{I}^+_{G^k}\cup\mathcal{I}^-_{G^k})} 0\nabla G_l(x^k) - \sum_{l\in\mathcal{I}_{H^*}\setminus(\mathcal{I}^+_{H^k}\cup\mathcal{I}^-_{H^k})} 0\nabla H_l(x^k) + O(\epsilon^k) \\
= {} & M^T(x^k) \begin{pmatrix} \nu^k \\ \pi^k \\ \delta^k \\ \gamma^k \end{pmatrix} + O(\epsilon^k),
\end{aligned} \tag{4.4}
$$

where δ^k, γ^k are given by

$$
\delta^k = \begin{pmatrix} \mu_{1,I}^k \\ \mu_{1,II}^k \\ -\eta_{III}^k \circ (H_{III}(x^k) - t^k \cdot e_{III}) \\ \eta_{IV}^k \circ (H_{IV}(x^k) - t^k \cdot e_{IV}) \\ 0_V \\ \mu_{1,VI}^k \\ \eta_{VII}^k \circ (H_{VII}(x^k) - t^k \cdot e_{VII}) \\ 0_{VIII} \end{pmatrix}, \tag{4.5}
$$

$$\gamma^k = \begin{pmatrix} \mu^k_{2,I} \\ -\eta^k_{II} \circ (G_{II}(x^k) - t^k \cdot e_{II}) \\ \mu^k_{2,III} \\ 0_{IV} \\ \eta^k_V \circ (G_V(x^k) - t^k \cdot e_V) \\ \mu^k_{2,IX} \\ \eta^k_X \circ (G_X(x^k) - t^k \cdot e_X) \\ 0_{XI} \end{pmatrix}, \tag{4.6}$$

and

$$\begin{aligned} I &= (\mathcal{I}^+_{G^kH^k} \setminus \mathcal{I}^-_{G^kH^k}) \cap \mathcal{I}^*, & & \\ II &= \mathcal{I}^+_{G^k} \cap \mathcal{I}^-_{H^k} \cap \mathcal{I}^*, & III &= \mathcal{I}^+_{H^k} \cap \mathcal{I}^-_{G^k} \cap \mathcal{I}^*, \\ IV &= (\mathcal{I}^-_{G^k} \setminus \mathcal{I}^+_{H^k}) \cap \mathcal{I}^*, & V &= (\mathcal{I}^-_{H^k} \setminus \mathcal{I}^+_{G^k}) \cap \mathcal{I}^*, \\ VI &= \mathcal{I}^+_{G^k} \setminus \mathcal{I}_{H^*}, & VII &= \mathcal{I}^-_{G^k} \setminus \mathcal{I}_{H^*}, \\ VIII &= \mathcal{I}^+_{H^k} \setminus \mathcal{I}_{G^*}, & IX &= \mathcal{I}^-_{H^k} \setminus \mathcal{I}_{G^*}, \\ X &= \mathcal{I}^* \setminus (I^+_{G^k} \cup \mathcal{I}^-_{G^k}), & XI &= \mathcal{I}^* \setminus (I^+_{H^k} \cup \mathcal{I}^-_{H^k}). \end{aligned} \tag{4.7}$$

The matrix $M(x^k)$ converges to the matrix $M(x^*)$ which has full column rank, because the multipliers $\nu^k,\ \pi^k,\ \delta^k,\ \gamma^k$ are unique, see Theorem 3.3, and converge, respectively, to the unique MPCC multipliers $\nu^*,\ \pi^*,\ \mu^*_1,\ \mu^*_2$ at x^*. Thus, the weak stationarity conditions of the MPCC (1.1) are satisfied at x^*. The rest of the proof is to show that either $\mu^*_{1,l}\mu^*_{2,l} > 0$, or $\mu^*_{1,l}\mu^*_{2,l} = 0$ for each $l \in \mathcal{I}^*$. For such l, we have five possible cases, that is, the first five sets in (4.7):

In the first case, we have $\delta^k_l = \mu^k_{l,1} \geq 0$ and $\gamma^k_l = \mu^k_{l,2} \geq 0$, then $\delta^k_l \to \mu^*_{l,1} \geq 0$ and $\gamma^k_l \to \mu^*_{l,2} \geq 0$ as $t^k \to 0$. The second and the third case, we have, respectively, $(\delta^k_l = \mu^k_{1,l} \geq 0, \gamma^k_l = 2t^k\eta^k_l \geq 0)$ and $(\delta^k_l = 2t^k\eta^k_l \geq 0, \gamma^k_l = \mu^k_{2,l} \geq 0)$, then δ^k_l and γ^k_l converge to the positive values.

In the fourth and the fifth case we have one of the two multipliers δ^k_l or γ^k_l is equal to 0. Then $\delta^k_l\gamma^k_l = 0$. Consequently, x^* is M-stationary.

The second statement follows immediately from the first claim. Indeed, under the asymptotically weakly nondegenerate assumption we have, respectively, $-t^k \leq H_l(x^k) \leq t^k$ for $l \in (\mathcal{I}^-_{G^k} \setminus \mathcal{I}^+_{H^k}) \cap \mathcal{I}^*$ and $-t^k \leq G_l(x^k) \leq t^k$ for $l \in (\mathcal{I}^-_{H^k} \setminus \mathcal{I}^+_{G^k}) \cap \mathcal{I}^*$. Then, the multiplier formulas in the fourth and fifth case will be, respectively, $\delta^k_l = -\eta^k_l(H_l(x^k) - t^k) \geq 0$, $\gamma^k_l = 0$ and $\delta^k_l = 0$, $\gamma^k_l = -\eta^k_l(G_l(x^k) - t^k) \geq 0$. Therefore, x^* is strongly stationary for the original MPCC. $\square$

Remark. If the ULSC assumption is considered, the fourth and the fifth case in the part (1) cannot occur, which yields that $\mu^*_{l,1} \geq 0$ and $\mu^*_{l,2} \geq 0$.

The asymptotically weak nondegeneracy hypothesis is not easy to bypass.

Example 3. The point $(x, y) = (1, 0)$ is a solution of the program

$$\begin{array}{ll} \min & x^2 - xy + \frac{1}{3}y^2 - 2x, \\ \text{s.t.} & \\ & x \geq 0, \quad y \geq 0, \\ & xy \leq 0, \end{array} \tag{4.8}$$

which is a strongly stationary point. However, the stationary points of the regularized scheme (NLP(t))

$$\begin{array}{ll} \min & x^2 - xy + \frac{1}{3}y^2 - 2x, \\ \text{s.t.} & \\ & x \geq -t, \quad y \geq -t, \\ & (x - t)(y - t) \leq 0, \end{array}$$

are $(x, y) = (1, 0)$ and $(x, y) = (t, \frac{3t}{2})$. The first stationary point converges to the optimal solution, while the second converges to $(0, 0)$ which is M-stationary point with multipliers $\mu_1 = -2$ of $x \geq 0$ and $\mu_2 = 0$ of $y \geq 0$. The point $(t, \frac{3t}{2})$ is not an asymptotically weak nondegenerate point.

The following example shows that the asymptotically weak nondegeneracy condition is weaker than the upper level strict complementarity condition employed in the literature [15, 16, 28].

Example 4. The origin with multipliers $(\mu_1^*, \mu_2^*) = (0, 0)$ is the unique strongly stationary point of the program

$$\begin{array}{ll} \min & (x+y)^2 + y^2, \\ \text{s.t.} & \\ & x + y \geq 0, \quad y \geq 0, \\ & y(x+y) \leq 0. \end{array} \tag{4.9}$$

We note that the upper level strict complementarity is not satisfied at $(x^*, y^*) = (0, 0)$ because $\mu_1^* \mu_2^* = 0$. For a small $t > 0$, the regularized problem NLP(t) of the program (4.9) has $(x, y, \mu_1, \mu_2, \eta) = (t, 0, 0, 0, 2)$ as a stationary point which satisfies the weak nondegeneracy condition. The point (x, y, δ, γ) with $\delta = -\eta(y - t) = 2t$ and $\gamma = -\eta(x + y - t) = 0$ converges to $(x^*, y^*, \mu_1^*, \mu_2^*) = (0, 0, 0, 0)$ as t tending to zero.

5. Characterization of attractors.

In this section we describe a situation where the sequence of local minimizers $\{x^k\}$ of NLP(t^k) with $\{t^k\} \searrow 0$ is attracted to a local minimizer of the MPCC. Consider the following example:

$$\begin{array}{ll} \min & x(1-y^2), \\ \text{s.t.} & \\ & x \geq 0, \quad y \geq 0, \\ & xy \leq 0. \end{array}$$

The local and global minimizers of MPCC are the whole positive y-axis. However, the stationary points of our regularized problem NLP(t), for a small value of $t < 1$, are

1. $(x,\ y) = (t,\ -t)$, and $(\mu_1,\ \mu_2,\ \eta) = \left(0,\ 2t^2,\ \frac{1-t^2}{2t}\right)$;
2. $(x,\ y) = (t,\ 0)$, and $(\mu_1,\ \mu_2,\ \eta) = (0,\ 0,\ 1/t)$;
3. $(x,\ y) = (-t,\ t)$, and $(\mu_1,\ \mu_2,\ \eta) = (1 - t^2,\ 0,\ t)$;

which all converge to the same point $(x^*,\ y^*) = (0,\ 0)$. This shows that further conditions, in addition to MPCC-LICQ, are necessary to guarantee that a local minimizer is a limit point of a sequence of stationary point x^k of NLP(t) with $t = t^k$.

The next theorem shows that whether x^* is a strict local minimum of the MPCC, the regularization method may generate a convergent sequence $\{x^k\} \to x^*$. We denote by $B(x^*, r)$ the closed ball centered at x^* with radius r.

THEOREM 5.1. *Let x^* be a locally unique B-stationary point of the MPCC. Then, there exists $r > 0$ and $\bar{t} > 0$ such that for each $t \in (0, \bar{t}]$ there exists a local minimum of the regularized problem NLP(t) in the ball $B(x^*, r)$.*

Proof. Since x^* is a strict local minimum of the MPCC, then there exists a $r > 0$ such that for all feasible point $x \neq x^*$ of the MPCC in $B(x^*, r)$, we have $f(x^*) < f(x)$, which implies that there exists a $\alpha > 0$ such that

$$f(x^*) + \alpha \leq f(x), \tag{5.1}$$

for all feasible point x which satisfies $\|x - x^*\| = r$. Now, we show that for any $\beta \in (0, \alpha)$ there exists a $\bar{t} > 0$ such that for any $t \in (0, \bar{t}\,]$ we have

$$f(x^*) + \beta \leq f(x),$$

for any feasible point x for the regularization problem NLP(t) that lies on the boundary of $B(x^*, r)$. Suppose by contradiction that there exists a sequence of feasible points $\{x^k\}$ of the regularized problem NLP(t), with $t = t^k$, and a positive scalar γ in $(0, \alpha)$ such that

$$f(x^k) < f(x^*) + \gamma, \text{ for all } k,$$

with $\|x^k - x^*\| = r$. Let $\bar{x}$ be a limit of the sequence $\{x^k\}$ as $t^k \to 0$. By the continuity of the objective and constraint functions, the point $\bar{x}$ is feasible for the MPCC and we have

$$\lim_{k\to\infty} f(x^k) = f(\bar{x}).$$

Therefore,

$$f(\bar{x}) \leq f(x^*) + \gamma;$$

this yields

$$f(\bar{x}) \leq f(x^*) + \gamma < f(x^*) + \alpha,$$

which is a contradiction with (5.1). Hence, for any $\beta \in (0, \alpha)$ there exists $\bar{t} \in (0, t^*]$ such that

$$f(x) \geq f(x^*) + \beta,$$

for any $t \in (0, \bar{t}]$ and any feasible point x for the the regularization problem NLP(t) with $\|x - x^*\| = r$. Thus

$$f(x) > f(x^*),$$

for any x feasible for the NLP(t) that lies on the boundary of $B(x^*, r)$. Since x^* is feasible for the problem NLP(t) with $\|x - x^*\| \leq r$ and it is in the interior of $B(x^*, r)$, we conclude that the global minimum $x(t)$ of the NLP(t) with $\|x - x^*\| \leq r$ lies in the interior of $B(x^*, r)$. Consequently, $x(t)$ is a minimum local of the NLP(t) in the interior of $B(x^*, r)$. ☐

We give now the sufficient condition for the sequence generated by the regularization (1.4) to be attracted to a B-stationarity point x^*. In addition to MPCC-LICQ, we assume that the MPCC satisfy the strong second-order sufficient condition (MPCC-SSOSC) at x^*, as used in [28],

$$d^T \nabla^2_{xx} \mathcal{L}(x^*, \nu^*, \pi^*, \mu_1^*, \mu_2^*) d > 0,$$

for every nonvanishing d with

$$\begin{array}{ll} \nabla_x g_i(x^*)^T d = 0, & i : \nu_i^* > 0, \\ \nabla_x h(x^*)^T d = 0, & \\ \nabla_x G_j(x^*)^T d = 0, & j : \mu_{1,j}^* \neq 0, \\ \nabla_x H_k(x^*)^T d = 0, & k : \mu_{2,k}^* \neq 0. \end{array}$$

THEOREM 5.2. *Suppose that x^* is a B-stationary point of the MPCC at which MPCC-LICQ and MPCC-SSOSC hold. Let Θ be an infinite set of t in a sufficiently small neighborhood of zero, and $S(t)$, with $t \in \Theta$, be a set of stationary points $x(t)$ of*

(1.4) *which are asymptotically weakly nondegenerate. Then, there exists $r > 0$ such that $\emptyset \neq S(t) \cap B(x^*, r) \longrightarrow x^*$, as $t \in \Theta$ and $t \searrow 0$.*

Proof. Before starting the proof of the theorem, we mention that we showed in Theorem 2.4 that MPCC-LICQ implies the LICQ holds at any feasible point of NLP(t) for a small $t > 0$, except the points x such that $G_l(x) = H_l(x) = t$ for some l. But we showed in Theorem 3.3 if these points are stationary, the Lagrange multipliers exist. The MPCC-LICQ and MPCC-SSOSC assumptions and the B-stationary imply that x^* is a strict local minimum point of MPCC [28]. It follows from Theorem 5.1 that there exist $r > 0$ and $\bar{t} > 0$ such that for each $t \in (0, \bar{t}\,]$ there exist a local minimum of the regularized problem NLP(t) in the ball $B(x^*, r)$. This implies that $S(t) \cap B(x^*, r) \neq \emptyset$. Since the stationary points of NLP(t) are supposed asymptotically weakly nondegenerate, it follows from Theorem 4.2 that the sequence of stationary points of NLP(t) in the ball $B(x^*, r)$ converges to a B-stationary point $\bar{x}$ of MPCC; thus $\bar{x} = x^*$. ◻

Notice that if the ULSC assumption is supposed to hold at x^*, we have the same result in the above theorem, since the ULSC implies the asymptotically weak nondegeneracy condition.

6. Extension. In this section, we present a generalized scheme $GNLP(t)$ of the regularization $NLP(t)$

$$GNLP(t) \quad \begin{cases} \min & f(x) \\ \text{s.t.} & \\ & g(x) \leq 0, \quad h(x) = 0, \\ & G_l(x) \geq \tau_{1,l} t, \\ & H_l(x) \geq \tau_{1,l} t, \\ & (G_l(x) - \tau_{2,l} t)(H_l(x) - \tau_{2,l} t) \leq 0, \\ & l = 1, 2, \ldots, n_c, \end{cases}$$

where τ_1 and τ_2 are two constant vectors.

The advantage of this regularization is to find easily an initial point x^0 strictly feasible for GNLP(t) with $t = t_0$. It suffices that x^0 satisfies $g(x^0) < 0$ and $h(x^0) = 0$ and to choose the vectors τ_1 and τ_2 such that

$$(6.1) \qquad \begin{aligned} & \tau_{1,l} > -\min\left\{\frac{G_l(x_0)}{t_0}, \frac{H_l(x_0)}{t_0}\right\}, \\ \text{and} \quad & \\ & \min\left\{\frac{G_l(x_0)}{t_0}, \frac{H_l(x_0)}{t_0}\right\} < \tau_{2,l} < \max\left\{\frac{G_l(x_0)}{t_0}, \frac{H_l(x_0)}{t_0}\right\}, \\ & l = 1, 2, \ldots, n_c. \end{aligned}$$

We show that the convergence results of section 4 remain valid for this generalized regularization. In fact, we adapt the proof of the Theorem 4.2 to this generalized scheme, and the multipliers δ and γ will be written as follows:

$$\delta^k = \begin{pmatrix} \mu^k_{1,I} \\ \mu^k_{1,II} \\ -\eta^k_{III} \circ (H_{III}(x^k) - t^k \cdot \tau_{2,III}) \\ \eta^k_{IV} \circ (H_{IV}(x^k) - t^k \cdot \tau_{2,IV}) \\ 0_V \\ \mu^k_{1,VI} \\ \eta^k_{VII} \circ (H_{VII}(x^k) - t^k \cdot \tau_{2,VII}) \\ 0_{VIII} \end{pmatrix},$$

$$\gamma^k = \begin{pmatrix} \mu^k_{2,I} \\ -\eta^k_{II} \circ (G_{II}(x^k) - t^k \cdot \tau_{2,II}) \\ \mu^k_{2,III} \\ 0_{IV} \\ \eta^k_V \circ (G_V(x^k) - t^k \cdot \tau_{2,V}) \\ \mu^k_{2,IX} \\ \eta^k_X \circ (G_X(x^k) - t^k \cdot \tau_{2,X}) \\ 0_{XI} \end{pmatrix}.$$

We show that either $\mu^*_{1,l}, \mu^*_{2,l} > 0$, or $\mu^*_{1,l}\mu^*_{2,l} = 0$ for each $l \in \mathcal{I}^*$. To this end, we have to prove it for each case of the five cases that we have. In the first case, we have $\delta^k_l = \mu^k_{l,1} \geq 0$ and $\gamma^k_l = \mu^k_{l,2} \geq 0$, then $\delta^k_l \to \mu^*_{l,1} \geq 0$ and $\gamma^k_l \to \mu^*_{l,2} \geq 0$ as $t^k \to 0$. In the second and third cases, we have, respectively, $(\delta^k_l = \mu^k_{1,l} \geq 0, \gamma^k_l = -\eta^k_l(-\tau_{1,l} - \tau_{2,l})t^k \geq 0)$ and $(\delta^k_l = -\eta^k_l(-\tau_{1,l} - \tau_{2,l})t^k \geq 0, \gamma^k_l = \mu^k_{2,l} \geq 0)$. Since τ_1 and τ_2 are chosen as in (6.1), then $\tau_1 + \tau_2 \in \boldsymbol{R}^{n_c}_+$; thus δ^k_l and γ^k_l converge to the positive values. In the fourth and the fifth case we have either $\delta^k_l = 0$ or $\gamma^k_l = 0$. Then $\delta^k_l\gamma^k_l = 0$. Consequently, x^* is M-stationary. Now, we suppose that the sequence of stationary points of GNLP(t) satisfies the asymptotically weak nondegeneracy condition. This implies that for each $l \in (\mathcal{I}^-_{G^k} \setminus \mathcal{I}^+_{H^k}) \cap \mathcal{I}^*$ we have $\delta^k_l = -\eta^k_l(H_l(x^k) - \tau_{2,l}t^k) \geq 0, \quad \gamma^k_l = 0$ because $-\tau_{2,l}t^k \leq H_l(x^k) \leq \tau_{2,l}t^k$, and for each $l \in (\mathcal{I}^-_{H^k} \setminus \mathcal{I}^+_{G^k}) \cap \mathcal{I}^*$ we have $\delta^k_l = 0, \quad \gamma^k_l = -\eta^k_l(G_l(x^k) - \tau_{2,l}t^k) \geq 0$ because $-\tau_{2,l}t^k \leq G_l(x^k) \leq \tau_{2,l}t^k$. Then, the multipliers in the fourth and fifth case are positive, which implies that $\mu^*_{l,1} \geq 0$ and $\mu^*_{l,2} \geq 0$. Therefore, x^* is a B-stationary point for MPCC.

7. How to solve the regularization.

In this section, we briefly present a global algorithm for solving the regularization scheme, and we report some numerical experiments on the tests of the MacMPEC collection.

7.1. Algorithm and formulation details.

Our primary interest in this paper is to propose a regularization method that possesses strong convergence properties and enables us to compute a solution of the MPCC problem by solving a sequence of nonlinear programs. We note that the geometric shape of the approximation region $G_l(x) \geq -t, H_l(x) \geq -t, (G_l(x) - t)(H_l(x) - t) \leq 0$ is a union of two boxes that intersect at a point x which satisfies $G_l(x) = H_l(x) = t$; it would not be easy to apply the interior-point approach for solving the regularized problem NLP(t) especially if the initial point (x^0, t^0) is in one box and the solution is in the other one. In order to demonstrate that our approach is useful from a practical point of view, we present a general scheme of an algorithm which uses the active set method for solving the regularization.

The formulation of the algorithm, Figure 7.1, is general and the rule for updating the relaxation parameter t is not discussed yet. The classical update strategy is that we solve the problem NLP(t) for each fixed t. The process is then repeated as $t \searrow 0$. However, the question is how a sequence of parameters $\{t^k\}$ can be updated such that $t^k \searrow 0$. To this end, we solve a penalization approach based on the regularization NLP(t). By introducing the slack variables, the problem (1.1) can be equivalently

```
Algorithm 1: General Scheme of a Practical Algorithm for MPCCs

Initialization: Choose a positive sequence $\{t^k\} \longrightarrow 0$ and a stopping complementarity tolerance $\epsilon^{'}$. Let $x^0$ be an initial feasible point of NLP($t^0$) and set $k = 1$.
Repeat
    a). Choose stopping tolerance $\epsilon^k$ (stationarity tolerance);
    b). Apply the active set algorithm to approximately solve the problem NLP($t^k$),
        until the conditions (2.7) are satisfied for some $x^k$ with Lagrange multipliers
        $(\nu^k, \pi^k, \mu_1^k, \mu_2^k, \eta^k)$.
    c). Let $k \longleftarrow k + 1$,
Until the stopping criterion for the MPCC holds.
```

Fig. 7.1. *General description of the algorithm for solving* (1.4).

written in the form

$$
\begin{array}{ll}
\min & f(x) \\
\text{s.t.} & \\
& g(x) + s = 0, \quad s \geq 0, \\
& h(x) = 0, \\
& G_l(x) - y_{1,l} = 0, \quad H_l(x) - y_{2,l} = 0, \\
& y_{1,l} \geq 0, \quad y_{2,l} \geq 0, \\
& y_{1,l} y_{2,l} \leq 0, \\
& l = 1, 2, \ldots, n_c.
\end{array}
\tag{7.1}
$$

This problem has the same properties as (1.1). Instead of solving (7.1), we solve the following penalized problem:

$$
\begin{array}{ll}
\min & \Psi_{r,\rho}(x, y, s, t) = f(x) + \frac{1}{2r}\phi(x, y, s) + \rho t \\
\text{s.t.} & \\
& s \geq 0, \\
& y_{1,l} + t \geq 0, \quad y_{2,l} + t \geq 0, \\
& (y_{1,l} - t)(y_{2,l} - t) \leq 0, \\
& l = 1, 2, \ldots, n_c,
\end{array}
\tag{7.2}
$$

where $\phi(x, y, s) = \|(g(x) + s, h(x), G(x) - y_1, H(x) - y_2)\|^2$, with r and ρ are the positive penalty parameters. The process is then to decrease r to zero and to increase ρ to infinity, if necessary, which involves that the variable t decreases to zero. The difference between the elastic mode [2, 10] and our formulation is in the nature of the penalization and relaxation of some constraints. The strategy of the elastic mode as in [1, 2] is to move the complementarity terms $G_l(x)H_l(x) = 0, l = 1, 2 \ldots, n_c$ from the constraints to the objective function, by adding an l_1-penalty function and relax the other constraints as follows: $g(x) \leq \zeta e_{n_i}, \quad \zeta e_{n_i} \geq h(x) \geq -\zeta e_{n_i}$, where ζ is the elastic variable which is also penalized. Our penalty method is applied to a new regularization scheme which relaxes the complementarity constraints and keeps them explicit as constraints. With our technique, the sequence of first-order points (x^k, t^k) of the problem (1.4) has an accumulation point that is M-stationary under MPCC-LICQ and strongly stationary if the penalty parameter ρ^k is bounded. The similar convergence properties have been deduced with second-order solutions of the regularized problem in the case of the elastic mode [2].

To solve the problem (7.2) we apply an active set method with a more flexible updating of the penalty parameter ρ. The main steps performed by an active set

Algorithm 2: Active set regularization approach for MPCCs

Step 0.: Set $z^0 = (x^0, y^0, s^0, t^0)$ with $(x^0, y^0, s^0,) \in \mathbf{R^n} \times \mathbf{R^{2n_c}} \times \mathbf{R^{n_i}}$ and $t^0 \in \mathbf{R}^+$ an initial feasible point (y^0, s^0, t^0) of (7.2). Choose an initial penalty parameters $r^0 > 0, \rho^0 > 0$, constants $\sigma > 0, m \geq 1, \tau, \zeta \in (0,1), \kappa \in (0,1)$ and the stopping tolerances $\epsilon > 0, \epsilon^{'} > 0$. Let $j = 0, k = 1$.

Step 1. Starting from $(\tilde{x}^j, \tilde{y}^j, \tilde{s}^j, \tilde{t}^j) = (x^k, y^k, s^k, t^k)$, using the active set method to solve approximately the problem (7.2). If the iterate $(\tilde{x}^{k_j}, \tilde{y}^{k_j}, \tilde{s}^{k_j}, \tilde{t}^{k_j})$ satisfies the ϵ^k-stationary conditions of (7.2), go to the step 2. Otherwise,

If $\tilde{t}^{k_j} > \epsilon^k_{relax}$ and

$$\tilde{t}^{k_j} > \zeta \max\left\{\tilde{t}^{k_j - 1}, \ldots, \tilde{t}^{k_j - m}\right\}, \tag{7.3}$$

then set $\rho^k \leftarrow \sigma \rho^k$.

$(x^k, y^k, s^k, t^k) = (\tilde{x}^{k_j}, \tilde{y}^{k_j}, \tilde{s}^{k_j}, \tilde{t}^{k_j})$, $j \leftarrow j + 1$ and go to Step 1.

Step 2. Set $(x^{k+1}, y^{k+1}, s^{k+1}, t^{k+1}) = (\tilde{x}^{k_j}, \tilde{y}^{k_j}, \tilde{s}^{k_j}, \tilde{t}^{k_j})$. If $\epsilon^k < \epsilon$ and $\epsilon^k_{relax} < \epsilon^{'}$, stop; else set $r^{k+1} = \kappa r^k$, $\epsilon^{k+1}_{relax} = \min\left\{\phi^{k+1}, \tau r^{k+1}\right\}$ and

$$\rho^{k+1} = \begin{cases} \sigma \rho^k, & \text{if} \quad t^{k+1} \leq \epsilon^k_{relax}, \\ \rho^k, & \text{otherwise.} \end{cases}$$

Let $k \leftarrow k + 1$, and go to Step 1.

FIG. 7.2. *Description of the Active Set Algorithm for MPCC.*

method, the inner algorithm, are as follows: (i) Determine the set of the active constraints $\mathcal{W}$ and the active submatrix. (ii) Define a descent direction, (iii) if some relaxing rule allows it, relax one or more constraints leading to the computation of a new direction, and (iv) perform a line search along the direction. For our case, this algorithm is slightly modified to define $\mathcal{W}$. In particular, the relaxed complementarity constraint $(y_{1,l} - t)(y_{2,l} - t) \leq 0$ is considered active if one or both of its terms $y_{1,l} - t$ and $y_{2,l} - t$ vanish and neither is considered in the maximum steplength computation. Our updating strategy of the penalty parameter ρ is different from those used by the classical penalty method. Indeed, the traditional penalty method updates ρ only after the problem (7.2) is solved and the relaxation variable t is decreased sufficiently, while in our algorithm, Figure 7.2, we use the information on the current relaxation variable t and some previous iterations of the active set algorithm to update the penalty parameter ρ. This strategy is used in the context of the interior-penalty method in [20], and it is as follows: If the relaxation parameter t is relatively small according to some tolerance ϵ_{relax}, the penalty parameter ρ is not increased. Otherwise, we look back at some previous iterations and check whether the current relaxation variable is less than a fraction of the maximum value of the m previous iterations.

The analysis of the inner algorithm is beyond the scope of this paper. We assume that the active set algorithm is always successful and algorithm 2 is able to proceed to step 2 for each r^k. To ensure the global convergence of the active set method we used the constraint relaxation rule of Dembo and Sahi [7].

Next, we use the following notations to denote the active index set of the penalized problem (7.2):

$$\mathcal{I}_{s^k} = \left\{i : \ s^k_i = 0\right\}, \quad \mathcal{I}^{\pm}_{y^k_j} = \left\{l : \ y^k_{j,l} \pm t^k = 0\right\}, \quad \mathcal{I}_{y^k_j} = \left\{l : \ y^k_{j,l} = 0\right\}, \ j = 1, 2.$$

Now we show that as $k \to \infty$, the ϵ^k-stationary points of (7.2) converge to a strong stationary point of MPCC if the penalty parameter ρ^k is bounded.

Proposition 7.1. *Let $\{r^k\}$ and $\{\rho^k\}$ be, respectively, a decreasing and a nondecreasing penalty parameter sequences with k, and let (x^k, y^k, s^k, t^k) be a ϵ^k-stationary point of* (7.2) *for each (r^k, ρ^k), with $\epsilon^k \searrow 0$. Suppose that (x^*, y^*, s^*, t^*) is a cluster point of $\{(x^k, y^k, s^k, t^k)\}$ that is feasible for* (7.1)*. Assume that the MPCC-LICQ holds at x^* for* (7.1)*. Then,*

1. *x^* is an M-stationary point for* (1.1).
2. *If $\{\rho^k\}$ is bounded, then x^* is a strong stationary point for* (1.1).

Proof. The first part has been given by Theorem 4.2 but its proof will be slightly modified. Without loss of generality, we assume that $\{(x^k, y^k, s^k, t^k)\} \to (x^*, y^*, s^*, t^*)$ and $\{\epsilon^k\} \to 0$. Let $(\nu^k, \mu_1^k, \mu_2^k, \eta^k)$ be the Lagrange multipliers of (7.2) at (x^k, y^k, s^k, t^k) for given ρ^k and r^k. The ϵ^k-stationary conditions of (7.2) at (x^k, y^k, s^k, t^k) are the following:

$$\begin{aligned}\nabla_x f(x^k) = \; & \sum_{i=1}^{n_i} -\frac{g_i(x^k)+s_i^k}{r^k}\nabla_x g_i(x^k) + \sum_{j=1}^{n_e} -\frac{h_j(x^k)}{r^k}\nabla_x h_j(x^k) \\ & -\sum_{l=1}^{n_c}\frac{G_l(x^k)-y_{1,l}^k}{r^k}\nabla_x G_l(x^k) - \sum_{l=1}^{n_c}\frac{H_l(x^k)-y_{2,l}^k}{r^k}\nabla_x H_l(x^k) + O(\epsilon^k),\end{aligned} \tag{7.4}$$

$$\rho^k = \sum_{l\in\mathcal{I}^+_{y_1^k}} \mu_{1,l}^k + \sum_{l\in\mathcal{I}^+_{y_2^k}} \mu_{2,l}^k + \sum_{l\in\mathcal{I}^-_{y_1^k}} \eta_l^k(y_{2,l}^k - t^k) + \sum_{l\in\mathcal{I}^-_{y_2^k}} \eta_l^k(y_{1,l}^k - t^k) + O(\epsilon^k), \tag{7.5}$$

$$\begin{aligned}\mu_{1,l}^k &= \frac{y_{1,l}^k - G_l(x^k)}{r^k} + O(\epsilon^k), \quad l \in \mathcal{I}^+_{y_1^k},\\ \mu_{2,l}^k &= \frac{y_{2,l}^k - H_l(x^k)}{r^k} + O(\epsilon^k), \quad l \in \mathcal{I}^+_{y_2^k},\\ -\eta_l^k(y_{2,l}^k - t^k) &= \frac{y_{1,l}^k - G_l(x^k)}{r^k} + O(\epsilon^k), \quad l \in \mathcal{I}^-_{y_1^k},\\ -\eta_l^k(y_{1,l}^k - t^k) &= \frac{y_{2,l}^k - H_l(x^k)}{r^k} + O(\epsilon^k), \quad l \in \mathcal{I}^-_{y_2^k},\\ \nu_i^k &= \frac{g_i(x^k)+s_i^k}{r^k} + O(\epsilon^k), \qquad i \in \mathcal{I}_{s^k}.\end{aligned} \tag{7.6}$$

By the same reasoning as in the proof of Theorem 4.2, the multipliers ν^k, δ^k, γ^k converge, respectively, to the unique MPCC multipliers ν^*, δ^*, γ^* of the problem (7.1) at (x^*, y^*, s^*, t^*), and we also have

$$\begin{aligned}\lim_{k\in\mathcal{K}\subseteq\mathbf{N}} \frac{g_i(x^k)+s_i^k}{r^k} + O(\epsilon^k) = \; & \nu_i^*, & \lim_{k\in\mathcal{K}\subseteq\mathbf{N}} \frac{h_j(x^k)}{r^k} + O(\epsilon^k) = \; & \pi_j^*,\\ \lim_{k\in\mathcal{K}\subseteq\mathbf{N}} \frac{y_{1,l}^k - G_l(x^k)}{r^k} + O(\epsilon^k) = \; & \delta_l^*, & \lim_{k\in\mathcal{K}\subseteq\mathbf{N}} \frac{y_{2,l}^k - H_l(x^k)}{r^k} + O(\epsilon^k) = \; & \gamma_l^*.\end{aligned} \tag{7.7}$$

Thus, the limit point (x^*, y^*, s^*, t^*) is M-stationary for the problem (7.1). By the feasibility assumption, we have $t^* = 0$ and $\phi(x^*, y^*, s^*) = 0$, which implies that $\mathcal{I}_{y_1^*} = \mathcal{I}_{G^*}, \mathcal{I}_{y_2^*} = \mathcal{I}_{H^*}$, and $\mathcal{I}_{s^*} = \mathcal{I}_{g^*}$. Therefore, x^* is M-stationary point for the problem (1.1).

2) Suppose that $\{\rho^k\}$ is bounded, then there is a $\hat{k}$ such that $\rho^k = \rho^{\hat{k}}$ and $t^{\hat{k}} = 0$ for all $k \geq \hat{k}$. Hence, for all $k \geq \hat{k}$, we have $\mathcal{I}^+_{y_1^k} = \mathcal{I}^-_{y_1^k}$ and $\mathcal{I}^+_{y_2^k} = \mathcal{I}^-_{y_2^k}$, which implies

Table 7.1
Solutions and optimal values of the problem (1.2).

	ρ	$\rho^1 = 0.25$	$\rho^2 = 0.5$	$\rho^* = 1$
	(z^0, t^0) f^0	(z^1, t^1) f^1	(z^2, t^2) f^2	(z^*, t^*) f^*
case 1	$(-0.5, 0.5, 0.5)$ 1.25	$(0.69, 0.98, 0.69)$ 0.04653	$(0.52, 1.00, 0.52)$ 0.11459	$(0.00, 0.99, 0.00)$ 0.5
case 2	$(-0.3, 0.5, 0.5)$ 0.97	$(0.69, 0.98, 0.69)$ 0.04658	$(0.52, 1.00, 0.52)$ 0.11358	$(0.00, 0.99, 0.00)$ 0.5
case 3	$(0.4, 2, 0.5)$ 0.68	$(0.67, 0.98, 0.67)$ 0.05297	$(0.51, 1.02, 0.51)$ 0.12201	$(0.00, 0.99, 0.00)$ 0.5
case 4	$(0.5, 0.5, 0.5)$ 0.25	$(0.64, 1.06, 0.64)$ 0.06652	$(0.48, 0.92, 0.48)$ 0.13814	$(0.00, 0.99, 0.00)$ 0.5
case 5	$(0.5, -0.5, 0.5)$ 1.25	$(0.98, 0.69, 0.69)$ 0.04653	$(1.00, 0.52, 0.52)$ 0.11459	$(0.99, 0.00, 0.00)$ 0.5
case 6	$(1, 0.5, 0.5)$ 0.125	$(1.06, 0.65, 0.65)$ 0.05908	$(1.00, 0.49, 0.49)$ 0.12893	$(0.00, 0.99, 0.00)$ 0.5

$\delta_l^k = \mu_{1,l}^k - \eta_l^k y_{2,k}$ and $\gamma_l^k = \mu_{2,l}^k - \eta_l^k y_{1,k}$. Equation (7.5) can be written as follows:

$$\rho^k - \Big(\sum_{l\in\mathcal{I}_{y_1^k}} \delta_l^k + \sum_{l\in\mathcal{I}_{y_2^k}} \gamma_l^k\Big) = 2\Big(\sum_{l\in\mathcal{I}_{y_1^k}} \eta_l^k y_{2,l}^k + \sum_{l\in\mathcal{I}_{y_2^k}} \eta_l^k y_{1,l}^k\Big) + O(\epsilon^k). \tag{7.8}$$

Then, the first part of the equation (7.8) will be bounded as $k \to +\infty$, because $\{\rho^k\}$ is bounded and $(\nu^*, \delta^*, \gamma^*)$ is bounded by MPCC-LICQ assumption. This implies that the second part of (7.8) is bounded, for $k \geq \hat{k}$ sufficiently large. Since $t^k = 0$ then, for $k \geq \hat{k}$ sufficiently large, we have $\eta_l^k y_{2,l}^k = \eta_l^k y_{1,l}^k = 0$, for all $l \in \mathcal{I}_{y_1^k} \cap \mathcal{I}_{y_2^k}$. Hence, for all $k \geq \hat{k}$ sufficiently large, $\delta_l^k = \mu_{1,l}^k \geq 0$ and $\gamma_l^k = \mu_{2,l}^k \geq 0$ with $l \in \mathcal{I}_{y_1^k} \cap \mathcal{I}_{y_2^k}$. Thus, we have $\delta_l^k \to \delta_l^* \geq 0$ and $\gamma_l^k \to \gamma_l^* \geq 0$ for any $l \in \mathcal{I}_{y_1^*} \cap \mathcal{I}_{y_2^*}$, so $(x^*, y_1^*, y_2^*, s^*, 0)$ is strongly stationary for the problem (7.2). By the feasibility assumption, the point x^* is strongly stationary for (1.1). □

We note that the cluster point of the stationary points generated by the regularization method is feasible for the original problem (7.1), but this is not automatically true for the penalty method (7.2). This seems unavoidable for exterior penalty methods. A common assumption used to avoid infeasible cluster points is related to coercivity conditions on the underlying functions.

7.2. Numerical illustrations. We wrote an experimental code implementing Algorithm 2. No attempt to refine or optimize the code was made. The purpose of this section is not to test the code, but to validate our theoretical convergence results. Nevertheless, several problems from MacMPEC collection [19] are solved; see Table 7.2 below. We first apply the algorithm to Example (1.2) with different values (positions) of the starting point, and we show that, unlike the smooth regularizations [8, 21, 28], our regularization is not attracted by the C-stationary point. The initial parameters in algorithm are selected as $\rho^1 = 0.25$, $\sigma = 2$, and $\epsilon^k = 2^{-k} \times 10^{-1}$. The optimal solutions are $(1, 0)$ and $(0, 1)$. Our algorithm solves this problem successfully at $\rho^* = 1$ and does not make the change in the working set in cases 4 and 6, while in the other cases there is only one change as illustrated in Table 7.1. (z^k, t^k) is the point (x, y, s, t) at the kth iterate and f^k the value of the objective function at this point.

These results confirm that the algorithm converges to a strong stationary point of MPCC (1.2). For more illustrations, we chose some problems from MacMPEC collec-

Table 7.2
Numerical results on some problems from MacMPEC collection.

Name	n	n_i	n_e	p	f^*	ϕ^*	ρ^*	nbW
bard1	5	8	1	3	24.999	2.07e-10	64	9
bard3	6	8	3	1	−18.685	1.38e-06	128	7
bilevel3	11	13	6	3	−12.6842	0.28e-06	8	13
dempe	3	2	1	1	31.249	0.14e-06	8	3
design-c-1	12	9	6	3	1.8605	0.10e-07	8	8
design-c-4	22	23	10	8	0.17e-04	0.10e-07	64	9
desilva	6	8	2	2	−1.000	0.20e-07	2	6
ex9.2.1	10	14	5	4	24.98265	0.22e-06	32	17
ex9.2.2	9	14	4	3	99.98847	0.70e-06	16	18
ex9.2.3	14	21	8	4	4.99579	0.40e-05	16	22
ex9.2.4	8	9	5	2	0.49750	0.30e-06	2	9
ex9.2.5	8	11	4	3	4.96044	0.40e-04	4	9
ex9.2.6	16	22	6	6	−1.25251	0.30e-05	4	28
ex9.2.7	10	14	5	4	24.82729	0.80e-04	32	17
ex9.2.8	6	9	3	2	1.49951	0.90e-06	2	14
ex9.2.9	9	13	5	3	1.99583	0.11e-04	2	13
gnash10	13	26	4	8	−230.8742	0.99e-05	128	65
gnash11	13	26	4	8	−129.9389	0.529e-06	256	69
gnash12	13	26	4	8	−36.9331	0.10e-07	256	32
gnash13	13	26	4	8	−7.0629	0.24e-06	256	49
jr1	2	2	0	1	0.48198	0.33e-05	2	2
jr2	2	2	0	1	0.49726	0.74e-05	2	2
kth1	2	3	0	1	0.0	0.10e-15	2	1
kth2	2	3	0	1	0.0	0.10e-15	2	1
kth3	2	3	0	1	0.5	0.10e-15	2	1
nash1	6	8	2	2	0.00	0.10e-07	16	4
outrata31	5	10	0	4	3.2064	0.38e-06	4	13
outrata32	5	10	0	4	3.4439	0.68e-05	8	10
portfl-i-1	87	98	13	12	0.69e-04	0.69e-04	4096	36
qpec-100-1	105	202	0	100	0.1222	0.26e-04	256	105
qpec-100-4	120	204	0	100	−3.189	0.12e-04	256	215
qpec-200-2	220	404	0	200	−24.307	6.22e-04	160	208
ralph2	2	3	0	1	0.0	0.10e-15	2	1
scholtes1	3	3	0	1	2.00	0.171e-27	2	3
scholtes2	3	3	0	1	14.98522	0.61e-07	4	1
scholtes3	2	3	0	1	0.0	0.10e-15	4	1
scholtes4	3	5	0	1	1	0.12e-04	4096	5
scholtes5	3	5	0	1	1.0	0.10e-15	4	4
stackelberg	3	5	1	1	−3267.37490	0.90e-04	8	4

tion [19]; these problems are transformed in the form (7.2) and coded in Scilab [27]. Table 7.2 summarizes the results, where n, n_i, n_e, and p are the numbers of variables, general inequality constraints, general equality constraints, and complementarity constraints, respectively. f^* is the optimal value of the objective function at the solution. ϕ^* measures the feasibility residual in the optimal point (x^*, y^*, s^*, t^*). ρ^* is the value of the penalty parameter when the algorithm terminates; we start with $\rho^0 = 1$. The last column, nbW, indicates the number of working set changes.

We note from Table 7.2 that algorithm 2 does well on almost all problems of relatively small and medium sizes. In particular, algorithm 2 converges to an approximate strong stationary point of MPCC for all the solved problems except $ex9.2.2$ and $Scholtes4$ which do not possess a strong stationary point. The algorithm has trouble with problem $ex9.2.2$, and it was unable to reach the accuracy. These first numerical results are interesting since they confirm our theoretical convergence properties and show that the algorithm based on the active set approach seems a relevant

Table 7.3
Solutions and optimal values of the problem (7.9).

iteration	(x_1, x_2, x_3, t)	PR	ϕ	$\Psi_{r,\rho}$	$max\lambda$	ρ
1	$(-0.86, 0.99, 0.00, 0.86)$	0.68000	0.15e-04	0.223	0.55	0.1
2	$(-0.99, 0.99, 0.00, 0.99)$	0.62e-04	0.10e-05	0.4	1.00	0.4
3	$(-0.99, 0.99, 0.00, 0.99)$	0.62e-03	0.10e-04	0.8	1.00	0.8
4	$(-0.00, 0.99, 0.00, 0.00)$	0.40e-01	0.40e-06	0.99	26.1	51.2
5	$(-0.00, 0.99, 0.00, 0.00)$	0.10e-07	0.10e-08	0.99	26.1	51.2

way to overcome the absence of constraint qualification at degenerate points of the regularized problem.

7.3. Discussion. In this subsection, we discuss the active set method applied to the original formulation of MPCC and the proposed regularization, and we explain that the method does not encounter the difficulty with the degenerate points of our regularization. We also discuss briefly the active set method related to the decomposition method.

The nonregularity of the feasible points of MPCC (1.1) has two major drawbacks on the direct application of the active set method. On the one hand, the set of the Lagrange multipliers may be empty, and on the other hand, the linearization of the constraints can be inconsistent arbitrarily close to a stationary point as the following example illustrates:

$$\begin{array}{ll} \text{min} & x_1 + x_2 \\ \text{s.t.} & \\ & x_2^2 - 1 \geq 0, \\ & 0 \leq x_1 \perp x_2 \geq 0. \end{array} \tag{7.9}$$

Its solution is $x^* = (0, 1)$ with NLP multipliers $\nu^* = 0.5$ of $x_2^2 - 1 \geq 0$, $\mu_1^* = 1$ of $x_1 \geq 0$, and $\eta^* = 0$ of $x_1 x_2 \leq 0$. This solution is a strongly stationary point of (7.9). The linearization of the constraints about a point $x^0 = (\epsilon_1, 1 - \epsilon_2)$, with $\epsilon_1, \epsilon_2 > 0$, close to the solution x^* gives a quadratic program (QP) that is inconsistent, see [10]. To overcome this difficulty the authors modified the pure SQP method by including a restoration phase that can be invoked if QP is inconsistent. This procedure allows one to find a next iterate x^{next} with $x_1^{next} x_2^{next} = 0$, but the convergence results established are of a local nature and the global convergence is not fully explored in [10]. By starting with the initial point $(x_1, x_2, s, t) = (1, 0.5, 1, 0.5)$, Algorithm 2 finds the optimal solution of (7.9) in four iterations. Table 7.3 summarizes the results given by the algorithm where the notations PR and $max\lambda$, respectively, mean the primal residual and the maximum of the Lagrange multipliers.

The global convergence of SQP algorithm with elastic mode has been studied in [2]. It was shown that the generated sequence has an accumulation point which is C-stationary under MPCC-LICQ and M-stationary point if the generated sequence is a sequence of inexact second-order points. In contrast, without second-order assumptions, the sequence generated by our approach has a cluster point which is M-stationary under the MPCC-LICQ and strongly stationary if the penalty parameter ρ is bounded.

In contrast to the MPCC formulation, we showed that the Lagrange multipliers exist at any feasible point $z = (x, y, s, t)$ of the regularization scheme (1.5) and they are unique for all feasible points z such that $(y, t) \notin \mathcal{D}_y = \{(y, t) | \exists l : y_{i,l} - t = 0, i = 1, 2\}$. Therefore, the active set method behaved well with the proposed regularization. In the degeneracy case, $\mathcal{D}_y \neq \emptyset$, the set of the Lagrange multipliers corresponding to the points z such that $(y,t) \in \mathcal{D}_y$ is unbounded, but this lack of regularity cannot create problems when applying the active set method. Indeed, let z be a stationary point for the problem (7.2) such that $(y,t) \in \mathcal{D}_y$, and we assume without loss of generality that there is a unique l_0 such that $y_{1,l_0} - t = y_{2,l_0} - t = 0$. In this case we have $\mu_{1,l_0} = \mu_{2,l_0} = 0$ and regardless the value of the η_{l_0} the quantities $\eta_{l_0}(y_{1,l_0} - t)$ and $\eta_{l_0}(y_{2,l_0} - t)$ vanish. Therefore, equations (7.4) and (7.5) become as follows:

$$\begin{aligned}
\nabla_x f(x) = & \sum_{i=1}^{n_i} -\frac{g_i(x)+s_i}{r}\nabla_x g_i(x) + \sum_{j=1}^{n_e} -\frac{h_j(x)}{r}\nabla_x h_j(x) \\
& - \sum_{l=1, l\neq l_0}^{n_c} \frac{G_l(x)-y_{1,l}}{r}\nabla_x G_l(x) - \frac{H_l(x)-y_{2,l}}{r}\nabla_x H_l(x) + O(\epsilon), \\
\rho = \sum_{l\in\mathcal{I}_{y_1}^+} \mu_{1,l} + & \sum_{l\in\mathcal{I}_{y_2}^+} \mu_{2,l} + \sum_{l\in\mathcal{I}_{y_1}^-\setminus\{l_0\}} \eta_l(y_{2,l}-t) + \sum_{l\in\mathcal{I}_{y_2}^-\setminus\{l_0\}} \eta_l(y_{1,l}-t) + O(\epsilon).
\end{aligned}$$

Thus, the relaxed constraints related to the complementarity constraint l_0 do not appear in the stationarity conditions of the penalized problem (7.2). This means that the point z, with $y_{1,l_0} - t = y_{2,l_0} - t = 0$, is a nondegenerate stationary point for a relaxed problem

$$\tag{7.10}
\begin{array}{ll}
\min & \Psi_{r,\rho}(x,y,s,t) = f(x) + \frac{1}{2r}\phi(x,y,s) + \rho t \\
\text{s.t.} & \\
& s \geq 0, \\
& y_{1,l} + t \geq 0, \quad y_{2,l} + t \geq 0, \\
& (y_{1,l}-t)(y_{2,l}-t) \leq 0, \\
& l \in \{1,2,\ldots,n_c\} \setminus \{l_0\}.
\end{array}$$

Consequently, the degenerate case of the penalization (7.2) can be treated as a nondegenerate case for the problem (7.10) which is a relaxed problem of (7.2). We note that in all problems tested from MacMPEC collection [19], we did not encounter such a case. We conjecture that this case may occur only for some pathological cases of the MPCC problems. On the other hand, the inconsistency difficulty for solving the regularization scheme (1.5) does not appear for the active set method and the problem (7.9) is successfully solved.

Another aspect of the active set for MPCC is related to the idea of identifying active constraints in inequality constrained optimization. It aims to specify two subsets $\mathcal{I}_G$ and $\mathcal{I}_H$ with $\mathcal{I}_G \cup \mathcal{I}_H = \{1,2,\ldots,n_c\}$ to decompose the complementarity constraints according to $\mathcal{I}_G$ and $\mathcal{I}_H$, and to compute a stationary point to the subproblem:

$$\begin{array}{lllll}
\min & f(x) & & & \\
\text{s.t.} & & & & \\
& g(x) \leq 0, & h(x) = 0, & & \\
& G_l(x) = 0, & l \in \mathcal{I}_G, & G_l(x) \geq 0, & l \in \mathcal{I}_H \setminus \mathcal{I}_G, \\
& H_l(x) = 0, & l \in \mathcal{I}_H, & H_l(x) \geq 0, & l \in \mathcal{I}_G \setminus \mathcal{I}_H.
\end{array}$$

However, the important question is how to choose $\mathcal{I}_G$ and $\mathcal{I}_H$ effectively, especially when the iterates are far from the solution. The linear case, of course, is very special because correct identification is easier and can often be obtained (relatively) far from a solution, but in the nonlinear case, correct identification is in general quite local. Recent developments in decomposition methods for the linear case show that the global convergence to a M-stationary point can be guaranteed by assuming MPCC-LICQ everywhere (uniform MPCC-LICQ) [13], while the convergence to B-stationary point can be achieved under an additional assumption that every M-stationary point is isolated [14]. We note that our regularization approach is not based on the decomposition method.

8. Conclusion. In summary, the feasible domain of an MPCC is thin (no interior) and nonconvex-nonsmooth. Usual regularization both smoothen and thicken the domain. Smoothing introduces spurious solutions. We proposed a thick, but still nonconvex-nonsmooth regularization that avoids such spurious solutions. Despite its nonsmooth-nonconvex nature, we proved that our regularized problems possess KKT multipliers, and regularized solutions are shown to converge to strong stationary points under weaker assumptions than other published regularizations, in particular without using any second order optimality condition. The proposed regularization has neither the inconsistency difficulty nor the decomposition problem and it is useful from a practical standpoint. The numerical experiments confirm the convergence results and demonstrate that the active set approach is an adequate method for solving this regularization.

Acknowledgments. The authors are grateful to the anonymous referees and the Associate Editor, Masao Fukushima, for their careful reading of the manuscript. The paper has benefited from their suggestions and detailed comments.

REFERENCES

[1] M. Anitescu, *On using the elastic mode in nonlinear programming approaches to mathematical programming with complementarity constraints*, SIAM J. Optim., 15 (2005), pp. 1203–1236.

[2] M. Anitescu, P. Tseng, and S. J. Wright, *Elastic-mode algorithms for mathematical programs with equilibrium constraints: Global convergence and stationarity properties*, Math. Program., 110 (2007), pp. 337–371.

[3] J. F. Bard, *Convex two level optimization*, Math. Program., 40 (1988), pp. 15–27.

[4] H. Y. Benson, A. Sen, D. F. Shanno, and R. J. Vanderbei, *Interior-point algorithm, penalty methods and equilibrium problems*, Comput. Optim. Appl., 34 (2006), pp. 155–182.

[5] X. Chen and M. Florian, *The nonlinear bilevel programming problem: Formulations, regularity and optimality conditions*, Optimization, 32 (1995), pp. 193–209.

[6] X. Chen and M. Fukushima, *A smoothing method for mathematical programming with p-matrix linear complementarity constraints*, Comput. Optim. Appl., 27 (2004), pp. 223–246.

[7] R. S. Dembo and S. Sahi, *A convergence framework for constrained nonlinear optimization*, School of Organization and Management Working Paper, Series B #69, Yale University, New Haven, CT, 1983.

[8] A. V. Demiguel, M. P. Friedlander, F. J Nogales, and S. Scholtes, *A two-sided relaxation scheme for mathematical programs with equilibrium constraints*, SIAM J. Optim., 16 (2005), pp. 587–609.

[9] F. Facchinei, H. Jiang, and L. Qi, *A smoothing method for mathematical programming with equilibrium constraints*, Math. Program., 85 (1999), pp. 107–134.

[10] R. Fletcher, S. Leyffer, D. Ralph, and S. Scholtes, *Local convergence of SQP methods for mathematical programs with equilibrium constraints*, J. Optim., 17 (2006), pp. 259–286.

[11] M. Fukushima, Z. Q. Luo, and J. S. Pang, *A globally convergence sequential quadratic programming algorithm for mathematical programming with linear complementarity constraints*, Comput. Optim. Appl., 10 (1998), pp. 5–34.

[12] M. Fukushima and J. S. Pang, *Convergence of a smoothing continuation method for mathematical programming with complementarity constraints*, Ill-posed Variational Problems and Regularization Technique, Lecture Notes in Economics and Mathematical Systems 477, M. Théra and R. Tichatschke, eds., Springer-Verlag, Berlin/Heidelberg, 1999, pp. 99–110.

[13] M. Fukushima and P. Tseng, *An implementable active-set algorithm for computing a B-stationary point of a mathematical programming with complementarity constraints*, SIAM J. Optim., 12 (2002), pp. 724–739.

[14] M. Fukushima and P. Tseng, *An implementable active-set algorithm for computing a B-stationary point of a mathematical programming with complementarity constraints, ERRATUM*, SIAM J. Optim., 17 (2007), pp. 1253–1257.

[15] X. Hu and D. Ralph, *Convergence of a penalty method for mathematical programming with equilibrium constraints*, J. Optim. Theory Appl., 123 (2004), pp. 365–390.

[16] X. X. Huang, X. Q. Yang, and D. L. Zhu, *A sequential smooth penalization approach to mathematical programming with complementarity constraints*, Numer. Funct. Anal. Optim., 27 (2006), pp. 71–98.

[17] H. Jiang and D. Ralph, *Smooth sequential quadratic programming methods for mathematical programming with linear complementarity constraints*, SIAM J. Optim., 10 (2000), pp. 779–808.

[18] C. Kanzow, *Some noninterior continuation methods for linear complementarity problems*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 851–868.

[19] S. Leyffer, *MacMPEC: AMPL collection of MPECs*, web page, www.mcs.anl.gov/~leyffer/MacMPEC/, 2000.

[20] S. Leyffer, G. Lopez-Calva, and J. Nocedal, *Interior methods for mathematical programming with equilibrium constraints*, SIAM J. Optim., 17 (2006), pp. 52–77.

[21] G. H. Lin and M. Fukushima, *A modified relaxation scheme for mathematical programs with complementarity constraints*, Ann. Oper. Res., 133 (2005), pp. 63–84.

[22] X. Liu and J. Sun, *Generalized stationary points and an interior point method for mathematical programs with equilibrium constraints*, Math. Program., 101 (2004), pp. 231–261.

[23] Z. Q. Luo, J. S. Pang, and D. Ralph, *Mathematical Programming with Equilibrium Constraints*, Cambridge University Press, Cambridge, UK, 1996.

[24] O. L. Mangasarian, *Nonlinear Programming*, McGraw-Hill, New York, 1969.

[25] D. Ralph and S. J. Wright, *Some properties of regularization and penalization schemes for MPECs*, Optim. Methods Softw., 19 (2004), pp. 527–556.

[26] H. Scheel and S. Scholtes, *Mathematical programming with complementarity constraints: Stationarity, optimality and sensitivity*, Math. Oper. Res., 25 (2000), pp. 1–22.

[27] Scilab group, *Scilab* 4.1.2, web page, http://www.scilab.org, 2008.

[28] S. Scholtes, *Convergence properties of a regularization scheme for mathematical programming with complementarity constraints*, SIAM J. Optim., 11 (2001), pp. 918–936.

SIAM J. Optim.
Vol. 20, No. 1, pp. 104–129

STABILITY ANALYSIS OF OPTIMAL CONTROL PROBLEMS WITH A SECOND-ORDER STATE CONSTRAINT[*]

AUDREY HERMANT[†]

Abstract. This paper gives stability results for nonlinear optimal control problems subject to a regular state constraint of second-order. The strengthened Legendre–Clebsch condition is assumed to hold, and no assumption on the structure of the contact set is made. Under a weak second-order sufficient condition (taking into account the active constraints), we show that the solutions are Lipschitz continuous w.r.t. the perturbation parameter in the L^2 norm, and Hölder continuous in the L^∞ norm. We use a generalized implicit function theorem in metric spaces by Dontchev and Hager [*SIAM J. Control Optim.*, 36 (1998), pp. 698–718]. The difficulty is that multipliers associated with second-order state constraints have a low regularity (they are only bounded measures). We obtain Lipschitz stability of a "primitive" of the state constraint multiplier.

Key words. optimal control, second-order state constraint, stability analysis, alternative formulation, sufficient second-order optimality condition, uniform quadratic growth, strong regularity

AMS subject classifications. 49K40, 34H05, 90C31, 49K15

DOI. 10.1137/070707993

1. Introduction.

This paper deals with stability analysis of nonlinear optimal control problems of an ordinary differential equation with a second-order state constraint. State constraints of second-order occur naturally in applications: For example, in the problem of the atmospheric re-entry of a space shuttle, with the bank angle as control, the constraints on the thermal flux, normal acceleration, and dynamic pressure are second-order state constraints; see [7]. Stability and sensitivity analysis of solutions of optimal control problems is of high interest for the study of numerical methods, such as, e.g., continuation algorithms (see [4]), and to analyze the convergence of discretization schemes and obtain errors estimates (see, e.g., [10]).

For a class of general constrained optimization problems in Banach spaces, when the derivative of the constraint is "onto" and a second-order sufficient condition holds, Lipschitz stability of solutions and multipliers can be obtained by application of Robinson's strong regularity theory [27] to the first-order optimality system. For optimal control problems, this theory does not apply because of the well-known *two-norm discrepancy* (see [24]). Stability results for optimal control problems using variants of Robinson's strong regularity in order to deal with the two-norm approach have been obtained in [8, 17, 11] for control constraints, and [19] for mixed control-state constraints.

Lipschitz stability results for state constraints of first-order have been obtained by Malanowski [18] and Dontchev and Hager [9]. The difficulty of pure state constraints is the low regularity of multipliers, which are bounded Borel measures. These multipliers can be identified with functions of bounded variation, and for first-order state constraints, it is known that, under standard hypothesis, they are more regular (they are Lipschitz continuous functions; see Hager [14]). This additional regularity of multipliers is strongly used in the analysis in [18] and [9]. In those two papers, strong

[*] Received by the editors November 12, 2007; accepted for publication (in revised form) September 4, 2008; published electronically March 25, 2009.
http://www.siam.org/journals/siopt/20-1/70799.html

[†] CMAP, École Polytechnique, INRIA Saclay Île-de-France, Route de Saclay, 91128 Palaiseau, France (hermant@cmap.polytechnique.fr).

second-order sufficient conditions were used (which do not take into account the active constraints). The sufficient condition was recently weakened by Malanowski [21, 20].

For higher-order state constraints, the multipliers associated with the state constraints are only measures, and are not continuous w.r.t. the perturbation parameter (for the total variation norm). For this reason, the frameworks of [18] or [9] are not directly applicable. The only stability and sensitivity results known for state constraints of higher-order are based on the shooting approach; see Malanowski and Maurer [22] and Bonnans and Hermant [5]. Such results require strong assumptions on the structure of the contact set.

The main result of this paper is a stability result for regular second-order state constraints, with no assumption on the structure of the contact set. The control is assumed to be continuous and the strengthened Legendre–Clebsch condition to hold. We use a generalized implicit function theorem in metric spaces by Dontchev and Hager [9], applied to a system equivalent to the first-order optimality condition (the *alternative formulation*). This formulation involves *alternative multipliers* that are "integrals" of the original state constraint multipliers, and therefore are more regular. We obtain Lipschitz continuity of solutions and alternative multipliers in the L^2 norm, and Hölder continuity in the L^∞ norm, under a weak second-order sufficient condition taking into account the active constraints.

This paper is organized as follows. In section 2, the problem, optimality conditions, assumptions, and the admissible class of perturbations are introduced. In section 3, the second-order sufficient optimality condition is presented. In section 4, the main stability results for the nonlinear optimal control problem are given. Section 5 is devoted to stability analysis of linear-quadratic problems, which is used to prove the main theorem in section 6. Finally, the conclusion and comments are given in section 7.

2. Preliminaries.

We consider the following optimal control problem:

$$(2.1) \qquad (\mathcal{P}) \qquad \min_{(u,y)\in\mathcal{U}\times\mathcal{Y}} \int_0^T \ell(u(t),y(t))\mathrm{d}t + \phi(y(T))$$

$$(2.2) \quad \text{subject to (s.t.)} \qquad \dot{y}(t) \;=\; f(u(t),y(t)) \quad \text{for a.a. } t\in[0,T], \quad y(0)=y_0,$$

$$(2.3) \qquad g(y(t)) \;\leq\; 0 \quad \forall t\in[0,T]$$

with the control and state spaces $\mathcal{U} := L^\infty(0,T;\mathbb{R}^m)$ and $\mathcal{Y} := W^{1,\infty}(0,T;\mathbb{R}^n)$. The following assumptions are assumed to hold throughout this paper and will not be repeated in its various results.

(A0) The data $\ell : \mathbb{R}^m\times\mathbb{R}^n \to \mathbb{R}$, $\phi : \mathbb{R}^n \to \mathbb{R}$ (resp., $f : \mathbb{R}^m\times\mathbb{R}^n \to \mathbb{R}^n$, $g : \mathbb{R}^n \to \mathbb{R}$) are C^2 (resp., C^3, C^4) mappings, with locally Lipschitz continuous second-order (resp., third-order, fourth-order) derivatives, and f is Lipschitz continuous.

(A1) The initial condition $y_0 \in \mathbb{R}^n$ satisfies $g(y_0) < 0$.

We consider in this paper state constraints of *second-order*. This means that the first-order time derivative $g^{(1)} : \mathbb{R}^m\times\mathbb{R}^n \to \mathbb{R}$ of the constraint, defined by

$$g^{(1)}(u,y) := g_y(y)f(u,y),$$

does not depend on the control variable u, i.e., $g^{(1)}_u \equiv 0$ (hence, we write $g^{(1)}(y) = g^{(1)}(u,y)$), and the second-order time derivative $g^{(2)} : \mathbb{R}^m\times\mathbb{R}^n \to \mathbb{R}$, defined by

$$g^{(2)}(u,y) := g^{(1)}_y(y)f(u,y),$$

depends explicitly on the control, i.e., $g_u^{(2)} \not\equiv 0$.

Remark 2.1. For linear-quadratic control problems of type (5.1)–(5.4) (see section 5), with dynamics given by $\dot{z}(t) = A(t)z(t) + B(t)v(t)$ and state constraint by $C(t)z(t) + d(t) \leq 0$, the state constraint is of second-order means that $C(t)B(t) \equiv 0$ on $[0,T]$ and $(\dot{C}(t) + C(t)A(t))B(t) \not\equiv 0$.

Remark 2.2. In this paper the state constraint is assumed to be scalar-valued for simplicity. The results are directly generalizable to several state constraints $g_1, \dots, g_r$ of second-order (and even of *higher-order* [23, 15] $q_i \geq 2$ for $i = 1, \dots, r$; see Remark 2.3 further) under the assumption (see [23, 3]) that the gradients of the nearly active constraints $\nabla_u g_i^{(q_i)}(u,y)$ are uniformly linearly independent along the trajectory.

Notation. We denote by subscripts Fréchet derivatives w.r.t. the variables u, y, i.e., $f_y(u,y) = D_y f(u,y)$, $f_{yy}(u,y) = D^2_{yy} f(u,y)$, etc. The derivative with respect to the time is denoted by a dot, i.e., $\dot{y} = \frac{\mathrm{d}y}{\mathrm{d}t} = y^{(1)}$. The set of row vectors of dimension n is denoted by $\mathbb{R}^{n*}$. Adjoint or transpose operators are denoted by the symbol $^\top$. The Euclidean norm is denoted by $|\cdot|$. By $L^r(0,T)$ we denote the Lebesgue space of measurable functions such that $\|u\|_r := (\int_0^T |u(t)|^r \mathrm{d}t)^{1/r} < \infty$ for $1 \leq r < \infty$, $\|u\|_\infty := \mathrm{supess}_{[0,T]} |u(t)| < \infty$. The space $W^{s,r}(0,T)$ denotes the Sobolev space of functions having their s first weak derivatives in $L^r(0,T)$, with the norm $\|u\|_{s,r} := \sum_{j=0}^s \|u^{(j)}\|_r$. We denote by H^s the space $W^{s,2}$. The space of continuous functions over $[0,T]$ and its dual space, the space of bounded Borel measures, are denoted, respectively, by $C[0,T]$ and $\mathcal{M}[0,T]$. The set of nonnegative measures is denoted by $\mathcal{M}_+[0,T]$. The space of functions of bounded variation over $[0,T]$ is denoted by $BV[0,T]$, and the set of normalized BV functions vanishing at T is denoted by $BV_T[0,T]$. Functions of bounded variation are without loss of generality (w.l.o.g.) assumed to be right-continuous. We identify the elements of $\mathcal{M}[0,T]$ with the distributional derivatives $\mathrm{d}\eta$ of functions η in $BV_T[0,T]$. The support and the total variation of the measure $\mathrm{d}\eta \in \mathcal{M}[0,T]$ are denoted, respectively, by $\mathrm{supp}(\mathrm{d}\eta)$ and $|\mathrm{d}\eta|_{\mathcal{M}}$. The duality product over $\mathcal{M}[0,T] \times C[0,T]$ is denoted by $\langle \mathrm{d}\eta, x\rangle = \int_0^T x(t)\mathrm{d}\eta(t)$. We denote by $B_X(x,\rho)$ (resp., B_X) the open ball of the space X with center x and radius ρ (resp., the open unit ball of the space X). We write B_r for B_{L^r}, $r = 2, \infty$.

We call a *trajectory* an element $(u,y) \in \mathcal{U} \times \mathcal{Y}$ satisfying the state equation (2.2). A trajectory satisfying the state constraint (2.3) is said to be *feasible*. The *contact set* of a feasible trajectory is defined by

$$I(g(y)) := \{t \in [0,T] : g(y(t)) = 0\}. \tag{2.4}$$

Under assumption (A0), the mapping $\mathcal{U} \to \mathcal{Y}$, $u \mapsto y_u$, where y_u is the unique solution of the state equation (2.2), is well defined. This leads us to the following abstract formulation of $(\mathcal{P})$:

$$\min_{u \in \mathcal{U}} J(u), \qquad G(u) \in K, \tag{2.5}$$

with the cost function $J(u) := \int_0^T \ell(u, y_u)\mathrm{d}t + \phi(y_u(T))$, the constraint mapping $G(u) := g(y_u)$, and the constraint cone $K := C_-[0,T]$ is the cone of continuous functions taking nonpositive values over $[0,T]$. The polar cone to K, denoted by K^-, is the set of nonnegative measures $\mathcal{M}_+[0,T]$.

Finally, throughout this paper the time argument $t \in [0,T]$ is often omitted when there is no ambiguity.

2.1. Optimality conditions and assumptions. Let us first recall the well-known first-order necessary optimality condition of problem $(\mathcal{P})$. The *Hamiltonian* $H : \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^{n*} \to \mathbb{R}$ is defined by

$$H(u, y, p) := \ell(u, y) + pf(u, y). \tag{2.6}$$

We say that a feasible trajectory (u, y) is a *stationary point* of $(\mathcal{P})$, if there exists $(p, \eta) \in BV([0, T]; \mathbb{R}^{n*}) \times BV_T[0, T]$ such that

$$-\mathrm{d}p = H_y(u, y, p)\mathrm{d}t + g_y(y)\mathrm{d}\eta, \qquad p(T) = \phi_y(y(T)), \tag{2.7}$$

$$0 = H_u(u(t), y(t), p(t)) \qquad \text{a.e. on } [0, T], \tag{2.8}$$

$$\mathrm{d}\eta \in N_K(g(y)). \tag{2.9}$$

Here $N_K(g(y))$ denotes the normal cone to K at point $g(y)$ (in the sense of convex analysis). If $g(y) \in K$, then $N_K(g(y))$ is the set of nonnegative measures in $\mathcal{M}_+[0, T]$ having their support included in the contact set (2.4); otherwise $N_K(g(y))$ is empty.

The *Lagrangian* $L : \mathcal{U} \times \mathcal{M}[0, T] \to \mathbb{R}$ of problem (2.5) is defined by

$$L(u, \eta) := J(u) + \langle \mathrm{d}\eta, G(u)\rangle = J(u) + \int_0^T g(y_u(t))\mathrm{d}\eta(t). \tag{2.10}$$

We may write the first-order optimality condition as follows: $(u, y = y_u)$ is a stationary point of $(\mathcal{P})$ iff there exists $\eta \in BV_T[0, T]$ such that

$$D_u L(u, \eta) = 0, \qquad \mathrm{d}\eta \in N_K(G(u)). \tag{2.11}$$

The costate p is then obtained in function of u, $y = y_u$ and η as the unique solution in $BV([0, T]; \mathbb{R}^{n*})$ of the costate equation (2.7).

Robinson's constraint qualification [25, 26] for problem $(\mathcal{P})$ in abstract form (2.5) is as follows:

$$\exists\, \varepsilon > 0, \qquad \varepsilon B_{C[0,T]} \subset G(u) + DG(u)\mathcal{U} - K. \tag{2.12}$$

This condition is equivalent to the existence of some $v \in \mathcal{U}$ such that

$$DG(u)v < 0 \qquad \text{on } I(g(y)).$$

It is well known that a local solution (weak minimum) of $(\mathcal{P})$ satisfying (2.12) is a stationary point of $(\mathcal{P})$.

Alternative formulation. For the stability analysis, it will be convenient to write the optimality condition using alternative multipliers η^2 and p^2, uniquely related to (p, η) in the following way:

$$\eta^1(t) := \int_{(t,T]} \mathrm{d}\eta(s) = -\eta(t), \qquad \eta^2(t) := \int_t^T \eta^1(s)\mathrm{d}s, \tag{2.13}$$

$$p^2(t) := p(t) - \eta^1(t)g_y(y(t)) - \eta^2(t)g_y^{(1)}(y(t)), \qquad t \in [0, T]. \tag{2.14}$$

We see that η^2 belongs to the set $BV_T^2[0, T]$, defined by

$$BV_T^2[0, T] := \{\xi \in W^{1,\infty}(0, T) : \xi(T) = 0,\ \dot{\xi} \in BV_T[0, T]\}. \tag{2.15}$$

Define the *alternative Hamiltonian* $\tilde{H} : \mathbb{R}^m \times \mathbb{R}^n \times \mathbb{R}^{n*} \times \mathbb{R} \to \mathbb{R}$ by

$$\tilde{H}(u, y, p^2, \eta^2) := H(u, y, p^2) + \eta^2 g^{(2)}(u, y), \tag{2.16}$$

where H is the classical Hamiltonian (2.6). Using these alternative multipliers, it is not difficult to see by a direct calculation (see [23] or [3, Lemma 3.4]) that a feasible trajectory (u, y) is a stationary point of $(\mathcal{P})$ iff there exists $(p^2, \eta^2) \in W^{1,\infty}(0, T; \mathbb{R}^{n*}) \times BV_T^2[0, T]$ such that

$$-\dot{p}^2 = \tilde{H}_y(u, y, p^2, \eta^2), \qquad p^2(T) = \phi_y(y(T)), \tag{2.17}$$

$$0 = \tilde{H}_u(u, y, p^2, \eta^2) \qquad \text{a.e. on } [0, T], \tag{2.18}$$

$$\mathrm{d}\dot{\eta}^2 \in N_K(g(y)). \tag{2.19}$$

The definition of these multipliers p^2, η^2 is inspired by the ones used in the alternative formulation for the shooting algorithm (see [23, 15, 22, 5]) though p^2, η^2 are continuous over $[0, T]$ while the ones in the shooting algorithm have jumps.

Remark 2.3. The results of this paper have a natural generalization to a state constraint of higher-order $q > 2$, considering in the analysis alternative multipliers (η^q, p^q) of order q defined below and the resulting alternative formulation of optimality condition of order q. These alternative multipliers of order q, $\eta^q \in BV_T^q[0, T]$ with

$$BV_T^q[0, T] := \{\xi \in W^{q-1,\infty}(0, T) : \xi^{(j)}(T) = 0 \ \forall j = 0, \ldots, q-2, \ \xi^{(q-1)} \in BV_T[0, T]\}$$

and $p^q \in W^{1,\infty}(0, T; \mathbb{R}^{n*})$ are defined by

$$\eta^1(t) := \int_{(t,T]} \mathrm{d}\eta(s), \qquad \eta^j(t) := \int_t^T \eta^{j-1}(s)\mathrm{d}s, \quad j = 2, \ldots, q,$$

$$p^q(t) := p(t) - \sum_{j=1}^{q} \eta^j(t) g_y^{(j-1)}(y(t)).$$

Assumptions. Let $(\bar{u}, \bar{y})$ be a local solution of $(\mathcal{P})$. We denote by $\Omega := I(g(\bar{y}))$ the contact set of the trajectory $(\bar{u}, \bar{y})$, and for a small $\sigma > 0$, let Ω_σ denote a neighborhood of the contact set

$$\Omega_\sigma := \{t \in [0, T] : \mathrm{dist}\{t, \Omega\} < \sigma\}. \tag{2.20}$$

We assume that $(\bar{u}, \bar{y})$ satisfies the assumption below.

(A2) The state constraint is a regular second-order state constraint; i.e., $g_u^{(1)} \equiv 0$ and

$$\exists\, \beta, \sigma > 0, \quad |g_u^{(2)}(\bar{u}(t), \bar{y}(t))| \geq \beta \quad \text{for a.a. } t \in \Omega_\sigma. \tag{2.21}$$

In view of (A1), it will be assumed w.l.o.g. in what follows that σ is small enough so that

$$\Omega_\sigma \subset [a, T] \qquad \text{for some } a > 0. \tag{2.22}$$

Given $v \in L^r(0, T; \mathbb{R}^m)$, $1 \leq r \leq \infty$, we denote by z_v the unique solution in $W^{1,r}(0, T; \mathbb{R}^n)$ of the linearized state equation

$$\dot{z}_v(t) = f_y(\bar{u}(t), \bar{y}(t))z_v(t) + f_u(\bar{u}(t), \bar{y}(t))v(t) \quad \text{a.e. on } [0, T], \quad z_v(0) = 0. \tag{2.23}$$

Note that the derivative of the constraint mapping is given by $DG(\bar{u})v = g_y(\bar{y})z_v$.

LEMMA 2.4. *Let $(\bar{u}, \bar{y})$ be a feasible trajectory of* $(\mathcal{P})$ *satisfying* (A2). *Then for all $r \in [1, +\infty]$ and all $\varepsilon \in (0, \sigma)$, with the σ of* (2.21) *satisfying* (2.22), *the linear mapping*

$$L^r(0,T;\mathbb{R}^m) \to W^{2,r}(\Omega_\varepsilon), \qquad v \mapsto (g_y(\bar{y}(\cdot))z_v(\cdot))|_{\Omega_\varepsilon}, \tag{2.24}$$

where $|_{\Omega_\varepsilon}$ denotes the restriction to the set Ω_ε, is onto, and therefore has a bounded right inverse by the open mapping theorem.

If u is continuous over $[0,T]$, then Lemma 2.4 is satisfied with $\varepsilon = \sigma$.

Proof. We recall only the main ideas of the proof, given in [3, Lemma 2.2]. We have that

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\{g_y(\bar{y}(t))z_v(t)\} &= g_y^{(1)}(\bar{y}(t))z_v(t), \\ \frac{\mathrm{d}^2}{\mathrm{d}t^2}\{g_y(\bar{y}(t))z_v(t)\} &= g_y^{(2)}(\bar{u}(t),\bar{y}(t))z_v(t) + g_u^{(2)}(\bar{u}(t),\bar{y}(t))v(t). \end{aligned}$$

Since by (A1) and hypothesis (2.21), $g_u^{(2)}(\bar{u}(t),\bar{y}(t))$ is nonsingular on a left neighborhood of Ω_ε, the result follows from Gronwall's lemma. □

By the above lemma, assumption (A2) (together with (A1)) implies that $(\bar{u}, \bar{y})$ satisfies Robinson's constraint qualification (2.12), and hence $(\bar{u}, \bar{y})$ is a stationary point of $(\mathcal{P})$, with multipliers $(\bar{p}, \bar{\eta})$. Moreover, Lemma 2.4 implies that the multipliers $(\bar{p}, \bar{\eta})$ associated with $(\bar{u}, \bar{y})$ are unique. We assume, in addition, the following:

(A3) $\bar{u}$ is continuous on $[0,T]$, and the strengthened Legendre–Clebsch condition holds:

$$\exists\, \alpha > 0,\ v^\top H_{uu}(\bar{u}(t),\bar{y}(t),\bar{p}(t))v \geq \alpha|v|^2 \quad \forall t \in [0,T]\ \forall v \in \mathbb{R}^m. \tag{2.25}$$

Remark 2.5. A stronger assumption than (2.25), which *implies* the continuity of $\bar{u}$ (see [3, Proposition 3.1]), is the uniform strong convexity of the Hamiltonian:

$$\exists\, \alpha > 0,\ v^\top H_{uu}(\hat{u},\bar{y}(t),\bar{p}(t))v \geq \alpha|v|^2 \quad \forall t \in [0,T]\ \forall \hat{u}, v \in \mathbb{R}^m.$$

Denote by $\bar{p}^2$ and $\bar{\eta}^2$ the alternative multipliers related to $\bar{p}$ and $\bar{\eta}$ by (2.13)–(2.14). Assumption (2.25) can be rewritten, using the alternative multipliers $\bar{p}^2$ and $\bar{\eta}^2$ instead of $\bar{p}$ and $\bar{\eta}$ and the alternative Hamiltonian (2.16), by

$$\exists\, \alpha > 0,\ v^\top \tilde{H}_{uu}(\bar{u}(t),\bar{y}(t),\bar{p}^2(t),\bar{\eta}^2(t))v \geq \alpha|v|^2 \quad \forall t \in [0,T]\ \forall v \in \mathbb{R}^m. \tag{2.26}$$

LEMMA 2.6. *Let $(\bar{u}, \bar{y})$ be a stationary point of* $(\mathcal{P})$ *satisfying* (A2)–(A3). *Then $\bar{u} \in W^{1,\infty}(0,T;\mathbb{R}^m)$.*

Proof. By (A3), implying (2.26), and the implicit function theorem applied to relation (2.18), there exists a C^1 function Υ such that $\bar{u}(t) = \Upsilon(\bar{y}(t),\bar{p}^2(t),\bar{\eta}^2(t))$. Since $\bar{y}, \bar{p}^2, \bar{\eta}^2 \in W^{1,\infty}$, it follows from the chain rule that $\bar{u} \in W^{1,\infty}$. □

Remark 2.7. More precisely, we have that under the assumptions of Lemma 2.6, $\bar{u} \in BV^2([0,T];\mathbb{R}^m)$, where $BV^2[0,T] := \{u \in W^{1,\infty}(0,T) : \dot{u} \in BV[0,T]\}$. Indeed, differentiation of (2.18) w.r.t. time shows that (omitting arguments $(\bar{u},\bar{y},\bar{p}^2,\bar{\eta}^2)$)

$$0 \;=\; \tilde{H}_{uu}\dot{\bar{u}} + \tilde{H}_{uy}f - \tilde{H}_y f_u + \dot{\bar{\eta}}^2 g_u^{(2)}.$$

Since $\dot{\bar{\eta}}^2 = \bar{\eta} \in BV_T[0,T]$ and $\tilde{H}_{uu}$ is uniformly invertible by (2.26), we obtain the result.

2.2. Perturbed optimal control problem. We consider perturbed problems in the following form:

$$(2.27)\qquad (\mathcal{P}^\mu) \qquad \min_{(u,y)\in\mathcal{U}\times\mathcal{Y}} \int_0^T \ell^\mu(u(t),y(t))\mathrm{d}t + \phi^\mu(y(T))$$

$$(2.28)\qquad \text{s.t.} \qquad \dot{y}(t) \;=\; f^\mu(u(t),y(t)) \quad \text{a.e. on } [0,T], \quad y(0) = y_0^\mu,$$

$$(2.29)\qquad g^\mu(y(t)) \;\leq\; 0 \quad \forall t\in[0,T].$$

Here μ is the perturbation parameter, belonging to an open subset M_0 of a Banach space M.

DEFINITION 2.8. *We say that* $(\mathcal{P}^\mu)$ *is a* stable extension *of* $(\mathcal{P})$, *if the following hold:*

(i) *There exists* $\bar{\mu}\in M_0$ *such that* $(\mathcal{P}^{\bar{\mu}}) \equiv (\mathcal{P})$.

(ii) *The mappings* $\mathbb{R}^m\times\mathbb{R}^n\times M_0 \to \mathbb{R}$, $(u,y,\mu)\mapsto \ell^\mu(u,y)$; $\mathbb{R}^n\times M_0\to\mathbb{R}$, $(y,\mu)\mapsto\phi^\mu(y)$; $M_0\to\mathbb{R}^n$, $\mu\mapsto y_0^\mu$ *(resp.,* $\mathbb{R}^m\times\mathbb{R}^n\times M_0\to\mathbb{R}^n$, $(u,y,\mu)\mapsto f^\mu(u,y)$; $\mathbb{R}^n\times M_0\to\mathbb{R}$, $(y,\mu)\mapsto g^\mu(y)$*) are of class* C^2 *(resp.,* C^3, C^4*), with locally Lipschitz continuous second-order (resp., third-order, fourth-order) derivatives, uniformly w.r.t.* $\mu\in M_0$.

(iii) *The dynamics* f^μ *is uniformly Lipschitz continuous over* $\mathbb{R}^m\times\mathbb{R}^n$ *for all* $\mu\in M_0$.

(iv) *The state constraint is not of first-order, i.e.,* $(g^\mu)_u^{(1)}(u,y)\equiv 0$ *for all* $(u,y,\mu)\in\mathbb{R}^m\times\mathbb{R}^n\times M_0$.

Given a stable extension $(\mathcal{P}^\mu)$ and $(u,\mu)\in\mathcal{U}\times M_0$, we denote by y_u^μ the unique solution in $\mathcal{Y}$ of the state equation (2.28), and we have the abstract formulation of $(\mathcal{P}^\mu)$

$$(2.30)\qquad \min_{u\in\mathcal{U}} J^\mu(u), \qquad G^\mu(u)\in K,$$

with $J^\mu(u) := \int_0^T \ell^\mu(u,y_u^\mu)\mathrm{d}t + \phi^\mu(y_u^\mu(T))$ and $G^\mu(u) := g^\mu(y_u^\mu)$. When we refer to the data of the reference problem $(\mathcal{P})$, we often omit the superscript $\bar{\mu}$.

3. Second-order sufficient optimality condition.

Let $(\bar{u},\bar{y})$ be a stationary point of $(\mathcal{P})$, with multipliers $(\bar{p},\bar{\eta})$. Let $\mathcal{V} := L^2(0,T;\mathbb{R}^m)$. The quadratic form involved in the second-order optimality conditions, defined over $\mathcal{V}$, is as follows:

$$(3.1)\qquad \begin{aligned}\mathcal{Q}(v) \;:=\; & \int_0^T D^2_{(u,y)^2}H(\bar{u},\bar{y},\bar{p})(v,z_v)^2\mathrm{d}t \;+\; \phi_{yy}(\bar{y}(T))(z_v(T),z_v(T)) \\ & +\; \int_0^T g_{yy}(\bar{y})(z_v,z_v)\mathrm{d}\bar{\eta}.\end{aligned}$$

Recall that z_v is the solution of the linearized state equation (2.23). Here the notation $D^2_{(u,y)^2}H(\bar{u},\bar{y},\bar{p})(v,z_v)^2$ stands for $D^2_{(u,y)(u,y)}H(\bar{u},\bar{y},\bar{p})((v,z_v),(v,z_v))$. The critical cone $\mathcal{C}(\bar{u})$ is the set of $v\in\mathcal{V}$ satisfying

$$(3.2)\qquad g_y(\bar{y}(t))z_v(t) = 0 \quad \text{on supp}(\mathrm{d}\bar{\eta}),$$

$$(3.3)\qquad g_y(\bar{y}(t))z_v(t) \leq 0 \quad \text{on } I(g(\bar{y}))\setminus\text{supp}(\mathrm{d}\bar{\eta}).$$

A sufficient second-order optimality condition for $(\mathcal{P})$ is (see [2, Theorem 18] for scalar-valued control and constraint and [3, Theorem 6.1] for vector-valued ones)

$$(3.4)\qquad \mathcal{Q}(v) > 0 \qquad \forall v\in\mathcal{C}(\bar{u})\setminus\{0\}.$$

When the strengthened Legendre–Clebsch condition (2.25) holds, (3.4) implies that $(\bar{u}, \bar{y})$ is a local solution of $(\mathcal{P})$ satisfying the second-order growth condition

$$\exists\, c, \rho > 0, \quad J(u) \geq J(\bar{u}) + c\|u - \bar{u}\|_2^2 \quad \forall u \in \mathcal{U} : G(u) \in K, \ \|u - \bar{u}\|_\infty < \rho. \tag{3.5}$$

This condition involves two norms: L^2 for the growth condition, and L^∞ for the neighborhood.

We will use, in the stability analysis, a natural strengthening of the sufficient condition (3.4), omitting the inequality constraint (3.3) in the critical cone. So let the extended critical cone $\hat{\mathcal{C}}(\bar{u})$ be defined as the set of $v \in \mathcal{V}$ satisfying (3.2) (and hence, $\mathcal{C}(\bar{u}) \subset \hat{\mathcal{C}}(\bar{u})$). The strong second-order sufficient condition used in the stability analysis is as follows:

$$\mathcal{Q}(v) > 0 \qquad \forall v \in \hat{\mathcal{C}}(\bar{u}) \setminus \{0\}. \tag{3.6}$$

Although we call the above condition the *strong* second-order sufficient condition (in comparison with (3.4)), it takes into account the active constraints so it is weaker than the second-order sufficient condition used in [9] that assumes the strict positivity of $\mathcal{Q}$ over the whole space $\mathcal{V} \setminus \{0\}$.

The strengthened Legendre–Clebsch condition (2.25) implies (see [6, Proposition 3.76(i)]) that the quadratic form $\mathcal{Q}$ is a *Legendre form* (see [16]), i.e., a weakly lower semicontinuous (weakly l.s.c.) quadratic form with the property that if a sequence v_n weakly converges to v in L^2 ($v_n \rightharpoonup v$) and if $Q(v_n) \to Q(v)$, then $v_n \to v$ strongly.

Lemma 3.1. *Let* $(\bar{u}, \bar{y})$ *be a stationary point of* $(\mathcal{P})$. *An equivalent expression for the quadratic form* $\mathcal{Q}$ *defined by* (3.1), *using the alternative multipliers* $(\bar{p}^2, \bar{\eta}^2)$ *given by* (2.13)–(2.14) *instead of* $(\bar{p}, \bar{\eta})$ *and the alternative Hamiltonian* (2.16), *is*

$$\mathcal{Q}(v) \;=\; \int_0^T D^2_{(u,y)^2} \tilde{H}(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2)(v, z_v)^2 \mathrm{d}t \;+\; \phi_{yy}(\bar{y}(T))(z_v(T), z_v(T)). \tag{3.7}$$

Proof. Let $v \in \mathcal{V}$. Denote by $\tilde{\mathcal{Q}}(v)$ the right-hand side of (3.7) and set $\Delta := \tilde{\mathcal{Q}}(v) - \mathcal{Q}(v)$. In view of the relations (2.13)–(2.14) between $(\bar{p}^2, \bar{\eta}^2)$ and $(\bar{p}, \bar{\eta})$, we have

$$\begin{aligned}
\Delta &= \int_0^T (\bar{p}^2 - \bar{p}) D^2 f(\bar{u}, \bar{y})(v, z_v)^2 \mathrm{d}t + \int_0^T D^2 g^{(2)}(\bar{u}, \bar{y})(v, z_v)^2 \bar{\eta}^2 \mathrm{d}t \\
&\quad - \int_0^T g_{yy}(\bar{y})(z_v, z_v)\mathrm{d}\bar{\eta} \\
&= -\int_0^T \bar{\eta}^1 g_y(\bar{y}) D^2 f(\bar{u}, \bar{y})(v, z_v)^2 \mathrm{d}t - \int_0^T \bar{\eta}^2 g_y^{(1)}(\bar{y}) D^2 f(\bar{u}, \bar{y})(v, z_v)^2 \mathrm{d}t \\
&\quad + \int_0^T D^2 g^{(2)}(\bar{u}, \bar{y})(v, z_v)^2 \bar{\eta}^2 \mathrm{d}t - \int_0^T g_{yy}(\bar{y})(z_v, z_v)\mathrm{d}\bar{\eta}.
\end{aligned}$$

The integration by parts formula in BV [12, p. 154] shows that (the calculus is analogous to Lemma 3.6 in [5])

$$\begin{aligned}
\int_0^T g_{yy}(\bar{y})(z_v, z_v)\mathrm{d}\bar{\eta} &= \int_0^T \frac{\mathrm{d}}{\mathrm{d}t}\{g_{yy}(\bar{y})(z_v, z_v)\}\bar{\eta}^1 \mathrm{d}t + [g_{yy}(\bar{y})(z_v, z_v)\bar{\eta}^1]_0^T \\
&= \int_0^T \{g_{yyy}(\bar{y})(f, z_v, z_v) + 2g_{yy}(\bar{y})(Df(\bar{u}, \bar{y})(v, z_v), z_v)\}\bar{\eta}^1 \mathrm{d}t \\
&= \int_0^T g_{yy}^{(1)}(\bar{y})(z_v, z_v)\bar{\eta}^1 \mathrm{d}t - \int_0^T g_y(\bar{y}) D^2 f(\bar{u}, \bar{y})(v, z_v)^2 \bar{\eta}^1 \mathrm{d}t.
\end{aligned}$$

Similarly, we obtain that

$$\int_0^T g_{yy}^{(1)}(\bar{y})(z_v, z_v)\bar{\eta}^1 \mathrm{d}t = \int_0^T D^2 g^{(2)}(\bar{u}, \bar{y})(v, z_v)^2 \bar{\eta}^2 \mathrm{d}t - \int_0^T g_y^{(1)}(\bar{y}) D^2 f(\bar{u}, \bar{y})(v, z_v)^2 \bar{\eta}^2 \mathrm{d}t.$$

Summing the two above equalities, we obtain that $\Delta = 0$, which completes the proof. □

4. Stability analysis for the nonlinear problem.

According to Definition 5.16 in [6], adapted to our optimal control framework, we consider the following definition of uniform second-order growth condition.

DEFINITION 4.1. *Let $(\bar{u}, \bar{y})$ be a stationary point of $(\mathcal{P})$. We say that the* uniform second-order (or quadratic) growth *condition holds if, for all stable extensions $(\mathcal{P}^\mu)$ of $(\mathcal{P})$, there exists $c, \rho > 0$ and a neighborhood $\mathcal{N}$ of $\bar{\mu}$, such that for any stationary point (u^μ, y^μ) of $(\mathcal{P}^\mu)$ with $\mu \in \mathcal{N}$ and $\|u^\mu - \bar{u}\|_\infty < \rho$,*

$$J^\mu(u) \geq J^\mu(u^\mu) + c\|u - u^\mu\|_2^2 \quad \forall u \in \mathcal{U} : G^\mu(u) \in K, \ \|u - \bar{u}\|_\infty < \rho. \tag{4.1}$$

The next proposition (proved in subsection 4.2) shows that the strong second-order sufficient condition (3.6) implies the uniform second-order growth condition. Therefore, if a stationary point for the perturbed problem $(\mathcal{P}^\mu)$ exists, then the latter is *locally unique* in a L^∞-neighborhood of $\bar{u}$, and is a local solution of $(\mathcal{P}^\mu)$.

PROPOSITION 4.2. *Let $(\bar{u}, \bar{y})$ be a stationary point of $(\mathcal{P})$ satisfying* (A2)–(A3) *and the strong second-order sufficient condition* (3.6)*. Then the uniform second-order growth condition holds.*

The difficult part in the stability analysis here is to prove the *existence* of a stationary point for the perturbed problem. For some general optimization problems, Robinson's constraint qualification (2.12) and the uniform quadratic growth condition imply, for a certain class of perturbations, the existence of a stationary point for the perturbed problem; see Bonnans and Shapiro [6, Theorem 5.17]. The proof uses Ekeland's variational principle [13]. However, this result does not apply to our nonlinear optimal control problem, due to the *two-norms discrepancy*, but it does apply to linear-quadratic problems (see the proof of Theorem 5.4). For the general nonlinear problem, in order to obtain the existence of a stationary point for the perturbed problem, we need to use a variant of Robinson's strong regularity theory [27].

The main result of this paper is the next theorem (proved in section 6).

THEOREM 4.3. *Let $(\bar{u}, \bar{y})$ be a local solution of $(\mathcal{P})$, satisfying* (A2)–(A3) *and the strong second-order sufficient condition* (3.6)*. Then for all stable extensions $(\mathcal{P}^\mu)$ of $(\mathcal{P})$, there exist $c, \rho, \kappa, \tilde{\kappa} > 0$ and a neighborhood $\mathcal{N}$ of $\bar{\mu}$, such that for all $\mu \in \mathcal{N}$, $(\mathcal{P}^\mu)$ has a unique stationary point (u^μ, y^μ) with $\|u^\mu - \bar{u}\|_\infty < \rho$ and unique associated alternative multipliers $(p^{2,\mu}, \eta^{2,\mu})$, and for all $\mu, \mu' \in \mathcal{N}$,*

$$\|u^\mu - u^{\mu'}\|_2, \ \|y^\mu - y^{\mu'}\|_{1,2}, \ \|p^{2,\mu} - p^{2,\mu'}\|_{1,2}, \ \|\eta^{2,\mu} - \eta^{2,\mu'}\|_2 \leq \kappa\|\mu - \mu'\|, \tag{4.2}$$

$$\|u^\mu - u^{\mu'}\|_\infty, \|y^\mu - y^{\mu'}\|_{1,\infty}, \|p^{2,\mu} - p^{2,\mu'}\|_{1,\infty}, \|\eta^{2,\mu} - \eta^{2,\mu'}\|_\infty \leq \tilde{\kappa}\|\mu - \mu'\|^{2/3}. \tag{4.3}$$

Moreover, (u^μ, y^μ) is a local solution of $(\mathcal{P}^\mu)$ satisfying the uniform quadratic growth condition (4.1).

The above theorem is obtained by application of a generalized implicit function theorem by Dontchev and Hager [9] (Theorem 4.8 of this paper) to the alternative formulation (2.17)–(2.19) in suitable functional spaces described in subsection 4.3. In order to show that the main assumption of this theorem is satisfied (assumption (iv) of Theorem 4.8), we have to show that a perturbed linear-quadratic optimal control problem has a unique solution which is Lipschitz continuous w.r.t. the parameter. For this, we will use Proposition 4.2 (or more precisely, its analogous statement adapted to linear-quadratic problems). Before giving the proof of Proposition 4.2, we first need to study the stability of multipliers (Proposition 4.4).

4.1. Stability of multipliers. The next result shows that under the constraint qualification (A2), the stability of multipliers could be deduced from the stability of solutions. Given $r \in [1, +\infty]$, we denote by $\| \cdot \|_{2,r*}$ the norm of the dual space to $W^{2,r}(0,T)$, i.e., for $\mathrm{d}\eta \in \mathcal{M}[0,T]$ we have

$$\|\mathrm{d}\eta\|_{2,r*} := \sup \left\{ \frac{|\int_0^T \Phi(t)\mathrm{d}\eta(t)|}{\|\Phi\|_{2,r}},\ \Phi \in W^{2,r}(0,T), \Phi \not\equiv 0 \right\}.$$

PROPOSITION 4.4. *Let* $(\bar{u}, \bar{y})$ *be a stationary point of* $(\mathcal{P})$ *satisfying* (A2)*. Then for every stable extension* $(\mathcal{P}^\mu)$ *of* $(\mathcal{P})$*, there exists* $\nu > 0$ *such that for every stationary point* (u, y) *of* $(\mathcal{P}^\mu)$*, with (unique) associated multipliers* (p, η) *and alternative multipliers* (p^2, η^2) *given by* (2.13)–(2.14)*, the following hold:*

(i) *If* $\|\mu - \bar{\mu}\|, \|u - \bar{u}\|_\infty < \nu$*, then* $\mathrm{d}\eta$ *is uniformly bounded in* $\mathcal{M}[0,T]$*.*

(ii) *There exists* $\kappa > 0$ *such that, for all* $\|\mu - \bar{\mu}\|, \|u - \bar{u}\|_\infty < \nu$*, we have*

$$\|\mathrm{d}\eta - \mathrm{d}\bar{\eta}\|_{2,1*},\ \|\eta^2 - \bar{\eta}^2\|_\infty \ \le\ \kappa(\|u - \bar{u}\|_\infty + \|\mu - \bar{\mu}\|).$$

Moreover, when $\|\mu - \bar{\mu}\|, \|u - \bar{u}\|_\infty \to 0$*:*

(iii) $\mathrm{d}\eta$ *weakly-* converges to* $\mathrm{d}\bar{\eta}$ *(*$\mathrm{d}\eta \overset{*}{\rightharpoonup} \mathrm{d}\bar{\eta}$*) in* $\mathcal{M}[0,T]$*;*

(iv) $\eta^1 \to \bar{\eta}^1$ *in* L^1*;*

(v) p^2 *and* η^2 *converge uniformly to* $\bar{p}^2$ *and* $\bar{\eta}^2$*, respectively.*

The proof of the above proposition uses the lemma below.

LEMMA 4.5. *For all* $1 \le r < \infty$*, with* $r' := r/(r-1)$ *(*$1' = \infty$*), there exists a positive constant* C *such that*

$$\|\xi\|_{r'} \ \le\ C\|\mathrm{d}\dot{\xi}\|_{2,r*} \qquad \forall \xi \in BV_T^2[0,T]. \tag{4.4}$$

Proof. Let $\varphi \in L^r(0,T)$. Set $\Phi^1(t) := \int_0^t \varphi(s)\mathrm{d}s$ and $\Phi(t) := \int_0^t \Phi^1(s)\mathrm{d}s$. Then $\Phi \in W^{2,r}(0,T)$, and $\|\Phi\|_{2,r} \le C\|\varphi\|_r$, with $C = 1 + T/\sqrt[r]{r} + (T/\sqrt[r]{r})^2$. Since $\xi(T) = \dot{\xi}(T) = 0$, the integration by parts formula in BV [12, p. 154] implies that, for all $\xi \in BV_T^2[0,T]$,

$$\int_0^T \varphi(t)\xi(t)\mathrm{d}t \ =\ -\int_0^T \Phi^1(t)\dot{\xi}(t)\mathrm{d}t \ =\ \int_0^T \Phi(t)\mathrm{d}\dot{\xi}(t).$$

Therefore,

$$\|\xi\|_{r'} = \sup_{\varphi \in L^r, \varphi \not\equiv 0} \frac{|\int_0^T \varphi(t)\xi(t)\mathrm{d}t|}{\|\varphi\|_r} \le C \sup_{\Phi \in W^{2,r}, \Phi \not\equiv 0} \frac{|\int_0^T \Phi(t)\mathrm{d}\dot{\xi}(t)|}{\|\Phi\|_{2,r}},$$

which gives the result. □

Proof of Proposition 4.4. Let $(\mathcal{P}^\mu)$ be a stable extension of $(\mathcal{P})$. Note first that for $\|\mu - \bar{\mu}\|$ and $\|u - \bar{u}\|_\infty$ small enough, assumptions (A1) and (A2) hold for $(\mathcal{P}^\mu)$.

This implies the uniqueness of the multipliers (p,η) associated with a stationary point (u,y) of $(\mathcal{P}^\mu)$. Since $(\bar{u},\bar{y})$ satisfies Robinson's constraint qualification (2.12), point (i) follows from [6, Proposition 4.43].

Let us show (ii). Since $(u, y=y_u^\mu)$ is a stationary point of $(\mathcal{P}^\mu)$, we have that

$$DJ^\mu(u) + DG^\mu(u)^\top \mathrm{d}\eta = 0, \qquad \mathrm{d}\eta \in N_K(G^\mu(u)).$$

It follows that $DG(\bar{u})^\top(\mathrm{d}\bar{\eta} - \mathrm{d}\eta) = DJ^\mu(u) - DJ(\bar{u}) + (DG^\mu(u) - DG(\bar{u}))^\top \mathrm{d}\eta$, and hence, for all $v \in L^1(0,T)$,

$$\langle \mathrm{d}\bar{\eta} - \mathrm{d}\eta, DG(\bar{u})v\rangle = (DJ^\mu(u) - DJ(\bar{u}))v + \langle \mathrm{d}\eta, (DG^\mu(u) - DG(\bar{u}))v\rangle. \tag{4.5}$$

Fix $\varepsilon \in (0,\sigma)$ with the σ of (2.21) satisfying (2.22). By Lemma 2.4, the linear mapping defined in (2.24) for $r=1$ is onto. Since $DG(\bar{u})v = g_y(\bar{y})z_v$, by the open mapping theorem, there exists a constant $C_1 > 0$ such that, for all $\Phi \in W^{2,1}(0,T)$, there exists $v \in L^1(0,T)$ such that $DG(\bar{u})v = \Phi$ on Ω_ε and $\|v\|_1 \le C_1\|\Phi\|_{2,1}$. For $\|\mu - \bar{\mu}\|, \|u-\bar{u}\|_\infty$ small enough, the contact set $I(g^\mu(y))$, and hence the support of the measure $\mathrm{d}\eta$, are included in the set Ω_ε. Therefore, $\langle \mathrm{d}\eta - \mathrm{d}\bar{\eta}, DG(\bar{u})v\rangle = \langle \mathrm{d}\eta - \mathrm{d}\bar{\eta}, \Phi\rangle$. Consequently, by (4.5),

$$|\langle \mathrm{d}\eta - \mathrm{d}\bar{\eta}, \Phi\rangle| \le |(DJ^\mu(u) - DJ(\bar{u}))v| + |\mathrm{d}\eta|_{\mathcal{M}}\|(DG^\mu(u) - DG(\bar{u}))v\|_\infty.$$

By point (i), $|\mathrm{d}\eta|_{\mathcal{M}}$ is uniformly bounded, and it is not difficult to check that

$$|(DJ^\mu(u) - DJ(\bar{u}))v|,\ \|(DG^\mu(u) - DG(\bar{u}))v\|_\infty \ \le\ C(\|u-\bar{u}\|_\infty + \|\mu - \bar{\mu}\|)\|v\|_1,$$

where C denotes (possibly different) positive constants. Therefore, we obtain that

$$\begin{aligned} |\langle \mathrm{d}\eta - \mathrm{d}\bar{\eta}, \Phi\rangle| &\le C(\|u-\bar{u}\|_\infty + \|\mu - \bar{\mu}\|)\|v\|_1 \\ &\le CC_1(\|u-\bar{u}\|_\infty + \|\mu - \bar{\mu}\|)\|\Phi\|_{2,1}. \end{aligned}$$

Consequently, $\|\mathrm{d}\eta - \mathrm{d}\bar{\eta}\|_{2,1*} \le CC_1(\|u-\bar{u}\|_\infty + \|\mu - \bar{\mu}\|)$, and since, by Lemma 4.5, $\|\eta^2 - \bar{\eta}^2\|_\infty \le C\|\mathrm{d}\eta - \mathrm{d}\bar{\eta}\|_{2,1*}$, this proves (ii).

Now consider a sequence $\mu_n \to \bar{\mu}$, and let (u_n, y_n) be a stationary point of $(\mathcal{P}^{\mu_n})$ such that $u_n \to \bar{u}$ in L^∞, with (unique) multipliers (p_n, η_n) and alternative multipliers (p_n^2, η_n^2). Since $W^{2,1}(0,T)$ is dense in $C[0,T]$, we deduce easily from point (ii) that $\mathrm{d}\eta_n \overset{*}{\rightharpoonup} \mathrm{d}\bar{\eta}$ in $\mathcal{M}[0,T]$, which shows (iii). By the compactness theorem in BV [1, Theorem 3.23], it follows that $\eta_n^1 \to \bar{\eta}^1$ in L^1, which shows (iv). Finally, since η^2 is given by (2.13), (iv) implies that $\eta_n^2 \to \bar{\eta}^2$ uniformly. By (2.17) and Gronwall's lemma, we conclude that $p_n^2 \to \bar{p}^2$ in $W^{1,\infty}$, which achieves the proof of (v). □

4.2. The uniform second-order growth condition (proof of Proposition 4.2). The proof of Proposition 4.2 uses the auxiliary result below. Given $A, B \subset [0,T]$, denote by $\mathrm{exc}\{A,B\}$ the *Hausdorff excess* of A over B, defined by

$$\mathrm{exc}\{A,B\} \ := \ \sup_{t\in A}\inf_{s\in B}|t-s|, \tag{4.6}$$

with the convention $\mathrm{exc}\{\emptyset, B\} = 0$.

Lemma 4.6. *Let $\mathrm{d}\bar{\eta} \in \mathcal{M}[0,T]$, and a sequence $(\mathrm{d}\eta_n) \subset \mathcal{M}[0,T]$ be such that $\mathrm{d}\eta_n$ weakly-* converges to $\mathrm{d}\bar{\eta}$ in $\mathcal{M}[0,T]$. Then $e_n := \mathrm{exc}\{\mathrm{supp}(\mathrm{d}\bar{\eta}), \mathrm{supp}(\mathrm{d}\eta_n)\}$ converges to zero when $n \to +\infty$.*

Proof. The result follows from classical compactness arguments. By contradiction, assume that the result is false. Then there exist $\varepsilon_0 > 0$ and a subsequence, still denoted by $\mathrm{d}\eta_n$, such that for all $n \in \mathbb{N}^*$, $e_n > \varepsilon_0$, i.e., there exists $t_n \in \operatorname{supp}(\mathrm{d}\bar{\eta})$ such that for all $s \in \operatorname{supp}(\mathrm{d}\eta_n)$, $|t_n - s| > \varepsilon_0$. Since the sequence $(t_n)_{n\in\mathbb{N}^*} \subset [0,T]$ is bounded, assume w.l.o.g. that $t_n \to \bar{t} \in [0,T]$. Since $\operatorname{supp}(\mathrm{d}\bar{\eta})$ is closed, $\bar{t} \in \operatorname{supp}(\mathrm{d}\bar{\eta})$. For n large enough, $|t_n - \bar{t}| < \varepsilon_0/2$, and hence, for all $s \in \operatorname{supp}(\mathrm{d}\eta_n)$, $|\bar{t} - s| \geq |t_n - s| - |t_n - \bar{t}| > \varepsilon_0/2$. Let φ be a continuous function, with support in $[\bar{t} - \varepsilon_0/2, \bar{t} + \varepsilon_0/2]$, and such that $\int_0^T \varphi \mathrm{d}\bar{\eta} \neq 0$. Since $\operatorname{dist}\{\bar{t}, \operatorname{supp}(\mathrm{d}\eta_n)\} > \varepsilon_0/2$ for all large enough n, $\int_0^T \varphi \mathrm{d}\eta_n = 0$. But $\mathrm{d}\eta_n \overset{*}{\rightharpoonup} \mathrm{d}\bar{\eta}$, implying that $\int_0^T \varphi \mathrm{d}\eta_n \to \int_0^T \varphi \mathrm{d}\bar{\eta}$, which gives the desired contradiction. □

Remark 4.7. We may equivalently reformulate Lemma 4.6 as follows: if $\mathrm{d}\eta_n$ weakly-* converges to $\mathrm{d}\bar{\eta}$ in $\mathcal{M}[0,T]$, then

$$\operatorname{supp}(\mathrm{d}\bar{\eta}) \subset \limsup_{n\to+\infty} \operatorname{supp}(\mathrm{d}\eta_n),$$

where the lim sup is in the sense of Painlevé–Kuratowski.

Proof of Proposition 4.2. We argue by contradiction. If the uniform second-order growth condition does not hold, there exist a stable extension $(\mathcal{P}^\mu)$, a sequence $\mu_n \to \bar{\mu}$, a stationary point (u_n, y_n) of $(\mathcal{P}^{\mu_n})$ such that $u_n \to \bar{u}$ in L^∞, with multipliers (p_n, η_n) and alternative multipliers (p_n^2, η_n^2), and a feasible point $(\hat{u}_n, \hat{y}_n)$ of $(\mathcal{P}^{\mu_n})$ such that

$$J^{\mu_n}(\hat{u}_n) < J^{\mu_n}(u_n) + o(\|\hat{u}_n - u_n\|_2^2). \tag{4.7}$$

Introducing the Lagrangian of $(\mathcal{P}^\mu)$, $L^\mu(u,\eta) = J^\mu(u) + \langle \mathrm{d}\eta, G^\mu(u)\rangle$, and using that $\mathrm{d}\eta_n \in N_K(G^{\mu_n}(u_n))$, (4.7) implies that

$$L^{\mu_n}(\hat{u}_n, \eta_n) - L^{\mu_n}(u_n, \eta_n) \leq J^{\mu_n}(\hat{u}_n) - J^{\mu_n}(u_n) < o(\|\hat{u}_n - u_n\|_2^2).$$

Set $\varepsilon_n := \|\hat{u}_n - u_n\|_2 \to 0$ and $v_n := \varepsilon_n^{-1}(\hat{u}_n - u_n)$. A second-order expansion of the Lagrangian shows that $L^{\mu_n}(\hat{u}_n, \eta_n) - L^{\mu_n}(u_n, \eta_n) = \varepsilon_n^2 \mathcal{Q}^{\mu_n}(v_n) + o(\varepsilon_n^2)$, where the quadratic form $\mathcal{Q}^{\mu_n}$ is defined like (3.1) for the stationary point (u_n, y_n) of $(\mathcal{P}^{\mu_n})$. Therefore, dividing the above inequality by ε_n^2, we obtain that

$$\mathcal{Q}^{\mu_n}(v_n) \leq o(1). \tag{4.8}$$

Since $\|v_n\|_2 = 1$ for all n, taking a subsequence if necessary, we may assume w.l.o.g. that $v_n \rightharpoonup \bar{v}$ weakly in L^2 for some $\bar{v} \in \mathcal{V}$ when $n \to +\infty$. Since, by Lemma 3.1, $\mathcal{Q}^{\mu_n}$ can also be expressed by (3.7), and $(u_n, y_n, p_n^2, \eta_n^2) \to (\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2)$ uniformly by Proposition 4.4(v), and since v_n is bounded in L^2, it follows that $\mathcal{Q}^{\mu_n}(v_n) - \mathcal{Q}(v_n) \to 0$. Therefore, writing that $\mathcal{Q}^{\mu_n}(v_n) = \mathcal{Q}(v_n) + (\mathcal{Q}^{\mu_n}(v_n) - \mathcal{Q}(v_n))$, and using that $\mathcal{Q}$ is a Legendre form and hence weakly l.s.c., we obtain by (4.8) that

$$\mathcal{Q}(\bar{v}) \leq 0. \tag{4.9}$$

Moreover, since $v_n \rightharpoonup \bar{v}$ weakly in L^2, and $(u_n, y_n) \to (\bar{u}, \bar{y})$ uniformly, the linearized state z_n, the solution of

$$\dot{z}_n = f_y^{\mu_n}(u_n, y_n) z_n + f_u^{\mu_n}(u_n, y_n) v_n \quad \text{a.e. on } [0,T], \quad z_n(0) = 0$$

converges weakly to $\bar{z} := z_{\bar{v}}$ in H^1, and hence uniformly. Since $G^{\mu_n}(\hat{u}_n) \in K$, we have that $0 \geq G^{\mu_n}(\hat{u}_n) - G^{\mu_n}(u_n) = \varepsilon_n DG^{\mu_n}(u_n) v_n + \varepsilon_n r_n$ on $\operatorname{supp}(\mathrm{d}\eta_n)$, with $\|r_n\|_\infty = \mathcal{O}(\varepsilon_n)$. Since $DG^{\mu_n}(u_n) v_n = g_y^{\mu_n}(y_n) z_n$, it follows that

$$g_y^{\mu_n}(y_n) z_n + r_n \leq 0 \quad \text{on } \operatorname{supp}(\mathrm{d}\eta_n). \tag{4.10}$$

Since $\frac{\mathrm{d}}{\mathrm{d}t} g_y^{\mu_n}(y_n(t))z_n(t) = (g^{\mu_n})_y^{(1)}(y_n)z_n$ is uniformly bounded over $[0,T]$, the functions $g_y^{\mu_n}(y_n)z_n$ are uniformly Lipschitz continuous over $[0,T]$. Therefore,

$$\begin{aligned}\sup_{\mathrm{supp}(\mathrm{d}\bar{\eta})} g_y(\bar{y})\bar{z} &\le \|g_y(\bar{y})\bar{z} - g_y^{\mu_n}(y_n)z_n\|_\infty + \|(g^{\mu_n})_y^{(1)}(y_n)z_n\|_\infty e_n + \sup_{\mathrm{supp}(\mathrm{d}\eta_n)} g_y^{\mu_n}(y_n)z_n \\ &\le o(1) + \mathcal{O}(e_n) + \mathcal{O}(\varepsilon_n),\end{aligned}$$

where $e_n := \mathrm{exc}\{\mathrm{supp}(\mathrm{d}\bar{\eta}), \mathrm{supp}(\mathrm{d}\eta_n)\}$ is defined by (4.6). Since $\mathrm{d}\eta_n \overset{*}{\rightharpoonup} \mathrm{d}\bar{\eta}$ by Proposition 4.4(iii), it follows from Lemma 4.6 that $e_n \to 0$. Therefore, we obtain that

$$g_y(\bar{y})\bar{z} \le 0 \qquad \text{on } \mathrm{supp}(\mathrm{d}\bar{\eta}). \tag{4.11}$$

In addition, by (4.7), $DJ^{\mu_n}(u_n)v_n \le \mathcal{O}(\varepsilon_n)$. Since $DJ^{\mu_n}(u_n) + DG^{\mu_n}(u_n)^\top \mathrm{d}\eta_n = 0$, it follows that $\langle \mathrm{d}\eta_n, DG^{\mu_n}(u_n)v_n\rangle = \int_0^T g_y^{\mu_n}(y_n)z_n \mathrm{d}\eta_n \ge \mathcal{O}(\varepsilon_n)$. Since $\mathrm{d}\eta_n \overset{*}{\rightharpoonup} \mathrm{d}\bar{\eta}$ and $g_y^{\mu_n}(y_n)z_n \to g_y(\bar{y})\bar{z}$ uniformly, we obtain that $\int_0^T g_y(\bar{y})\bar{z}\mathrm{d}\bar{\eta} \ge 0$. Using that $\mathrm{d}\bar{\eta} \ge 0$, (4.11) implies that

$$g_y(\bar{y})\bar{z} = 0 \qquad \text{on } \mathrm{supp}(\mathrm{d}\bar{\eta}),$$

i.e., $\bar{v} \in \hat{\mathcal{C}}(\bar{u})$. The strong second-order sufficient condition (3.6) and (4.9) imply then that $\bar{v} = 0$. But then $\mathcal{Q}(\bar{v}) = 0$, and $\mathcal{Q}(v_n) \to \mathcal{Q}(\bar{v})$. Since $\mathcal{Q}$ is a Legendre form, we deduce that $v_n \to \bar{v} = 0$ strongly in L^2, contradicting that $\|v_n\|_2 = 1$ for all n. □

4.3. The strong regularity framework.

We use the following generalized implicit function theorem in metric spaces by Dontchev and Hager [9], which is a variant of Robinson's strong regularity [27].

THEOREM 4.8 (see [9, Theorem 2.2]). *Let X be a complete metric space, $\tilde{X}$ a closed subset of X, W a linear metric space, Δ a subset of W, P a metric space, and $\mathcal{F} : X \times P \to W$, $\mathcal{N} : X \to 2^W$, $\mathcal{L} : X \to W$. Assume that $\mathcal{L}$ is continuous and that there exists $(\bar{x}, \bar{\mu}) \in \tilde{X} \times P$ such that the following hold:*

(i) $\mathcal{F}(\bar{x}, \bar{\mu}) \in \mathcal{N}(\bar{x})$.

(ii) $\mathcal{F}(\bar{x}, \cdot)$ *is continuous at* $\bar{\mu}$.

(iii) $\Psi^\mu := \mathcal{F}(\cdot, \mu) - \mathcal{L}(\cdot)$ *is strictly stationary at $x = \bar{x}$, uniformly in μ near $\bar{\mu}$, i.e., for all $\varepsilon > 0$, there exists $\nu > 0$ such that if $\|x_i - \bar{x}\|_X,\ \|\mu - \bar{\mu}\| \le \nu$, $i = 1, 2$,*

$$\|\Psi^\mu(x_1) - \Psi^\mu(x_2)\|_W \le \varepsilon \|x_1 - x_2\|_X. \tag{4.12}$$

(iv) *For all $\delta \in \Delta$, there exists a unique solution $x \in \tilde{X}$ of*

$$\delta \in \mathcal{L}(x) - \mathcal{N}(x), \tag{4.13}$$

and there exists $\lambda > 0$ such that, with x_δ the unique solution associated with δ,

$$\|x_\delta - x_{\delta'}\|_X \le \lambda \|\delta - \delta'\|_W \quad \forall\, \delta, \delta' \in \Delta.$$

(v) $\mathcal{F} - \mathcal{L}$ *maps a neighborhood of $(\bar{x}, \bar{\mu})$ into Δ.*

Then for all $\lambda_+ > \lambda$, there exist neighborhoods $\mathcal{X}$ of $\bar{x}$ in $\tilde{X}$ and $\mathcal{W}$ of $\bar{\mu}$, such that for each $\mu \in \mathcal{W}$, there exists a unique $x \in \mathcal{X}$ satisfying $\mathcal{F}(x, \mu) \in \mathcal{N}(x)$; moreover, for each $\mu_i \in \mathcal{W}$, $i = 1, 2$, if x_i denotes the $x \in \mathcal{X}$ associated with μ_i, then

$$\|x_2 - x_1\|_X \le \lambda_+ \|\mathcal{F}(x_1, \mu_1) - \mathcal{F}(x_1, \mu_2)\|_W. \tag{4.14}$$

In [9], the theorem is stated with $\tilde{X} = X$, but remains true if we replace the complete metric space X by any closed subset $\tilde{X}$ of X, equipped with the metric of X, since $\tilde{X}$ remains a complete metric space.

This theorem was used for stability analysis of optimal control problems subject to first-order state constraints in [9]. In what follows, we describe a suitable framework to apply Theorem 4.8 for second-order state constraints.

Remark 4.9. Our choice of functional spaces to apply Theorem 4.8 differs from that of [9] or [18] in the spaces for the state constraint and state constraint multiplier. Whereas in [9, 18] the state constraint is seen in $W^{1,\infty}$, we consider here rather the state constraint in the space of continuous functions $C[0,T]$. Another natural choice for the space of second-order state constraints would be $W^{2,\infty}$ since the constraint is "onto" in this space (Lemma 2.4). The reason for considering here the constraint in $C[0,T]$ is to have multipliers in $\mathcal{M}[0,T]$ instead of in the dual space of $W^{1,\infty}$ or $W^{2,\infty}$. For first-order state constraints it can be shown (see [14]) that the state constraint multiplier η lies in $W^{1,\infty}$ (and therefore a suitable choice for the state constraint multiplier space is the space Lip_k defined below), but this is no more true for higher-order state constraints. Note that since $W^{2,\infty} \subset W^{1,\infty} \subset C[0,T]$ with continuous and dense embeddings, and the constraint is "onto" in $W^{2,\infty}$ by Lemma 2.4, the multipliers in the three possible formulations are one-to-one.

Notation. In order to apply Theorem 4.8 to prove Theorem 4.3 in sections 5 and 6, we use the following notation. Given $k, l, r, \varrho, k' > 0$, define the spaces

$$Lip_k(0,T) := \{u \in W^{1,\infty}(0,T) : \|\dot{u}\|_\infty \le k\},$$

$$BV^2_{T,l}[0,T] := \{\xi \in BV^2_T[0,T] : |\mathrm{d}\dot{\xi}|_{\mathcal{M}} \le l\},$$

$$X := Lip_k(0,T;\mathbb{R}^m) \times BV^2_{T,l}[0,T], \tag{4.15}$$

$$\tilde{X} := \{x = (u,\xi) \in X : \|u - \bar{u}\|_2 \le r\}, \tag{4.16}$$

$$W := L^2(0,T;\mathbb{R}^{m*}) \times H^2(0,T) \tag{4.17}$$

equipped with its standard norm $\|\delta\|_W := \|\gamma\|_2 + \|\zeta\|_{2,2}$ for $\delta = (\gamma,\zeta) \in W$,

$$\Delta := \{\delta \in Lip_{k'}(0,T;\mathbb{R}^{m*}) \times H^2(0,T),\ \|\delta\|_W \le \varrho\}, \tag{4.18}$$

$$P\ :\ \text{closed neighborhood of } \bar{\mu}, \text{ contained in } M_0,$$

and mappings

- $\mathcal{F} : X \times P \to W$,

$$\mathcal{F}(x,\mu) := \begin{pmatrix} \tilde{H}^\mu_u(u, y^\mu_u, p^{2,\mu}_{u,\eta^2}, \eta^2) \\ g^\mu(y^\mu_u) \end{pmatrix},$$

  where $\tilde{H}^\mu$ is the alternative Hamiltonian (2.16) of $(\mathcal{P}^\mu)$, y^μ_u is the solution of the state equation (2.28), and $p^{2,\mu}_{u,\eta^2}$ is the solution of the alternative costate equation (2.17) for $(\mathcal{P}^\mu)$, i.e.,

$$-\dot{p}^{2,\mu}_{u,\eta^2} = \tilde{H}^\mu_y(u, y^\mu_u, p^{2,\mu}_{u,\eta^2}, \eta^2) \ \text{ a.e. on } [0,T], \quad p^{2,\mu}_{u,\eta^2}(T) = \phi^\mu_y(y^\mu_u(T)). \tag{4.19}$$

- $\mathcal{N} : X \to 2^W$, $\mathcal{N}(x) = \{0\} \times (N_{K^-}(\mathrm{d}\dot{\eta}^2) \cap H^2(0,T))$, where

$$N_{K^-}(\mathrm{d}\dot{\eta}^2) = \begin{cases} \{\varphi \in C_-[0,T] : \langle \mathrm{d}\dot{\eta}^2, \varphi\rangle = 0\} & \text{if } \mathrm{d}\dot{\eta}^2 \ge 0, \\ \emptyset & \text{otherwise.} \end{cases}$$

- $\mathcal{L} : X \to W$,

$$\mathcal{L}(x) := \mathcal{F}(\bar{x}, \bar{\mu}) - D_x \mathcal{F}(\bar{x}, \bar{\mu})(x - \bar{x}). \tag{4.20}$$

By Lemma 2.6, we have that $(\bar{u}, \bar{\eta}^2) \in X$ for sufficiently large k, l.

LEMMA 4.10. *Equipped with the norm*

$$\|(u, \xi)\|_X := \|u\|_2 + \|\xi\|_2, \tag{4.21}$$

X is a complete metric space, and

$$\|u\|_\infty \;\leq\; \max\{\sqrt{3/T}\|u\|_2, \sqrt[3]{3k}\|u\|_2^{2/3}\} \qquad \forall u \in Lip_k(0, T). \tag{4.22}$$

Proof. It was shown in [9, Lemma 3.2] that the space $(Lip_k(0, T), \|\cdot\|_2)$ is a complete metric space, and the estimate (4.22) follows from [9, Lemma 3.1]. We show now that $(BV^2_{T,l}[0, T], \|\cdot\|_2)$ is complete as well. Let (ξ_n) be a Cauchy sequence in $BV^2_{T,l}[0, T]$ (for the norm $\|\cdot\|_2$). Since $L^2(0, T)$ is complete, there exists $\tilde{\xi} \in L^2(0, T)$ such that $\xi_n \to \tilde{\xi}$ in L^2. Let us show that the limit point $\tilde{\xi}$ lies in $BV^2_{T,l}[0, T]$. We have that $|\mathrm{d}\dot{\xi}_n|_{\mathcal{M}} \leq l$ for all n, and since $\dot{\xi}_n(T) = 0$, the sequence $(\dot{\xi}_n)$ is bounded in BV for the norm $\|\eta\|_{BV} := \|\eta\|_1 + |\mathrm{d}\eta|_{\mathcal{M}}$. Therefore, by the compactness theorem in BV [1, Theorem 3.23], there exists a subsequence $\xi_{\psi(n)}$ and $\zeta \in BV[0, T]$ such that $\mathrm{d}\dot{\xi}_{\psi(n)} \overset{*}{\rightharpoonup} \mathrm{d}\zeta$ weakly-* in $\mathcal{M}[0, T]$ and $\dot{\xi}_{\psi(n)} \to \zeta$ in L^1. Moreover, using the integration by parts formula in BV [12, p. 154], we obtain that

$$T\zeta(T) = \int_0^T (\zeta(t) - \dot{\xi}_{\psi(n)}(t))\mathrm{d}t + \int_0^T s(\mathrm{d}\zeta(s) - \mathrm{d}\dot{\xi}_{\psi(n)}(s)) \;\to\; 0,$$

and hence $\zeta(T) = 0$. Setting $\hat{\xi}(t) := -\int_t^T \zeta(s)\mathrm{d}s$, we have that $\hat{\xi} \in BV^2_T[0, T]$, and $\xi_{\psi(n)} \to \hat{\xi}$ in L^∞ and a fortiori in L^2. We deduce that necessarily, $\hat{\xi} = \tilde{\xi} \in BV^2_T[0, T]$, the whole sequence $(\mathrm{d}\dot{\xi}_n)$ weakly-* converges to $\mathrm{d}\dot{\tilde{\xi}}$ in $\mathcal{M}[0, T]$, and then

$$|\mathrm{d}\dot{\tilde{\xi}}|_{\mathcal{M}} \;\leq\; \liminf |\mathrm{d}\dot{\xi}_n|_{\mathcal{M}} \;\leq\; l.$$

This shows that $\tilde{\xi} \in BV^2_{T,l}[0, T]$, and hence $(BV^2_{T,l}[0, T], \|\cdot\|_2)$ is a complete metric space. This achieves the proof. □

Note that for all $\xi \in BV^2_{T,l}[0, T]$, we have that $|\mathrm{d}\dot{\xi}|_{\mathcal{M}} \leq l$, and since $\dot{\xi}(T) = 0$, it follows that $\|\dot{\xi}\|_\infty \leq l$, and hence $BV^2_{T,l}[0, T] \subset Lip_l(0, T)$. Therefore, we deduce from (4.22) that

$$\|\xi\|_\infty \;\leq\; \max\{\sqrt{3/T}\|\xi\|_2, \sqrt[3]{3l}\|\xi\|_2^{2/3}\} \qquad \forall \xi \in BV^2_{T,l}[0, T]. \tag{4.23}$$

The space $\tilde{X}$ defined by (4.16) is a closed subset of X, and hence, by Lemma 4.10, $\tilde{X}$ equipped with the norm of X (4.21) is a complete metric space. We need to work with $\tilde{X}$ instead of X in order to obtain the *uniqueness* of a solution of (4.13) in $\tilde{X}$, for small enough $r > 0$. The space of sufficiently smooth variations $\Delta \subset W$, in assumptions (iv) and (v) of Theorem 4.8, is defined by (4.18).

Given a stable extension $(\mathcal{P}^\mu)$ of $(\mathcal{P})$, our formulation is the following: For μ in the neighborhood of $\bar{\mu}$, find $x = (u, \eta^2) \in \tilde{X}$ solution of

$$\mathcal{F}(x, \mu) \in \mathcal{N}(x), \tag{4.24}$$

where $\mathcal{F}$ and $\mathcal{N}$ are defined as above. Then (u, y^μ_u) is a stationary point of $(\mathcal{P}^\mu)$ with alternative multipliers $(p^{2,\mu}_{u,\eta^2}, \eta^2)$ iff $x = (u, \eta^2)$ is solution of (4.24).

5. Stability analysis of linear-quadratic problems. The verification of assumption (iv) of Theorem 4.8 is strongly related to stability analysis of linear-quadratic optimal control problems with a second-order state constraint, which we study in this section. Since these results have their own interest, they are stated independently of the rest of this paper. The problem under consideration is of the form

$$\text{(5.1)}\quad (\mathcal{P}_\delta)\quad \min_{(v,z)\in\mathcal{V}\times\mathcal{Z}} \tfrac{1}{2}\int_0^T (v(t)^\top S(t)v(t) + 2v(t)^\top R(t)z(t) + z(t)^\top Q(t)z(t))\mathrm{d}t$$

$$\text{(5.2)}\qquad + \int_0^T (a(t)z(t) + (b(t)-\gamma(t))v(t))\mathrm{d}t + \tfrac{1}{2}z(T)^\top \Phi z(T)$$

$$\text{(5.3)}\qquad \text{s.t.}\quad \dot{z}(t) = A(t)z(t) + B(t)v(t) \quad \text{a.e. on } [0,T],\quad z(0)=0,$$

$$\text{(5.4)}\qquad C(t)z(t) + d(t) - \zeta(t) \le 0 \quad \text{on } [0,T].$$

The perturbation parameter is here $\delta = (\gamma,\zeta) \in W = L^2(0,T;\mathbb{R}^{m*}) \times H^2(0,T)$, with the norm $\|\delta\|_W = \|\gamma\|_2 + \|\zeta\|_{2,2}$. The control and state spaces for the linearized problem are $\mathcal{V} := L^2(0,T;\mathbb{R}^m)$ and $\mathcal{Z} := H^1(0,T;\mathbb{R}^n)$. The matrix and vectors $S(\cdot), R(\cdot), Q(\cdot), a(\cdot), b(\cdot), A(\cdot), B(\cdot), C(\cdot), d(\cdot)$, of appropriate dimensions, are Lipschitz continuous functions of time. In addition, $C(\cdot)$ and $d(\cdot)$ lie in the space $W^{3,\infty}$. The matrix S and Q are symmetric. We assume, in addition, in this section that (recall (A1))

$$\text{(5.5)}\qquad d(0) < 0.$$

Given $v \in \mathcal{V}$, we denote by z_v the unique solution in $\mathcal{Z}$ of the linearized state equation (5.3). Then we may write $(\mathcal{P}_\delta)$ as follows:

$$(\mathcal{P}_\delta)\qquad \min_{v\in\mathcal{V}} \mathcal{J}^\delta(v),\quad \Gamma^\delta(v) \in K,$$

with $\mathcal{J}^\delta(v) := \int_0^T \{\frac{1}{2}(v^\top Sv + 2v^\top Rz_v + z_v^\top Qz_v) + az + (b-\gamma)v\}\mathrm{d}t + \frac{1}{2}z_v(T)^\top \Phi z_v(T)$, $\Gamma^\delta(v) := Cz_v + d - \zeta$, and $K = C_-[0,T]$.

Assume that $C(t)B(t) \equiv 0$ on $[0,T]$ (state constraint of second-order), and define the matrix

$$C_1(t) := \dot{C}(t) + C(t)A(t),\quad C_2(t) := \dot{C}_1(t) + C_1(t)A(t),\quad N_2(t) := C_1(t)B(t).$$

Then for all $v \in \mathcal{V}$, we have that

$$\frac{\mathrm{d}}{\mathrm{d}t}\{C(t)z_v(t)\} = C_1(t)z_v(t),\qquad \frac{\mathrm{d}^2}{\mathrm{d}t^2}\{C(t)z_v(t)\} = C_2(t)z_v(t) + N_2(t)v(t).$$

The alternative multipliers $(\pi^2,\eta^2) \in W^{1,\infty}(0,T;\mathbb{R}^{n*}) \times BV_T^2[0,T]$ for the linear-quadratic problem are defined by

$$\text{(5.6)}\qquad \eta^1(t) := \int_{(t,T]} \mathrm{d}\eta(s),\qquad \eta^2(t) := \int_t^T \eta^1(s)\mathrm{d}s,$$

$$\text{(5.7)}\qquad \pi^2(t) := \pi(t) - \eta^1(t)C(t) - \eta^2(t)C_1(t),\qquad t\in[0,T].$$

Let $(\bar{v}, \bar{z} = z_{\bar{v}})$ be a stationary point of $(\mathcal{P}_0)$, with multipliers $(\bar{\pi},\bar{\eta})$ and alternative multipliers $(\bar{\pi}^2,\bar{\eta}^2)$. Denote the contact set by $\Omega := \{t\in[0,T] : C(t)\bar{z}(t) + d(t) = 0\}$, and a neighborhood of the contact set by $\Omega_\sigma := \{t\in[0,T] : \mathrm{dist}\{t,\Omega\} < \sigma\}$ for $\sigma > 0$. For linear-quadratic problems, assumptions (A2)–(A3) may be rewritten as follows:

($\tilde{\text{A2}}$) The state constraint is a regular second-order state constraint, i.e., $C(t)B(t) \equiv 0$ on $[0,T]$, and there exists $\beta, \sigma > 0$ (σ satisfying (2.22)) such that

$$|N_2(t)| \geq \beta \quad \text{ on } \Omega_\sigma.$$

($\tilde{\text{A3}}$) The matrix $S(t)$ is uniformly positive definite over $[0,T]$, i.e.,

$$\exists\, \alpha > 0, \quad v^\top S(t) v \geq \alpha |v|^2 \quad \forall t \in [0,T]\ \forall v \in \mathbb{R}^m.$$

Note that by Remark 2.5, ($\tilde{\text{A3}}$) is equivalent to (A3). Assumption ($\tilde{\text{A2}}$), with (5.5), implies the following (cf. Lemma 2.4).

LEMMA 5.1. *Assume that* ($\tilde{\text{A2}}$) *holds. Then there exists a positive constant c such that for all $\varphi \in H^2(0,T)$, there exists $v \in \mathcal{V}$ satisfying*

$$C(t) z_v(t) = \varphi(t) \quad \textit{on } \Omega_\sigma \qquad \textit{and} \qquad \|v\|_2 \leq c \|\varphi\|_{2,2}. \tag{5.8}$$

Therefore ($\tilde{\text{A2}}$) and (5.5) imply that Robinson's constraint qualification holds, and that the multipliers associated with $(\bar{v}, \bar{z})$ are unique.

Propositions 5.2 and 5.3 hold for a larger set of perturbations, more precisely for $\delta = (\gamma, \zeta) \in \hat{W}$, where

$$\hat{W} := L^2(0,T;\mathbb{R}^m) \times C[0,T],$$

equipped with its standard norm $\|\delta\|_{\hat{W}} := \|\gamma\|_2 + \|\zeta\|_\infty$. We have, of course, $W \subset \hat{W}$ with continuous embedding. Identical to Proposition 4.4, we obtain the stability of multipliers for linear-quadratic problems (with a slightly modified statement).

PROPOSITION 5.2. *Let $(\bar{v}, \bar{z})$ be a stationary point of $(\mathcal{P}_0)$ satisfying* ($\tilde{\text{A2}}$). *Then there exists $\nu > 0$ such that for every stationary point (v, z) of $(\mathcal{P}_\delta)$, with (unique) multipliers (π, η) and alternative multipliers (π^2, η^2) defined by (5.7)–(5.6), the following hold:*

(i) *If $\|\delta\|_{\hat{W}}, \|v - \bar{v}\|_2 < \nu$, then $\mathrm{d}\eta$ is uniformly bounded in $\mathcal{M}[0,T]$.*

(ii) *There exists $\kappa > 0$ such that, for all $\|\delta\|_{\hat{W}}, \|v - \bar{v}\|_2 < \nu$, we have*

$$\|\mathrm{d}\eta - \mathrm{d}\bar{\eta}\|_{2,2*},\ \|\eta^2 - \bar{\eta}^2\|_2 \leq \kappa (\|v - \bar{v}\|_2 + \|\delta\|_{\hat{W}}).$$

Moreover, when $\|\delta\|_{\hat{W}}, \|v - \bar{v}\|_2 \to 0$:

(iii) *$\mathrm{d}\eta$ weakly-* converges to $\mathrm{d}\bar{\eta}$ in $\mathcal{M}[0,T]$;*

(iv) *$\eta^1 \to \bar{\eta}^1$ in L^1;*

(v) *π^2 and η^2 converge uniformly to $\bar{\pi}^2$ and $\bar{\eta}^2$, respectively.*

Second-order optimality conditions. Let $\tilde{\mathcal{Q}}$ denote the quadratic part of the cost $\mathcal{J}^\delta$ (independent of δ):

$$\begin{aligned} \tilde{\mathcal{Q}}(v) \quad &= \quad \tfrac{1}{2} \int_0^T (v(t)^\top S(t) v(t) + 2 v(t)^\top R(t) z_v(t) + z_v(t)^\top Q(t) z_v(t)) \mathrm{d}t \\ &\quad + \tfrac{1}{2} z_v(T)^\top \Phi z_v(T). \end{aligned} \tag{5.9}$$

The strong second-order sufficient condition is

$$\tilde{\mathcal{Q}}(v) > 0 \qquad \forall v \in \mathcal{V} \setminus \{0\} \text{ such that } C(t) z_v(t) = 0 \text{ on } \operatorname{supp}(\mathrm{d}\bar{\eta}). \tag{5.10}$$

Identical to Proposition 4.2, we obtain that the second-order sufficient condition (5.10) implies the uniform second-order growth condition for the perturbed problems $(\mathcal{P}_\delta)$ (here again the statement is slightly modified).

Proposition 5.3. *Let $(\bar{v}, \bar{z})$ be a stationary point of $(\mathcal{P}_0)$ satisfying* $(\tilde{\mathrm{A}2})$–$(\tilde{\mathrm{A}3})$ *and the strong second-order sufficient condition* (5.10). *Then there exist $c, \rho > 0$ and a neighborhood $\mathcal{W}$ of 0 in $\hat{W}$, such that for all $\delta \in \mathcal{W}$ and any stationary point (v_δ, z_δ) of $(\mathcal{P}_\delta)$ with $\|v_\delta - \bar{v}\|_2 < \rho$,*

$$\mathcal{J}^\delta(v) \geq \mathcal{J}^\delta(v_\delta) + c\|v - v_\delta\|_2^2 \quad \forall\, v \in \mathcal{V} : \Gamma^\delta(v) \in K,\ \|v - \bar{v}\|_2 < \rho. \tag{5.11}$$

Stability analysis. The main result of this section is the theorem below. The key point is to show the existence of a stationary point for the perturbed linear-quadratic problem under the weak second-order sufficient condition (5.10), where the active constraints are taken into account. To this end, the uniform growth condition (Proposition 5.3), together with an abstract theorem from Bonnans and Shapiro [6, Theorem 5.17 and Remark 5.19], is used.

Theorem 5.4. *Let $(\bar{v}, \bar{z})$ be a stationary point of $(\mathcal{P}_0)$ satisfying* $(\tilde{\mathrm{A}2})$–$(\tilde{\mathrm{A}3})$ *and the strong second-order sufficient condition* (5.10). *Then there exist $c, \rho, \lambda > 0$ and a neighborhood $\mathcal{W}$ of 0 in W, such that for all $\delta \in \mathcal{W}$, $(\mathcal{P}_\delta)$ has a unique stationary point (v_δ, z_{v_δ}) with $\|v_\delta - \bar{v}\|_2 < \rho$ and unique associated alternative multipliers $(\pi^2_\delta, \eta^2_\delta)$, and*

$$\|v_\delta - v_{\delta'}\|_2 + \|\eta^2_\delta - \eta^2_{\delta'}\|_2 \ \leq\ \lambda\|\delta - \delta'\|_W \quad \forall\, \delta, \delta' \in \mathcal{W}. \tag{5.12}$$

Moreover, (v_δ, z_{v_δ}) is a local solution of $(\mathcal{P}_\delta)$ satisfying the uniform quadratic growth condition (5.11).

Proof. Let us show the existence of a stationary point of problem $(\mathcal{P}_\delta)$. We may write $(\mathcal{P}_\delta)$ as

$$(\mathcal{P}_\delta) \quad \min_{v \in \mathcal{V}}\ \tfrac{1}{2}\langle v, \mathcal{A}v\rangle + \langle b, v\rangle - \langle \gamma, v\rangle \qquad \text{s.t.} \qquad \mathcal{C}v + d - \zeta \in K,$$

where $\mathcal{A}$ is the continuous, self-adjoint bilinear operator over $\mathcal{V}$ associated with the quadratic form (5.9), b is an element in $\mathcal{V}^* \equiv \mathcal{V}$, $\mathcal{C} : v \mapsto Cz_v$ is a linear continuous operator $\mathcal{V} \to C[0, T]$, and $d \in H^2(0, T)$. Here, without ambiguity, we also denote by $\langle \cdot, \cdot \rangle$ the scalar product over $\mathcal{V}$.

Step 1. Reduction to a fixed feasible set. Let us first consider perturbations of the cost function only, i.e., consider the problem $(\mathcal{P}_\gamma)$ defined by

$$(\mathcal{P}_\gamma) \quad \min_{v \in \mathcal{V}}\ \tfrac{1}{2}\langle v, \mathcal{A}v\rangle + \langle b, v\rangle - \langle \gamma, v\rangle \qquad \text{s.t.} \qquad \mathcal{C}v + d \in K.$$

By Proposition 5.3, the uniform second-order growth condition holds for $(\mathcal{P}_\gamma)$, so does Robinson's constraint qualification by $(\tilde{\mathrm{A}2})$, and the perturbed problem $(\mathcal{P}_\gamma)$ includes the so-called *tilt perturbation* (see [6, p. 416]), i.e., additive perturbations of the cost function of type $-\langle \gamma, v\rangle$ with $\gamma \in \mathcal{V}^*$. Therefore, it follows from [6, Theorem 5.17 and Remark 5.19], since the feasible set of $(\mathcal{P}_\gamma)$ is constant, that there exist $\rho_1, \rho_2 > 0$ and a constant $\lambda > 0$, such that for all $\gamma \in B_2(0, \rho_2)$, $(\mathcal{P}_\gamma)$ has a unique stationary point v_γ in $B_2(\bar{v}, \rho_1)$, and

$$\|v_\gamma - v_{\gamma'}\|_2 \ \leq\ \lambda\|\gamma - \gamma'\|_2 \quad \forall\, \gamma, \gamma' \in B_2(0, \rho_2). \tag{5.13}$$

We have of course that $\bar{v} = v_0$.

Step 2. Existence of a stationary point of $(\mathcal{P}_\delta)$. Now let $\delta = (\gamma, \zeta) \in W$. By Lemma 5.1, there exists $v_\zeta \in \mathcal{V}$ such that

$$(\mathcal{C}v_\zeta)(t) = \zeta(t) \quad \text{on } \Omega_\sigma \qquad \text{and} \qquad \|v_\zeta\|_2 \leq c\|\zeta\|_{2,2}.$$

Set $\tilde{\gamma} := \gamma - \mathcal{A}v_\zeta$. We have that $\|\tilde{\gamma}\|_2 \leq \|\gamma\|_2 + c\|\mathcal{A}\|\|\zeta\|_{2,2} < \rho_2$ if $\|\delta\|_W$ is small enough. Therefore, there exists a (unique) stationary point $v_{\tilde{\gamma}} \in B_2(\bar{v}, \rho_1)$ of $(\mathcal{P}_{\tilde{\gamma}})$, with multiplier $\mathrm{d}\eta_{\tilde{\gamma}} \in \mathcal{M}[0,T]$, satisfying the first-order optimality condition

$$\begin{cases} \mathcal{A}v_{\tilde{\gamma}} + b - \tilde{\gamma} + \mathcal{C}^\top \mathrm{d}\eta_{\tilde{\gamma}} = 0, \\ \mathcal{C}v_{\tilde{\gamma}} + d \leq 0 \text{ on } [0,T], \quad \mathrm{d}\eta_{\tilde{\gamma}} \geq 0, \quad \langle \mathrm{d}\eta_{\tilde{\gamma}}, \mathcal{C}v_{\tilde{\gamma}} + d\rangle = 0. \end{cases} \tag{5.14}$$

Since $\|\mathcal{C}v_{\tilde{\gamma}} - \mathcal{C}\bar{v}\|_\infty \leq \|\mathcal{C}\|\|v_{\tilde{\gamma}} - \bar{v}\|_2 \leq \lambda\|\mathcal{C}\|\|\tilde{\gamma}\|_2$ by (5.13), if $\|\delta\|_W$ is small enough, then the contact set of $\mathcal{C}v_{\tilde{\gamma}} + d$ is included in Ω_σ, and hence

$$\operatorname{supp}(\mathrm{d}\eta_{\tilde{\gamma}}) \subset \Omega_\sigma. \tag{5.15}$$

Let $v_\delta := v_{\tilde{\gamma}} + v_\zeta$ and $\mathrm{d}\eta_\delta := \mathrm{d}\eta_{\tilde{\gamma}}$. Note that there exists a constant $a > 0$ such that $(\mathcal{C}\bar{v})(t) + d(t) < -a$ on $[0,T] \setminus \Omega_\sigma$. Therefore, on $[0,T] \setminus \Omega_\sigma$, we obtain that (we denote in what follows by C different positive constants)

$$\begin{aligned} \mathcal{C}v_\delta + d - \zeta &= \mathcal{C}\bar{v} + d - \zeta + \mathcal{C}v_\zeta + \mathcal{C}(v_{\tilde{\gamma}} - \bar{v}) \\ &\leq -a + \|\zeta\|_\infty + \|\mathcal{C}v_\zeta\|_\infty + \|\mathcal{C}(v_{\tilde{\gamma}} - \bar{v})\|_\infty \\ &\leq -a + C\|\zeta\|_{2,2} + \|\mathcal{C}\|\|v_\zeta\|_2 + \|\mathcal{C}\|\|v_{\tilde{\gamma}} - \bar{v}\|_2 \\ &\leq -a + (C + c\|\mathcal{C}\|)\|\zeta\|_{2,2} + \lambda\|\mathcal{C}\|\|\tilde{\gamma}\|_2 \;\leq\; -a + C\|\delta\|_W, \end{aligned}$$

and hence, if $\|\delta\|_W$ is small enough, then $\mathcal{C}v_\delta + d - \zeta < 0$ on $[0,T] \setminus \Omega_\sigma$. On Ω_σ, we have that $\mathcal{C}v_\delta + d - \zeta = \mathcal{C}v_{\tilde{\gamma}} + d \leq 0$. Therefore, using (5.14) and (5.15), v_δ obviously satisfies

$$\begin{cases} \mathcal{A}v_\delta + b - \gamma + \mathcal{C}^\top \mathrm{d}\eta_\delta = 0, \\ \mathcal{C}v_\delta + d - \zeta \leq 0 \text{ on } [0,T], \quad \mathrm{d}\eta_\delta \geq 0, \quad \langle \mathrm{d}\eta_\delta, \mathcal{C}v_\delta + d - \zeta\rangle = 0, \end{cases}$$

i.e., v_δ is a stationary point of $(\mathcal{P}_\delta)$, with multiplier $\mathrm{d}\eta_\delta$. Consequently, for $\rho_3 > 0$ small enough, reducing ρ_1 if necessary, $(\mathcal{P}_\delta)$ has, for all $\delta \in B_W(0, \rho_3)$, a (*necessarily unique* by Proposition 5.3) stationary point $v_\delta \in B_2(\bar{v}, \rho_1)$, with (unique) multiplier $\mathrm{d}\eta_\delta$. That (v_δ, z_{v_δ}) is a local solution of $(\mathcal{P}_\delta)$ satisfying the uniform growth condition (5.11) follows then from Proposition 5.3.

Step 3. Lipschitz continuity of the stationary point. Let $\delta_i = (\gamma_i, \zeta_i) \in B_W(0, \rho_3)$, $i = 1, 2$, and v_{ζ_i} be such that

$$\mathcal{C}v_{\zeta_i} = \zeta_i \text{ on } \Omega_\sigma, \; i = 1, 2, \quad \text{and} \quad \|v_{\zeta_1}\|_2 \leq c\|\zeta_1\|_{2,2}, \; \|v_{\zeta_1} - v_{\zeta_2}\|_2 \leq c\|\zeta_1 - \zeta_2\|_{2,2}.$$

It follows that $\|v_{\zeta_2}\|_2 \leq c(2\|\zeta_1\|_{2,2} + \|\zeta_2\|_{2,2}) < 3c\rho_3$. Setting $\tilde{\gamma}_i := \gamma_i - \mathcal{A}v_{\zeta_i}$, we obtain as before that if ρ_3 is small enough, then the unique stationary point v_i of $(\mathcal{P}_{\delta_i})$ is given by $v_i = v_{\zeta_i} + v_{\tilde{\gamma}_i}$. Therefore, using (5.13),

$$\begin{aligned} \|v_1 - v_2\|_2 &\leq \|v_{\zeta_1} - v_{\zeta_2}\|_2 + \lambda\|\tilde{\gamma}_1 - \tilde{\gamma}_2\|_2 \\ &\leq c(1 + \lambda\|\mathcal{A}\|)\|\zeta_1 - \zeta_2\|_{2,2} + \lambda\|\gamma_1 - \gamma_2\|_2 \\ &\leq C\|\delta_1 - \delta_2\|_W. \end{aligned} \tag{5.16}$$

Step 4. Lipschitz continuity of the alternative multiplier η^2_δ given by (5.6). Using the above notation, denote by $\mathrm{d}\eta_i$ the (unique) multiplier associated with v_i, and by η^2_i the associated alternative multiplier. Since $-\mathcal{C}^\top(\mathrm{d}\eta_2 - \mathrm{d}\eta_1) = \mathcal{A}(v_2 - v_1) + \gamma_2 - \gamma_1$, we have, for all $v \in \mathcal{V}$,

$$|\langle \mathrm{d}\eta_2 - \mathrm{d}\eta_1, \mathcal{C}v\rangle| \;\leq\; (\|\mathcal{A}\|\|v_2 - v_1\|_2 + \|\gamma_2 - \gamma_1\|_2)\|v\|_2. \tag{5.17}$$

By Lemma 5.1, for all $\varphi \in H^2(0,T)$, there exists $v \in \mathcal{V}$ such that $\mathcal{C}v = \varphi$ on Ω_σ and $\|v\|_2 \le c\|\varphi\|_{2,2}$. It follows from (5.15) that $\int_0^T \varphi(t)(\mathrm{d}\eta_2(t) - \mathrm{d}\eta_1(t)) = \langle \mathrm{d}\eta_2 - \mathrm{d}\eta_1, \mathcal{C}v\rangle$. Therefore, we obtain in view of (5.17) that

$$\|\mathrm{d}\eta_2 - \mathrm{d}\eta_1\|_{2,2*} = \sup_{\varphi \in H^2, \varphi \not\equiv 0} \frac{|\int_0^T \varphi(t)(\mathrm{d}\eta_2(t) - \mathrm{d}\eta_1(t))|}{\|\varphi\|_{2,2}} \le c(\|\mathcal{A}\|\|v_2 - v_1\|_2 + \|\gamma_2 - \gamma_1\|_2).$$

Since $\|\eta_2^2 - \eta_1^2\|_2 \le C\|\mathrm{d}\eta_2 - \mathrm{d}\eta_1\|_{2,2*}$ by Lemma 4.5, the above estimate, together with (5.16), shows the existence of a constant $\lambda > 0$ such that (5.12) holds and achieves the proof of Theorem 5.4. □

6. Proof of Theorem 4.3. In order to prove Theorem 4.3, we have to show that assumptions (iii), (iv), and (v) of Theorem 4.8 are satisfied, which is done, respectively, in Lemmas 6.1 to 6.3. Throughout this section, the assumptions of Theorem 4.3 are assumed to hold. We consider a stable extension $(\mathcal{P}^\mu)$ of $(\mathcal{P})$, and we use the notations defined in subsection 4.3. Moreover, the following notations are used throughout this section (time dependence is omitted):

$$\begin{gathered}
S := \tilde{H}_{uu}(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2), \qquad R := \tilde{H}_{uy}(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2), \qquad Q := \tilde{H}_{yy}(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2),\\
A := f_y(\bar{u}, \bar{y}), \qquad B := f_u(\bar{u}, \bar{y}), \qquad \Phi := \phi_{yy}(\bar{y}(T)),\\
C := g_y(\bar{y}), \qquad d := g(\bar{y}), \qquad C_1 = g_y^{(1)}(\bar{y}),\\
C_2 := g_y^{(2)}(\bar{u}, \bar{y}), \qquad N_2 := g_u^{(2)}(\bar{u}, \bar{y}), \qquad a := -C_2\bar{\eta}^2, \qquad b := -N_2\bar{\eta}^2.
\end{gathered}$$

All the above quantities are bounded and Lipschitz continuous over $[0,T]$.

Let us first make explicit the expression of the derivative $D_x\mathcal{F}(\bar{x}, \bar{\mu})(x - \bar{x})$ involved in the definition (4.20) of $\mathcal{L}(x)$, with $x = (u, \eta^2)$ and $\bar{x} = (\bar{u}, \bar{\eta}^2)$. Note that the Fréchet derivative of the mapping $(u, \mu) \mapsto y_u^\mu$ w.r.t. u in direction v is the solution $z_{u,v}^\mu$ of

$$\dot{z}_{u,v}^\mu = f_y^\mu(u, y_u^\mu) z_{u,v}^\mu + f_u^\mu(u, y_u^\mu) v, \qquad z_{u,v}^\mu(0) = 0$$

and that of the mapping $(x, \mu) \mapsto p_x^{2,\mu}$ (recall that $p_x^{2,\mu}$ is the solution of (4.19)) w.r.t. $x = (u, \eta^2)$ in direction $h = (v, \xi)$ is the solution $\pi_{x,h}^{2,\mu}$ of (omitting the arguments $(u, y_u^\mu, p_x^{2,\mu}, \eta^2)$)

$$\begin{aligned}
-\dot{\pi}_{x,h}^{2,\mu} &= \tilde{H}_{yu}^\mu v + \tilde{H}_{yy}^\mu z_{u,v}^\mu + \pi_{x,h}^{2,\mu} f_y^\mu + \xi (g^\mu)_y^{(2)},\\
\pi_{x,h}^{2,\mu}(T) &= \phi_{yy}^\mu(y_u^\mu(T)) z_{u,v}^\mu(T).
\end{aligned} \tag{6.1}$$

Applications of Gronwall's lemma shows that, for μ in a neighborhood of $\bar{\mu}$, $x = (u, \eta^2)$ in a L^∞-neighborhood of $\bar{x} = (\bar{u}, \bar{\eta}^2)$ and a direction $h = (v, \xi) \in X$,

$$\|z_{u,v}^\mu\|_\infty = \mathcal{O}(\|v\|_2), \qquad \|\pi_{x,h}^{2,\mu}\|_\infty = \mathcal{O}(\|h\|_X), \tag{6.2}$$

$$\|z_{u,v}^\mu - z_{\bar{u},v}^{\bar{\mu}}\|_\infty = \mathcal{O}(\|u - \bar{u}\|_2 + \|\mu - \bar{\mu}\|)\|v\|_2, \tag{6.3}$$

$$\|\pi_{x,h}^{2,\mu} - \pi_{\bar{x},h}^{2,\bar{\mu}}\|_\infty = \mathcal{O}(\|x - \bar{x}\|_X + \|\mu - \bar{\mu}\|)\|h\|_X. \tag{6.4}$$

By the chain rule, we obtain that

$$D_x\mathcal{F}(\bar{x}, \bar{\mu})(x - \bar{x}) = \begin{pmatrix} S(u - \bar{u}) + R z_{u - \bar{u}} + \pi_{u - \bar{u}, \eta^2 - \bar{\eta}^2}^2 B + (\eta^2 - \bar{\eta}^2) N_2 \\ C z_{u - \bar{u}} \end{pmatrix},$$

where $z_{u-\bar{u}} := z^{\bar{\mu}}_{\bar{u},u-\bar{u}}$ is the solution of (5.3) for $v = u-\bar{u}$, and $\pi^2_{u-\bar{u},\eta^2-\bar{\eta}^2} := \pi^{2,\bar{\mu}}_{\bar{x},(x-\bar{x})}$ is the solution of (6.1) for $(v,\xi) = (u-\bar{u}, \eta^2-\bar{\eta}^2)$:

$$-\dot{\pi}^2_{v,\xi} = R^\top v + Qz_v + \pi^2_{v,\xi}A + \xi C_2, \qquad \pi^2_{v,\xi}(T) = \Phi z_v(T).$$

Set $v := u-\bar{u}$, and let $\delta = (\gamma,\zeta) \in \Delta$. Then (4.13) has a unique solution $x = (u,\eta^2) \in \tilde{X}$ iff the system of equations below has a unique solution (v,z,π^2,η^2) with $(\bar{u}+v,\eta^2) \in \tilde{X}$:

$$\begin{aligned}
\dot{z} &= Az + Bv, \qquad z(0) = 0,\\
-\dot{\pi}^2 &= R^\top v + Qz + \pi^2 A + \eta^2 C_2 - \bar{\eta}^2 C_2, \qquad \pi^2(T) = \Phi z(T),\\
0 &= Sv + Rz + \pi^2 B + \eta^2 N_2 - \bar{\eta}^2 N_2 - \gamma,\\
0 &\geq d + Cz - \zeta, \qquad \mathrm{d}\dot{\eta}^2 \geq 0, \qquad \langle \mathrm{d}\dot{\eta}^2, d + Cz - \zeta\rangle = 0.
\end{aligned}$$

We recognize the first-order necessary optimality condition of linear-quadratic problem $(\mathcal{P}_\delta)$ *in its alternative form.* That is, setting $\mathrm{d}\eta = \mathrm{d}\dot{\eta}^2$ and $\pi = \pi^2 - C\dot{\eta}^2 + C_1\eta^2$, we recover the "classical" optimality conditions of $(\mathcal{P}_\delta)$ (note that $C_1 = \dot{C} + CA$, $C_2 = \dot{C}_1 + C_1A$, $N_2 = C_1B$, and $CB = g^{(1)}_u(\bar{u},\bar{y}) \equiv 0$):

$$\begin{aligned}
\dot{z} &= Az + Bv, \qquad z(0) = 0,\\
-\mathrm{d}\dot{\pi} &= (R^\top v + Qz + \pi A - \bar{\eta}^2 C_2)\mathrm{d}t + C\mathrm{d}\eta, \qquad \pi(T) = \Phi z(T),\\
0 &= Sv + Rz + \pi B - \bar{\eta}^2 N_2 - \gamma,\\
0 &\geq d + Cz - \zeta, \qquad \mathrm{d}\eta \geq 0, \qquad \langle \mathrm{d}\eta, d + Cz - \zeta\rangle = 0.
\end{aligned}$$

We see then that $(\bar{v},\bar{z}) := 0$ is a stationary point of $(\mathcal{P}_0)$, with alternative multipliers $\bar{\pi}^2 := 0$ and $\bar{\eta}^2$, and classical multipliers $\bar{\pi} := -C\dot{\bar{\eta}}^2 + C_1\bar{\eta}^2$ and $\bar{\eta} = \dot{\bar{\eta}}^2$. The second-order optimality condition (3.6), with the quadratic cost expressed by (3.7), is precisely the condition (5.10) and implies that $(\bar{v},\bar{z}) = 0$ is a local solution of $(\mathcal{P}_0)$.

The verifications of assumptions (iii) and (v) in Lemmas 6.1 and 6.3 are only technical, and for assumption (iv) in Lemma 6.2, we use Theorem 5.4.

LEMMA 6.1. *The mapping $\Psi^\mu = \mathcal{F}(\cdot,\mu) - \mathcal{L}(\cdot)$ is strictly stationary at $x = \bar{x}$, uniformly in μ near $\bar{\mu}$.*

Proof. Let $x_1, x_2 \in X$ and $\mu \in P$. We have that

$$\begin{aligned}
\Psi^\mu(x_1) - \Psi^\mu(x_2) &= \mathcal{F}(x_1,\mu) - \mathcal{F}(x_2,\mu) - D_x\mathcal{F}(\bar{x},\bar{\mu})(x_1-x_2)\\
&= \int_0^1 (D_x\mathcal{F}(\theta x_1 + (1-\theta)x_2,\mu) - D_x\mathcal{F}(\bar{x},\bar{\mu}))\mathrm{d}\theta(x_1-x_2).
\end{aligned}$$

Let $x = (u,\eta^2) \in \tilde{X}$. Then by (4.22)–(4.23), if x is close to $\bar{x} = (\bar{u},\bar{\eta}^2)$ for the norm of X, this implies that (u,η^2) belongs to a L^∞-neighborhood of $(\bar{u},\bar{\eta}^2)$. Hence, y^μ_u and $p^{2,\mu}_{u,\eta^2}$ remain also uniformly bounded for μ in a neighborhood of $\bar{\mu}$. Let $x_i = (u_i,\eta^2_i) \in X$, $i = 1,2$, and given $\theta \in [0,1]$, write $x_\theta := \theta x_1 + (1-\theta)x_2$ and similarly for the other variables. Set

$$\begin{pmatrix} r_1 \\ r_2 \end{pmatrix} := (D_x\mathcal{F}(x_\theta,\mu) - D_x\mathcal{F}(\bar{x},\bar{\mu}))(x_1-x_2).$$

Let us express the first row r_1. Denoting by $(\cdot)$ the arguments $(u_\theta, y^\mu_{u_\theta}, p^{2,\mu}_{x_\theta}, \eta^2_\theta)$, we obtain that

$$\begin{aligned}
r_1 = {}&(\tilde{H}^\mu_{uu}(\cdot) - S)(u_1-u_2) + (\tilde{H}^\mu_{uy}(\cdot)z^\mu_{u_\theta,u_1-u_2} - Rz^{\bar{\mu}}_{\bar{u},u_1-u_2})\\
&+ (\pi^{2,\mu}_{x_\theta,x_1-x_2}f^\mu_u(\cdot) - \pi^{2,\bar{\mu}}_{\bar{x},x_1-x_2}B) + (\eta^2_1 - \eta^2_2)((g^\mu)^{(2)}_u(\cdot) - N_2).
\end{aligned}$$

For (u_i, η_i^2) in a L^∞-neighborhood of $(\bar{u}, \bar{\eta}^2)$ and μ in the neighborhood of $\bar{\mu}$, we have that $\tilde{H}_{uu}^\mu(\cdot) - S = \tilde{H}_{uu}^\mu(u_\theta, y_{u_\theta}^\mu, p_{x_\theta}^{2,\mu}, \eta_\theta^2) - \tilde{H}_{uu}^{\bar{\mu}}(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2)$ is arbitrarily small in the L^∞ norm, and similarly for the terms involving the other derivatives, $\tilde{H}_{uy}^\mu$, f_u^μ, and $(g^\mu)_u^{(2)}$. Therefore, given any $\varepsilon > 0$, for $\|x_i - \bar{x}\|_X, \|\mu - \bar{\mu}\|$ small enough,

$$\begin{aligned} \|r_1\|_2 \le \varepsilon(&\|u_1 - u_2\|_2 + \|z_{u_\theta,u_1-u_2}^\mu\|_2 + \|\pi_{x_\theta,x_1-x_2}^{2,\mu}\|_2 + \|\eta_1^2 - \eta_2^2\|_2) \\ &+ \|R\|_\infty \|z_{u_\theta,u_1-u_2}^\mu - z_{\bar{u},u_1-u_2}^{\bar{\mu}}\|_2 + \|B\|_\infty \|\pi_{x_\theta,x_1-x_2}^{2,\mu} - \pi_{\bar{x},x_1-x_2}^{2,\bar{\mu}}\|_2. \end{aligned}$$

Using (6.2)–(6.4) with $x = x_\theta$ and $h = x_1 - x_2$, we obtain that $\|r_1\|_2 \le \varepsilon \|x_1 - x_2\|_X$, whenever x_1, x_2 are close enough to $\bar{x}$ in X and μ is close enough to $\bar{\mu}$. For the second row r_2, we have that

$$\begin{aligned} r_2 &= g_y^\mu(y_{u_\theta}^\mu) z_{u_\theta,u_1-u_2}^\mu - g_y^{\bar{\mu}}(\bar{y}) z_{\bar{u},u_1-u_2}^{\bar{\mu}}, \\ \dot{r}_2 &= (g^\mu)_y^{(1)}(y_{u_\theta}^\mu) z_{u_\theta,u_1-u_2}^\mu - (g^{\bar{\mu}})_y^{(1)}(\bar{y}) z_{\bar{u},u_1-u_2}^{\bar{\mu}}, \\ \ddot{r}_2 &= ((g^\mu)_u^{(2)}(u_\theta, y_{u_\theta}^\mu) - (g^{\bar{\mu}})_u^{(2)}(\bar{u}, \bar{y}))(u_1 - u_2) \\ &\quad + (g^\mu)_y^{(2)}(u_\theta, y_{u_\theta}^\mu) z_{u_\theta,u_1-u_2}^\mu - (g^{\bar{\mu}})_y^{(2)}(\bar{u}, \bar{y}) z_{\bar{u},u_1-u_2}^{\bar{\mu}}. \end{aligned}$$

Therefore, we conclude with the same arguments that $\|r_2\|_{2,2} \le \varepsilon \|u_1 - u_2\|_2$, whenever $\|x_i - \bar{x}\|_X$, $i = 1, 2$, and $\|\mu - \bar{\mu}\|$ are small enough. This shows the desired property. □

LEMMA 6.2. *For k sufficiently large w.r.t. l in definition* (4.15) *of the space X, r small enough in definition* (4.16) *of the space $\tilde{X}$, and small enough positive constants ϱ and k' in definition* (4.18) *of the set Δ,* (4.13) *has a unique solution $x_\delta = (u_\delta, \eta_\delta^2)$ in $\tilde{X}$, for all $\delta \in \Delta$, and this solution is Lipschitz continuous w.r.t. δ.*

Proof. We have that $x = (u, \eta^2)$ is solution of (4.13) iff $(v := u - \bar{u}, z_v)$ is solution of the first-order optimality condition of $(\mathcal{P}_\delta)$ with alternative multipliers $\pi_{v,\eta^2-\bar{\eta}^2}^2$ and η^2. By the hypotheses of Theorem 4.3, $(\bar{v}, \bar{z}) = 0$ is a stationary point of $(\mathcal{P}_0)$ satisfying the assumptions of Theorem 5.4. Choose ϱ small enough, so that $B_W(0, \varrho)$ is included in the neighborhood $\mathcal{W}$ of Theorem 5.4. By this theorem, for all $\delta \in B_W(0, \varrho)$, $(\mathcal{P}_\delta)$ has a unique stationary point (v_δ, z_{v_δ}) with $\|v_\delta\|_2 < \rho$ and unique associated alternative multipliers $(\pi_{v_\delta,\eta_\delta^2-\bar{\eta}^2}^2, \eta_\delta^2)$. Therefore, (4.13) has a unique solution $(u_\delta := \bar{u} + v_\delta, \eta_\delta^2)$ with $\|u_\delta - \bar{u}\|_2 < \rho$. We have to show that $(u_\delta, \eta_\delta^2)$ belongs to the space $\tilde{X}$. Throughout the proof, we denote by C different positive constants.

By Proposition 5.2(i), shrinking ϱ if necessary, we immediately obtain that η_δ^2 belongs to the space $BV_{T,l}^2[0, T]$, for large enough l. Therefore, by (4.23) and (5.12), for all $\delta \in B_W(0, \varrho)$,

$$\|\eta_\delta^2 - \bar{\eta}^2\|_\infty \;\le\; \sqrt[3]{6l} \|\eta_\delta^2 - \bar{\eta}^2\|_2^{2/3} \;\le\; \sqrt[3]{6l} \lambda^{2/3} \|\delta\|_W^{2/3}.$$

For $\delta = (\gamma, \zeta) \in \Delta$ (then $\gamma \in Lip_{k'}$), let us show now that $u_\delta = \bar{u} + v_\delta \in Lip_k$. From the first-order alternative optimality condition of $(\mathcal{P}_\delta)$, we have that

$$S v_\delta + R z_{v_\delta} + \pi_{v_\delta,\eta_\delta^2-\bar{\eta}^2}^2 B + N_2(\eta_\delta^2 - \bar{\eta}^2) - \gamma = 0. \tag{6.5}$$

Since S is uniformly invertible by (A3), using (6.2), (5.12), and (4.22), we deduce that

$$\begin{aligned} \|v_\delta\|_\infty &\le C(\|z_{v_\delta}\|_\infty + \|\pi_{v_\delta,\eta_\delta^2-\bar{\eta}^2}^2\|_\infty + \|\eta_\delta^2 - \bar{\eta}^2\|_\infty) + \|\gamma\|_\infty \\ &\le C(2\lambda\|\delta\|_W + \sqrt[3]{6l}\lambda^{2/3}\|\delta\|_W^{2/3}) + \sqrt[3]{3k'}\|\gamma\|_2^{2/3} \\ &\le (C(l) + \sqrt[3]{3k'})\|\delta\|_W^{2/3}. \end{aligned}$$

We denote here and in what follows by $C(l)$ different positive constants that depend on l (but not on k). Since $\gamma \in Lip_{k'}$, $\eta^2_\delta, \bar{\eta}^2 \in BV^2_{T,l} \subset Lip_l$, $z_{v_\delta}, \pi^2_{v_\delta,\eta^2_\delta-\bar{\eta}^2} \in W^{1,\infty}$, S, R, B, N_2 are Lipschitz continuous, and S is uniformly invertible, we can differentiate (6.5) in time and get

$$S\dot{v}_\delta + \dot{S}v_\delta + R\dot{z}_{v_\delta} + \dot{R}z_{v_\delta} + \dot{\pi}^2_{v_\delta,\eta^2_\delta-\bar{\eta}^2}B + \pi^2_{v_\delta,\eta^2_\delta-\bar{\eta}^2}\dot{B} + N_2(\dot{\eta}^2 - \dot{\bar{\eta}}^2) + \dot{N}_2(\eta^2 - \bar{\eta}^2) - \dot{\gamma} = 0.$$

Since $\|z_{v_\delta}\|_\infty$, $\|\pi^2_{v_\delta,\eta^2_\delta-\bar{\eta}^2}\|_\infty$, $\|\dot{z}_{v_\delta}\|_\infty$, $\|\dot{\pi}^2_{v_\delta,\eta^2_\delta-\bar{\eta}^2}\|_\infty \leq C(\|v_\delta\|_\infty + \|\eta^2_\delta - \bar{\eta}^2\|_\infty)$, and S has the inverse uniformly bounded over $[0,T]$, whereas $\|\dot{\eta}^2_\delta\|_\infty, \|\dot{\bar{\eta}}^2\|_\infty \leq l$, we obtain that

$$\begin{aligned}\|\dot{v}_\delta\|_\infty &\leq C(\|v_\delta\|_\infty + \|\eta^2_\delta - \bar{\eta}^2\|_\infty + \|\dot{\eta}^2_\delta - \dot{\bar{\eta}}^2\|_\infty) + \|\dot{\gamma}\|_\infty \\ &\leq (C(l) + C\sqrt[3]{3k'})\|\delta\|_W^{2/3} + 2Cl + k'.\end{aligned}$$

Therefore, we have that $\|\dot{v}_\delta\|_\infty \leq k/2$ if, fixing a suitable l, we take k so large that $k > \max\{4Cl; 2\|\dot{\bar{u}}\|_\infty\}$, and choose ϱ and k' in (4.18) small enough. It follows that the solution $x_\delta = (u_\delta = \bar{u} + v_\delta, \eta^2_\delta)$ of (4.13) belongs to the space X. In addition, if we choose $r = \rho$, with the ρ of Theorem 5.4, then $x_\delta \in \tilde{X}$ for $\|\delta\|_W$ small enough, and is the unique solution of (4.13) in $\tilde{X}$. Moreover, by Theorem 5.4,

$$\|u_\delta - u_{\delta'}\|_2 + \|\eta^2_\delta - \eta^2_{\delta'}\|_2 \;\leq\; \lambda\|\delta - \delta'\|_W \quad \forall\, \delta, \delta' \in \Delta.$$

This achieves the proof of assumption (iv) of Theorem 4.8. □

Lemma 6.3. *There exists a neighborhood of* $(\bar{x}, \bar{\mu})$*, such that* $\mathcal{F}(x,\mu) - \mathcal{L}(x)$ *belongs to* Δ *for all* (x,μ) *in this neighborhood.*

Proof. We have to show that for $\|x - \bar{x}\|_X, \|\mu - \bar{\mu}\|$ small enough, $\mathcal{F}(x,\mu) - \mathcal{L}(x) \in \Delta$, where Δ is our set of smooth variations defined by (4.18). Throughout this proof, we denote by C different positive constants. For $\theta \in [0,1]$, set $x_\theta := \theta x + (1-\theta)\bar{x}$ and similarly define μ_θ. We have that

$$\begin{aligned}\mathcal{F}(x,\mu) - \mathcal{L}(x) &= \mathcal{F}(x,\mu) - \mathcal{F}(\bar{x},\bar{\mu}) - D_x\mathcal{F}(\bar{x},\bar{\mu})(x - \bar{x}) \\ &= \int_0^1 (D_x\mathcal{F}(x_\theta,\mu_\theta) - D_x\mathcal{F}(\bar{x},\bar{\mu}))\mathrm{d}\theta(x - \bar{x}) \\ &\quad + \int_0^1 D_\mu\mathcal{F}(x_\theta,\mu_\theta)\mathrm{d}\theta(\mu - \bar{\mu}) \;=:\; \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}.\end{aligned}$$

Let us show that $\|r_1\|_2 + \|r_2\|_{2,2} \leq \varrho$ and $\|\dot{r}_1\|_\infty \leq k'$ for $\|x - \bar{x}\|_X$ and $\|\mu - \bar{\mu}\|$ small enough. By the arguments of Lemma 6.1, given any $\varepsilon > 0$, for $\|x - \bar{x}\|_X$ and $\|\mu - \bar{\mu}\|$ small enough, we have that $\| \int_0^1 (D_x\mathcal{F}(x_\theta,\mu_\theta) - D_x\mathcal{F}(\bar{x},\bar{\mu}))\mathrm{d}\theta(x - \bar{x})\|_W \leq \varepsilon\|x - \bar{x}\|_X$. Moreover, since $D_\mu\mathcal{F}(x,\mu)$ is uniformly bounded for (x,μ) in a neighborhood of $(\bar{x},\bar{\mu})$ by definition of a stable extension, we deduce that

$$\|r_1\|_2 + \|r_2\|_{2,2} \;\leq\; \varepsilon\|x - \bar{x}\|_X + C\|\mu - \bar{\mu}\| \;\leq\; \varrho \tag{6.6}$$

for $\|x - \bar{x}\|_X$ and $\|\mu - \bar{\mu}\|$ small enough. Making now explicit the expression of r_1, we obtain that (recall the notations $S = \tilde{H}^{\bar{\mu}}_{uu}$, $R = \tilde{H}^{\bar{\mu}}_{uy}$, $B = f^{\bar{\mu}}_u$, $N_2 = (g^{\bar{\mu}})^{(2)}_u$)

$$\begin{aligned}r_1 &= \tilde{H}^\mu_u(u, y^\mu_u, p^{2,\mu}_{u,\eta^2}, \eta^2) - \tilde{H}^{\bar{\mu}}_u(\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2) - S(u - \bar{u}) - Rz_{u-\bar{u}} \\ &\quad - \pi^2_{u-\bar{u},\eta^2-\bar{\eta}^2}B - N_2(\eta^2 - \bar{\eta}^2).\end{aligned}$$

Time derivation yields (omitting arguments and reorganizing the terms)

$$\begin{aligned}\dot{r}_1 = {} & (\tilde{H}^{\mu}_{uu} - \tilde{H}^{\bar{\mu}}_{uu})\dot{u} + (\tilde{H}^{\mu}_{uy} f^{\mu} - \tilde{H}^{\bar{\mu}}_{uy} f^{\bar{\mu}}) - (\tilde{H}^{\mu}_{y} f^{\mu}_{u} - \tilde{H}^{\bar{\mu}}_{y} f^{\bar{\mu}}_{u}) + ((g^{\mu})^{(2)}_{u} - (g^{\bar{\mu}})^{(2)}_{u})\dot{\eta}^2 \\ & - R\dot{z}_{u-\bar{u}} - \dot{\pi}^2_{u-\bar{u},\eta^2-\bar{\eta}^2} B - \dot{S}(u-\bar{u}) - \dot{R} z_{u-\bar{u}} - \pi^2_{u-\bar{u},\eta^2-\bar{\eta}^2} \dot{B} - \dot{N}_2(\eta^2 - \bar{\eta}^2).\end{aligned}$$

For (u, η^2) close to $(\bar{u}, \bar{\eta}^2)$ in X, and μ in a neighborhood of $\bar{\mu}$, we have by (4.22)–(4.23) that $\|(u, y^{\mu}_{u}, p^{2,\mu}_{u,\eta^2}, \eta^2) - (\bar{u}, \bar{y}, \bar{p}^2, \bar{\eta}^2)\|_\infty$ is arbitrarily small, and hence, by continuity of $\tilde{H}^{\mu}_{uu}$, etc., given any $\varepsilon > 0$, we obtain that

$$\begin{aligned}\|\dot{r}_1\|_\infty & \leq \varepsilon(\|\dot{u}\|_\infty + \|\dot{\eta}^2\|_\infty + 1) + C(\|\dot{z}_{u-\bar{u}}\|_\infty + \|\dot{\pi}^2_{u-\bar{u},\eta^2-\bar{\eta}^2}\|_\infty) \\ & \qquad + C(\|u-\bar{u}\|_\infty + \|z_{u-\bar{u}}\|_\infty + \|\pi^2_{u-\bar{u},\eta^2-\bar{\eta}^2}\|_\infty + \|\eta^2 - \bar{\eta}^2\|_\infty) \\ & \leq \varepsilon(k + l + 1) + C(\|u - \bar{u}\|_\infty + \|\eta^2 - \bar{\eta}^2\|_\infty) \\ & \leq \varepsilon(k + l + 1) + C(\sqrt[3]{6k} + \sqrt[3]{6l})\|x - \bar{x}\|_X^{2/3} \ \leq \ k',\end{aligned}$$

if $\|x - \bar{x}\|_X$ and $\|\mu - \bar{\mu}\|$ are small enough. It follows that $r_1 \in Lip_{k'}(0, T; \mathbb{R}^m)$, and with (6.6), this achieves the proof. □

Proof of Theorem 4.3. We apply Theorem 4.8 with the spaces X, $\tilde{X}$, W, Δ, P and mappings $\mathcal{F}$, $\mathcal{N}$, $\mathcal{L}$ defined in subsection 4.3. We set $\bar{x} := (\bar{u}, \bar{\eta}^2)$. The assumptions (i) and (ii) of Theorem 4.8 are obviously fulfilled by our hypotheses and the definition of a stable extension. For an appropriate choice of the constants k, l, r, k', ϱ involved in the definition of the spaces X, $\tilde{X}$, and Δ, assumptions (iii), (iv), and (v) hold by Lemmas 6.1, 6.2, and 6.3, respectively. It follows that for all μ in a neighborhood of $\bar{\mu}$, there exist a unique stationary point (u^{μ}, y^{μ}) of $(\mathcal{P}^{\mu})$ and unique associated alternative multipliers $(p^{2,\mu}, \eta^{2,\mu})$ with $(u^{\mu}, \eta^{2,\mu})$ in a X-neighborhood of $\bar{x}$, and (4.14) is satisfied. Since, by definition of a stable extension, $\mathcal{F}$ is Lipschitz continuous w.r.t. μ, uniformly w.r.t. x, this implies that (4.2) holds, while (4.3) follows from (4.22)–(4.23). Finally, by (4.3), taking if necessary a smaller neighborhood of $\bar{\mu}$, u^{μ} belongs to the L^∞-neighborhood of $\bar{u}$ on which the uniform quadratic growth condition holds (Proposition 4.2). Therefore, (u^{μ}, y^{μ}) is the unique stationary point of $(\mathcal{P}^{\mu})$ with u^{μ} in a L^∞-neighborhood of $\bar{u}$ and is a local solution of $(\mathcal{P}^{\mu})$ satisfying (4.1). □

7. Conclusion and remarks.

In this paper, we obtain for the first time stability results for optimal control problems with a state constraint of order greater than one without any assumption on the structure of the contact set. For this we use a generalized implicit function theorem in metric spaces [9] applied to a system equivalent to the first-order optimality condition, involving *alternative multipliers* obtained by integrating the original state constraint multiplier. In the stability analysis of linear-quadratic problems, we use [6, Theorem 5.17] to obtain the existence of a stationary point for the perturbed problem under a weak second-order sufficient condition taking into account the active constraints. In this way the method for weakening the second-order sufficient condition is different from the method used in [21, 20].

Due to the low regularity of state constraint multipliers, we use a framework that differs from the ones used for first-order state constraints in [18] or in [9] in the choice of the spaces for the state constraint and state constraint multiplier. We keep the idea of [9] to use as control space the space of Lipschitz continuous functions with a bound on the Lipschitz constant.

Though the analysis is restricted to a scalar state constraint of second-order, the framework and results presented in this paper have a natural extension to several state constraints of orders ≥ 2 (see Remarks 2.2 and 2.3). Taking into account both components of first-order and higher-order is more delicate since then the arguments used in [18, 9, 20] and in the present paper would have to be combined.

Making additional assumptions on the structure of the contact set, L^∞ Lipschitz stability of solutions can be obtained (see [22, 5]) improving (4.3), as it is the case for first-order state constraints (see [9, section 4]). In [22, 5] it was also shown using a shooting approach that the solutions are directionally differentiable w.r.t. the parameter. It would be interesting as well to obtain sensitivity results without assumption on the structure of the contact set, extending to higher-order state constraints the sensivity results obtained by Malanowski [18] for state constraints of first-order.

Finally, let us note that the second-order sufficient condition (3.6) used in the stability analysis might be weakened by taking into account the curvature term of the constraint (see [2, Theorem 27], [3, Theorem 6.1], and [5, Theorem 4.3]).

Acknowledgments. The author thanks J. F. Bonnans for his comments on the manuscript and the anonymous referees for their useful remarks.

REFERENCES

[1] L. Ambrosio, N. Fusco, and D. Pallara, *Functions of Bounded Variation and Free Discontinuity Problems*, Oxford Mathematical Monographs, The Clarendon Press, Oxford University Press, New York, 2000.

[2] J. F. Bonnans and A. Hermant, *No gap second order optimality conditions for optimal control problems with a single state constraint and control*, Math. Program., 117 (2009), pp. 21–50.

[3] J. F. Bonnans and A. Hermant, *Second-order analysis for optimal control problems with pure state constraints and mixed control-state constraints*, Ann. Inst. H. Poincaré Anal. Non Linéaire, 26 (2009), pp. 561–598.

[4] J. F. Bonnans and A. Hermant, *Stability and sensitivity analysis for optimal control problems with a first-order state constraint and application to continuation methods*, ESAIM Control Optim. Calc. Var., 14 (2008), pp. 825–863.

[5] J. F. Bonnans and A. Hermant, *Well-posedness of the shooting algorithm for state constrained optimal control problems with a single constraint and control*, SIAM J. Control Optim., 46 (2007), pp. 1398–1430.

[6] J. F. Bonnans and A. Shapiro, *Perturbation Analysis of Optimization Problems*, Springer-Verlag, New York, 2000.

[7] B. Bonnard, L. Faubourg, and E. Trélat, *Optimal control of the atmospheric arc of a space shuttle and numerical simulations with multiple-shooting method*, Math. Models Methods Appl. Sci., 15 (2005), pp. 109–140.

[8] A. L. Dontchev and W. W. Hager, *Lipschitzian stability in nonlinear control and optimization*, SIAM J. Control Optim., 31 (1993), pp. 569–603.

[9] A. L. Dontchev and W. W. Hager, *Lipschitzian stability for state constrained nonlinear optimal control*, SIAM J. Control Optim., 36 (1998), pp. 698–718.

[10] A. L. Dontchev and W. W. Hager, *The Euler approximation in state constrained optimal control*, Math. Comp., 70 (2001), pp. 173–203.

[11] A. L. Dontchev, W. W. Hager, A. B. Poore, and B. Yang, *Optimality, stability, and convergence in nonlinear control*, Appl. Math. Optim., 31 (1995), pp. 297–326.

[12] N. Dunford and J. Schwartz, *Linear Operators*, Vols. I and II, Interscience, New York, 1958, 1963.

[13] I. Ekeland, *Nonconvex minimization problems*, Bull. Amer. Math. Soc. (N.S.), 1 (1979), pp. 443–474.

[14] W. W. Hager, *Lipschitz continuity for constrained processes*, SIAM J. Control Optim., 17 (1979), pp. 321–338.

[15] R. F. Hartl, S. P. Sethi, and R. G. Vickson, *A survey of the maximum principles for optimal control problems with state constraints*, SIAM Rev., 37 (1995), pp. 181–218.

[16] A. D. Ioffe and V. M. Tihomirov, *Theory of Extremal Problems*, North–Holland, Amsterdam, 1979.

[17] K. Malanowski, *Two-norm approach in stability and sensitivity analysis of optimization and optimal control problems*, Adv. Math. Sci. Appl., 2 (1993), pp. 397–443.

[18] K. Malanowski, *Stability and sensitivity of solutions to nonlinear optimal control problems*, Appl. Math. Optim., 32 (1995), pp. 111–141.

[19] K. Malanowski, *Stability and sensitivity analysis for optimal control problems with control-state constraints*, Dissertationes Math. (Rozprawy Mat.), 394 (2001), 51 pp.

[20] K. Malanowski, *Stability analysis for nonlinear optimal control problems subject to state constraints*, SIAM J. Optim., 18 (2007), pp. 926–945.

[21] K. Malanowski, *Sufficient optimality conditions in stability analysis for state-constrained optimal control*, Appl. Math. Optim., 55 (2007), pp. 255–271.

[22] K. Malanowski and H. Maurer, *Sensitivity analysis for optimal control problems subject to higher order state constraints, Optimization with data perturbations*, II, Ann. Oper. Res., 101 (2001), pp. 43–73.

[23] H. Mäurer, *On the Minimum Principle for Optimal Control Problems with State Constraints*, Schriftenreihe des Rechenzentrum 41, Universität Münster, Münster, Germany, 1979.

[24] H. Mäurer, *First and second order sufficient optimality conditions in mathematical programming and optimal control*, Math. Programming Stud., 14 (1981), pp. 163–177.

[25] S. M. Robinson, *First order conditions for general nonlinear optimization*, SIAM J. Appl. Math., 30 (1976), pp. 597–607.

[26] S. M. Robinson, *Stability theorems for systems of inequalities.* II*: Differentiable nonlinear systems*, SIAM J. Numer. Anal., 13 (1976), pp. 497–513.

[27] S. M. Robinson, *Strongly regular generalized equations*, Math. Oper. Res., 5 (1980), pp. 43–62.

SIAM J. OPTIM.
Vol. 20, No. 1, pp. 130–155

A SEQUENTIAL CONVEX SEMIDEFINITE PROGRAMMING ALGORITHM WITH AN APPLICATION TO MULTIPLE-LOAD FREE MATERIAL OPTIMIZATION[*]

M. STINGL[†], M. KOČVARA[‡], AND G. LEUGERING[†]

Abstract. A new method for the efficient solution of a class of convex semidefinite programming (SDP) problems is introduced. The method extends the sequential convex programming (SCP) concept to optimization problems with matrix variables. The basic idea of the new method is to approximate the original optimization problem by a sequence of subproblems, in which nonlinear functions (defined in matrix variables) are approximated by block separable convex functions. The subproblems are semidefinite programs with a favorable structure which can be efficiently solved by existing SDP software. The new method is shown to be globally convergent. The article is concluded by a series of numerical experiments with free material optimization problems demonstrating the effectiveness of the generalized SCP approach.

Key words. structural optimization, material optimization, semidefinite programming, sequential convex programming

AMS subject classifications. 74B05, 74P05, 90C90, 90C25, 90C22

DOI. 10.1137/070711281

1. Introduction.

Free material optimization (FMO) is a branch of structural optimization that has gained more and more interest in recent years. It represents a generalization of so-called topology optimization (see [3]) that, nowadays, is being routinely used in the industry. FMO has been successfully used for conceptual design of aircraft components; the most prominent example is the design of ribs in the leading edge of Airbus A380 [12]. The underlying FMO model was introduced in [2] and [21] and has been studied in several further articles such as, for example, [4, 33]. The optimization variable is the material tensor which is allowed to vary from point to point. The method is supported by powerful optimization and numerical techniques, which allow for scenarios with complex bodies and fine finite-element meshes. Rather than solving the (primal) FMO problem directly, the most successful method for the solution of multiple load FMO problems is based on dualization of the original problem and leads to large scale semidefinite programming problems [4]. The dual method has been implemented in a software package MOPED which has been recently applied to real-world applications. Nevertheless, the dual semidefinite approach has two major disadvantages. First of all, the computational complexity of the method depends cubically on the number of load cases [14]. This makes the approach impractical for three-dimensional (3D) problems with more than a few (typically 3–5) load cases. Moreover, it is almost impossible to apply the dual approach to extended (multidisciplinary) FMO problems. This is a serious drawback, because additional

[*] Received by the editors December 18, 2007; accepted for publication (in revised form) October 20, 2008; published electronically March 25, 2009. This work has been partially supported by the EU Commission in the Sixth Framework Program, Project 30717 PLATO-N, and by grant A1075402 of the Czech Academy of Sciences (MK).

http://www.siam.org/journals/siopt/20-1/71128.html

[†] Institute of Applied Mathematics, University of Erlangen, Martensstr. 3, 91058 Erlangen, Germany (stingl@am.uni-erlangen.de, leugering@am.uni-erlangen.de).

[‡] School of Mathematics, University of Birmingham, Birmingham B15 2TT, UK and Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 182 08 Praha 8, Czech Republic (kocvara@maths.bham.ac.uk).

constraints like, for instance, displacement-based constraints play an important role in many real-world applications (compare [12, 15]). A direct treatment of the primal problem seems to avoid both of these difficulties, but, unfortunately, no successful algorithmic concept has been found for the solution of this problem so far.

On the other hand, during the last two decades powerful optimization methods have been developed for the solution of topology optimization problems, based on the so-called SIMP (solid isotropic material with penalization) approach (see [3]), a related field of structural optimization. The most successful methods CONLIN [8], the method of moving asymptotes [26, 27], and the sequential convex programming method [31, 32] are all based on separable convex first-order approximations of nonlinear functions. The mathematical structure of SIMP-based optimization problems is closely related to the structure of the primal FMO problem. The only significant difference is that the design variables in FMO (material matrices/tensors) are defined in matrix spaces, while the variables in SIMP-based problems (density, thickness) are typically of real type. Motivated by this fact, we propose a new optimization method, which generalizes the sequential convex approximation concept to functions defined on matrix spaces. In the scope of this article we investigate theoretical as well as numerical aspects of the new method. Moreover, we demonstrate by numerical experiments that the new method offers a viable alternative and supplement to existing methods in the field of material optimization.

This article is structured as follows: In section 2, we define the basic problem statement and provide a brief motivation of the optimization method discussed in this article. In section 3, we introduce so-called separable hyperbolic approximations of functions defined on matrix spaces. In section 4, these approximations are used to construct a globally convergent algorithm for the solution of certain convex semidefinite programs. Then, in section 5, we describe an algorithm used for the efficient solution of separable convex semidefinite programs, which appeared as subproblems in section 4. In section 6, we briefly repeat the FMO model. Finally, in section 7, we present algorithmic details along with extensive numerical studies by means of FMO problems.

Throughout this article we use the following notation: We denote by $\mathbb{S}^d$ the space of symmetric $d{\times}d$-matrices equipped with the standard inner product $\langle \cdot, \cdot \rangle_{\mathbb{S}^d}$ defined by $\langle A, B \rangle_{\mathbb{S}^d} := \mathrm{Tr}(AB)$ for any pair of matrices $A, B \in \mathbb{S}^d$. We denote by $\mathbb{S}^d_+$ the cone of all positive semidefinite matrices in $\mathbb{S}^d$ and use the abbreviation $A \succcurlyeq_{\mathbb{S}^d} 0$ for matrices $A \in \mathbb{S}^d_+$. Moreover, for $A, B \in \mathbb{S}^d$, we say that $A \succcurlyeq_{\mathbb{S}^d} B$ if and only if $A - B \succcurlyeq_{\mathbb{S}^d} 0$ and similarly for $A \preccurlyeq_{\mathbb{S}^d} B$. Further we make use of the operator $\mathrm{svec} : \mathbb{S}^d \to \mathbb{R}^{\hat{d}}$ with $\hat{d} := d(d+1)/2$, which maps a matrix $A \in \mathbb{S}^d$ with entries $(a_{i,j})_{i,j=1}^d$ to the vector

$$(a_{1,1}, a_{2,1}, a_{2,2}, a_{3,1}, a_{3,2}, a_{3,3}, \dots, a_{d,1}, a_{d,2}, \dots, a_{d,d})$$

(notice the slightly nonstandard definition of this operator). Along with this operator we define $\mathrm{smat} : \mathbb{R}^{\hat{d}} \to \mathbb{S}^d$ as the inverse operator of svec.

2. Motivation.

Our aim is to solve the following generic optimization problem:

$$(\mathcal{P}) \qquad \begin{aligned} &\min_{Y \in \mathbb{S}} f(Y) \\ &\text{subject to} \\ &\quad g_k(Y) \le 0, \quad k = 1, 2, \dots, K, \\ &\quad \underline{Y_i} \preccurlyeq_{\mathbb{S}^{d_i}} Y_i \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}, \quad i = 1, 2, \dots, m, \end{aligned}$$

with

$$\mathbb{S} = \mathbb{S}^{d_1} \times \mathbb{S}^{d_2} \times \cdots \times \mathbb{S}^{d_m} \text{ and } (d_1, d_2, \ldots, d_m) \in \mathbb{N}^m .$$

We assume that, in general, m is large (10^3–10^5) and d_i are small (2–10). That is, we have many small-size matrix variables and matrix constraints.

In what follows $\mathcal{F}$ denotes the feasible domain of problem $(\mathcal{P})$. Throughout the paper we make the following assumptions:

(A1) $f : \mathbb{S} \to \mathbb{R}$ is convex. Moreover, f is the maximum over a finite set of twice continuously differentiable functions. Therefore we may write

$$f(Y) = \max_{\ell \in \mathcal{I}_{\max}} f_\ell(Y)$$

for some index set $\mathcal{I}_{\max}$.

(A2) The functions $g_k : \mathbb{S} \to \mathbb{R}$ $(k = 1, 2, \ldots, K)$ are continuously differentiable and convex so that $\mathcal{F}$ is convex.

(A3) The interior of $\mathcal{F}$ is nonempty.

Problems of type $(\mathcal{P})$ arise in various applications. Our main motivation is to solve the FMO problems described in detail in section 6. However, other applications can be found, e.g., in spline approximation [1] and sparse semidefinite programming (SDP) relaxation of polynomial optimization problems [28].

Of course, one could try to apply an existing linear or nonlinear (convex) SDP solver directly in order to solve $(\mathcal{P})$. However, in our FMO application, the Hessian of the objective function is a full matrix and, given the dimension of the problem, it is prohibitive to use second-order methods with explicit derivative calculations. Experiments with an SDP solver avoiding the calculation and storage of explicit second-order derivatives by Krylov-type methods (see [16]) led to only moderate success. For this reason we decided to develop an SDP solver that is solely based on first-order information.

We have opted for a generalization of the so-called sequential convex approximation methods. These methods proved to be extremely efficient when solving (standard) nonlinear optimization problems arising in the field or structural optimization. The most prominent (and well-known in the structural optimization community) representatives are CONLIN [8] by Fleury, the method of moving asymptotes (MMA) [26, 27] by Svanberg, and the sequential convex programming method (SCP) [31, 32] by Zillober. All of these methods are based on separable convex first-order approximations of nonlinear functions. The mathematical structure of structural optimization problems is closely related to the structure of our FMO problem. The only significant difference is that the design variables in FMO are matrices of dimension 3 × 3 or 6 × 6, while the variables in structural optimization problems (density, thickness) are typically real vectors. Motivated by this fact, we propose a generalization of the sequential convex approximation concept to functions defined on matrix spaces.

We should emphasize that, while the sequential convex approximation methods are efficient when solving structural optimization problems, they are still first-order methods. Although they were applied to few other problems, too (see [30]), no systematic study or benchmark testing has been done for general large-scale nonlinear optimization problems, as to our knowledge. A brief comparison with an SQP code can be found in [22], using the Hock–Schittkowski collection of 306 small-scale test problems (with up to 100 variables). Within a relatively low accuracy of the stopping criteria, the SCP code solved 93% of all problems (compared to 100% success of the

SQP code), while the number of function evaluations was about twice as high as for the SQP code. As with other first-order methods, examples can be found for which the method will behave poorly. Therefore, we do not want to make an impression that our generalization is a universal cure for large-scale (nonlinear) SDP problems. But, as we will see in section 7, it *is* efficient when solving the FMO problems.

3. A block-separable convex approximation scheme. In this section, we will define block-separable convex approximations of continuously differentiable functions

$$f:\ \mathbb{S}\to\mathbb{R},\ \text{where } \mathbb{S}=\mathbb{S}^{d_1}\times\mathbb{S}^{d_2}\times\cdots\times\mathbb{S}^{d_m} \text{ and } (d_1,d_2,\dots,d_m)\in\mathbb{N}^m. \tag{3.1}$$

Let $\mathcal{I}=\{1,2,\dots,m\}$. On $\mathbb{S}$ we define the inner product $\langle\cdot,\cdot\rangle_{\mathbb{S}}:=\sum_{i\in\mathcal{I}}\langle\cdot,\cdot\rangle_{\mathbb{S}^{d_i}}$, where $\langle\cdot,\cdot\rangle_{\mathbb{S}^{d_i}}$ is the standard inner product in $\mathbb{S}^{d_i}$ $(i\in\mathcal{I})$. Moreover, we denote by $\|\cdot\|_{\mathbb{S}}$ the norm induced by $\langle\cdot,\cdot\rangle_{\mathbb{S}}$. Finally, we denote the directional derivatives of f of first and second order in directions $V,W\in\mathbb{S}$ by $\frac{\partial}{\partial Y}f(Y;V)$ and $\frac{\partial^2}{\partial Y\partial Y}f(Y;V,W)$, respectively.

Definition 3.1. *We call an approximation $g:\mathbb{S}\to\mathbb{R}$ of a function f of type* (3.1) *a* convex first-order approximation *at $\bar{Y}=(\bar{Y}_1,\dots,\bar{Y}_m)\in\mathbb{S}$ if the following assumptions are satisfied:*

(A4) *$g(\bar{Y})=f(\bar{Y})$,*

(A5) *$\frac{\partial}{\partial Y_i}g(\bar{Y})=\frac{\partial}{\partial Y_i}f(\bar{Y})$ for all $i\in I$,*

(A6) *g is convex.*

In the following, we construct a local block separable *convex first-order approximation* scheme for functions of type f. Our construction can be considered a generalization of the classic MMA-type approximations defined, for example, in [26, 32].

We start with the following definitions.

Definition 3.2. *Let $f:\mathbb{S}\to\mathbb{R}$ be continuously differentiable on a subset $B\subset\mathbb{S}$. For all $i\in\mathcal{I}$ we define differential operators entrywise by*

$$\left(\nabla^i f\right)_{\ell,j}:=\left(\frac{\partial f}{\partial Y_i}\right)_{\ell,j},\quad 1\le l,j\le d_i,$$

and denote by $\nabla^i_+f(\bar{Y})$ and $\nabla^i_-f(\bar{Y})$ the projections of $\nabla^i f(\bar{Y})$ onto $\mathbb{S}^{d_i}_+$ and $\mathbb{S}^{d_i}_-$, respectively.

Definition 3.3. *Let $f:\mathbb{S}\to\mathbb{R}$ be continuously differentiable on a subset $B\subset\mathbb{S}$ and $\bar{Y}=(\bar{Y}_1,\bar{Y}_2,\dots,\bar{Y}_m)\in B$. Moreover, let* asymptotes *$L=(L_1,L_2,\dots,L_m)^\top$ and $U=(U_1,U_2,\dots,U_m)^\top$ be given such that*

$$L_i\prec_{\mathbb{S}^{d_i}_+}\bar{Y}_i\prec_{\mathbb{S}^{d_i}_+}U_i\ \ \textit{for all } i\in\mathcal{I}$$

and $\tau:=\{\tau_1,\tau_2,\dots,\tau_m\}$ be a set of nonnegative real parameters. Then we define the hyperbolic approximation $f^{L,U,\tau}_{\bar{Y}}$ of f at $\bar{Y}$ *as*

$$\begin{aligned} f^{L,U,\tau}_{\bar{Y}}(Y):=&\ f(\bar{Y})\\ &+\sum_{i=1}^m\left\langle\nabla^i_+f(\bar{Y}),(U_i-\bar{Y}_i)(U_i-Y_i)^{-1}(U_i-\bar{Y}_i)-(U_i-\bar{Y}_i)\right\rangle_{\mathbb{S}^{d_i}}\\ &-\sum_{i=1}^m\left\langle\nabla^i_-f(\bar{Y}),(\bar{Y}_i-L_i)(Y_i-L_i)^{-1}(\bar{Y}_i-L_i)-(\bar{Y}_i-L_i)\right\rangle_{\mathbb{S}^{d_i}}\\ &+\sum_{i=1}^m\tau_i\left\langle(Y_i-\bar{Y}_i)^2,(U_i-Y_i)^{-1}+(Y_i-L_i)^{-1}\right\rangle_{\mathbb{S}^{d_i}}. \end{aligned}\tag{3.2}$$

The following theorem says that (3.2) is a convex approximation in the sense of Definition 3.1.

THEOREM 3.4.

(a) $f_{\bar{Y}}^{L,U,\tau}$ *satisfies assumptions* (A4)–(A6).

(b) $f_{\bar{Y}}^{L,U,\tau}$ *is separable w.r.t. the matrix variables* $Y_1, Y_2, \ldots, Y_m$.

(c) *Let* B *be a compact subset of* $\mathbb{S}$, $\overline{\tau} \geq \tau_i \geq \underline{\tau} > 0$ *for all* $i \in \mathcal{I}$, *and compact sets of asymptotes* $\mathcal{L}$ *and* $\mathcal{U}$ *satisfy the following condition:*

$$\text{(AS)} \qquad \forall \left\{ \begin{array}{c} L \in \mathcal{L} \\ U \in \mathcal{U} \end{array} \right\} \exists \mu > 0 : \forall i \in \mathcal{I}\ \forall Y \in B : \left\{ \begin{array}{c} Y_i - L_i \\ U_i - Y_i \end{array} \right\} \succeq \mu I_{\mathbb{S}^{d_i}}.$$

Then $f_{\bar{Y}}^{L,U,\tau}$ *is strongly convex on* B *and there is a common constant* $\underline{\nu} > 0$ *such that for all* $L \in \mathcal{L}, U \in \mathcal{U}$, *and* $\bar{Y} \in B$

$$\frac{\partial}{\partial Y} f_{\bar{Y}}^{L,U,\tau}(Y;\ X - Y) + \underline{\nu}\|X - Y\| \leq f_{\bar{Y}}^{L,U,\tau}(X) - f_{\bar{Y}}^{L,U,\tau}(Y)$$

for all $X, Y \in B$. *Moreover, the second-order derivative of* $f_{\bar{Y}}^{L,U,\tau}$ *is uniformly bounded for all* $L \in \mathcal{L}, U \in \mathcal{U}$, *and* $\bar{Y} \in B$ *in the sense that there is a constant* $\overline{\nu} > 0$ *such that*

$$\frac{\partial^2}{\partial Y \partial Y} f_{\bar{Y}}^{L,U,\tau}(Y;\ D, D) \leq \overline{\nu}\|D\|^2$$

for all $Y \in B$ *and all* $D \in \mathbb{S}$.

Proof.

(A4) For $Y := \bar{Y}$ we have for all $i \in \mathcal{I}$

$$\begin{aligned} &\left\langle \nabla_+^i f(\bar{Y}), (U_i - \bar{Y}_i)(U_i - \bar{Y}_i)^{-1}(U_i - \bar{Y}_i) - (U_i - \bar{Y}_i) \right\rangle_{\mathbb{S}^{d_i}} \\ &\qquad\qquad = \left\langle \nabla_+^i f(\bar{Y}), (U_i - \bar{Y}_i) - (U_i - \bar{Y}_i) \right\rangle_{\mathbb{S}^{d_i}} = 0. \end{aligned}$$

Consequently, the first sum in (3.2) vanishes. Analogously, we show that the second sum vanishes and, with $\left\langle (\bar{Y}_i - \bar{Y}_i)^2, (U_i - Y_i)^{-1} + (Y_i - L_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} = 0$, we conclude that $f_{\bar{Y}}^{L,U,\tau}(\bar{Y}) = f(\bar{Y})$.

(A5) Differentiating $f^{L,U,\tau}$ w.r.t. Y_i we obtain

$$\begin{aligned} \frac{\partial}{\partial Y_i} f_{\bar{Y}}^{L,U,\tau}(Y) &= (U_i - Y_i)^{-1}(U_i - \bar{Y}_i)\nabla_+^i f(\bar{Y})(U_i - \bar{Y}_i)(U_i - Y_i)^{-1} \\ &\quad + (Y_i - L_i)^{-1}(\bar{Y}_i - L_i)\nabla_-^i f(\bar{Y})(\bar{Y}_i - L_i)(Y_i - L_i)^{-1} \\ &\quad + \tau_i \left(I_{\mathbb{S}^{d_i}} - (U_i - Y_i)^{-1}(U_i - \bar{Y}_i)^2(U_i - Y_i)^{-1} \right) \\ &\quad + \tau_i \left(I_{\mathbb{S}^{d_i}} - (Y_i - L_i)^{-1}(\bar{Y}_i - L_i)^2(Y_i - L_i)^{-1} \right). \end{aligned} \tag{3.3}$$

Substituting $Y := \bar{Y}$ in (3.3), we obtain

$$\frac{\partial}{\partial Y_i} f_{\bar{Y}}^{L,U,\tau}(\bar{Y}) = \nabla_+^i f(\bar{Y}) + \nabla_-^i f(\bar{Y}) = \nabla^i f(\bar{Y}) = \frac{\partial}{\partial Y_i} f(\bar{Y}).$$

(A6) Before we show convexity, we prove separability (assertion (b)). This follows immediately from (3.3), as

$$\frac{\partial}{\partial Y_j}\left(\frac{\partial}{\partial Y_i} f_{\bar{Y}}^{L,U,\tau}(Y)\right) = 0.$$

Now, in order to prove convexity, it is sufficient to show that for all $i \in \mathcal{I}$, for all $Y \in \mathbb{S}$, and for an arbitrary direction $D \in \mathbb{S}^{d_i}$,

$$\frac{\partial^2}{\partial Y_i \partial Y_i} f_{\bar{Y}}^{L,U,\tau}(Y;\ D, D) = \left\langle \nabla^i \left\langle \nabla^i f_{\bar{Y}}^{L,U,\tau}(Y), D \right\rangle_{\mathbb{S}^{d_i}}, D \right\rangle_{\mathbb{S}^{d_i}} \geq 0. \tag{3.4}$$

Introducing the abbreviations $B_i^+ := (U_i - \bar{Y}_i)\nabla_+^i f(\bar{Y})(U_i - \bar{Y}_i)$ and $B_i^- := (\bar{Y}_i - L_i)\nabla_-^i f(\bar{Y})(\bar{Y}_i - L_i)$ it follows from (3.3) that

$$\begin{aligned}
&\left\langle \nabla^i \left\langle \nabla^i f_{\bar{Y}}^{L,U,\tau}(Y), D \right\rangle_{\mathbb{S}^{d_i}}, D \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad = 2 \left\langle D(U_i - Y_i)^{-1} D, (U_i - Y_i)^{-1} B_i^+ (U_i - Y_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad\quad + 2 \left\langle D(Y_i - L_i)^{-1} D, (Y_i - L_i)^{-1} (-B_i^-) (Y_i - L_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad\quad + 2\tau_i \left\langle D(U_i - Y_i)^{-1} D, (U_i - Y_i)^{-1} (U_i - \bar{Y}_i)^2 (U_i - Y_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad\quad + 2\tau_i \left\langle D(Y_i - L_i)^{-1} D, (Y_i - L_i)^{-1} (\bar{Y}_i - L_i)^2 (Y_i - L_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} .
\end{aligned} \tag{3.5}$$

Given that the matrices $B_i^+, -B_i^-, U_i - Y_i, Y_i - L_i, (U_i - \bar{Y}_i)^2$, and $(\bar{Y}_i - L_i)^2$ are all positive semidefinite, we observe that all terms in (3.5) are nonnegative. Consequently, the estimate (3.4) holds true and $f_{\bar{Y}}^{L,U,\tau}$ is convex. Finally, given that (AS) holds for the compact sets $\mathcal{L}$ and $\mathcal{U}$, we use (3.5) to show that

$$\begin{aligned}
&\left\langle \nabla^i \left\langle \nabla^i f_{\bar{Y}}^{L,U,\tau}(Y), D \right\rangle_{\mathbb{S}^{d_i}}, D \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad \geq 2\tau_i \left\langle D(U_i - Y_i)^{-1} D, (U_i - Y_i)^{-1} (U_i - \bar{Y}_i)^2 (U_i - Y_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad\quad + 2\tau_i \left\langle D(Y_i - L_i)^{-1} D, (Y_i - L_i)^{-1} (\bar{Y}_i - L_i)^2 (Y_i - L_i)^{-1} \right\rangle_{\mathbb{S}^{d_i}} \\
&\quad \geq 4\tau_i \gamma^3 \mu^2 \left\langle D, D \right\rangle_{\mathbb{S}^{d_i}} ,
\end{aligned}$$

where γ is an upper bound for the maximal possible difference of eigenvalues of arbitrary elements in the compact sets B and $\mathcal{L}$ or B and $\mathcal{U}$, respectively. Now the assertion of part (c) follows with $\underline{\nu} := m 4 \underline{\tau} \gamma^3 \mu^2$. The uniform boundedness of the second-order derivatives follows in an analogous way from (3.5) with $\overline{\nu} := m\gamma^2\mu^3(2 \max_{i \in \mathcal{I}, \overline{Y} \in B} \|\nabla^i f(\bar{Y})\| + 4\overline{\tau})$. □

Remark 3.1. Assumption (AS) in Theorem 3.4 essentially says that the eigenvalues of all iterates have to remain bounded away from the asymptotes by a small positive number. This is crucial for the strong convexity proof in part (c) of Theorem 3.4. A practical choice of asymptotes is briefly discussed in section 7. For more sophisticated choices of asymptotes in the nonlinear programming case we refer to [5, 9, 26, 32].

4. A globally convergent algorithm based on hyperbolic approximations.

In the framework of this section we use the local hyperbolic approximations defined in section 3 in order to establish a solution scheme for our generic convex optimization problem $(\mathcal{P})$:

$$(\mathcal{P}) \qquad \begin{aligned}
&\min_{Y \in \mathbb{S}} f(Y) \\
&\text{subject to} \\
&\qquad g_k(Y) \leq 0, \quad k = 1, 2, \ldots, K, \\
&\qquad \underline{Y_i} \preccurlyeq_{\mathbb{S}^{d_i}} Y_i \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}, \quad i = 1, 2, \ldots, m\,;
\end{aligned}$$

here $\mathbb{S}$ is defined as in (3.1). Recall that $\mathcal{F}$ denotes the feasible domain of problem $(\mathcal{P})$. In addition to (A1)–(A6), we will make the following assumption:

(A7) The compact sets $\mathcal{L}$ and $\mathcal{U}$ satisfy property (AS) for the compact set $\mathcal{F}$.

Given an iteration index j and an associated feasible point Y^j of problem $(\mathcal{P})$, we define a local hyperbolic approximation of f as

$$f^j(Y) := \max_{\ell \in \mathcal{I}_{\max}} f^j_\ell(Y) := \max_{\ell \in \mathcal{I}_{\max}} (f_\ell)\,{}^{L^j,U^j,\tau^j}_{Y^j}(Y).$$

Using this function, we further define a local approximation of $(\mathcal{P})$ close to Y^j as follows:

$$(\mathcal{P}^j) \qquad \begin{aligned} &\min_{Y \in \mathbb{S}} f^j(Y) \\ &\text{subject to} \\ &\quad g_k(Y) \le 0, \quad k = 1, 2, \ldots, K, \\ &\quad \underline{Y_i}^j \preccurlyeq_{\mathbb{S}^{d_i}} Y_i \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}^j, \quad i = 1, 2, \ldots, m. \end{aligned}$$

Here the bounds $\underline{Y_i}^j, \overline{Y_i}^j$ are chosen to be compatible with the following assumption:

(A8) $\underline{Y_i} \preccurlyeq_{\mathbb{S}^{d_i}} \underline{Y_i}^j \preccurlyeq_{\mathbb{S}^{d_i}} Y_i^j \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}^j \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}$ for all $i = 1, 2, \ldots, m$.

We denote the feasible domain of problem $(\mathcal{P}^j)$ by $\mathcal{F}^j$. By construction the following corollary is an immediate consequence of Theorem 3.4.

COROLLARY 4.1.

(a) $f^j(Y^j) = f(Y^j)$.

(b) *The subdifferentials of* f^j *and* f *coincide at* Y^j*, i.e.,* $\partial f^j(Y^j) = \partial f(Y^j)$.

(c) f^j *is convex.*

(d) f^j *is separable w.r.t.* $Y_1, Y_2, \ldots Y_m$.

(e) *Let* $\overline{\tau} \ge \tau_1, \tau_2, \ldots, \tau_m \ge \underline{\tau} > 0$. *Let further compact sets of asymptotes* $\mathcal{L}$ *and* $\mathcal{U}$ *be given such that property* (AS) *holds for* $\mathcal{F}^j$. *Then* f^j *is strongly convex on* $\mathcal{F}^j$. *Moreover, there is a common constant* $\underline{\nu} > 0$ *such that for all* j

$$\frac{\partial}{\partial Y} f^j(Y;\ X - Y) + \underline{\nu}\|X - Y\| \le f^j(X) - f^j(Y)$$

for all $X, Y \in \mathcal{F}^j$.

Proof. All assertions but the last one follow directly from Theorem 3.4. It remains to show that $\mathcal{F}^j$ is compact and convex. The convexity follows directly from assumption (A2). Moreover, each domain $\mathcal{F}^j$ is compact as a closed subset of the domain $\left\{Y \in \mathbb{S} \mid \underline{Y_i} \preccurlyeq_{\mathbb{S}^{d_i}} Y_i \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}, i = 1, 2, \ldots, m\right\}$. □

The following proposition states some basic properties of $(\mathcal{P}^j)$.

PROPOSITION 4.2. *Each subproblem* $(\mathcal{P}^j)$ *has a unique solution* Y^{j+1}. *Associated with* Y^{j+1} *there exist Lagrangian multipliers* (v^{j+1}, V^{j+1}) *such that* $(Y^{j+1}, v^{j+1}, V^{j+1})$ *is a KKT point of* $(\mathcal{P}^j)$.

Proof. The existence and the uniqueness of a solution follows from the strong convexity of the objective function f^j on the compact set $\mathcal{F}^j$. Furthermore, assumption (A3) implies that the Slater condition holds for $(\mathcal{P}^j)$. Consequently a KKT point exists. □

Now we are able to present the basic algorithm for the solution of $(\mathcal{P})$.

ALGORITHM 4.3. *Let initial points* $Y^1 \in \mathbb{S}$ *and initial multipliers* $(v^1, V^1) \in \mathbb{R}^k_+ \times \mathbb{S}_+$ *be given.*

(1) *Put* $j = 1$.

(2) *Choose asymptotes* $L^j \in \mathcal{L}, U^j \in \mathcal{U}$, *and* $\overline{\tau} \ge \tau^j_1, \tau^j_2, \ldots, \tau^j_m \ge \underline{\tau} > 0$.

(3) *Solve problem* $(\mathcal{P}^j)$. *Denote the solution by* $Y^+ \in \mathbb{S}$ *and the associated Lagrangian multipliers by* $(v^+, V^+) \in \mathbb{R}^k_+ \times \mathbb{S}_+$.

(4) *Choose* $\alpha^j = \min\{1, \hat{\alpha}\}$*, where* $\hat{\alpha} = \operatorname{argmin}_{\alpha \in \mathbb{R}_+} f(Y^j + \alpha(Y^+ - Y^j))$.

(5) $\left(Y^{j+1}, v^{j+1}, V^{j+1}\right) = \left(Y^j, v^j, V^j\right) + \alpha^j \left((Y^+, v^+, V^+) - \left(Y^j, v^j, V^j\right)\right)$.

(6) *If* Y^{j+1} *is stationary for problem* $(\mathcal{P})$*, STOP; otherwise put* $j = j + 1$ *and GOTO* (2).

Possible choices of asymptotes (step 2) will be discussed in section 7. In section 5, we will propose an algorithm for the efficient solution of the subproblem in step 3. Practical implementations of the line search in step 4 as well as the stopping criterion in step 6 will be given in section 7. Before we state the central convergence result for Algorithm 4.3, we make one more assumption:

(A9) The multiplier estimates generated by Algorithm 4.3 stay bounded.

Note that in [31] the assertion of assumption (A9) is proven to hold for standard inequality constrained nonlinear programs under the LICQ condition.

Theorem 4.4. *If* f *is bounded from below in* $\mathcal{F}$*, then assume that* (A1)–(A3) *and* (A7)–(A6) *are satisfied. Then either Algorithm* 4.3 *stops at a global minimizer of* $(\mathcal{P})$ *or the sequence* $\{Y^j\}_{j>0}$ *generated by Algorithm* 4.3 *has at least one accumulation point and each accumulation point is a global minimizer of* $(\mathcal{P})$.

In order to be able to prove the convergence theorem, we make use of the following lemmas.

Lemma 4.5. *Let* $\mathcal{A}_{f^{(j)}}(Y) := \{\ell \mid f_\ell^{(j)}(Y) = f^{(j)}(Y)\} \subset \mathcal{I}_{\max}$. *Then*

$$\partial f^{(j)}(Y) = \operatorname{conv}\{\nabla f_\ell^{(j)}(Y) \mid \ell \in \mathcal{A}(Y)\}, \tag{4.1}$$

where $\partial f^{(j)}(Y)$ *denotes the subdifferential of* $f^{(j)}$ *at* Y *and* conv *is the convex hull.*

Proof. Formula (4.1) is a direct consequence of Corollary 4.3.2 in [10]. □

Lemma 4.6. *If* $Y^j \in \mathcal{F}^j$ *is not stationary for* $(\mathcal{P})$*, then the direction* $D^j := Y^+ - Y^j$ *is a descent direction for* f *at* Y^j.

Proof. Using the fact that Y^+ is a unique minimizer of $(\mathcal{P}^j)$, we obtain the following from Corollary 4.1(e):

$$\frac{\partial}{\partial Y} f^j(Y^j; D^j) + \underline{\nu}\|D^j\| \leq f(Y^+) - f(Y^j) \leq 0. \tag{4.2}$$

Consequently, we obtain the following from the first-order approximation properties of f^j:

$$\frac{\partial}{\partial Y} f(Y^j; D^j) < 0. \qquad \square$$

Lemma 4.7. *Algorithm* 4.3 *generates a sequence of feasible points* $Y^1, Y^2, \ldots$ *with*

$$f(Y^{j+1}) \leq f(Y^j).$$

Proof. The convexity of $\mathcal{F}^j$ and the fact that $\mathcal{F}^j \subset \mathcal{F}$ imply that all iterates remain feasible. Let us now consider a subproblem at an arbitrary iteration j. Then we have $f(Y^j) = f^j(Y^j)$. From assumption $(A8)$ we know that Y^j is a feasible point of problem $(\mathcal{P}^j)$. Consequently, it follows from Lemma 4.6 that $D^j := Y^+ - Y^j$ is a descent direction for f at Y^j. Now the assertion follows by construction of the line search defined in step 4 of Algorithm 4.3. □

Lemma 4.8. *Let* $Y^* \in \mathcal{F}$ *be an accumulation point of the sequence generated by Algorithm* 4.3 *applied to* $(\mathcal{P})$*. Then* Y^* *is an unconstrained minimizer of* f *or the line search in step* 4 *of Algorithm* 4.3 *returns the result* $\alpha^j = 1$ *for almost all* $j > 0$.

Proof. Given an arbitrary element $g^j \in \partial f(Y^j)$ with $\langle -g^j, D^j\rangle = \frac{\partial}{\partial Y} f(Y^j; D^j)$ it follows from (4.2) that

$$\frac{\langle -g^j, D^j\rangle_{\mathbb{S}}}{\|D^j\|_{\mathbb{S}}} \geq \underline{\nu} > 0. \tag{4.3}$$

From (4.3) and the fact that $\|g^j\|_{\mathbb{S}}$ is bounded on the compact set $\mathcal{F}$, we obtain the existence of $\gamma > 0$ such that

$$\frac{\langle -g^j, D^j\rangle_{\mathbb{S}}}{\|D^j\|_{\mathbb{S}}\|g^j\|_{\mathbb{S}}} \geq \gamma.$$

Consequently, the cosine of the angle between $\partial f(Y^j)$ and the descent direction D^j is strictly bounded from zero. Suppose now that Algorithm 4.3 generates infinitely many iterations j_s with $\operatorname{argmin}_{\alpha\in\mathbb{R}_+} f(Y^{j_s} + \alpha Y^+) \in [0; 1)$. Then it follows from the Theorem of Zoutendijk (see [19, Thm. 3.2]) that $\|g_j\|_{\mathbb{S}} \to 0$. Hence we obtain $0 \in \partial f(Y^*)$ and conclude that $Y^* \in \mathcal{F}$ is a minimizer of the convex function f. □

Now we are able to finish the proof of Theorem 4.4.

Proof. Suppose that Algorithm 4.3 does not stop at a stationary point. Then, according to Lemma 4.7, it generates an infinite sequence with $\{Y^k\}_{k>0}$ in $\mathcal{F}$ such that $\{f(Y^k)\}_{k>0}$ is monotonically decreasing. As f is bounded from below on the compact set $\mathcal{F}$, the sequence $\{f(Y^k)\}_{k>0}$ converges. Therefore, there is at least one accumulation point Y^* of the sequence $\{Y^k\}_{k>0}$.

Next we will show that Y^* is a first-order critical point and thus a global optimizer for problem $(\mathcal{P})$. Proposition 4.2 guarantees that step 3 of Algorithm 4.3 is well-defined. Moreover, from Lemma 4.8 we have, after finitely many iterations,

$$\left(Y^+, v^+, V^+\right) = \left(Y^{j+1}, v^{j+1}, V^{j+1}\right).$$

Lemma 4.7 and the fact that (Y^+, v^+, V^+) is a KKT point of problem $(\mathcal{P}^j)$ imply the existence of an index $\bar{j} > 0$ and a subsequence $\{Y^{j_s}\}_{j_s > \bar{j}}$ such that

- $Y^{j_s} \in \mathcal{F}$,
- $v^{j_s} \geq 0, V^{j_s} \succeq 0$,
- $g(Y^{j_s})^\top v^{j_s} = 0$,
- $\langle Y_i^{j_s} - \underline{Y}_i, V_i^{j_s}\rangle = 0,\ \ \langle \overline{Y}_i - Y_i^{j_s}, V_{i+m}^{j_s}\rangle = 0$ for all $i = 1, 2, \ldots, m$.

Hence Y^* is feasible. Moreover, making use of assumption (A9), we conclude that there exist nonnegative multipliers (v^*, V^*) such that the triple (Y^*, v^*, V^*) satisfies the complementary slackness condition for $(\mathcal{P})$. In order to complete the proof, we have to show that

$$\operatorname{dist}\ \left(\{0\}, \partial_Y L(Y^{j_s}, v^{j_s}, V^{j_s})\right) \to 0 \tag{4.4}$$

for $j_s \to \infty$, where L denotes the Lagrangian function associated with problem $(\mathcal{P})$ and

$$\operatorname{dist}(A, B) := \min_{X\in A, Y\in B} \|X - Y\|_{\mathbb{S}}$$

for two sets $A, B \subset \mathbb{S}$. Denoting by L^{j_s} the Lagrangian function associated with problem $(\mathcal{P}^{j_s})$, the following estimate holds true:

$$\begin{aligned}
&\operatorname{dist}\ \big(0, \partial_Y L(Y^{j_s}, v^{j_s}, V^{j_s})\big) \\
&\qquad = \operatorname{dist}\ \big(0, \partial_Y f^{j_s}(Y^{j_s}) - \partial_Y f^{j_s}(Y^{j_s}) + \partial_Y L(Y^{j_s}, v^{j_s}, V^{j_s})\big) \\
&\qquad \leq \operatorname{dist}\ \big(0, \partial_Y f(Y^{j_s}) - \partial_Y f^{j_s}(Y^{j_s})\big) + \operatorname{dist}\ \big(0, \partial_Y L^{j_s}(Y^{j_s}, v^{j_s}, V^{j_s})\big) \\
&\qquad = \operatorname{dist}\ \big(\partial_Y f^{j_s}(Y^{j_s}), \partial_Y f(Y^{j_s})\big) \\
&\qquad \leq \operatorname{dist}\ \big(\partial_Y f^{j_s}(Y^{j_s-1}), \partial_Y f^{j_s}(Y^{j_s})\big) + \operatorname{dist}\ \big(\partial_Y f(Y^{j_s-1}), \partial_Y f(Y^{j_s})\big)\, .
\end{aligned} \tag{4.5}$$

From the continuity of f we have (see, e.g., [10])

$$\operatorname{dist}\ \big(\partial_Y f(Y^{j_s-1}), \partial_Y f(Y^{j_s})\big) \to 0$$

for $j_s \to \infty$. Moreover, we conclude from the uniform boundedness of the second-order directional derivatives stated in Theorem 3.4(c) that for all $\ell \in \mathcal{I}_{\max}$

$$\|\nabla f_\ell^{j_s}(Y^{j-1}) - \nabla f_\ell^{j_s}(Y^j)\| \to 0$$

for $j_s \to \infty$. Now, applying Lemma 4.5, we obtain

$$\operatorname{dist}\ \big(\partial_Y f^{j_s}(Y^{j_s-1}), \partial_Y f^{j_s}(Y^{j_s})\big) \to 0$$

for $j_s \to \infty$. Consequently, (4.4) holds, Y^* is a first-order critical point, and the proof of Theorem 4.4 is complete. □

5. A modified barrier algorithm for the solution of subproblems.

In order to solve the subproblems defined in section 4 numerically, we rewrite them in terms of real variables $x = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_m)^\top \in \mathbb{R}^{\hat{d}_1} \times \mathbb{R}^{\hat{d}_2} \times \cdots \times \mathbb{R}^{\hat{d_m}} = \mathbb{R}^{\hat{d}}$. This yields the following problem:

$$\begin{aligned}
&\min_{x \in \mathbb{R}^{\hat{d}}} \max_{\ell \in \mathcal{I}_{\max}} \tilde{f}^{L,U,\tau}_{\operatorname{smat}(\bar{x}),\ell}(x) \\
&\text{subject to} \\
&\qquad g_k(\operatorname{smat}(x)) \leq 0, \quad k \in \{1, 2, \dots, K\}, \\
&\qquad \underline{Y_i} \preccurlyeq_{\mathbb{S}^{d_i}} \operatorname{smat}(\bar{x}_i) \preccurlyeq_{\mathbb{S}^{d_i}} \overline{Y_i}, \quad i \in \mathcal{I}.
\end{aligned} \tag{5.1}$$

In this section, we work with the following additional assumption:

(A10) the functions $g_k : \mathbb{R}^{\hat{d}} \to \mathbb{R}$ $(k = 1, 2, \dots, K)$ are separable w.r.t. $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_m$.

The algorithm used to solve subproblems of the form (5.1) is based on a generalized augmented Lagrangian method for the solution of nonlinear (semidefinite) programs and described in detail in [13, 25]. We briefly recall the basics here and show how this algorithm can be adapted to take advantage of the special structure of problem (5.1).

The algorithm described in [13, 25] is designed for the solution of general nonlinear semidefinite optimization problems of the form

$$\begin{aligned}
&\min_{x \in \mathbb{R}^n} f(x) \\
&\text{subject to} \\
&\qquad \mathcal{G}_j(x) \preccurlyeq 0, \quad j \in \mathcal{J} = \{1, 2, \dots, J\}\,,
\end{aligned} \tag{5.2}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ and $\mathcal{G}_j(x) : \mathbb{R}^n \to \mathbb{S}^{m_j}$ $(j \in \mathcal{J})$ are twice continuously differentiable mappings. The algorithm is based on a choice of smooth modified barrier functions

$\Phi_p : \mathbb{S}^{m_j} \to \mathbb{S}^{m_j}$ $(j \in \mathcal{J})$, depending on a parameter $p > 0$, that satisfy a number of assumptions (see [13]) guaranteeing, in particular, that

$$\mathcal{G}_j(x) \preccurlyeq 0 \Leftrightarrow \Phi_p(\mathcal{G}_j(x)) \preccurlyeq 0 \qquad \forall j \in \mathcal{J}\,.$$

Thus, for any $p > 0$, problem (5.2) has the same solution as the following "augmented" problem

$$\begin{aligned} &\min_{x\in\mathbb{R}^n} f(x) \\ &\text{subject to} \\ &\qquad \Phi_p(\mathcal{G}_j(x)) \preccurlyeq 0, \quad j \in \mathcal{J}\,. \end{aligned} \tag{5.3}$$

A typical choice of Φ_p is

$$\Phi_p(\mathcal{A}(x)) = -p^2(\mathcal{A}(x) - pI)^{-1} - pI\,. \tag{5.4}$$

The Lagrangian of (5.3) can be viewed as a (generalized) augmented Lagrangian of (5.2):

$$F(x, U, p) = f(x) + \sum_{j\in\mathcal{J}} \langle U_j, \Phi_p\left(\mathcal{G}_j(x)\right)\rangle_{\mathbb{S}_m}\,; \tag{5.5}$$

here $U = (U_1, U_2, \ldots, U_J)^\top \in \mathbb{S}^{m_1} \times \mathbb{S}^{m_2} \times \cdots \times \mathbb{S}^{m_J}$ are Lagrangian multipliers associated with the inequality constraints. Defining $\mathcal{G} := (\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_J)^\top$ and $\Phi_p\mathcal{G} := (\Phi_p(\mathcal{G}_1), \Phi_p(\mathcal{G}_2), \ldots, \Phi_p(\mathcal{G}_J))^\top$, the augmented Lagrangian algorithm is defined as follows.

ALGORITHM 5.1. *Let x^1 and U^1 be given. Let $p^1 > 0, \alpha^1 > 0$. For $k = 1, 2, \ldots$ repeat until a stopping criterion is reached:*

(1) *Find $x^{\ell+1}$ satisfying $\|\nabla_x F(x^{\ell+1}, U^\ell, p^\ell)\| \le \alpha^\ell$.*

(2) $U^{\ell+1} = D_{\mathcal{G}}\Phi_p(\mathcal{G}(x^{\ell+1}); U^\ell)$.

(3) $p^{\ell+1} \le p^\ell\,, \quad \alpha^{\ell+1} < \alpha^\ell\,.$

The unconstrained minimization problem in step (1) is approximately solved by the damped Newton method. Multiplier and penalty update strategies as well as local and global convergence properties under standard assumptions are studied extensively in [25]. Let us mention only that, imposing standard assumptions, one can prove that any cluster point of the sequence $\{(x^\ell, U^\ell)\}_{\ell>0}$ generated by Algorithm 5.1 is a KKT point of problem (5.2). The proof given in [25] is an extension of results by Polyak [20] and Breitfeld and Shanno [6].

Identifying $\mathbb{S}_1$ with $\mathbb{R}$ and introducing a slack variable, it is easy to see that problem (5.1) can be written in the form (5.2). Thus Algorithm 5.1 is directly applicable to problem (5.1). In the following, we will demonstrate how the separable structure of (5.1) can be exploited by the damped Newton method applied in step (1) of Algorithm 5.1:

Each search direction at a point $\hat{x}$ is computed as a solution of a linear system of the form

$$\nabla_x^2 F(\hat{x}, U^\ell, p^\ell)d = -\nabla_x F(\hat{x}, U^\ell, p^\ell). \tag{5.6}$$

In order to understand the structure of the system matrix $\nabla_x^2 F(\hat{x}, U^\ell, p^\ell)$, we state F explicitly for problem (5.1) (assuming $|\mathcal{I}_{\max}| = 1$ for simplicity):

$$\begin{aligned} F(x, u^\ell, U^\ell, p^\ell) = {} & f^{L,U,\tau}_{\mathrm{smat}(\bar{x})}(\mathrm{smat}(x)) + \sum_{k \in \mathcal{K}_1} u_j^\ell \varphi_{p^\ell}(g_j(x)) + \sum_{k \in \mathcal{K}_0} u_j^\ell \varphi_{p^\ell}(g_j(x)) \\ & + \sum_{j=1,\ldots,m} \left\langle U_j^\ell, \Phi_{p^\ell}(\mathrm{smat}(x) - \underline{Y}) \right\rangle \\ & + \sum_{j=1,\ldots,m} \left\langle U_{j+m}^\ell, \Phi_{p^\ell}(\overline{Y} - \mathrm{smat}(x)) \right\rangle ; \end{aligned}$$

here u^ℓ are Lagrangian multipliers associated with real valued constraints, φ_{p^ℓ} is the scalar version of Φ_{p^ℓ},

$$\mathcal{K}_1 := \{k \in \{1, 2, \ldots, K\} \mid g_k \text{ depends on exactly one matrix variable}\} ,$$

and $\mathcal{K}_0 = \{1, 2, \ldots, K\} \setminus \mathcal{K}_1$. Now we define

$$q(x) := \sum_{k \in \mathcal{K}_0} u_k^\ell \varphi_{p^\ell}(g_j(x)), \qquad r(x) := F(x, u^\ell, U^\ell, p^\ell) - q(x).$$

Obviously, $r(x)$ is separable w.r.t. $\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m$. On the other hand, we obtain the following for $q(x)$:

$$\nabla_x^2 q(x) = \sum_{k \in \mathcal{K}_0} u_k^\ell \left(\varphi''_{p^\ell}(g_k(x)) \nabla_x g_k(x) \nabla_x g_k(x)^\top + \varphi'_{p^\ell}(g_k(x)) \nabla^2_{xx} g_k(x) \right) .$$

Now, defining β as a vector with entries $\beta_k^\ell(x) := 1/(u_k^\ell \varphi'_{p^\ell}(g_k(x)))$ $(k \in \mathcal{K}_0)$,

$$H^\ell(x) := \nabla^2_{xx} r(x) + \sum_{k \in \mathcal{K}_0} u_k^\ell \varphi'_{p^\ell}(g_k(x)) \nabla^2_{xx} g_k(x)$$

and denoting by $A^\ell(x)$ the matrix with columns $\nabla_x g_k(x)$ $(k \in \mathcal{K}_0)$ we are able to prove the following result.

Proposition 5.2. *Any vector $d \in \mathbb{R}^{\hat{n}}$ satisfying the equation*

$$\begin{pmatrix} H^\ell(x) & A^\ell(x) \\ A^{\ell,\top}(x) & -\mathrm{diag}(\beta) \end{pmatrix} \begin{pmatrix} d \\ y \end{pmatrix} = \begin{pmatrix} -\nabla_x F(\hat{x}, U^\ell, p^\ell) \\ 0 \end{pmatrix} \tag{5.7}$$

is a solution of the linear system (5.6)*. Moreover, the matrix $H^\ell(x)$ is separable w.r.t. $\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m$.*

Proof. From the second line in system (5.7) we see that

$$y = \mathrm{diag}(\beta)^{-1} A^{\ell,\top}(x) d. \tag{5.8}$$

Now, substituting y by the right-hand side of (5.8), the first row of (5.7) becomes

$$\left(H^\ell(x) + A^\ell(x) \mathrm{diag}(\beta)^{-1} A^{\ell,\top}(x) \right) d = -\nabla_x F(\hat{x}, U^\ell, p^\ell),$$

but this is exactly system (5.6). □

Depending on the cardinality of the set $\mathcal{K}_0$, the system (5.7) is solved directly or by the following strategy:

1. Compute $(H^\ell(x))^{-1} \nabla_x F(\hat{x}, U^\ell, p^\ell)$ and $(H^\ell(x))^{-1} A^\ell(x)$.
2. Solve the system

$$\left(A^{\ell,\top}(x) (H^\ell(x))^{-1} A^\ell(x) - \mathrm{diag}(\beta) \right) y = (H^\ell(x))^{-1} \nabla_x F(\hat{x}, U^\ell, p^\ell).$$

3. Compute $d = -(H^\ell(x))^{-1} A^\ell(x) y - (H^\ell(x))^{-1} \nabla_x F(\hat{x}, U^\ell, p^\ell)$.

The second variant can be viewed as a generalization of the dual technique used in the original MMA paper [26].

6. FMO.

We briefly introduce the FMO problem.

Let $\Omega \subset \mathbb{R}^2$ be a 2D bounded domain[1] with a Lipschitz boundary. By $u(x) = (u_1(x), u_2(x))$ we denote the displacement vector at a point x of the body under an external load and by

$$e_{ij}(u(x)) = \frac{1}{2}\left(\frac{\partial u_i(x)}{\partial x_j} + \frac{\partial u_j(x)}{\partial x_i}\right) \quad \text{for } i,j = 1,2$$

the associated (small-)strain tensor. We assume that our system is governed by linear Hooke's law; i.e., the stress is a linear function of the strain

$$\sigma_{ij}(x) = E_{ijk\ell}(x) e_{k\ell}(u(x)) \qquad \text{(in tensor notation)},$$

where E is the elastic stiffness tensor. The symmetries of E allow us to write the second order tensors e and σ as vectors

$$e = (e_{11}, e_{22}, \sqrt{2}e_{12})^T \in \mathbb{R}^3, \ \sigma = (\sigma_{11}, \sigma_{22}, \sqrt{2}\sigma_{12})^T \in \mathbb{R}^3 .$$

Correspondingly, the fourth order tensor E can be written as a symmetric (sym.) 3×3 matrix

$$E = \begin{pmatrix} E_{1111} & E_{1122} & \sqrt{2}E_{1112} \\ & E_{2222} & \sqrt{2}E_{2212} \\ \text{sym.} & & 2E_{1212} \end{pmatrix}. \tag{6.1}$$

In this notation, Hooke's law reads as $\sigma(x) = E(x)e(u(x))$.

Given a set of external load functions $f_\ell \in [L_2(\Gamma)]^2,\ \ell \in \mathcal{L}_{lc} = \{1, 2, \ldots, n_{lc}\}$, where Γ is a part of $\partial\Omega$ that is not fixed by Dirichlet boundary conditions, we are able to state for each load case $\ell \in \mathcal{L}_{lc}$ a basic boundary value problem of the following type:

$$\text{Find } u_\ell \in [H^1(\Omega)]^2, \text{ such that} \tag{6.2}$$
$$\begin{array}{rcll} \operatorname{div}(\sigma) & = & 0 & \text{in} \quad \Omega, \\ \sigma \cdot n & = & f_\ell & \text{on} \quad \Gamma, \\ u_\ell & = & 0 & \text{on} \quad \Gamma_0, \\ \sigma & = & E \cdot e(u_\ell) & \text{in} \quad \Omega\,. \end{array}$$

Here Γ and Γ_0 are open disjunctive subsets of $\partial\Omega$. Applying Green's formula, we obtain the following so-called weak equilibrium equation:

$$\text{Find } u_\ell \in \mathcal{V} \text{ such that} \tag{6.3}$$
$$\int_\Omega \langle E(x)e(u_\ell(x)), e(v(x))\rangle \mathrm{d}x = \int_\Gamma f_\ell(x) \cdot v(x) \mathrm{d}x \quad \forall v \in \mathcal{V},$$

where $\mathcal{V} = \{u \in [H^1(\Omega)]^2 \,|\, u = 0 \text{ on } \Gamma_0\} \supset [H_0^1(\Omega)]^2$ reflects the Dirichlet boundary conditions.

[1] The entire presentation is given for 2D bodies to keep the notation simple. Analogously, all this can be done for 3D solids.

In FMO, the design variable is the elastic stiffness tensor E which is a function of the space variable x (see [2, 21]). The only constraints on E are that it is physically reasonable, i.e., that E is symmetric and positive semidefinite. This gives rise to the following definition:

$$\mathcal{E}_0 := \left\{ E \in L^\infty(\Omega)^{3\times 3} \mid E = E^\top, E \succeq \underline{\rho} I \text{ a.e. in } \Omega \right\},$$

where $\underline{\rho} \in \mathbb{R}^+$ is a suitable nonnegative number and I denotes the identity matrix. The choice of L^∞ is due to the fact that we allow for maximal-material/minimal-material situations. A frequently used measure for the stiffness of the material tensor is its trace. In order to avoid arbitrarily stiff material, we add pointwise stiffness restrictions of the form $\mathrm{Tr}(E) \leq \overline{\rho}$, where $\overline{\rho}$ is a finite real number. Accordingly, we define the *set of admissible materials* as

$$\mathcal{E} := \left\{ E \in L^\infty(\Omega)^{3\times 3} \mid E = E^\top, E \succeq \underline{\rho} I, \mathrm{Tr}(E) \leq \overline{\rho} \text{ a.e. in } \Omega \right\}.$$

The following result is an immediate consequence of the definition of $\mathcal{E}$ (see [17]).

LEMMA 6.1. *If $\underline{\rho} > 0$, the bilinear form*

$$a_E : \mathcal{V} \times \mathcal{V} \to \mathbb{R}, (w, v) \mapsto \int_\Omega \langle E(x) e(w(x)), e(v(x)) \rangle \mathrm{d}x$$

is $\mathcal{V}$-elliptic and bounded for all $E \in \mathcal{E}$.

Now we are able to present the *worst-case multiple-load FMO problem*:

$$\tag{6.4} \begin{aligned} &\inf_{\substack{u\in\mathcal{V},\\ E\in\mathcal{E}}} \max_{\ell\in\mathcal{L}_{lc}} \int_\Gamma f_\ell(x) \cdot u_\ell(x) \mathrm{d}x \\ &\text{subject to} \\ &\qquad u_1, u_2, \ldots, u_{n_{lc}} \text{ solve equilibrium equations of form (6.3)}, \\ &\qquad v(E) \leq \bar{v}\,. \end{aligned}$$

Here the volume $v(E)$ is defined as $\int_\Omega \mathrm{Tr}(E)\mathrm{d}x$ and $\bar{v} \in \mathbb{R}$ is an upper bound on overall resources. Moreover, the objective, the so-called worst-case compliance functional, measures how well the structure can carry the loads f_ℓ, $\ell \in \mathcal{L}_{lc}$. As an alternative to problem (6.4), one can also consider the *weighted multiple-load FMO problem*

$$\tag{6.5} \begin{aligned} &\inf_{\substack{u\in\mathcal{V},\\ E\in\mathcal{E}}} \sum_{\ell\in\mathcal{L}_{lc}} w_\ell \int_\Gamma f_\ell(x) \cdot u_\ell(x) \mathrm{d}x \\ &\text{subject to} \\ &\qquad u_1, u_2, \ldots, u_{n_{lc}} \text{ solve equilibrium equations of form (6.3)}, \\ &\qquad v(E) \leq \bar{v}\,; \end{aligned}$$

here the values $w_\ell \in \mathbb{R}_+$ ($\ell \in \mathcal{L}_{lc}$) are given weights of the associated load cases. Note that for $\ell = 1$ (*single load FMO problem*) both problems coincide.

The major concern of this article is to find an efficient procedure for the numerical solution of the above FMO problems. Therefore, we continue directly with the presentation of the discrete counterparts of problems (6.4) and (6.5). For a more detailed analysis of the infinite dimensional problems the interested reader is referred to [4, 29].

The most successful approach proposed for the numerical solution of problems (6.4) and (6.5) is based on dualization and subsequent discretization; see [4, 29]. Having many advantages, this strategy turns out to have two major drawbacks:

- The computational complexity depends cubically on the number of load cases (see [14]). This makes the approach impractical for 3D problems with more than a few (typically 3–5) load cases.
- It is difficult to apply the dual approach for problem statements extended by additional constraints on the design variable E or the state variable u (see, for instance, [12, 15]). Especially in the case of nonconvexity, the dual formulation may become useless due to the existence of a duality gap.

Motivated by this, we propose to solve a discretized version of problem (6.4) (or alternatively (6.5)) directly. We define the following finite element scheme, which is based on the discretization schemes used in [4, 29]:

The design space Ω is partitioned into m elements called $\Omega_i, i = 1, \dots, m$. For simplicity, we assume that all elements are of quadrilateral type of the same size $h \in \mathbb{R}$ (otherwise we use the standard isoparametric concept; see, for instance, [7]). We approximate the matrix function $E(x)$ by a function that is constant on each element, i.e., characterized by a vector of matrices $E = (E_1, \dots, E_m)$ of its element values. Hence the discrete counterpart of the set of admissible materials in algebraic form is

$$\widetilde{\mathcal{E}} = \left\{ E \in (\mathbb{S}^3)^m \;\middle|\; E_i \succeq \underline{\rho} I,\ \operatorname{Tr}(E_i) \le \overline{\rho},\quad i = 1, \dots, m \right\}. \tag{6.6}$$

Moreover, the discrete resource constraint takes the form

$$\sum_{i=1}^{m} \operatorname{Tr}(E_i) \le V,$$

where $V = h\bar{v}$ and $\bar{v}$ is the upper bound on resources introduced in (6.5). Further we assume that the displacement vectors $u_\ell(x)$ $(\ell \in \mathcal{L}_{lc})$ are approximated by continuous functions that are bilinear in each coordinate on every element. Such functions can be written as $u_\ell(x) = \sum_{i=1}^{n} u_{\ell,i} \vartheta_i(x)$ for all $\ell \in \mathcal{L}_{lc}$, where $u_{\ell,i}$ is the value of u_ℓ at the ith node and ϑ_i is a basis function with nodal interpolation property associated with the ith node (for details, see [7]). Now each admissible displacement function can be identified with a vector in $\mathbb{R}^n$, where $n = 2N - \#$(components of u_ℓ fixed by Dirichlet boundary conditions) and N is the number of nodes (vertices of the elements Ω_i) in the discrete design space. For the discussion on other approximation schemes and their relation to the Babuška–Brezzi condition, see, e.g., [23] and the references therein.

Next we derive the discrete counterpart of $a_E(\cdot, \cdot)$. Along with the family of basis functions $\vartheta_l, l = 1, \dots, n$, we define a 3×2 matrix

$$\bar{B}_j^T = \begin{pmatrix} \dfrac{\partial \vartheta_j}{\partial x_1} & 0 & \dfrac{1}{2}\dfrac{\partial \vartheta_j}{\partial x_2} \\ 0 & \dfrac{\partial \vartheta_j}{\partial x_2} & \dfrac{1}{2}\dfrac{\partial \vartheta_j}{\partial x_1} \end{pmatrix}$$

and associate with each element Ω_i a set $\mathcal{D}_i$ of nodes belonging to this element. We use a Gauss formula for the evaluation of the integral over each element Ω_i, assume that there are n_{ig} Gauss integration points on each element, and denote by $x_{i,k}^G$ the kth integration point associated with the ith element. Using this, we construct block matrices $B_{i,k} \in \mathbb{R}^{3 \times n}$ composed of (3×2) blocks $\bar{B}_j(x_{i,k}^G)$ at the jth position for every $j \in \mathcal{D}_i$ and zero blocks of the same size otherwise. Then the discrete counterpart of

$a_E(\cdot,\cdot)$, the *stiffness matrix* A, is

$$A(E) = \sum_{i=1}^{m} A_i(E), \quad A_i(E) = \sum_{k=1}^{n_{ig}} B_{i,k}^T E_i B_{i,k}\,. \tag{6.7}$$

Finally, assuming the load functions f_ℓ ($\ell \in \mathcal{L}_{lc}$) to be linear on each element and identifying each such function with a vector $f_\ell \in \mathbb{R}^n$, the discrete compliance functionals and equilibrium conditions read as

$$f_\ell^\top u, \quad A(E)u_\ell = f_\ell, \qquad \ell \in \mathcal{L}_{lc}, \tag{6.8}$$

respectively. Using the assumption $\underline{\rho} > 0$, it follows from Lemma 6.1 that $A(E)$ is strictly positive definite, and we are able to eliminate u_ℓ, $\ell \in \mathcal{L}_{lc}$, from the equations above and to rewrite the compliance functionals as

$$c_\ell(E) := f_\ell^\top A^{-1}(E) f_\ell \quad \text{for all } \ell \in \mathcal{L}_{lc}.$$

Thus, after discretization, problems (6.4) and (6.5) become

- *discrete worst-case multiple-load FMO problem*

$$\min_{E \in \tilde{\mathcal{E}}} \max_{\ell \in \mathcal{L}_{lc}} f_\ell^\top A^{-1}(E) f_\ell \quad \text{subject to} \quad \sum_{i=1}^{m} \operatorname{Tr}(E_i) \le V, \tag{6.9}$$

- *discrete weighted multiple-load FMO problem*

$$\min_{E \in \tilde{\mathcal{E}}} \sum_{\ell \in \mathcal{L}_{lc}} w_\ell f_\ell^\top A^{-1}(E) f_\ell \quad \text{subject to} \quad \sum_{i=1}^{m} \operatorname{Tr}(E_i) \le V. \tag{6.10}$$

Definition 6.2. *Let $\bar{x}_i \in \mathbb{R}^6$ for all $i = 1, \dots, m$, and define*

$$\begin{aligned} \tilde{c}_\ell :&(\mathbb{R}^6)^m \to \mathbb{R} \\ x = (\bar{x}_1^\top, \dots, \bar{x}_m^\top) &\mapsto c_\ell\left((\operatorname{smat}(\bar{x}_1), \dots, \operatorname{smat}(\bar{x}_m))\right). \end{aligned}$$

In the following lemma we summarize some useful properties of the compliance functionals c_ℓ ($k \in \mathcal{K}$).

Lemma 6.3. *For all $\ell \in \mathcal{L}_{lc}$ the following hold:*

(a) *c_ℓ is well-defined, infinitely often continuously differentiable, and convex on $\tilde{\mathcal{E}}$. Moreover, the formula*

$$\frac{\partial}{\partial E_i} c_\ell(E) = -\sum_{k=1}^{n_{ig}} B_{i,k} u_\ell(E) u_\ell(E)^\top B_{i,k}^\top, \qquad u_\ell(E) := A^{-1}(E) f_\ell,$$

holds true for all partial derivatives of c_ℓ and $\frac{\partial}{\partial E_i} c_\ell(E)$ is negative semidefinite for all $i = 1, 2, \dots, m$ and all $E \in \tilde{\mathcal{E}}$.

(b) *$\tilde{c}_\ell$ is infinitely often continuously differentiable and convex on*

$$\mathcal{X} := \left\{(\bar{x}_1, \dots, \bar{x}_m) \mid (\operatorname{smat}(\bar{x}_1), \dots, \operatorname{smat}(\bar{x}_m)) \in \tilde{\mathcal{E}}\right\}.$$

(c) *The Hessian of $\tilde{c}_\ell$ is dense.*

Proof. We start with the proof of assertion (b): The global stiffness matrix $A(E)$ can be written as a linear operator of the form

$$A(E_1, \dots, E_m) = \sum_{i=1}^{m} \sum_{j=1}^{6} (\bar{x}_i)_j A_i^{p(j),q(j)},$$

where $\bar{x}_i := \operatorname{svec}(E_i)$, $p(j)$ and $q(j)$ are the row and column indices, respectively, of the element $(\bar{x}_i)_j$ in the lower triangular part of the matrix E_i, and the matrices $A_i^{p(j),q(j)}$ are defined as

$$(6.11) \qquad A_i^{p(j),q(j)} = \begin{cases} \displaystyle\sum_{k=1}^{n_{ig}} (B_{i,k})_{p(j)}^\top (B_{i,k})_{q(j)}, & p(j) = q(j), \\ \displaystyle\sum_{k=1}^{n_{ig}} (B_{i,k})_{p(j)}^\top (B_{i,k})_{q(j)} + (B_{i,k})_{q(j)}^\top (B_{i,k})_{p(j)}, & p(j) \neq q(j), \end{cases}$$

with $(B_{i,k})_j$ denoting the jth row of $B_{i,k}$. Thus the mapping

$$\tilde{A}: \ (\mathbb{R}^6)^m \to \mathbb{S}^n, \ x \mapsto A\left(\operatorname{smat}(\bar{x}_1), \dots, \operatorname{smat}(\bar{x}_m)\right)$$

is linear in $x := (\bar{x}_1, \dots, \bar{x}_m)$. Moreover, it follows from Lemma 6.1 that $\tilde{A}(x)$ is positive definite for all $x \in \mathcal{X}$. Taking into account that the mapping $A \mapsto A^{-1}$ is convex (infinitely often continuously differentiable) on $\mathbb{S}_+^n$ (see [11]) and writing $\tilde{c}_\ell$ as

$$\tilde{c}_\ell(x) = \langle \tilde{A}^{-1}(x), f_\ell f_\ell^\top \rangle_{\mathbb{S}^n},$$

the assertion of part (b) follows from the fact that $\tilde{c}_\ell$ is the composition of a linear, a convex (infinitely often continuously differentiable), and a linear function.

Next we prove (a): The well definedness follows directly from Lemma 6.1. The differentiability and convexity follow from assertion (b). Using the matrices $A_i^{j,p}$ defined in (6.11) to rewrite $A_i(E) = \sum_{1 \le j \le p \le 3} (E_i)_{j,p} A_i^{j,p}$, we obtain

$$\frac{\partial}{\partial (E_i)_{j,p}} c_\ell(E) = -u_\ell(E)^\top \left(\frac{\partial}{\partial (E)_{j,p}} A(E) \right) u_\ell(E) = -u_\ell(E)^\top A_i^{j,p} u_\ell(E),$$

where $u_\ell(E) = A^{-1}(E) f_\ell$ and $(E_i)_{j,p}$ denotes the (j,p) element of the matrix E_i. Using the definition of $A_i^{j,p}$ we can now conclude that $\frac{\partial}{\partial E_i} c_\ell(E) = -\sum_{k=1}^{n_{ig}} B_{i,k} u_\ell(E) u_\ell (E)^\top B_{i,k}^\top$, and the negative semidefiniteness follows immediately from the dyadic structure of the matrix $u_\ell(E) u_\ell(E)^\top$.

Finally, we show assertion (c): Using the same arguments as the proof of part (a), we obtain

$$\frac{\partial^2}{\partial (x_i)_j \partial (x_p)_q} \tilde{c}_\ell(x) = 2 u_\ell(E)^\top A_i^j \ A^{-1}(E) \ A_p^q \ u_\ell(E),$$

which is in general a nonzero value. □

The following corollary is a direct consequence of Lemma 6.3.

Corollary 6.4. *Problems* (6.9) *and* (6.10) *are convex semidefinite programming problems.*

Remark 6.1. As a consequence of Corollary 6.4, one could try to apply an existing nonlinear (convex) SDP solver directly in order to solve (6.9) or (6.10). This is, however, not recommended in practice with any SDP solver using explicit second-order derivatives, due to Lemma 6.3(c). Experiments with an SDP solver avoiding the calculation and storage of explicit second-order derivatives by Krylov-type methods (see [16]) led to moderate success, as the effort for the approximation of $\nabla^2 \tilde{c}_\ell$ turned out to be too expensive. This was the reason why we decided to develop an SDP solver that is solely based on first-order information.

7. Numerical experiments.

The main goals of this section are to

- provide algorithmic details of our implementation of Algorithm 4.3,
- present the results of numerical experiments with Algorithm 4.3 applied to FMO problems of the form (6.9).

7.1. Algorithmic details.

The choice of the asymptotes. As a consequence of Lemma 6.3(a) the formula for the hyperbolic approximation of c_ℓ ($\ell \in \mathcal{L}_{lc}$) in the jth iteration reduces to

$$\begin{aligned}(c_\ell)^{L,\tau}_{E^j}(E) &:= c_\ell(E^j)\\ &\quad + \sum_{i=1}^{m} \left\langle \nabla^i c_\ell(E^j), (E_i^j - L_i)(E_i - L_i)^{-1}(E_i^j - L_i) - (E_i^j - L_i) \right\rangle_{\mathbb{S}^3}\\ &\quad + \sum_{i=1}^{m} \tau_i \left\langle (E_i - E_i^j)^2, (E_i - L_i)^{-1} \right\rangle_{\mathbb{S}^3} . \end{aligned} \tag{7.1}$$

For this reason we neglect the upper asymptotes U below. We have investigated two different types of schemes: a moving scheme and a constant scheme. For the moving scheme we used direct generalizations of the update rules recommended in [27, 32]. In the case of the constant scheme we simply used

$$L_i^j = L_0 \prec \underline{\rho} I \text{ for all } i \in \mathcal{I} \text{ and all iterates } j = 1, 2, 3, \dots .$$

The result of our experiments showed that (in sharp contrast to the original MMA/SCP approach) the moving scheme brings almost no benefit compared to the constant one. Additionally, the constant scheme has an important advantage: The feasible set of the subproblems can be kept during all iterations. This allows for an extensive use of warm starts when solving the inner convex semidefinite programs. As a consequence, we observed a significant reduction in the number of inner iterations. This is of particular importance, because the subproblem in FMO is much more expensive to solve than, for example, in SIMP-based problems.

For this reason we report only on experiments with the constant scheme of asymptotes. The most efficient constant choice we found was $L_0 = 0$. Note that the combination of Algorithm 4.3 with this simple choice applied to FMO-type problems can be interpreted as a direct generalization of CONLIN [8]. The latter choice leads to a further simplification of (7.1):

$$\begin{aligned}(c_\ell)^{\tau}_{E^j}(E) := c_\ell(E^j) &+ \sum_{i=1}^{m} \left\langle \nabla^i c_\ell(E^j), E_i^j E_i^{-1} E_i^j - E_i^j \right\rangle_{\mathbb{S}^3}\\ &+ \sum_{i=1}^{m} \tau_i \left\langle (E_i - E_i^j)^2, E_i^{-1} \right\rangle_{\mathbb{S}^3}, \quad \ell \in \mathcal{L}_{lc}.\end{aligned}$$

The subproblems. Using the constant asymptote scheme described above, the subproblems of (6.9) become

$$\begin{aligned} &\min_{E\in\widetilde{\mathcal{E}}}\max_{\ell\in\mathcal{L}_{lc}}(c_\ell)^{\tau}_{E^j}(E) \\ &\text{subject to} \\ &\qquad\sum_{i=1}^{m}\operatorname{Tr}(E_i)\le V\,. \end{aligned} \tag{7.2}$$

During all iterations, we solve the subproblems approximately. We use the following strategy: We start with a moderate accuracy of $\varepsilon = 10^{-3}$ for the KKT error of (7.2). Whenever the calculated search direction fails to be a descent direction, we decrease the precision by a constant factor. Lemmas 4.7 and 4.8 show that when we solve the subproblem exactly, we end up with a descent direction of sufficient decrease. Furthermore, it can be seen from the proof of Lemma 4.8 that the same holds true for a perturbed solution, provided its solution is close enough to the exact solution. Consequently, our simple strategy is guaranteed to terminate after a finite number of steps with an acceptable descent direction.

The line search. Instead of preforming an exact minimization in step (4) of Algorithm 4.3, we use a simple Armijo rule, guaranteeing sufficient descent. Our experience shows that, already after few outer iterates of Algorithm 4.3, the step length $\alpha^j = 1$ is accepted in almost all iterates.

The choice of τ. The parameters τ_i $(i \in \mathcal{I})$ are chosen such that the following condition remains valid throughout all iterations:

$$-\nabla^i c_\ell(E^j) + \tau_i I \succeq \delta I \qquad (i \in \mathcal{I})$$

for all $i \in \mathcal{I}$ and all $\ell \in \mathcal{L}_{lc}$. A typical choice for δ is 10^{-4}.

A practical stopping criterion. We use two stopping criteria for Algorithm 4.3. The first one is based on the relative difference of two successive objective function values. We consider this stopping criterion as achieved if the relative difference falls below some given threshold ϵ_1 (typically $\epsilon_1 = 10^{-8}$). The second stopping criterion is based on the following KKT-related error measures:

$$\begin{aligned} \mathrm{err}_1 &= \left\|\nabla L(Y^l, u^l, \underline{U}^l, \overline{U}^l)\right\|, \\ \mathrm{err}_2 &= \max\{g_k(Y^l) \mid k = 1, 2, \dots, K\}, \\ \mathrm{err}_3 &= \max\Big\{|u^l_k g_k(Y^l)|, |\langle \underline{U}^l_j, Y^l_j - \underline{Y}_j\rangle|, |\langle \overline{U}^l_j, \overline{Y}_j - Y^l_j\rangle| \;\Big|\; k = 1, \dots, K,\ j = 1, \dots, m\Big\}, \end{aligned}$$

where Y^l is the approximate solution at iterate l; L is the Lagrangian associated with problem $(\mathcal{P})$ defined in section 2; and u^l, $\underline{U}^l$, and $\overline{U}^l$ are the corresponding vectors of Lagrangian (matrix) multipliers associated with the constraint functions g_k and the lower and upper matrix bound constraints, respectively. Recall that the feasibility of Y^l w.r.t. the matrix bound constraints is maintained throughout all iterations. Now we define our second stopping criterion as

$$\frac{1}{3}\sum_{i=1}^{3}\mathrm{err}_i \le \epsilon_2, \tag{7.3}$$

where a typical value for ϵ_2 is $5 \cdot 10^{-5}$. Note that we stop only when both stopping criteria are satisfied simultaneously.

The code. We have implemented the new algorithm in the C programming language. In what follows we refer to the resulting code as Penscp.

7.2. Numerical studies with FMO problems. Before we start presenting the numerical results we take a brief look at the (theoretical) computational complexity of Algorithm 4.3, when applied to FMO problems of type (6.9).

Computational complexity. We split the whole process in two subtasks, namely, *model calculation* and *optimization.* The model calculation includes the evaluation of $c_\ell(E)$ ($\ell \in \mathcal{L}_{lc}$), the computation of all partial derivatives and some preassembling steps for the hyperbolic approximations. The optimization covers the solution of a subproblem of type (7.2). Clearly, the model calculation is dominated by the factorization of the global stiffness matrix $A(E)$. The factorization is performed by a sparse Cholesky method (see [18]) whose complexity depends linearly on the number of nonzero entries in $A(E)$. From this, (6.7), (6.8), and Lemma 6.3 we conclude the following: The computational complexity of the *model calculation* phase depends

- *linearly* on the number of elements in discretization,
- *linearly* on the number of load cases.

In the *optimization* phase, the most time-consuming steps are the calculation of gradients and Hessians of the hyperbolic approximations and the factorization of (5.7). Obviously, both types of operations depend

- *linearly* on the number of elements in discretization,
- *linearly* on the number of load cases.

Goals of the numerical experiments. The goals of the numerical experiments presented in the remainder of this section are

- numerical verification of the linear dependence of the computational complexity of Algorithm 4.3 on the number of elements,
- numerical verification of the linear dependence of the computational complexity of Algorithm 4.3 on the number of load cases,
- a comparsion with Moped3, the most recent implementation of the dual method described in [4],
- the effect of a SIMP-like preprocessing step.

All experiments have been performed on a Sun Opteron machine with 8 Gbyte of memory and processor speed of approximately 3 GHz.

2D examples. The goal of our first experiment is to verify the linear dependence of the computational complexity of Algorithm 4.3 on the number of elements in the finite element discretization. In order to do that, we solve several instances of the test problem depicted in Figure 7.1 with an increasing number of elements. Moreover, we compare the calculation times of Penscp and Moped3 on this example. The results are summarized in Table 7.1. The meaning of columns 1 to 6 is the following: the number of finite elements, the number of iterations performed by Penscp, the relative precision reached by Penscp (w.r.t. a high quality approximation of the accurate solution computed by Moped3), the KKT error given by (7.3), the computation time required by Penscp, and the computation time required by Moped3.

For both codes we observe for a factor of 4 in the number of elements a corresponding factor of 7–8 in the computational time. Thus the numerical experiment approximately confirms an almost linear growth. Optimal density distributions obtained from Experiment 1 are shown in Figures 7.2 and 7.3.

By means of our second experiment we try to verify the linear dependence of the computational complexity of Algorithm 4.3 on the number of load cases. Therefore, we solve the basic test with an increasing number of load cases. Again we compare

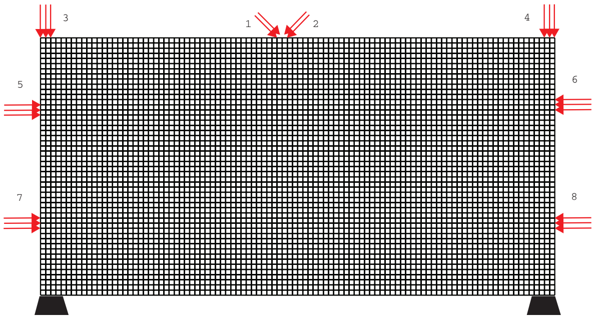


FIG. 7.1. *Basic test problem–mesh, boundary conditions, and forces.*

TABLE 7.1
Experiment 1. 1.250–20.000 *elements.*

FEs	Iter.	Precision	KKT error	Time in sec. (opt/mod)	Time in sec. MOPED3
1.250	622	5.0e-5	4.5e-5	256 (175/81)	153
5.000	489	1.2e-4	5.0e-5	1.027 (653/374)	996
20.000	522	1.3e-4	2.5e-5	7.878 (6.120/1.758)	8.732

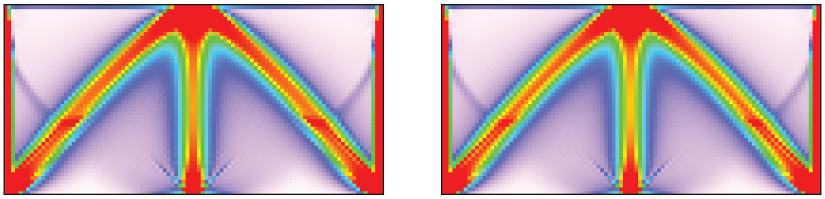


FIG. 7.2. 5000 *elements,* 4 *load cases.* MOPED3 *(left),* PENSCP *(right).*

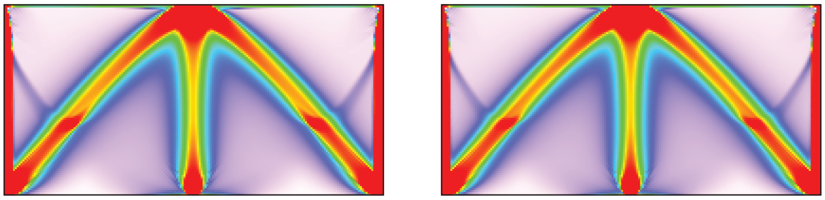


FIG. 7.3. 20.000 *elements,* 4 *load cases.* MOPED3 *(left),* PENSCP *(right).*

TABLE 7.2
Experiment 2. 2–8 *load cases.*

# LC	Iter.	Precision	KKT error	Time in sec. (opt/mod)	Time in sec. MOPED3
2	543	1.4e-4	5.0e-5	585 (423/162)	182
4	489	1.2e-4	5.0e-5	1.027 (653/374)	996
8	370	1.0e-4	2.5e-5	1.319 (749/570)	7.212

the results of PENSCP to MOPED3. The results are summarized in Table 7.2. Note that here and below # LC denotes the number of load cases.

Obviously, PENSCP shows a much better behavior than MOPED3 here. For a factor of 2 in the number of load cases we observe only a factor of 1.4–2 in the computational time. On the other hand, MOPED3 shows the predicted quadratic to

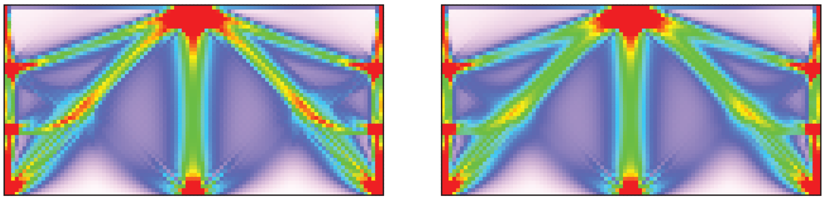


FIG. 7.4. 5.000 *elements,* 8 *load cases.* MOPED3 *(left),* PENSCP *(right).*

TABLE 7.3
Experiment 3. *Accuracy and speed of convergence.*

# Accurate digits	# Iterations	Rounded function value
1	2	2
2	10	1.5
3	107	1.50
4	288	1.499
5	727	1.4987
6	977	1.49873
7	1251	1.498732

cubic behavior: For a factor of 2 in the number of load cases we observe a factor of 6–8 in the computational time. Optimal density distributions obtained from this experiment are depicted in Figure 7.4.

In order to get an idea about the accuracy and the speed of convergence we can reach by our method, we tried to solve the problem depicted in Figure 7.1 to higher precision. This time we chose four load cases and a resolution of 1250 finite elements. The result is outlined in Table 7.3: We are able to compute seven digits of accuracy. In this example the error diminishes approximately with linear speed of convergence. We observed very similar results for experiments with different numbers of load cases and finer resolutions.

The goal of the fourth experiment is to test the effect of a preprocessing strategy on the comparatively high number of outer iterations required by PENSCP. We make use of the following *preprocessing strategy*:

1. Run a few (10–20) steps of Algorithm 4.3 for a SIMP model obtained from (7.2) by setting $E_i = \rho_i^2 I$ for all $i = 1, 2, \ldots, m$ and letting $\rho = (\rho_1, \rho_2, \ldots, \rho_m)^\top$ be the design variable.
2. Approximate material tensors using the formula

$$E_i \approx \frac{\rho_i}{\#LC} \sum_{\ell \in \mathcal{L}_{lc}} \bar{e}(u_\ell)\bar{e}(u_\ell)^T,$$

where u_ℓ are displacements associated with the intermediate densities calculated in step 1 and $\bar{e}(u_\ell)$ is a corresponding normalized small-strain tensor.

For a motivation of the above strategy we refer the reader to [33]. Using preprocessing, we computed our basic example with

- 5.000 finite elements and 8 load cases,
- 20.000 finite elements and 4 load cases.

PENSCP was stopped after 150 iterations in both cases. The resulting density plots are depicted in Figures 7.5 and 7.6.

It seems that the preprocessing strategy significantly improved the result after 150 iterations. In both cases we could save about 60 percent of the computation time.

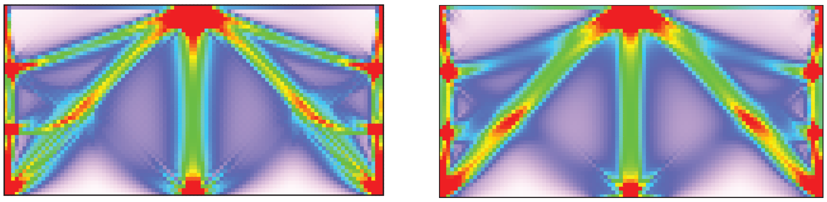


FIG. 7.5. 5.000 *elements,* 8 *LC, with preprocessing.* MOPED3 *(left),* PENSCP *(right).*

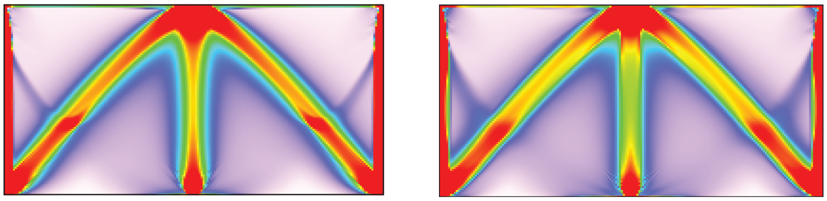


FIG. 7.6. 20.000 *elements,* 4 *LC, with preprocessing.* MOPED3 *(left),* PENSCP *(right).*

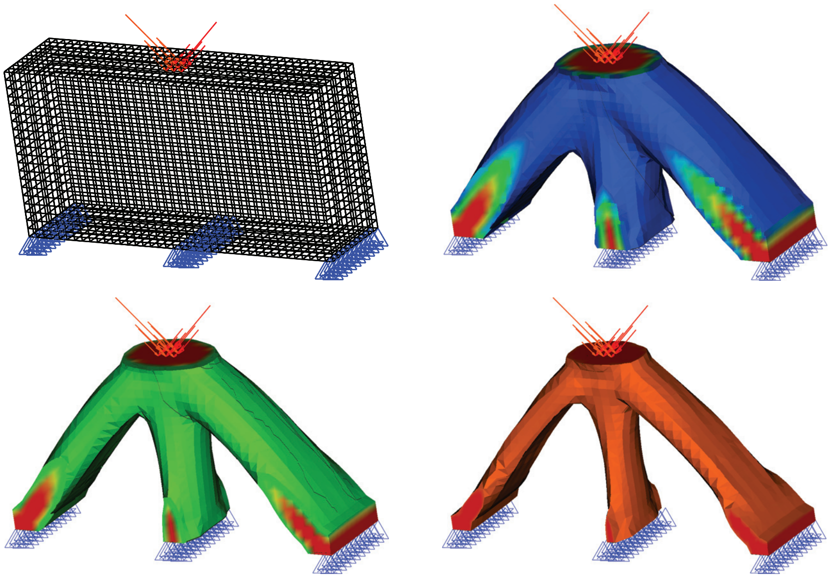


FIG. 7.7. *Problem setting.* ≈ 10.000 *elements,* 2 *LC (top left), unfiltered density result* PENSCP *(top right), and filtered density plots (bottom).*

3*D experiments.* We performed experiments with PENSCP on two 3D examples. In our first experiment we used a solid block discretized by approximately 10.000 finite elements. Moreover, we applied 2 load cases (see Figure 7.7, top left). PENSCP stopped after almost 500 iterations and approximately 1.5 hours of computation time. MOPED3 required already 8 hours. The problem setting as well as the density result generated by PENSCP can be seen in Figure 7.7.

In our second 3D experiment we used again a solid block, this time subjected to 4 load cases and discretized by approximately 20.000 finite elements. PENSCP generated

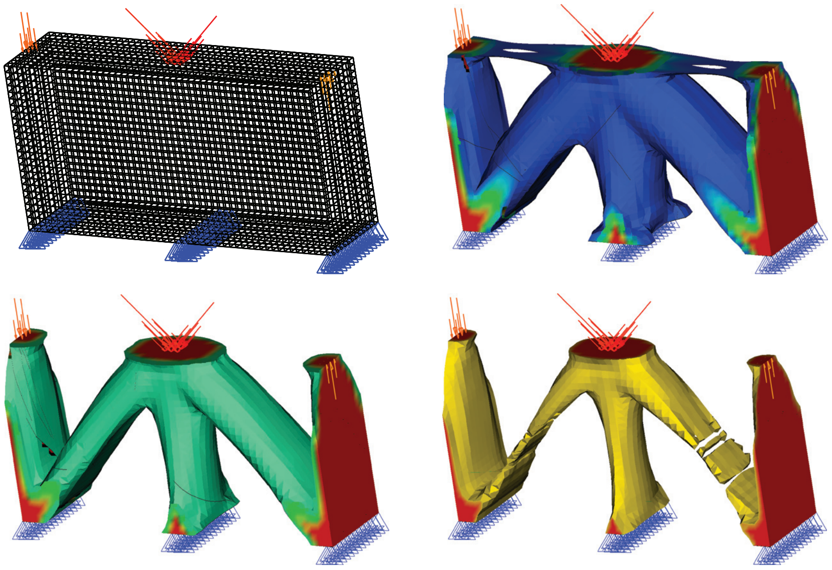


FIG. 7.8. *Problem setting.* ≈ 20.000 *elements,* 4 *LC (top left), unfiltered density result* PENSCP *(top right), and filtered density plots (bottom).*

a solution in approximately 4 hours. MOPED3 failed for this example, because the memory of 8 Gbyte was exceeded. An estimate based on the results from the 3D experiment described above yields a computation time of approximately two weeks for this example. The problem setting along with the density result computed by PENSCP is depicted in Figure 7.8.

8. Conclusion and outlook. We have developed a globally convergent method for the minimization of convex nonlinear functions defined on matrix spaces over convex sets described by (separable) convex constraints. The new method turned out to be particularly efficient when applied to FMO problems with multiple load cases. The key strategy of the new method is to replace the original optimization problem by a sequence of convex semidefinite programs. The structure of these semidefinite programs has a strong influence on the efficiency of the overall method. For example, we have seen that an efficient solution is possible if all constraints are separable or even linear. If this is not the case, or more generally, if one wants to deal with nonconvex functions, the situation is more involved. The authors are currently investigating a generalized algorithmic concept for this case (see [24]).

Acknowledgments. The authors would like to thank two anonymous referees for their helpful comments. The 3D figures were produced using the FMStudio with kind permission of RISC Software GmbH.

REFERENCES

[1] F. ALIZADEH, J. ECKSTEIN, N. NOYAN, AND G. RUDOLF, *Arrival rate approximation by nonnegative cubic splines*, Oper. Res., 56 (2008), pp. 140–156.

[2] M. P. BENDSØE, J. M. GUADES, R. HABER, P. PEDERSEN, AND J. E. TAYLOR, *An analytical model to predict optimal material properties in the context of optimal structural design*, J. Appl. Mech., 61 (1994), pp. 930–937.

[3] M. Bendsøe and O. Sigmund, *Topology Optimization. Theory, Methods and Applications*, Springer, Heidelberg, 2002.

[4] A. Ben-Tal, M. Kočvara, A. Nemirovski, and J. Zowe, *Free material design via semidefinite programming: The multi-load case with contact conditions*, SIAM J. Optim., 9 (1999), pp. 813–832.

[5] K.-U. Bletzinger, *Extended method of moving asymptotes based on second-order information*, Struct. Multidiscip. Optim., 5 (1993), pp. 175–183.

[6] M. Breitfeld and D. Shanno, *A Globally Convergent Penalty-barrier Algorithm for Nonlinear Programming and its Computational Performance*, Technical report, Rutcor Research Report, Rutgers University, Piscataway, NJ, 1994.

[7] P. G. Ciarlet, *The Finite Element Method for Elliptic Problems*, North–Holland, Amsterdam, New York, Oxford, 1978.

[8] C. Fleury, *CONLIN: An efficient dual optimizer based on convex approximation concepts*, Struct. Multidispl. Optim., 1 (1989), pp. 81–89.

[9] C. Fleury, *Efficient approximation concepts using second order information*, Internat. J. Numer. Methods Engrg., 28 (1989), pp. 2041–2058.

[10] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms* I, Springer, Berlin, Heidelberg, 1993.

[11] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1991.

[12] M. Kočvara, M. Stingl, and J. Zowe, *Free material optimization: Recent progress*, Optimization, 57 (2008), pp. 79–100.

[13] M. Kočvara and M. Stingl, *PENNON—a code for convex nonlinear and semidefinite programming*, Optim. Methods Softw., 18 (2003), pp. 317–333.

[14] M. Kočvara and M. Stingl, *The worst-case multiple-load fmo problem revisited*, in Proceedings of the International Union of Theoretical and Applied Mechanics (IUTAM) Symposium on Topological Design Optimization of Structures, Machines and Material: Status and Perspectives, Aalborg and Lyngby, Denmark, M.P. Bendsoe, N. Olhoff, and O. Sigmund, eds., Springer, 2006, pp. 403–411.

[15] M. Kočvara and M. Stingl, *Free material optimization: Towards the stress constraints*, Struct. Multidiscip. Optim., 33 (2007), pp. 323–335.

[16] M. Kočvara and M. Stingl, *On the solution of large-scale SDP problems by the modified barrier method using iterative solvers*, Math. Program. Ser. B, 109 (2007), pp. 413–444.

[17] J. Nečas and I. Hlaváček, *Mathematical Theory of Elastic and Elasto-Plastic Bodies: An Introduction*, Elsevier Science, Amsterdam, 1981.

[18] E. G. Ng and B. W. Peyton, *Block sparse Cholesky algorithms on advanced uniprocessor computers*, SIAM J. Sci. Comput., 14 (1993), pp. 1034–1056.

[19] J. Nocedal and S. Wright, *Numerical Optimization*, Springer Ser. Oper. Res., Springer, New York, 1999.

[20] R. Polyak, *Modified barrier functions: Theory and methods*, Math. Program., 54 (1992), pp. 177–222.

[21] U. Ringertz, *On finding the optimal distribution of material properties*, Struct. Optim. Multidispl., 5 (1993), pp. 265–267.

[22] K. Schittkowski and C. Zillober, *SQP versus SCP methods for nonlinear programming*, in Optimization and Control with Applications, Appl. Optim. 96, L. Qi, K. Teo, and XC. Yang, eds., Springer, 2005, pp. 305–330.

[23] O. Sigmund and J. Petersson, *Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima*, Struct. Optim. Multidispl., 16 (1998), pp. 68–75.

[24] M. Stingl, M. Kočvara, and G. Leugering, *A new method for the solution of multidisciplinary free material optimization problems*, Oberwolfach Rep., 13 (2008), pp. 647–648.

[25] M. Stingl, *On the Solution of Nonlinear Semidefinite Programs by Augmented Lagrangian Methods*, Ph.D. thesis, Institute of Applied Mathematics II, Friedrich-Alexander University of Erlangen-Nuremberg, Erlangen, Germany, 2006.

[26] K. Svanberg, *The method of moving asymptotes – a new method for structural optimization*, Internat. J. Numer. Methods Engrg., 24 (1987), pp. 359–373.

[27] K. Svanberg, *A class of globally convergent optimization methods based on conservative convex separable approximations*, SIAM J. Optim., 12 (2002), pp. 555–573.

[28] H. Waki, S. Kim, M. Kojima, and M. Muramatsu, *Sums of squares and semidefinite program relaxations for polynomial optimization problems with structured sparsity*, SIAM J. Optim., 17 (2006), pp. 218–242.

[29] R. Werner, *Free Material Optimization*, Ph.D. thesis, Institute of Applied Mathematics II, Friedrich-Alexander University of Erlangen-Nuremberg, Erlangen, Germany, 2000.

[30] C. Zillober, K. Schittkowski, and K. Moritzen, *Very large scale optimization by sequential convex programming*, Optim. Methods Softw., 19 (2004), pp. 103–120.

[31] C. Zillober, *Eine global konvergente Methode zur Lösung von Problemen aus der Strukturoptimierung*, Ph.D. thesis, Technische Universität München, München, Germany, 1992.

[32] C. Zillober, *Global convergence of a nonlinear programming method using convex approximations*, Numer. Algorithms, 27 (2001), pp. 256–289.

[33] J. Zowe, M. Kočvara, and M. Bendsøe, *Free material optimization via mathematical programming*, Math. Program. Ser. B, 79 (1997), pp. 445–466.

SIAM J. OPTIM.
Vol. 20, No. 1, pp. 156–171

THE SMOOTHED SPECTRAL ABSCISSA FOR ROBUST STABILITY OPTIMIZATION[*]

JORIS VANBIERVLIET[†], BART VANDEREYCKEN[†], WIM MICHIELS[‡],
STEFAN VANDEWALLE[†], AND MORITZ DIEHL[§]

Abstract. This paper concerns the stability optimization of (parameterized) matrices $A(x)$, a problem typically arising in the design of fixed-order or fixed-structured feedback controllers. It is well known that the minimization of the spectral abscissa function $\alpha(A)$ gives rise to very difficult optimization problems, since $\alpha(A)$ is not everywhere differentiable and even not everywhere Lipschitz. We therefore propose a new stability measure, namely, the *smoothed spectral abscissa* $\tilde{\alpha}_\epsilon(A)$, which is based on the inversion of a relaxed H_2-type cost function. The regularization parameter ϵ allows tuning the degree of smoothness. For ϵ approaching zero, the smoothed spectral abscissa converges towards the nonsmooth spectral abscissa from above so that $\tilde{\alpha}_\epsilon(A) \leq 0$ guarantees asymptotic stability. Evaluation of the smoothed spectral abscissa and its derivatives w.r.t. matrix parameters x can be performed at the cost of solving a primal-dual Lyapunov equation pair, allowing for an efficient integration into a derivative-based optimization framework. Two optimization problems are considered: On the one hand, the minimization of the smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A(x))$ as a function of the matrix parameters for a fixed value of ϵ, and, on the other hand, the maximization of ϵ such that the stability requirement $\tilde{\alpha}_\epsilon(A(x)) \leq 0$ is still satisfied. The latter problem can be interpreted as an H_2-norm minimization problem, and its solution additionally implies an upper bound on the corresponding H_∞-norm or a lower bound on the distance to instability. In both cases, additional equality and inequality constraints on the variables can be naturally taken into account in the optimization problem.

Key words. robust stability, Lyapunov equations, eigenvalue optimization, pseudospectra

AMS subject classifications. 93D09, 65K10, 49M20

DOI. 10.1137/070704034

1. Introduction.

Stability optimization of linear and nonlinear continuous-time dynamic systems is both a highly relevant and a difficult task. The optimization parameters often stem from a feedback controller, which can be used to optimize either a performance criterion or the asymptotic stability around a certain steady state. When robustness against perturbations of the system must be taken into account also, the resulting optimization problem becomes even more challenging.

Assuming an adequate parameterization of the desired feedback controller, the problem of finding a suitable steady state along with a stabilizing feedback controller can essentially be transformed into a nonlinear programming problem. By collecting all optimization variables in a vector x, we can summarize the described stability

[*] Received by the editors September 28, 2007; accepted for publication (in revised form) December 7, 2008; published electronically March 25, 2009. This research was supported in part by the Research Council K.U.Leuven, CoE EF/05/006 Optimization in Engineering (OPTEC) and presents results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors. Bart Vandereycken is a Research Assistant and Wim Michiels is a Postdoctoral Fellow of the Research Foundation – Flanders (FWO).

http://www.siam.org/journals/siopt/20-1/70403.html

[†] Dept. of Computer Science, K.U.Leuven, Celestijnenlaan 200A, 3001 Leuven, Belgium (joris.vanbiervliet@cs.kuleuven.be, bart.vandereycken@cs.kuleuven.be, stefan.vandewalle@cs.kuleuven.be).

[‡] Dept. of Mechanical Engineering, T. U. Eindhoven, Den Dolech 2, 5612 Eindhoven, The Netherlands and Dept. of Computer Science, K.U.Leuven, Belgium (wim.michiels@cs.kuleuven.be).

[§] Dept. of Electrical Engineering, K.U.Leuven, Kasteelpark Arenberg 10, 3001 Leuven, Belgium (moritz.diehl@esat.kuleuven.be).

optimization problem as

$$\min_{x} \; \Phi_{\mathbf{stab}}(A(x)), \quad \text{subject to (s.t.)} \quad g(x) = 0,\ h(x) \leq 0, \tag{1.1}$$

where $A(x)$ is the system matrix depending smoothly on x and the function $\Phi_{\mathbf{stab}}(\cdot)$ expresses our desire to optimize stability, under the given constraints. In the field of linear output feedback control, closed-loop system matrix $A(x)$ will typically be of the form $A + BKC$, with A the open-loop system matrix, B and C the input and output matrices, and K containing the controller parameters x to be optimized.

The most straightforward choice for the objective function $\Phi_{\mathbf{stab}}$ is related to the eigenvalues of A, namely, the spectral abscissa $\alpha(A)$. This value is defined as the real part of the rightmost eigenvalue of the spectrum $\Lambda(A) = \{z \in \mathbb{C} : \det(zI - A) = 0\}$, that is, $\alpha(A) = \sup\{\Re(z) : z \in \Lambda(A)\}$.

The spectral abscissa is, in general, a non-Lipschitz and nonconvex function of A [13, 14] and therefore typically a very hard function to optimize. Nonetheless, recent developments have led to algorithms that are able to tackle such nonsmooth objective functions [8, 12, 25, 26]. The extension to infinite-dimensional systems has been made in [29]. Still, the spectral abscissa is also known to perform quite poorly in terms of robustness against parameter uncertainties. A tiny perturbation or disturbance to a parameter of a system that was optimized in the spectral abscissa can possibly lead to instability.

For this reason, more robust approaches have been proposed. Amongst those, the most prominent are H_∞-optimization [1, 2, 7, 23, 24] and, closely related, the minimization of the pseudospectral abscissa [10, 28]. As these robust optimization formulations are connected to maximizing the distance to instability of the system under consideration, they inherently take the effect of perturbations into account in the stability measure. However, their objective functions still suffer from nonsmoothness and associated high computational costs in optimization. Throughout this paper, we will use standard notation α_ϵ for the pseudospectral abscissa, not to be confused with our symbol for the smoothed spectral abscissa, namely, $\tilde{\alpha}_\epsilon$. Another, albeit less well-known robustness measure, is the robust spectral abscissa, denoted by α_δ as in [8] and is based on Lyapunov variables.

The paper is organized as follows. In section 2, we define the smoothed spectral abscissa, and we outline its most important properties. Section 3 discusses how to efficiently compute this newly defined stability measure along with its derivatives. In section 4, we explain how the smoothed spectral abscissa can be used to formulate optimization problems dealing with robust stability, and section 5 draws a relation with the pseudospectral abscissa. Finally, we illustrate our stabilization method by treating two numerical examples in section 6.

2. The smoothed spectral abscissa. In this section, we introduce the notion of the smoothed spectral abscissa as a new stability measure that is not susceptible to nonsmoothness like the spectral abscissa and the H_∞-norm are. It can, in addition, be attributed with certain beneficial robustness properties. We will use several well-known principles from robust control for linear systems such as stability, H_2-norm, controllability, and observability. See, e.g., [30] for an introduction. At the basis of the smoothed spectral abscissa lies the following stability criterion.

LEMMA 2.1. *For any submultiplicative matrix norm* $\|\cdot\|$, *matrix* $A \in \mathbb{R}^{n\times n}$ *is Hurwitz stable if and only if integral* $\int_0^\infty \|\exp(At)\|^2\,\mathrm{d}t$ *is finite.*

Proof. Suppose $\int_0^\infty \|\exp(At)\|^2\,\mathrm{d}t$ is finite, then $\|\exp(At)\| \to 0$ for $t \to \infty$. It is well known that, for any norm $\|\cdot\|$, this is equivalent to $\alpha(A) < 0$; see, e.g., [21].

Conversely, suppose that $\alpha(A) < 0$ and let $\|\cdot\|$ be any submultiplicative norm, then there exists $0 < \gamma < \infty$ such that $\|\exp(At)\| \leq \gamma \exp(\alpha(A)t/2)\ \forall\ t \geq 0$; see, e.g., [16, Chap. 1, sect. 3]. From this, we can derive that $\int_0^\infty \|\exp(At)\|^2\,\mathrm{d}t \leq -\gamma^2/\alpha(A) < \infty$. □

Inspired by this observation, we let $f : \mathbb{R}^{n\times n} \times \mathbb{R}\cup\{\infty\} \to \mathbb{R}\cup\{\infty\}$ be the matrix function that uses Frobenius norm $\|M\|_{\mathrm{F}}^2 := \operatorname{trace}(M^{\mathrm{T}}M)$ and that takes as its arguments, next to the matrix A, also a real-valued relaxation parameter s:

$$f(A,s) := \int_0^\infty \|V\mathrm{e}^{(A-sI)t}U\|_{\mathrm{F}}^2\ \mathrm{d}t. \tag{2.1}$$

Here, matrices U and V are to be seen as respective input and output weighting matrices, with (A,U) controllable and (V,A) observable. It is easy to see that $f(A,s)$ is nothing else than the squared weighted and relaxed H_2-norm of a system, with transfer function $\mathbf{H}_s(z) = V\left(zI - (A-sI)\right)^{-1}U$, i.e.,

$$f(A,s) = \|\mathbf{H}_s\|_{\mathcal{H}_2}^2. \tag{2.2}$$

We continue with the following properties for the function $f(A,s)$.

Lemma 2.2. $\forall A \in \mathbb{R}^{n\times n} : \{f(A,s) : s > \alpha(A)\} = \mathbb{R}^+ \setminus \{0\}$.

Proof. If $s > \alpha(A)$, matrix $A - sI$ is stable, and therefore $f(A,s)$ is finite by Lemma 2.1. Additionally, $f(A,s)$ tends to infinity and to zero for $s \to \alpha(A)$ and $s \to \infty$, respectively. □

Lemma 2.3. $\forall s > \alpha(A) : \partial f(A,s)/\partial s < 0$ *and* $\partial^2 f(A,s)/\partial s^2 > 0$.

Proof. This can be verified by differentiating the integral in (2.1) with respect to s once and twice, respectively. □

These last two properties allow us to introduce the implicit function of the relation $f(A,s) = \epsilon^{-1}$ w.r.t. the relaxation argument s, as it is well defined on the whole domain, that is, for any $\epsilon > 0$ and for any matrix $A \in \mathbb{R}^{n\times n}$. We will call this function the “smoothed spectral abscissa,” analogously to the smoothed spectral radius for discrete time systems [15].

Definition 2.4. *The smoothed spectral abscissa is defined as the mapping* $\alpha : \mathbb{R}^{n\times n} \times \mathbb{R}^+ \setminus \{0\} \to \mathbb{R},\ (A,\epsilon) \mapsto \tilde{\alpha}_\epsilon(A)$ *that uniquely solves*

$$f(A,\tilde{\alpha}_\epsilon(A)) = \epsilon^{-1}. \tag{2.3}$$

Because $f(A,s)$ is analytic in both its arguments for any $s > \alpha(A)$, it follows from the implicit function theorem that $\tilde{\alpha}_\epsilon(A)$ is analytic on its whole domain $\epsilon > 0$, $A \in \mathbb{R}^{n\times n}$. Moreover, it has the following additional properties.

Theorem 2.5. $\tilde{\alpha}_\epsilon(A)$ *is an increasing function of* ϵ, *that is,* $\partial\tilde{\alpha}_\epsilon(A)/\partial\epsilon > 0$.

Proof. Differentiating (2.3) on both sides w.r.t. ϵ, we obtain

$$\frac{\mathrm{d}f(A,\tilde{\alpha}_\epsilon(A))}{\mathrm{d}\epsilon} = \frac{\partial f(A,s)}{\partial s}\frac{\partial\tilde{\alpha}_\epsilon(A)}{\partial\epsilon} = -\epsilon^{-2} < 0,$$

from which the proposition holds by Lemma 2.3. □

Theorem 2.6. $\forall\,\epsilon > 0 : \tilde{\alpha}_\epsilon(A) > \alpha(A)$ *and* $\lim_{\epsilon\to 0}\tilde{\alpha}_\epsilon(A) = \alpha(A)$.

Proof. These two properties follow from the fact that $f(A,s)$ is finite and descending for $s > \alpha(A)$ but tends to infinity as s approaches $\alpha(A)$. □

Also note that this last theorem implies that a nonpositive smoothed spectral abscissa guarantees that the underlying system is asymptotically stable. The above definition and properties are illustrated in Figure 2.1.

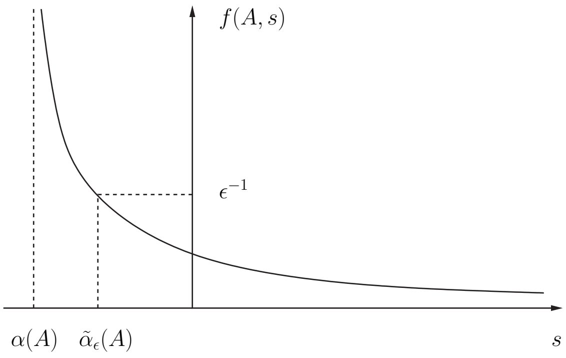


FIG. 2.1. *Typical behavior of function $f(A,s)$ as a function of s. The smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A)$ is the abscissa of the point where this function reaches ϵ^{-1}.*

3. Computing the smoothed spectral abscissa and its derivatives. Having defined the smoothed spectral abscissa, we now take a look at its computation. As explained in the previous section, this involves solving the smooth but nonlinear equation $f(A,s) = \epsilon^{-1}$ for s. Therefore, we first give some properties of function $f(A,s)$ regarding its evaluation and its derivatives.

LEMMA 3.1. *For all $s > \alpha(A)$, there exist symmetric $n \times n$ matrices P and Q such that*

$$f(A,s) = \operatorname{trace}(VPV^{\mathrm{T}}) = \operatorname{trace}(U^{\mathrm{T}}QU)\,, \tag{3.1a}$$

$$\frac{\partial f(A,s)}{\partial s} = -2\operatorname{trace}(QP) = -2\operatorname{trace}(PQ)\,, \tag{3.1b}$$

$$\frac{\partial f(A,s)}{\partial A} = 2QP, \tag{3.1c}$$

where P and Q satisfy the primal-dual Lyapunov equation pair

$$0 = L(P,A,U,s), \tag{3.2a}$$

$$0 = L(Q,A^{\mathrm{T}},V^{\mathrm{T}},s), \tag{3.2b}$$

with L defined as

$$L(P,A,U,s) \ := (A - sI)P + P(A - sI)^{\mathrm{T}} + UU^{\mathrm{T}}.$$

Proof. The first part follows immediately by writing out the Frobenius norm in (2.1):

$$f(A,s) = \operatorname{trace}\left(V \int_0^\infty \mathrm{e}^{(A-sI)t} UU^{\mathrm{T}} \mathrm{e}^{(A-sI)^{\mathrm{T}}t}\,\mathrm{d}t\ V^{\mathrm{T}}\right),$$

and, by the well-known fact that, since $A - sI$ is stable, the above integral can be identified as the trace of P, the solution of (3.2a) (see, for instance, [18, 30]). Note that solving dual Lyapunov equation (3.2b) computes a matrix Q that solves the dual integral

$$Q = \int_0^\infty \mathrm{e}^{(A-sI)^{\mathrm{T}}t}\, V^{\mathrm{T}}V\, \mathrm{e}^{(A-sI)t}\,\mathrm{d}t.$$

Since A is fixed in the partial derivative $\frac{\partial f(A,s)}{\partial s}$, we can regard f as a function of P, where P depends on s through the Lyapunov relation $L(P(s),A,U,s)$. Rather than computing this partial derivative directly as

$$\frac{\partial f(A,s)}{\partial s} = \frac{\mathrm{d}}{\mathrm{d}s}\operatorname{trace}(VP(s)V^{\mathrm{T}}) = \operatorname{trace}\left(V\frac{\mathrm{d}P}{\mathrm{d}s}V^{\mathrm{T}}\right),$$

with $\frac{\mathrm{d}P}{\mathrm{d}s}$ the solution of the Lyapunov equation $(A-sI)\frac{\mathrm{d}P}{\mathrm{d}s}+\frac{\mathrm{d}P}{\mathrm{d}s}(A-sI)^{\mathrm{T}}-2P=0$, we choose to use an adjoint differentiation technique. Vectorizing matrix P in an $n^2\times 1$ vector $p=\operatorname{vec}(P)$, we can write

$$\frac{\partial f}{\partial s} = \frac{\partial f}{\partial p}\frac{\partial p}{\partial s} = -\frac{\partial f}{\partial p}\left(\frac{\partial \ell}{\partial p}\right)^{-1}\frac{\partial \ell}{\partial s}, \tag{3.3}$$

where $\ell := \operatorname{vec}(L(P,A,U,s))$ represents the vectorized primal Lyapunov equation (3.2a). Making use of the fact that $\operatorname{vec}(MXN^T)=(N\otimes M)\operatorname{vec}(X)$, where $\otimes$ denotes the Kronecker product [17], we can make ℓ explicit in p and, as a result, arrive at the following $n^2\times n^2$ linear system:

$$\ell(p,A,U,s) = \frac{\partial \ell}{\partial p}p + \operatorname{vec}(UU^{\mathrm{T}}) = 0,$$

with $\frac{\partial \ell}{\partial p} = (A-sI)\otimes I + I\otimes(A-sI)$. For the dual Lyapunov equation, we similarly obtain

$$\ell(q,A^{\mathrm{T}},V^{\mathrm{T}},s) = \frac{\partial \ell}{\partial q}q + \operatorname{vec}(V^{\mathrm{T}}V) = 0,$$

with $\frac{\partial \ell}{\partial q} = (A-sI)^{\mathrm{T}}\otimes I + I\otimes(A-sI)^{\mathrm{T}}$. It is easily verified that $\frac{\partial \ell}{\partial q} = \frac{\partial \ell}{\partial p}^{\mathrm{T}}$. Replacing $\frac{\partial \ell}{\partial q}$ in the relation $\ell(q,A^{\mathrm{T}},V^{\mathrm{T}},s)=0$ and using, in addition, the fact that $\operatorname{vec}(V^{\mathrm{T}}V)$ equals $\frac{\partial f}{\partial p}^{\mathrm{T}}$, we find that

$$\frac{\partial \ell}{\partial p}^{\mathrm{T}} q + \frac{\partial f}{\partial p}^{\mathrm{T}} = 0 \quad\Leftrightarrow\quad q^{\mathrm{T}} = -\frac{\partial f}{\partial p}\left(\frac{\partial \ell}{\partial p}\right)^{-1}.$$

Combining this with (3.3), along with $\frac{\partial \ell}{\partial s} = -2p$, finally gives

$$\frac{\partial f}{\partial s} = q^{\mathrm{T}}(-2p) = -2\operatorname{vec}(Q)^{\mathrm{T}}\operatorname{vec}(P) = -2\operatorname{trace}(QP).$$

For the third part of the proof, i.e., the proof of the expression for the derivative w.r.t. A, we can use the same adjoint differentiation technique. Here, we again let f depend on vectorized matrix $p=\operatorname{vec}(P)$, which now depends on $a=\operatorname{vec}(A)$ according to relation $\ell(p(a),a,s)=0$. Using the previous results, we obtain the following expression for $\frac{\partial f}{\partial a} := \operatorname{vec}^{\mathrm{T}}\left(\frac{\partial f}{\partial A}\right)$:

$$\frac{\partial f}{\partial a} = \frac{\partial f}{\partial p}\frac{\partial p}{\partial a} = -\frac{\partial f}{\partial p}\left(\frac{\partial \ell}{\partial p}\right)^{-1}\frac{\partial \ell}{\partial a} = q^{\mathrm{T}}\frac{\partial \ell}{\partial a}. \tag{3.4}$$

To find $\frac{\partial \ell}{\partial a}$, we first have to make ℓ explicit in a, which yields

$$\ell(P,a,U,s) = \frac{\partial \ell}{\partial a}a + \operatorname{vec}(UU^{\mathrm{T}}) = 0, \quad \text{with} \quad \frac{\partial \ell}{\partial a} = (P\otimes I) + (I\otimes P)\,\Pi, \tag{3.5}$$

where Π denotes the symmetric permutation matrix that satisfies $\mathrm{vec}(A^{\mathrm{T}}) = \Pi\,\mathrm{vec}(A)$, e.g., $\Pi = S_{n,n}$ in [20]. Substituting in (3.4) gives

$$\mathrm{vec}^{\mathrm{T}}\left(\frac{\partial f}{\partial A}\right) = \left(\frac{\partial \ell}{\partial a}^{\mathrm{T}} q\right)^{\mathrm{T}} = \left[\mathrm{vec}(QP) + \Pi^{\mathrm{T}}\,\mathrm{vec}(PQ)\right]^{\mathrm{T}} = 2\,\mathrm{vec}^{\mathrm{T}}\left(QP\right) = \mathrm{vec}^{\mathrm{T}}\left(2QP\right).$$

By comparison of both sides, we finally obtain that

$$\frac{\partial f}{\partial A} = 2QP,$$

which concludes the proof. □

The relatively cheap computation of $f(A, s)$ and its derivative w.r.t. s enables us to efficiently solve the nonlinear equation $f(A, s) = \epsilon^{-1}$ by the use of standard root-finding methods and thus evaluate the smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A)$. Specifically, we can use a Dekker–Brent-type method [6], provided that we establish a root bracketing interval first, or Newton's method if we want to exploit the availability of the derivatives. For further elaboration on the computational issues involving the smoothed spectral abscissa, see section 6.3.

As we will want to use derivative-based optimization methods later on to exploit the smoothness of the smoothed spectral abscissa, we need to be able to compute also the derivative of $\tilde{\alpha}_\epsilon(A)$ w.r.t. A. Fortunately, this can be done at almost no extra cost. Indeed, the same ingredients that were needed for the evaluation of $\tilde{\alpha}_\epsilon(A)$, namely, the solutions P and Q of one primal-dual Lyapunov equation pair, give us direct access to the derivative of $\tilde{\alpha}_\epsilon(A)$ w.r.t. A, as expressed in the following theorem.

THEOREM 3.2. *For fixed ϵ, the derivative of the smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A)$ w.r.t. A equals*

$$\frac{\partial \tilde{\alpha}_\epsilon(A)}{\partial A} = \frac{QP}{\mathrm{trace}\,(QP)},$$

where P and Q satisfy the Lyapunov equation pair (3.2a)–(3.2b) *for $s = \tilde{\alpha}_\epsilon(A)$.*

Proof. Differentiating the implicit equation $f(A, s) = \epsilon^{-1}$ w.r.t. A and using the chain rule, we obtain

$$\frac{\partial \tilde{\alpha}_\epsilon(A)}{\partial A} = -\left(\frac{\partial f(A, s)}{\partial s}\right)^{-1}\left(\frac{\partial f(A, s)}{\partial A}\right).$$

Recalling (3.1b) and (3.1c) of Lemma 3.1, the result follows directly. □

Remark 1. Suppose A depends on an $m \times 1$ parameter vector x, then a direct approach to compute the derivatives w.r.t. to these parameters would require solving $m + 1$ Lyapunov equations with different right-hand sides, instead of $m + 1$ matrix multiplications of $\partial A/\partial x$ with $\partial \tilde{\alpha}_\epsilon/\partial A$.

4. Robust stability optimization. When it comes to algorithmic optimization, a first major advantage of the smoothed spectral abscissa criterion is that it is differentiable everywhere and that its derivatives can be computed efficiently. This allows us to use derivative-based methods without any restriction. Additionally, due to its differentiable dependence on A and its connection with the H_2-norm, it is expected to be a more robust measure for stability than the spectral abscissa. We will present two smooth formulations of stability optimization problem (1.1): one that focuses on mere stabilization and one that will turn out to perform an H_2-norm minimization.

The first variant is to simply choose a fixed $\epsilon > 0$ and then solve

$$\min_{x} \ \tilde{\alpha}_\epsilon(A(x)) \quad \text{s.t.} \quad g(x) = 0,\ h(x) \leq 0. \tag{4.1}$$

Here, $\tilde{\alpha}_\epsilon(A(x))$ is indirectly dependent on matrix parameter vector x, as it is implicitly defined as the solution of the relation $f(A(x), s) = \epsilon^{-1}$ w.r.t. s. By decoupling this implicit relation into a constraint, we can formulate the problem alternatively as

$$\min_{x} \ s, \quad \text{s.t.} \quad f(A(x), s) = \epsilon^{-1},\ g(x) = 0,\ h(x) \leq 0, \tag{4.2}$$

which is more amenable for an SQP optimization framework.

Should problem (4.1) or (4.2) not result in a negative optimal value for the chosen ϵ, then one can try again with a smaller ϵ. Note also that, if the sole goal is to achieve a stable system, one may terminate the optimization procedure once the smoothed spectral abscissa becomes smaller than zero.

In the minimization formulation of the smoothed spectral abscissa with fixed ϵ, the choice of ϵ is somewhat arbitrary. As indicated by Theorem 2.6, $\tilde{\alpha}_\epsilon(A)$ becomes smoother—and thus presumably a more robust measure for stability—with increasing values for $\epsilon > 0$. Thus, we might alternatively search for the largest ϵ so that the stability certificate $\tilde{\alpha}_\epsilon(A) \leq 0$ still holds. This leads to a second optimization problem:

$$\max_{x,\epsilon} \ \epsilon \quad \text{s.t.} \quad \tilde{\alpha}_\epsilon(A(x)) \leq 0 \quad \text{and} \quad g(x) = 0,\ h(x) \leq 0. \tag{4.3}$$

Since $\tilde{\alpha}_\epsilon(A)$ is a continuously growing function of ϵ, the constraint in problem (4.3) will always be active at its optimizer (x^*, ϵ^*). Hence, it is easily seen that the solution of the first problem (4.1), with ϵ fixed to ϵ^*, will be exactly zero and that, in addition, its minimizer will be the same as the one for problem (4.3), namely, x^*. Succinctly,

$$x^* = \arg\min_{x} \tilde{\alpha}_{\epsilon^*}(A(x)) \qquad \text{and} \qquad \tilde{\alpha}_{\epsilon^*}(A(x^*)) = 0.$$

Problem (4.3) can thus be solved by finding the ϵ for which the resulting minimal smoothed spectral abscissa is zero, which can be implemented by bisecting with respect to ϵ. The activity of the stability constraint also leads to the following nice interpretation of problem (4.3).

THEOREM 4.1. *Any solution x^* that solves problem* (4.3) *also solves the H_2-norm optimization of a system with transfer function $\mathbf{H}(x)(z) := V(zI - A(x))^{-1}U$, i.e.,*

$$x^* = \arg \ \min_{x} \|\mathbf{H}(x)\|_{\mathcal{H}_2}, \quad \textit{s.t.} \quad g(x) = 0,\ h(x) \leq 0,$$

and the solution $\|\mathbf{H}(x^*)\|_{\mathcal{H}_2}$ is equal to $\sqrt{1/\epsilon^*}$.

Proof. Taking the inverse of the objective function in problem (4.3) and incorporating the fact that the stability constraint will be active, this problem can be rewritten as the minimization of the function $f(A(x), 0)$, subject to the constraints g and h, and additionally restricting x to values for which $A(x)$ is stable. This is, by (2.2), equivalent to minimizing the squared H_2-norm of the system with transfer function $\mathbf{H}$. □

Remark 2. Solving problem (4.3) with the restriction $\tilde{\alpha}_\epsilon < s$ (with $s < 0$) would minimize the H_2-norm of a system with the shifted transfer function $\mathbf{H}_s$.

5. Relation with the pseudospectral abscissa. We will now draw a relationship between the smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A)$ and the pseudospectral abscissa $\alpha_\epsilon(A)$, the latter being defined as

$$\alpha_\epsilon(A) := \sup\{\Re(z) : z \in \Lambda_\epsilon(A)\}, \quad \text{where} \quad \Lambda_\epsilon(A) = \{\Lambda(X) : \|X - A\|_2 \leq \epsilon\}.$$

For this section, we take the restriction $U = V = I$, so the transfer function becomes $\mathbf{H}(z) = (zI - A)^{-1}$. Define the H_∞-norm as

$$\|\mathbf{H}\|_{\mathcal{H}_\infty} = \sup_{\Re(z)=0} \|\mathbf{H}(z)\|_2.$$

We then have the following well-known equivalency involving $\alpha_\epsilon(A)$ and the corresponding H_∞-norm [10]:

$$\alpha_\epsilon(A) < 0 \quad \Leftrightarrow \quad \|\mathbf{H}\|_{\mathcal{H}_\infty} < \epsilon^{-1}. \tag{5.1}$$

We can also interpret this in terms of the H_∞-norm of a shifted matrix $A - sI$. The relation then becomes

$$\alpha_\epsilon(A - sI) < 0 \quad \Leftrightarrow \quad \alpha_\epsilon(A) < s \quad \Leftrightarrow \quad \|\mathbf{H}_s\|_{\mathcal{H}_\infty} < \epsilon^{-1}, \tag{5.2}$$

where $\mathbf{H}_s(z) := (zI - (A - sI))^{-1} = ((z + s)I - A)^{-1}$. In other words, the pseudospectral abscissa is the minimal shift-to-the-left s for which the "shifted" H_∞-norm is smaller than ϵ^{-1}. Similarly as in Remark 2, if follows from (5.2) that the minimization of α_ϵ amounts to minimizing the H_∞-norm of the shifted system $\mathbf{H}_s$, where $s = \min \alpha_\epsilon$.

Going back to the definition of the smoothed spectral abscissa $\tilde{\alpha}_\epsilon(A)$ and taking into account that f is a decreasing function of s (Lemma 2.3), we derive a similar relation as we did in (5.2):

$$\tilde{\alpha}_\epsilon(A) < s \quad \Leftrightarrow \quad f(A, s) = \|\mathbf{H}_s\|^2_{\mathcal{H}_2} < \epsilon^{-1}. \tag{5.3}$$

Analogously, we can regard the smoothed spectral abscissa as the minimal shift s for which $\|\mathbf{H}_s\|^2_{\mathcal{H}_2}$ lies below the bound ϵ^{-1} (see also Figure 2.1). Thus, $\alpha_\epsilon(A)$ and $\tilde{\alpha}_\epsilon(A)$ are both relaxations of the spectral abscissa in the sense that they are both induced by placing a bound on a norm (H_∞ and H_2, respectively) that goes to infinity when approaching instability. This analogy enables us to relate these two robust stability measures.

THEOREM 5.1 (relation to pseudospectral abscissa). *For $s > \alpha(A)$ and for $U = V = I$, the following holds:*

$$\|\mathbf{H}_s\|_{\mathcal{H}_\infty} < 2\|\mathbf{H}_s\|^2_{\mathcal{H}_2} \tag{5.4a}$$

$$\alpha_{\epsilon/2}(A) < \tilde{\alpha}_\epsilon(A). \tag{5.4b}$$

Proof. The first inequality is based on [3], where $2\lambda_{\max}(Q^2)^{\frac{1}{2}}$ is established to be an upper bound on the H_∞-norm of an unweighted system with transfer function $\mathbf{H}_s$, where Q satisfies (3.2b). Since Q is a positive definite matrix, we can deduce from this the following:

$$\|\mathbf{H}_s\|_{\mathcal{H}_\infty} \leq 2\lambda_{\max}(Q) < 2\,\mathrm{trace}\,(Q)\,.$$

This proves (5.4a) directly by Lemma 3.1(a) and by (2.2). Suppose then, by (5.3), that, for $s = \tilde{\alpha}_\epsilon(A)$, it is true that $\|\mathbf{H}_s\|^2_{\mathcal{H}_2} = \epsilon^{-1}$. Using (5.4a) in connection with (5.2), assertion (5.4b) follows. ▯

This property has an important implication in terms of robust optimization. It shows that the squared H_2-norm constitutes an upper bound on the H_∞-norm, which is directly related to the distance to instability of a system. By minimizing the first norm, one could expect that the second norm should also go down.

On top of this rather intuitive result, (5.4b) together with (5.1) provides us with a guarantee w.r.t. the H_∞-norm once the smoothed spectral abscissa is negative. Indeed, if we have that $\tilde{\alpha}_\epsilon(A(x)) < 0$ for some x, we are not only sure that the system with system matrix $A(x)$ will be a stable one, but also that this system will have an H_∞-norm that is smaller than $2/\epsilon$. In other words, we can be certain that the distance to instability of the system will be at least $\epsilon/2$.

6. Numerical examples. We will now put theory into practice by treating two control examples using the smoothed spectral abscissa as the stability criterion. First, we will illustrate the theory behind the smoothed spectral abscissa by use of an academic example. Next, we will treat a more realistic example, namely, a turbo generator model. We conclude by making a comparison of the computational cost of the smoothed spectral abscissa in relation with other robust stability measures. All computations were done with MATLAB R2008a.

6.1. A simple state feedback controlled system. Consider the following two-parameter linear state feedback controlled system, with a closed-loop system matrix $A + BK$, and where

$$A = \begin{bmatrix} 0.1 & -0.03 & 0.2 \\ 0.2 & 0.05 & 0.01 \\ -0.06 & 0.2 & 0.07 \end{bmatrix}, \quad B = \frac{1}{2}\begin{bmatrix} -1 \\ -2 \\ 1 \end{bmatrix}, \quad K^{\mathrm{T}} = \begin{bmatrix} x_1 \\ x_2 \\ 1.4 \end{bmatrix}. \tag{6.1}$$

Figure 6.1 shows, as a function of the control parameters, the spectral abscissa ($\epsilon = 0$) in comparison with the smoothed spectral abscissae for three different smoothing levels ($\epsilon = 4, 8, 12 \cdot 10^{-3}$). For $\epsilon = 4 \cdot 10^{-3}$, the corresponding pseudospectral abscissa, i.e., with an ϵ half as large, is also plotted. In the left frame, $x_2 = 1.25$ is held fixed. In the right frame, both x_1 and x_2 are free, and the boundaries of the stability regions, that is, the regions where the respective measures are negative, are drawn. On both figures, we clearly observe the smooth behavior of $\tilde{\alpha}_\epsilon$ in contrast with the nonsmoothness of the spectral and pseudospectral abscissa.

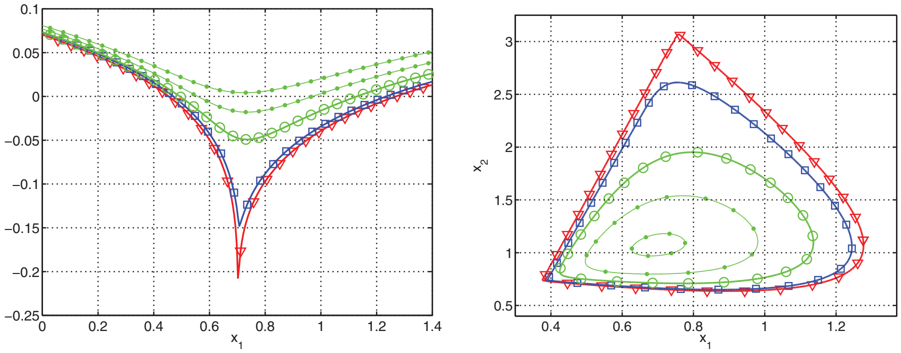


FIG. 6.1. *Evolution with respect to x_1 (left) and stability regions (right) of the spectral abscissa α (with ▽), pseudospectral abscissa $\alpha_{\epsilon/2}$ (with □), and smoothed spectral abscissa $\tilde{\alpha}_\epsilon$ (with ○) of the example in section 6.1 with smoothing parameter $\epsilon = 4 \cdot 10^{-3}$. In addition (with •), two smoothed spectral abscissae for $\epsilon = 8 \cdot 10^{-3}$ and $\epsilon = 12 \cdot 10^{-3}$.*

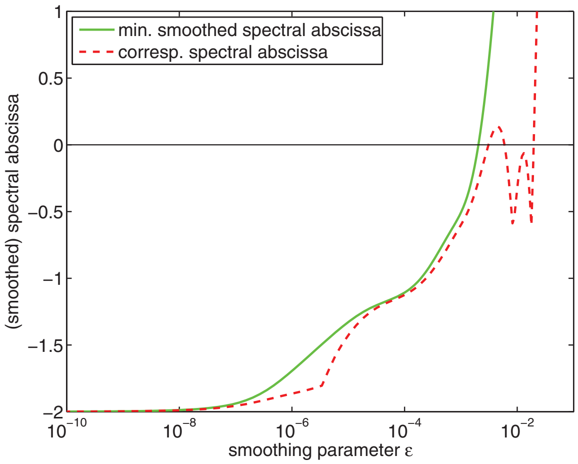


FIG. 6.2. *The minimal smoothed spectral abscissa $\tilde{\alpha}_\epsilon$ of the turbo generator model, for ϵ ranging from 10^{-10} to 10^{-1}, and the corresponding spectral abscissa α evaluated in each of the minimizers.*

The ordering of $\tilde{\alpha}_\epsilon$, $\alpha_{\epsilon/2}$, and α as stated by Theorems 2.6 and 5.1 is also confirmed. On the left, the curve of the smoothed spectral abscissa is everywhere above the other two curves and on the right, the $\tilde{\alpha}_\epsilon$-stability region is strictly contained within the stability regions of the pseudospectral and, consequently, also of the spectral abscissa.

6.2. Turbo generator model. Next, we treat problem "TG1" of Leibfritz's control problem database [19], which models a nuclear powered turbo generator by a linear system of dimension 10 with four control parameters. This system has already been used as an example in [9] for robust stability optimization using the pseudospectral abscissa in combination with the gradient sampling algorithm.

Figure 6.2 shows the behavior of the solutions to a minimization of the smoothed spectral abscissa $\tilde{\alpha}_\epsilon$ for a dense set of smoothing parameters ϵ between 10^{-1} and 10^{-10}, i.e., ranging from relatively large to very small. It is immediately verified that the minima of the smoothed spectral abscissa decrease monotonically for ϵ becoming smaller. Next to the minima, we also plotted the evolution of the corresponding spectral abscissae evaluated at each of these minimizers. We can see that, although α is always strictly smaller than $\tilde{\alpha}_\epsilon$, it is not guaranteed to decrease monotonically, which is, for example, the case for large ϵ. For $\epsilon \to 0$, however, the gap between the two becomes tighter and tighter. Of course, as the smoothed spectral abscissa converges to the spectral abscissa when ϵ approaches zero, the $\tilde{\alpha}_\epsilon$-minimization problem becomes more nonsmooth and thus harder.

To analyze this, Table 6.1 shows the results of a standard BFGS minimization of $\tilde{\alpha}_\epsilon$ for 11 selected values of ϵ and with random starting parameters x. Next to the resulting minima for each ϵ, the number of iterations (averaged out over ten random starting points) needed to solve the respective optimization problems is listed. For small ϵ and consequently, poor smoothing, this number becomes huge. However, having the smoothing parameter at hand to tune the level of smoothing, the amount of required iterations can be drastically decreased by following a homotopy strategy, namely, iteratively decreasing ϵ and each time using the minimizer of the previous problem as the starting point. This is confirmed in Table 6.1, where the number of iterations required for this homotopy strategy and the resulting minima are listed next to the ones for which random starting points were used.

Table 6.1

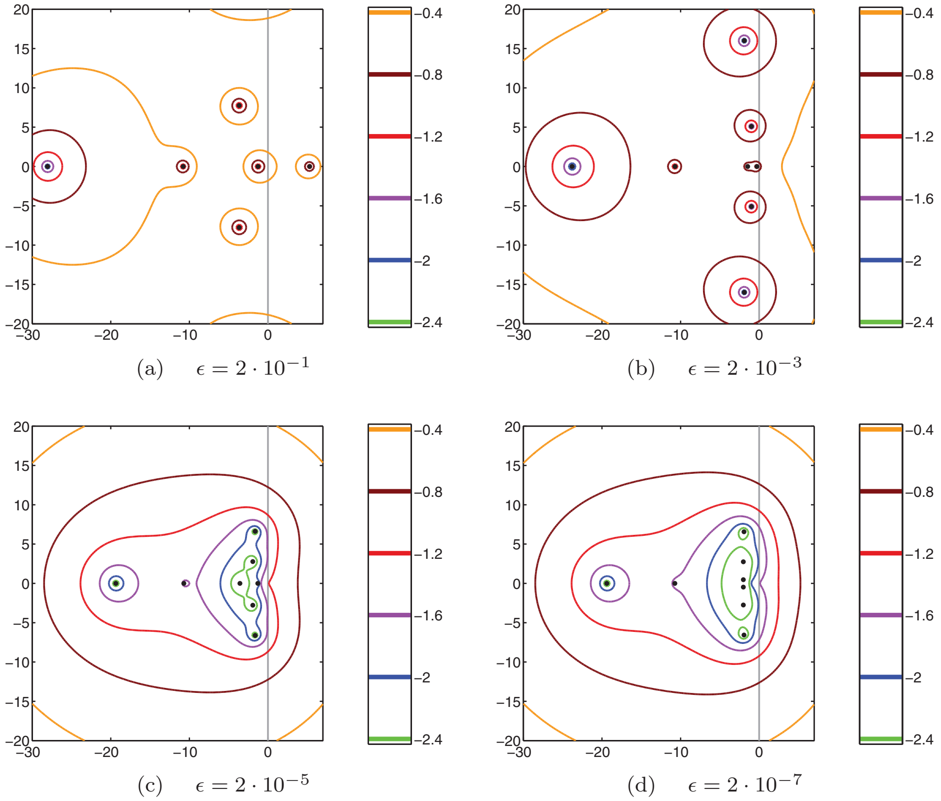
Solutions to the minimization of the smoothed spectral abscissa of the turbo generator model for 11 *designated* ϵ*-values (without homotopy/with homotopy) and the corresponding pseudospectral and spectral abscissae.*

$\log_{10}\epsilon$	$\min_x \tilde{\alpha}_\epsilon(x)$	Its.	$\alpha_{\epsilon/2}(x^*)$	$\alpha(x^*)$
0	86.920558/ 86.920558	35/ 32	4.170208	9.899472
-1	23.951317/ 23.951317	37/ 33	5.267947	5.447648
-2	3.925292/ 3.925292	29/ 29	-0.327800	-0.272926
-3	$-0.508181/-0.508181$	34/ 62	-0.600857	-0.598826
-4	$-1.107119/-1.107119$	69/ 54	-1.125056	-1.124731
-5	$-1.328287/-1.328287$	104/ 80	-1.427324	-1.426787
-6	$-1.694445/-1.694445$	102/ 65	-1.864124	-1.864049
-7	$-1.938475/-1.938475$	252/ 57	-1.955631	-1.955624
-8	$-1.987303/-1.987303$	292/125	-1.988801	-1.988801
-9	$-1.996336/-1.996246$	1522/ 51	-1.996542	-1.996542
-10	$-1.998646/-1.998587$	1827/ 47	-1.998680	-1.998680

It is known that minimization of the pseudospectral abscissa produces a balance between the asymptotic and the initial decay rate for different ϵ. In particular, minimizing α_ϵ amounts to the minimal spectral abscissa for $\epsilon \to 0$, and α_ϵ minimizes the H_∞-norm if ϵ is such that $\min_x \alpha_\epsilon = 0$; see [10]. In our case, we obtain a similar trade-off by minimizing the smoothed spectral abscissa. For ϵ going to zero, we also converge to the minimal spectral abscissa, and, for a particular value of ϵ, the H_2-norm is minimized. By the relation $\alpha_{\epsilon/2} < \tilde{\alpha}_\epsilon$, it is reasonable to expect that the pseudospectral abscissa, evaluated in the minimizers of $\min_x \tilde{\alpha}_\epsilon(x)$, will also be pushed down when $\tilde{\alpha}_\epsilon$ is minimized for increasingly smaller ϵ. This is confirmed by the fourth column in Table 6.1.

To study the behavior of the eigenvalues and corresponding pseudospectra in the minimizers of the smoothed spectral abscissa, Figures 6.3(a)–(d) depict the pseudospectra at four $\tilde{\alpha}_\epsilon$-minimizers, namely, for $\epsilon = 2 \cdot 10^{-1,-3,-5,-7}$. For the first value of ϵ, both the smoothed and spectral abscissa are positive and the minimizer is not stabilizing, as seen in the spectrum plotted in Figure 6.3(a). For $\epsilon = 2 \cdot 10^{-3}$, the minimal smoothed spectral abscissa equals $-0.0270\ldots$, which guarantees a stable system. As seen in Figure 6.3(b), the eigenvalues are indeed all in the left half complex plane. Since the minimum is very close to zero, we can expect $2 \cdot 10^{-3}$ to be close to the maximal ϵ for which a stabilizing solution can be found. Solving optimization problem (4.3) yields an optimal value for ϵ of $2.048 \cdot 10^{-3}$, which is indeed only slightly higher. Note that this corresponds to a minimal H_2-norm of approximately 22. A further decrease of ϵ results in smaller and smaller minimal smoothed spectral abscissae. As observed in the two bottom frames of Figure 6.3, the rightmost eigenvalues of the optimal spectra become more and more aligned on a vertical line, indicating convergence to the typical spectrum configuration for a minimized spectral abscissa. Figures 6.3(b)–(c)–(d) thus represent three instances out of the range of stabilizing solutions that compromise between a minimal spectral abscissa on the one hand and a minimal H_2-norm on the other hand.

From relation (5.2), we can deduce that the H_∞-norm equals γ^{-1} for which γ corresponds to pseudospectrum Λ_γ that is exactly contained in the left half complex plane. If we have a closer look at the last three frames of Figure 6.3, we see that this is the case for the pseudospectra with $\gamma = 10^{-1.2}$, $10^{-1.6}$, and 10^{-2}, indicating that the H_∞-norm grows as ϵ is decreased. So, although the H_∞-norm was not minimized here, the set of smoothed spectral abscissa minimizers appears to result in the same



(a) $\epsilon = 2 \cdot 10^{-1}$

(b) $\epsilon = 2 \cdot 10^{-3}$

(c) $\epsilon = 2 \cdot 10^{-5}$

(d) $\epsilon = 2 \cdot 10^{-7}$

FIG. 6.3. *Boundaries of the pseudospectra Λ_γ of the turbo generator model for values of γ as indicated in the right-hand side color bar. The frames correspond to four sets of controller parameters that were obtained by minimizing $\tilde{\alpha}_\epsilon$ with indicated ϵ.*

qualitative H_∞ behavior as would be the case for a range of pseudospectral abscissa minimizations; see [9].

Let us now investigate the H_2- and H_∞-norms of the two sets of stabilizing minimizers, one belonging to the smoothed spectral abscissa and the other to the pseudospectral abscissa. We denote them as functions $\chi_1^*(\epsilon)$ and $\chi_2^*(\epsilon)$, depending on the ϵ used in the respective minimizations. In order to be able to compare these two functions, we introduce $\epsilon_1(s)$ and $\epsilon_2(s)$ as the epsilons that yield s as minimum, i.e., such that

$$\min_x \tilde{\alpha}_{\epsilon_1(s)}(A(x)) = s,$$
$$\min_x \alpha_{\epsilon_2(s)}(A(x)) = s.$$

In this way, we obtain two new functions $x_1^*(s)$ and $x_2^*(s)$ as the respective minimizers of the smoothed and pseudospectral abscissa, with smoothing epsilons $\epsilon_1(s)$ and $\epsilon_2(s)$ and thus with minima $s \le 0$. Concisely put,

$$x_1^*(s) := \chi_1^*(\epsilon_1(s)) = \arg\min_x \tilde{\alpha}_{\epsilon_1(s)}(A(x)),$$
$$x_2^*(s) := \chi_2^*(\epsilon_2(s)) = \arg\min_x \alpha_{\epsilon_2(s)}(A(x)).$$

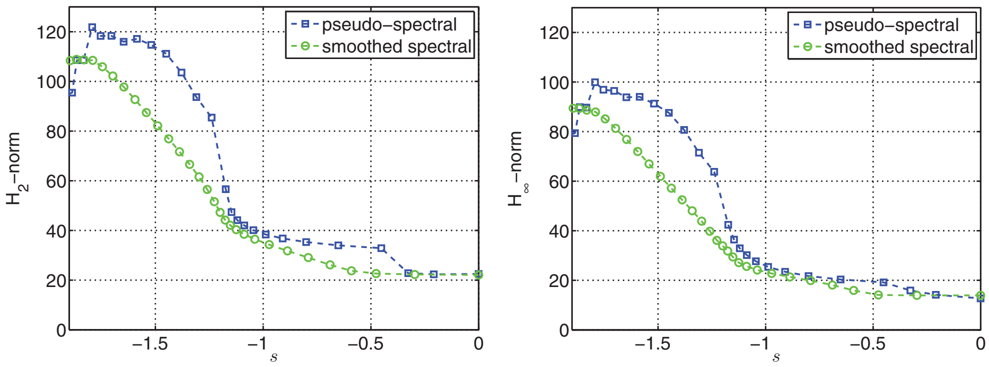


FIG. 6.4. *The H_2-norm (left) and the H_∞-norm (right) of the unshifted systems for minimizers $x_1^*(s)$ obtained by the minimization of the smoothed spectral abscissa (with ○) and for minimizers $x_2^*(s)$ obtained by the minimization of the pseudospectral abscissa (with □).*

According to Remark 2 following Theorem 4.1 and (5.2), $x_1^*(s)$ and $x_2^*(s)$, respectively, minimize the H_2-norm and H_∞-norm of a shifted system with transfer function $\mathbf{H}_s$. This justifies comparing x_1^* and x_2^* for the same s.

Because we are, in the end, interested only in the properties of the unshifted systems, we show in Figure 6.4, as a function of s, norms $\|zI - A(x_1^*(s))\|_{\mathcal{H}_2}$ and $\|zI - A(x_2^*(s))\|_{\mathcal{H}_2}$. In other words, we compare the H_2-norms of the *unshifted* transfer function, evaluated at the smoothed spectral abscissa minimizers $x_1^*(s)$ on the one hand and at the pseudospectral abscissa minimizers $x_2^*(s)$ on the other hand. In the left frame of Figure 6.4, we see that the H_2-norms of the smoothed spectral abscissa minimizers are everywhere smaller than those of the pseudospectral abscissa minimizers, except for s very close to the minimal spectral abscissa. For s close to zero, the difference between the two H_2-norms becomes very small and, for $s = 0$, the H_2-norm of the smoothed spectral abscissa minimizer is only just below the one of the pseudospectral abscissa minimizer. This implies that the optimal H_∞-minimizer, being the pseudospectral abscissa minimizer $x_2^*(0)$, is accompanied by an H_2-norm that is only slightly worse compared to the optimal H_2-norm.

We now make the same comparison for the H_∞-norm. In the right frame of Figure 6.4, we have plotted $\|zI - A(x_i^*(s))\|_{\mathcal{H}_\infty}$ for $i = 1, 2$. Again, the difference between the H_∞-norm evaluated at the pseudospectral abscissa minimizers and smoothed spectral abscissa minimizers is small for s between -1 and 0. For $s = 0$, the optimal H_∞-norm evaluated at x_2^* is naturally smaller than the H_∞-norm at x_1^*. Surprisingly though, for almost all of the other shifts, the H_∞-norms of the smoothed spectral abscissa minimizers are better than the H_∞-norms of the pseudospectral abscissa minimizers.

6.3. Computational cost. Finally, we compare the computational cost of the smoothed spectral abscissa $\tilde{\alpha}_\epsilon$ with two other robust stability measures: the pseudospectral abscissa α_ϵ and the robust spectral abscissa α_δ. Each of these measures can be used as $\Phi_{\mathbf{stab}}$ in (1.1) and as such, will be evaluated several times in the inner iterations of an optimization algorithm. Thus, the efficiency by which $\Phi_{\mathbf{stab}}$ can be evaluated has a direct influence on the overall efficiency of the optimization method for solving (1.1).

The details of the numerical methods used to compute each measure are listed in Table 6.2. From these, the criss-cross algorithm with a structure-preserving Hamil-

Table 6.2
Algorithms to compute the three stability measures.

$\Phi_{\mathbf{stab}}$	Algorithm	Convergence	Inner solve (software)
α_ϵ	criss-cross [11]	quadratic	Hamiltonian (Hapack based on [5])
α_δ	bisection [8]	linear	SDP (YALMIP [22], SeDuMi [27])
$\tilde{\alpha}_\epsilon$	Dekker–Brent	superlinear	Lyapunov (Bartels–Stewart [4])

Table 6.3
Timings and inner iterations of the three stability measures. The timings were done with 100 *samples, and the inner iterations are given as the number of inner solves in Table* 6.2*.*

Ex.	$\Phi_{\mathbf{stab}}$	ϵ, δ	Min		Mean		Max	
		(`logspace`)	(sec.)	(its.)	(sec.)	(its.)	(sec.)	(its.)
1	α_ϵ	`(-15,0,20)`	1.20e−03	(3)	1.79e−03	(4)	2.71e−03	(8)
	α_δ	`(-2,0,20)`	5.58e+00	(32)	6.18e+00	(32)	7.71e+00	(32)
	$\tilde{\alpha}_\epsilon$	`(-15,0,20)`	4.23e−03	(7)	7.92e−03	(16)	1.57e−02	(30)
2	α_ϵ	`(-12,0,20)`	2.16e−03	(3)	3.09e−03	(5)	5.49e−03	(8)
	α_δ	`(-03,0,20)`	2.99e+01	(54)	1.79e+02	(149)	1.63e+03	($\geq$999)
	$\tilde{\alpha}_\epsilon$	`(-15,0,20)`	6.31e−03	(10)	1.25e−02	(22)	2.20e−02	(39)

tonian eigenvalue solver is to be preferred. To the best of our knowledge, the listed bisection algorithm is the only known implementation for computing α_δ. Specifically, we bisect until an absolute tolerance of $10\epsilon_{\text{mach}}$ is satisfied and in each bisection step, we check the feasibility of an SDP with SeDuMi 1.1R3.

Regarding the smoothed spectral abscissa, we use Dekker–Brent, implemented by MATLAB's `fzero` with an absolute tolerance ϵ_{mach}, to find the unique root of the function $g(s) := 1/f(A,s) - \epsilon$. The reason for using the reciprocal instead of $f(A,s) - 1/\epsilon$ is that the former is better behaved numerically. Most of the time, we observed superlinear convergence. Recall that evaluating $f(A,s)$ involves solving a Lyapunov equation, which is done by the Bartels–Stewart algorithm, implemented by `lyap` in MATLAB.

We remark that our implementation for computing $\tilde{\alpha}_\epsilon$ is very preliminary, but it seems to work well for the model problems we tried. Besides some heuristics on setting up a bracketing interval, the procedure is quite robust. As far as efficiency goes, there is a lot of room for improvement. An obvious improvement is the inner loop of `fsolve` where $f(A,s)$ is evaluated for fixed A but different shifts s. Since we solve the Lyapunov equations independently for each shift, we do not make use of the fact that we can reuse the computed Schur factorizations in Bartels–Stewart. In exact arithmetic, only one factorization would suffice. Furthermore, using Dekker–Brent to solve $g(s) = 0$ has the benefit of robustness, but we sometimes need a lot of work to find a bracketing interval. Since $f(A,s)$ is smooth and convex, a safeguarded method based on Newton may be more efficient. However, it is beyond the scope of the current article to implement this.

In Table 6.3 we have summarized timings for the systems that we have examined earlier with control parameters x set to zero. However, since these three measures are quite different, comparing them is somewhat arbitrary. In order to have an impression of the computational cost, we computed each measure for a sensible range of its regularization parameter ϵ or δ. It is clear from the table that the pseudospectral and smoothed spectral abscissa are comparable in computational cost and that the robust spectral abscissa is orders of magnitudes slower.

7. Conclusions. A smooth relaxation of the nonsmooth spectral abscissa function was introduced as an alternative stability measure, with the advantage that derivative-based optimization techniques can readily be used for its optimization. Formulae for the efficient computation and derivative evaluation of the smoothed spectral abscissa were deduced based on the solution of a primal-dual Lyapunov equation pair.

Besides its direct minimization, which can be used to find stabilizing controllers, a second optimization formulation was shown to be applicable to solve fixed-order H_2-optimization problems. Moreover, a guaranteed bound on the distance to instability was established by relating the results to the H_∞-norm. The robust stabilization by use of these two optimization problems involving the smoothed spectral abscissa was illustrated with numerical examples, and also a comparative study of the computational complexity cost was made.

REFERENCES

[1] P. Apkarian and D. Noll, *Nonsmooth H-infinity synthesis*, IEEE Trans. Automat. Control, 51 (2006), pp. 71–86.

[2] P. Apkarian and D. Noll, *Nonsmooth optimization for multidisk H-infinity synthesis*, Eur. J. Control, 12 (2006), pp. 229–244.

[3] V. Balakrishnan and L. Vandenberghe, *Semidefinite programming duality and linear time-invariant systems*, IEEE Trans. Automat. Control, AC-48 (2003), pp. 30–41.

[4] R. H. Bartels and G. W. Stewart, *Solution of the matrix equation $AX + XB = C$*, Comm. ACM, 15 (1972), pp. 820–826.

[5] P. Benner, V. Mehrmann, and H. Xu, *A numerically stable, structure preserving method for computing the eigenvalues of real Hamiltonian or symplectic pencils*, Numer. Math., 78 (1998), pp. 329–358.

[6] R. P. Brent, *Algorithms for Minimization without Derivatives*, Prentice-Hall, Englewood Cliffs, NJ, 1973.

[7] J. V. Burke, D. Henrion, A. S. Lewis, and M. L. Overton, *HIFOO—A* MATLAB *package for fixed-order controller design and H-infinity optimization*, in Proceedings of the 5th IFAC Symposium on Robust Control Design, Toulouse, France, 2006.

[8] J. V. Burke, A. S. Lewis, and M. L. Overton, *Two numerical methods for optimizing matrix stability*, Linear Algebra Appl., 351 (2002), pp. 147–184.

[9] J. V. Burke, A. S. Lewis, and M. L. Overton, *A nonsmooth, nonconvex optimization approach to robust stabilization by static output feedback and low-order controllers*, in Proceedings of 4th IFAC Symposium on Robust Control Design, Milan, Italy, 2003, pp. 175–181.

[10] J. V. Burke, A. S. Lewis, and M. L. Overton, *Optimization and pseudospectra, with applications to robust stability*, SIAM J. Matrix Anal. Appl., 25 (2003), pp. 80–104.

[11] J. V. Burke, A. S. Lewis, and M. L. Overton, *Robust stability and a criss-cross algorithm for pseudospectra*, IMA J. Numer. Anal., 23 (2003), pp. 359–375.

[12] J. V. Burke, A. S. Lewis, and M. L. Overton, *A robust gradient sampling algorithm for nonsmooth, nonconvex optimization*, SIAM J. Optim., 15 (2005), pp. 751–779.

[13] J. V. Burke and M. L. Overton, *Differential properties of the spectral abscissa and the spectral radius for analytic matrix-valued mappings*, Nonlinear Anal., 23 (1994), pp. 467–488.

[14] J. V. Burke and M. L. Overton, *Variational analysis of non-Lipschitz spectral functions*, Math. Program., 90 (2001), pp. 317–352.

[15] M. Diehl, K. Mombaur, and D. Noll, *Stability Optimization of Hybrid Periodic Systems via a Smooth Criterion*, Technical report 07-97, ESAT-SISTA, K.U.Leuven, Belgium, 2007.

[16] S. K. Godunov, *Ordinary Differential Equations with Constant Coefficient*, Trans. Math. Monogr. 169, American Mathematical Society, Providence, RI, 1997.

[17] A. Graham, *Kronecker Products and Matrix Calculus With Applications*, Halsted Press, John Wiley and Sons, New York, 1981.

[18] P. Lancaster, *Explicit solutions of linear matrix equations*, SIAM Rev., 12 (1970), pp. 544–566.

[19] F. Leibfritz, *COMPl$_e$ib: COnstraint Matrix-optimization Problem Library – A Collection of Test Examples for Nonlinear Semidefinite Programs, Control System Design and Related Problems*, Technical report, Universität Trier, Trier, Germany, 2004.

[20] C. F. V. Loan, *The ubiquitous Kronecker product*, J. Comput. Appl. Math., 123 (2000), pp. 85–100.

[21] C. V. Loan, *The sensitivity of the matrix exponential*, SIAM J. Numer. Anal., 14 (1977), pp. 971–981.

[22] J. Löfberg, *YALMIP: A toolbox for modeling and optimization in* matlab, in Proceedings of the CACSD Conference, Taipei, Taiwan, 2004.

[23] M. Mammadov and R. Orsi, *H-infinity synthesis via a nonsmooth, nonconvex optimization approach*, Pac. J. Optim., 1 (2005), pp. 405–420.

[24] W. Michiels and D. Roose, *An eigenvalue based approach for the robust stabilization of linear time-delay systems*, Internat. J. Control, 76 (2003), pp. 678–686.

[25] D. Noll and P. Apkarian, *Spectral bundle methods for nonconvex maximum eigenvalue functions. Part* 1: *First-order methods*, Math. Program. Ser. B, 104 (2005), pp. 701–727.

[26] D. Noll and P. Apkarian, *Spectral bundle methods for nonconvex maximum eigenvalue functions. Part* 2: *Second-order methods*, Math. Program. Ser. B, 104 (2005), pp. 729–747.

[27] J. F. Sturm, *Using SeDuMi* 1.02*, a* matlab *toolbox for optimization over symmetric cones*, Optim. Methods Softw., 11-12 (1999), pp. 625–653.

[28] L. N. Trefethen and M. Embree, *Spectra and Pseudospectra – The Behavior of Nonnormal Matrices*, Princeton University Press, Princeton, NJ, 2005.

[29] J. Vanbiervliet, K. Verheyden, W. Michiels, and S. Vandewalle, *A nonsmooth optimisation approach for the stabilisation of time-delay systems*, ESAIM Control Optim. Calc. Var., 14 (2008), pp. 478–493.

[30] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice-Hall, Englewood Cliffs, NJ, 1996.

SIAM J. OPTIM.
Vol. 20, No. 1, pp. 172–191

BENCHMARKING DERIVATIVE-FREE OPTIMIZATION ALGORITHMS[*]

JORGE J. MORÉ[†] AND STEFAN M. WILD[‡]

Abstract. We propose *data profiles* as a tool for analyzing the performance of derivative-free optimization solvers when there are constraints on the computational budget. We use performance and data profiles, together with a convergence test that measures the decrease in function value, to analyze the performance of three solvers on sets of smooth, noisy, and piecewise-smooth problems. Our results provide estimates for the performance difference between these solvers, and show that on these problems, the model-based solver tested performs better than the two direct search solvers tested.

Key words. derivative-free optimization, benchmarking, performance evaluation, deterministic simulations, computational budget

AMS subject classifications. 90C56, 65Y20, 65K10

DOI. 10.1137/080724083

1. Introduction.

Derivative-free optimization has experienced a renewed interest over the past decade that has encouraged a new wave of theory and algorithms. While this research includes computational experiments that compare and explore the properties of these algorithms, there is no consensus on the benchmarking procedures that should be used to evaluate derivative-free algorithms.

We explore benchmarking procedures for derivative-free optimization algorithms when there is a limited computational budget. The focus of our work is the unconstrained optimization problem

$$\min \{ f(x) : x \in \mathbb{R}^n \}, \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ may be noisy or nondifferentiable and, in particular, in the case where the evaluation of f is computationally expensive. These expensive optimization problems arise in science and engineering because evaluation of the function f often requires a complex deterministic simulation based on solving the equations (for example, nonlinear eigenvalue problems, ordinary or partial differential equations) that describe the underlying physical phenomena. The computational noise associated with these complex simulations means that obtaining derivatives is difficult and unreliable. Moreover, these simulations often rely on legacy or proprietary codes and hence must be treated as black-box functions, necessitating a derivative-free optimization algorithm.

Several comparisons have been made of derivative-free algorithms on noisy optimization problems that arise in applications. In particular, we mention [7, 10, 14, 17, 22]. The most ambitious work in this direction [7] is a comparison of six derivative-free optimization algorithms on two variations of a groundwater problem specified

[*] Received by the editors May 13, 2008; accepted for publication (in revised form) December 8, 2008; published electronically March 25, 2009.
http://www.siam.org/journals/siopt/20-1/72408.html

[†] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439 (more@mcs.anl.gov). This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract DE-AC02-06CH11357.

[‡] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439 (wild@mcs.anl.gov). This work was supported by a DOE Computational Science Graduate Fellowship under grant number DE-FG02-97ER25308.

by a simulator. In this work algorithms are compared by their trajectories (plot of the best function value against the number of evaluations) until the solver satisfies a convergence test based on the resolution of the simulator. The work in [7] also addresses *hidden constraints*, regions where the function does not return a proper value, a setting in which we have not yet applied the methodology presented here.

Benchmarking derivative-free algorithms on selected applications with trajectory plots provide useful information to users with related applications. In particular, users can find the solver that delivers the largest reduction within a given computational budget. However, the conclusions in these computational studies do not readily extend to other applications. Further, when testing larger sets of problems it becomes increasingly difficult to understand the overall performance of solvers using a single trajectory plot for each problem.

Most researchers have relied on a selection of problems from the CUTEr [9] collection of optimization problems for their work on testing and comparing derivative-free algorithms. Work in this direction includes [3, 14, 16, 18, 20]. The performance data gathered in these studies is the number of function evaluations required to satisfy a convergence test when there is a limit μ_f on the number of function evaluations. The convergence test is sometimes related to the accuracy of the current iterate as an approximation to a solution, while in other cases it is related to a parameter in the algorithm. For example, a typical convergence test for trust region methods [3, 18, 20] requires that the trust region radius be smaller than a given tolerance.

Users with expensive function evaluations are often interested in a convergence test that measures the decrease in function value. In section 2 we propose the convergence test

$$f(x_0) - f(x) \geq (1-\tau)(f(x_0) - f_L), \tag{1.2}$$

where $\tau > 0$ is a tolerance, x_0 is the starting point for the problem, and f_L is computed for each problem as the smallest value of f obtained by any solver within a given number μ_f of function evaluations. This convergence test is well suited for derivative-free optimization because it is invariant to the affine transformation $f \mapsto \alpha f + \beta$ ($\alpha > 0$) and measures the function value reduction $f(x_0) - f(x)$ achieved by x relative to the best possible reduction $f(x_0) - f_L$.

The convergence test (1.2) was used by Marazzi and Nocedal [16] but with f_L set to an accurate estimate of f at a local minimizer obtained by a derivative-based solver. In section 2 we show that setting f_L to an accurate estimate of f at a minimizer is not appropriate when the evaluation of f is expensive, since no solver may be able to satisfy (1.2) within the user's computational budget.

We use performance profiles [5] with the convergence test (1.2) to evaluate the performance of derivative-free solvers. Instead of using a fixed value of τ, we use $\tau = 10^{-k}$ with $k \in \{1, 3, 5, 7\}$ so that a user can evaluate solver performance for different levels of accuracy. These performance profiles are useful to users who need to choose a solver that provides a given reduction in function value within a limit of μ_f function evaluations.

To the authors' knowledge, previous work with performance profiles has not varied the limit μ_f on the number of function evaluations and has used large values for μ_f. The underlying assumption has been that the long-term behavior of the algorithm is of utmost importance. This assumption is not likely to hold, however, if the evaluation of f is expensive.

Performance profiles were designed to compare solvers and thus use a performance ratio instead of the number of function evaluations required to solve a problem. As a result, performance profiles do not provide the percentage of problems that can be solved (for a given tolerance τ) with a given number of function evaluations. This information is essential to users with expensive optimization problems and thus an interest in the short-term behavior of algorithms. On the other hand, the *data profiles* of section 2 have been designed to provide this information.

The remainder of this paper is devoted to demonstrating the use of performance and data profiles for benchmarking derivative-free optimization solvers. Section 2 reviews the use of performance profiles with the convergence test (1.2) and defines data profiles.

Section 3 provides a brief overview of the solvers selected to illustrate the benchmarking process: The Nelder-Mead NMSMAX code [13], the pattern-search APPSPACK code [11], and the model-based trust region NEWUOA code [20]. Since the emphasis of this paper is on the benchmarking process, no attempt was made to assemble a large collection of solvers. The selection of solvers was guided mainly by a desire to examine the performance of a representative subset of derivative-free solvers.

Section 4 describes the benchmark problems used in the computational experiments. We use a selection of problems from the CUTEr [9] collection for the basic set, but since the functions f that describe the optimization problem are invariably smooth, with at least two continuous derivatives, we augment this basic set with noisy and piecewise-smooth problems derived from this basic set. The choice of noisy problems was guided by a desire to mimic simulation-based optimization problems.

The benchmarking results in section 5 show that data and performance profiles provide complementary information that measures the strengths and weaknesses of optimization solvers as a function of the computational budget. Data profiles are useful, in particular, to assess the short-term behavior of the algorithms. The results obtained from the benchmark problems of section 4 show that the model-based solver NEWUOA performs better than the direct search solvers NMSMAX and APPSPACK even for noisy and piecewise-smooth problems. These results also provide estimates for the performance differences between these solvers.

Standard disclaimers [5] in benchmarking studies apply to the results in section 5. In particular, all solvers were tested with the default options, so results may change if these defaults are changed. In a similar vein, our results apply only to the current version of these solvers and this family of test problems, and may change with future versions of these solvers and other families of problems.

2. Benchmarking derivative-free optimization solvers.

Performance profiles, introduced by Dolan and Moré [5], have proved to be an important tool for benchmarking optimization solvers. Dolan and Moré define a benchmark in terms of a set $\mathcal{P}$ of benchmark problems, a set $\mathcal{S}$ of optimization solvers, and a convergence test $\mathcal{T}$. Once these components of a benchmark are defined, performance profiles can be used to compare the performance of the solvers. In this section we first propose a convergence test for derivative-free optimization solvers and then examine the relevance of performance profiles for optimization problems with expensive function evaluations.

2.1. Performance profiles.

Performance profiles are defined in terms of a performance measure $t_{p,s} > 0$ obtained for each $p \in \mathcal{P}$ and $s \in \mathcal{S}$. For example, this measure could be based on the amount of computing time or the number of function evaluations required to satisfy the convergence test. Larger values of $t_{p,s}$ indicate worse performance. For any pair (p, s) of problem p and solver s, the performance ratio is defined by

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in \mathcal{S}\}}.$$

Note that the best solver for a particular problem attains the lower bound $r_{p,s} = 1$. The convention $r_{p,s} = \infty$ is used when solver s fails to satisfy the convergence test on problem p.

The *performance profile* of a solver $s \in \mathcal{S}$ is defined as the fraction of problems where the performance ratio is at most α, that is,

$$\rho_s(\alpha) = \frac{1}{|\mathcal{P}|}\text{size}\{p \in \mathcal{P} : r_{p,s} \leq \alpha\}, \tag{2.1}$$

where $|\mathcal{P}|$ denotes the cardinality of $\mathcal{P}$. Thus, a performance profile is the probability distribution for the ratio $r_{p,s}$. Performance profiles seek to capture how well the solver performs relative to the other solvers in $\mathcal{S}$ on the set of problems in $\mathcal{P}$. Note, in particular, that $\rho_s(1)$ is the fraction of problems for which solver $s \in \mathcal{S}$ performs the best and that for α sufficiently large, $\rho_s(\alpha)$ is the fraction of problems solved by $s \in \mathcal{S}$. In general, $\rho_s(\alpha)$ is the fraction of problems with a performance ratio $r_{p,s}$ bounded by α, and thus solvers with high values for $\rho_s(\alpha)$ are preferable.

Benchmarking gradient-based optimization solvers is reasonably straightforward once the convergence test is chosen. The convergence test is invariably based on the gradient, for example,

$$\|\nabla f(x)\| \leq \tau \|\nabla f(x_0)\|$$

for some $\tau > 0$ and norm $\|\cdot\|$. This convergence test is augmented by a limit on the amount of computing time or the number of function evaluations. The latter requirement is needed to catch solvers that are not able to solve a given problem.

Benchmarking gradient-based solvers is usually done with a fixed choice of tolerance τ that yields reasonably accurate solutions on the benchmark problems. The underlying assumption is that the performance of the solvers will not change significantly with other choices of the tolerance and that, in any case, users tend to be interested in solvers that can deliver high-accuracy solutions. In derivative-free optimization, however, users are interested in both low-accuracy and high-accuracy solutions. In practical situations, when the evaluation of f is expensive, a low-accuracy solution is all that can be obtained within the user's computational budget. Moreover, in these situations, the accuracy of the data may warrant only a low-accuracy solution.

Benchmarking derivative-free solvers requires a convergence test that does not depend on evaluation of the gradient. We propose to use the convergence test

$$f(x) \leq f_L + \tau(f(x_0) - f_L), \tag{2.2}$$

where $\tau > 0$ is a tolerance, x_0 is the starting point for the problem, and f_L is computed for each problem $p \in \mathcal{P}$ as the smallest value of f obtained by any solver within a given number μ_f of function evaluations. The convergence test (2.2) can also be written as

$$f(x_0) - f(x) \geq (1 - \tau)(f(x_0) - f_L),$$

and this shows that (2.2) requires that the reduction $f(x_0) - f(x)$ achieved by x be at least $1 - \tau$ times the best possible reduction $f(x_0) - f_L$.

The convergence test (2.2) was used by Elster and Neumaier [6] but with f_L set to an accurate estimate of f at a global minimizer. This test was also used by Marazzi and Nocedal [16] but with f_L set to an accurate estimate of f at a local minimizer obtained by a derivative-based solver. Setting f_L to an accurate estimate of f at a minimizer is not appropriate when the evaluation of f is expensive because no solver may be able to satisfy (2.2) within the user's computational budget. Even for problems with a cheap f, a derivative-free solver is not likely to achieve accuracy comparable to a derivative-based solver. On the other hand, if f_L is the smallest value of f obtained by any solver, then at least one solver will satisfy (2.2) for any $\tau \geq 0$.

An advantage of (2.2) is that it is invariant to the affine transformation $f \mapsto \alpha f + \beta$ where $\alpha > 0$. Hence, we can assume, for example, that $f_L = 0$ and $f(x_0) = 1$. There is no loss in generality in this assumption because derivative-free algorithms are invariant to the affine transformation $f \mapsto \alpha f + \beta$. Indeed, algorithms for gradient-based optimization (unconstrained and constrained) problems are also invariant to this affine transformation.

The tolerance $\tau \in [0, 1]$ in (2.2) represents the percentage decrease from the starting value $f(x_0)$. A value of $\tau = 0.1$ may represent a modest decrease, a reduction that is 90% of the total possible, while smaller values of τ correspond to larger decreases. As τ decreases, the accuracy of $f(x)$ as an approximation to f_L increases; the accuracy of x as an approximation to some minimizer depends on the growth of f in a neighborhood of the minimizer. As noted, users are interested in the performance of derivative-free solvers for both low-accuracy and high-accuracy solutions. A user's expectation of the decrease possible within their computational budget will vary from application to application.

The following new result relates the convergence test (2.2) to convergence results for gradient-based optimization solvers.

THEOREM 2.1. *Assume that $f : \mathbb{R}^n \mapsto \mathbb{R}$ is a strictly convex quadratic and that x^* is the unique minimizer of f. If $f_L = f(x^*)$, then $x \in \mathbb{R}^n$ satisfies the convergence test* (2.2) *if and only if*

$$\|\nabla f(x)\|_* \leq \tau^{1/2} \, \|\nabla f(x_0)\|_* \tag{2.3}$$

for the norm $\| \cdot \|_*$ defined by

$$\|v\|_* = \|G^{-\frac{1}{2}} \, v\|_2,$$

and G is the Hessian matrix of f.

Proof. Since f is a quadratic, G is the Hessian matrix of f, and x^* is the unique minimizer,

$$f(x) = f(x^*) + \tfrac{1}{2}(x - x^*)^T G(x - x^*).$$

Hence, the convergence test (2.2) holds if and only if

$$(x - x^*)^T G(x - x^*) \leq \tau(x_0 - x^*)^T G(x_0 - x^*),$$

which in terms of the square root $G^{\frac{1}{2}}$ is just

$$\|G^{\frac{1}{2}}(x - x^*)\|_2^2 \leq \tau \|G^{\frac{1}{2}}(x_0 - x^*)\|_2^2.$$

We obtain (2.3) by noting that since x^* is the minimizer of the quadratic f and G is the Hessian matrix, $\nabla f(x) = G(x - x^*)$. □

Other variations on Theorem 2.1 are of interest. For example, it is not difficult to show, by using the same proof techniques, that (2.2) is also equivalent to

$$\tfrac{1}{2}\|\nabla f(x)\|_*^2 \le \tau\,(f(x_0)-f(x^*)). \tag{2.4}$$

This inequality shows, in particular, that we can expect that the accuracy of x, as measured by the gradient norm $\|\nabla f(x)\|_*$, to increase with the square root of $f(x_0)-f(x^*)$.

Similar estimates hold for the error in x because $\nabla f(x) = G(x-x^*)$. Thus, in view of (2.3), the convergence test (2.2) is equivalent to

$$\|x-x^*\|_\diamond \le \tau^{1/2}\,\|x_0-x^*\|_\diamond,$$

where the norm $\|\cdot\|_\diamond$ is defined by

$$\|v\|_\diamond = \|G^{\frac{1}{2}}v\|_2.$$

In this case the accuracy of x in the $\|\cdot\|_\diamond$ norm increases with the distance of x_0 from x^* in the $\|\cdot\|_\diamond$ norm.

We now explore an extension of Theorem 2.1 to nonlinear functions that is valid for an arbitrary starting point x_0. The following result shows that the convergence test (2.2) is (asymptotically) the same as the convergence test (2.4).

LEMMA 2.2. *If $f:\mathbb{R}^n \mapsto \mathbb{R}$ is twice continuously differentiable in a neighborhood of a minimizer x^* with $\nabla^2 f(x^*)$ positive definite, then*

$$\lim_{x\to x^*}\frac{f(x)-f(x^*)}{\|\nabla f(x)\|_*^2} = \frac{1}{2}, \tag{2.5}$$

where the norm $\|\cdot\|_*$ is defined in Theorem 2.1 and $G=\nabla^2 f(x^*)$.

Proof. We first prove that

$$\lim_{x\to x^*}\frac{\|\nabla^2 f(x^*)^{1/2}(x-x^*)\|}{\|\nabla f(x)\|_*} = 1. \tag{2.6}$$

This result can be established by noting that since $\nabla^2 f$ is continuous at x^* and $\nabla f(x^*)=0$,

$$\nabla f(x) = \nabla^2 f(x^*)(x-x^*) + r_1(x), \qquad r_1(x) = o(\|x-x^*\|).$$

If λ_1 is the smallest eigenvalue of $\nabla^2 f(x^*)$, then this relationship implies, in particular, that

$$\|\nabla f(x)\|_* \ge \tfrac{1}{2}\lambda_1^{1/2}\|x-x^*\| \tag{2.7}$$

for all x near x^*. This inequality and the previous relationship prove (2.6). We can now complete the proof by noting that since $\nabla^2 f$ is continuous at x^* and $\nabla f(x^*)=0$,

$$f(x) = f(x^*) + \tfrac{1}{2}\|\nabla^2 f(x^*)^{1/2}(x-x^*)\|^2 + r_2(x), \qquad r_2(x) = o(\|x-x^*\|^2).$$

This relationship, together with (2.6) and (2.7), completes the proof. □

Lemma 2.2 shows that there is a neighborhood $N(x^*)$ of x^* such that if $x\in N(x^*)$ satisfies the convergence test (2.2) with $f_L = f(x^*)$, then

$$\|\nabla f(x)\|_* \le \gamma\,\tau^{1/2}\,(f(x_0)-f(x^*))^{1/2}, \tag{2.8}$$

where the constant γ is a slight overestimate of $2^{1/2}$. Conversely, if γ is a slight underestimate of $2^{1/2}$, then (2.8) implies that (2.2) holds in some neighborhood of x^*. Thus, in this sense, the gradient test (2.8) is asymptotically equivalent to (2.2) for smooth functions.

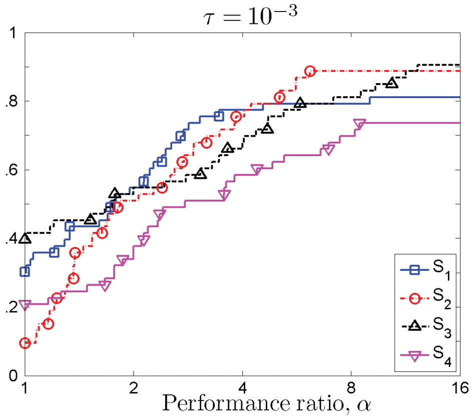


FIG. 2.1. *Sample performance profile $\rho_s(\alpha)$ (logarithmic scale) for derivative-free solvers.*

2.2. Data profiles. We can use performance profiles with the convergence test (2.2) to benchmark optimization solvers for problems with expensive function evaluations. In this case the performance measure $t_{p,s}$ is the number of function evaluations because this is assumed to be the dominant cost per iteration. Performance profiles provide an accurate view of the relative performance of solvers within a given number μ_f of function evaluations. Performance profiles do not, however, provide sufficient information for a user with an expensive optimization problem.

Figure 2.1 shows a typical performance profile for derivative-free optimization solvers with the convergence test (2.2) and $\tau = 10^{-3}$. Users generally are interested in the best solver, and for these problems and level of accuracy, solver S_3 has the best performance. However, it is also important to pay attention to the performance difference between solvers. For example, consider the performance profiles ρ_1 and ρ_4 at a performance ratio of $\alpha = 2$, $\rho_1(2) \approx 55\%$, and $\rho_4(2) \approx 35\%$. These profiles show that solver S_4 requires more than twice the number of function evaluations as solver S_1 on roughly 20% of the problems. This is a significant difference in performance.

The performance profiles in Figure 2.1 provide an accurate view of the performance of derivative-free solvers for $\tau = 10^{-3}$. However, these results were obtained with a limit of $\mu_f = 1300$ function evaluations and thus are not directly relevant to a user for which this limit exceeds their computational budget.

Users with expensive optimization problems are often interested in the performance of solvers as a function of the number of functions evaluations. In other words, these users are interested in *the percentage of problems that can be solved (for a given tolerance τ) with κ function evaluations.* We can obtain this information by letting $t_{p,s}$ be the number of function evaluations required to satisfy (2.2) for a given tolerance τ, since then

$$d_s(\alpha) = \frac{1}{|\mathcal{P}|}\text{size}\{p \in \mathcal{P} : t_{p,s} \le \alpha\}$$

is the percentage of problems that can be solved with α function evaluations. As usual, there is a limit μ_f on the total number of function evaluations, and $t_{p,s} = \infty$ if the convergence test (2.2) is not satisfied after μ_f evaluations.

Griffin and Kolda [12] were also interested in performance in terms of the number of functions evaluations and used plots of the total number of solved problems as a function of the number of (penalty) function evaluations to evaluate performance. They did not investigate how results changed if the convergence test was changed; their main concern was to evaluate the performance of their algorithm with respect to the penalty function.

This definition of d_s is independent of the number of variables in the problem $p \in \mathcal{P}$. This is not realistic because, in our experience, the number of function evaluations needed to satisfy a given convergence test is likely to grow as the number of variables increases. We thus define the *data profile* of a solver $s \in \mathcal{S}$ by

$$d_s(\alpha) = \frac{1}{|\mathcal{P}|}\text{size}\left\{p \in \mathcal{P} : \frac{t_{p,s}}{n_p + 1} \le \alpha\right\}, \tag{2.9}$$

where n_p is the number of variables in $p \in \mathcal{P}$. We refer to a plot of (2.9) as a data profile to acknowledge that its application is more general than the one used here and that our choice of scaling is for illustration only. For example, we note that the authors in [1] expect performance of stochastic global optimization algorithms to grow faster than linear in the dimension.

With this scaling, the unit of cost is $n_p + 1$ function evaluations. This is a convenient unit that can be easily translated into function evaluations. Another advantage of this unit of cost is that $d_s(\kappa)$ can then be interpreted as the percentage of problems that can be solved with the equivalent of κ *simplex gradient estimates*, $n_p + 1$ referring to the number of evaluations needed to compute a one-sided finite-difference estimate of the gradient.

Performance profiles (2.1) and data profiles (2.9) are cumulative distribution functions, and thus monotone increasing, step functions with a range in $[0, 1]$. However, performance profiles compare different solvers, while data profiles display the raw data. In particular, performance profiles do not provide the number of function evaluations required to solve any of the problems. Also note that the data profile for a given solver $s \in \mathcal{S}$ is independent of other solvers; this is not the case for performance profiles.

Data profiles are useful to users with a specific computational budget who need to choose a solver that is likely to reach a given reduction in function value. The user needs to express the computational budget in terms of simplex gradients and examine the values of the data profile d_s for all the solvers. For example, if the user has a budget of 50 simplex gradients, then the data profiles in Figure 2.2 show that solver S_3 solves 90% of the problems at this level of accuracy. This information is not available from the performance profiles in Figure 2.1.

We illustrate the differences between performance and data profiles with a synthetic case involving two solvers. Assume that solver S_1 requires k_1 simplex gradients to solve each of the first n_1 problems, but fails to solve the remaining n_2 problems. Similarly, assume that solver S_2 fails to solve the first n_1 problems, but solves each of the remaining n_2 problems with k_2 simplex gradients. Finally, assume that $n_1 < n_2$, and that $k_1 < k_2$. In this case,

$$\rho_1(\alpha) \equiv \frac{n_1}{n_1 + n_2}, \qquad \rho_2(\alpha) \equiv \frac{n_2}{n_1 + n_2},$$

for all $\alpha \ge 1$ if the maximum number of evaluations μ_f allows k_2 simplex gradients. Hence, $n_1 < n_2$ implies that $\rho_1 < \rho_2$, and thus solver S_2 is preferable. This is justifiable because S_2 solves more problems for all performance ratios. On the other

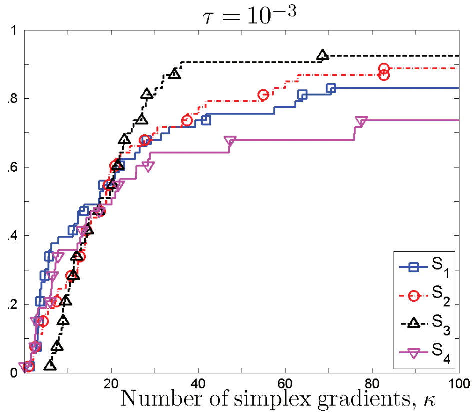


FIG. 2.2. *Sample data profile* $d_s(\kappa)$ *for derivative-free solvers.*

hand,

$$d_1(\alpha) = \begin{cases} 0, & \alpha \in [0, k_1) \\ \dfrac{n_1}{n_1 + n_2}, & \alpha \in [k_1, \infty) \end{cases} \qquad d_2(\alpha) = \begin{cases} 0, & \alpha \in [0, k_2) \\ \dfrac{n_2}{n_1 + n_2}, & \alpha \in [k_2, \infty) \end{cases}$$

In particular, $0 = d_2(k) < d_1(k)$ for all budgets of k simplex gradients where $k \in [k_1, k_2)$, and thus solver S_1 is preferable under these budget constraints. This choice is appropriate because S_2 is not able to solve any problems with less than k_2 simplex gradients.

This example illustrates an extreme case, but this can happen in practice. For example, the data profiles in Figure 2.2 show that solver S_2 outperforms S_1 with a computational budget of k simplex gradients where $k \in [20, 100]$, though the differences are small. On the other hand, the performance profiles in Figure 2.1 show that S_1 outperforms S_2.

One other connection between performance profiles and data profiles needs to be emphasized. The limiting value of $\rho_s(\alpha)$ as $\alpha \to \infty$ is the percentage of problems that can be solved with μ_f function evaluations. Thus,

$$d_s(\hat{\kappa}) = \lim_{\alpha \to \infty} \rho_s(\alpha), \tag{2.10}$$

where $\hat{\kappa}$ is the maximum number of simplex gradients performed in μ_f evaluations. Since the limiting value of ρ_s can be interpreted as the reliability of the solver, we see that (2.10) shows that the data profile d_s measures the reliability of the solver (for a given tolerance τ) as a function of the budget μ_f.

3. Derivative-free optimization solvers. The selection of solvers $\mathcal{S}$ that we use to illustrate the benchmarking process was guided by a desire to examine the performance of a representative subset of derivative-free solvers, and thus we included both direct search and model-based algorithms. Similarly, our selection of solvers was not guided by their theoretical properties. No attempt was made to assemble a large collection of solvers, although we did consider more than a dozen different solvers. Users interested in the performance of other solvers (including SID-PSM [4] and UOBYQA [19]) can find additional results at www.mcs.anl.gov/˜more/dfo. We note that some solvers were not tested because they require additional parameters outside the scope of this investigation, such as the requirement of bounds by imfil [8, 15].

We considered only solvers that are designed to solve unconstrained optimization problems using only function values, and with an implementation that is both serial and deterministic. We used an implementation of the Nelder–Mead method because this method is popular among application scientists. We also present results for the APPSPACK pattern search method because, in a comparison of six derivative-free methods, this code performed well in the benchmarking [7] of a groundwater problem. We used the model-based trust region code NEWUOA because this code performed well in a recent comparison [18] of model-based methods.

The NMSMAX code is an implementation of the Nelder–Mead method and is available from the Matrix Computation Toolbox [13]. Other implementations of the Nelder–Mead method exist, but this code performs well and has a reasonable default for the size of the initial simplex. All variations on the Nelder–Mead method update an initial simplex defined by $n+1$ points via a sequence of reflections, expansions, and contractions. Not all of the Nelder–Mead codes that we examined, however, allow the size of the initial simplex to be specified in the calling sequence. The NMSMAX code requires an initial starting point x_0, a limit on the number of function evaluations, and the choice of a starting simplex. The user can choose either a regular simplex or a right-angled simplex with sides along the coordinate axes. We used the right-angled simplex with the default value of

$$\Delta_0 = \max\{1, \|x_0\|_\infty\} \tag{3.1}$$

for the length of the sides. This default value performs well in our testing. The right-angled simplex was chosen to conform with the default initializations of the two other solvers.

The APPSPACK code [11] is an asynchronous parallel pattern search method designed for problems characterized by expensive function evaluations. The code can be run in serial mode, and this is the mode used in our computational experiments. This code requires an initial starting point x_0, a limit on the number of function evaluations, the choice of scaling for the starting pattern, and an initial step size. We used unit scaling with an initial step size Δ_0 defined by (3.1) so that the starting pattern was defined by the right-angled simplex with sides of length Δ_0.

The model-based trust region code NEWUOA [20, 21] uses a quadratic model obtained by interpolation of function values at a subset of m previous trial points; the geometry of these points is monitored and improved if necessary. We used $m = 2n+1$ as recommended by Powell [20]. The NEWUOA code requires an initial starting point x_0, a limit on the number of function evaluations, and the initial trust region radius. We used Δ_0 as in (3.1) for the initial trust region radius.

Our choice of initial settings ensures that all codes are given the same initial information. As a result, both NMSMAX and NEWUOA evaluate the function at the vertices of the right-angled simplex with sides of length Δ_0. The APPSPACK code, however, moves off this initial pattern as soon as a lower function value is obtained.

We effectively set all termination parameters to zero so that all codes terminate only when the limit on the number of function evaluations is exceeded. In a few cases the codes terminate early. This situation happens, for example, if the trust region radius (size of the simplex or pattern) is driven to zero. Since APPSPACK requires a strictly positive termination parameter for the final pattern size, we used 10^{-20} for this parameter.

Table 4.1
Distribution of problem dimensions.

n_p	2	3	4	5	6	7	8	9	10	11	12
Number of problems	5	6	5	4	4	5	6	5	4	4	5

4. Benchmark problems.

The benchmark problems we have selected highlight some of the properties of derivative-free solvers as they face different classes of optimization problems. We made no attempt to define a definitive set of benchmark problems, but these benchmark problems could serve as a starting point for further investigations. This test set is easily available, widely used, and allows us easily examine different types of problems.

Our benchmark set comprises 22 of the nonlinear least squares functions defined in the CUTEr [9] collection. Each function is defined by m components $f_1, \ldots, f_m$ of n variables and a standard starting point x_s.

The problems in the benchmark set $\mathcal{P}$ are defined by a vector (k_p, n_p, m_p, s_p) of integers. The integer k_p is a reference number for the underlying CUTEr function, n_p is the number of variables, m_p is the number of components, and $s_p \in \{0, 1\}$ defines the starting point via $x_0 = 10^{s_p} x_s$, where x_s is the standard starting point for this function. The use of $s_p = 1$ is helpful for testing solvers from a remote starting point because the standard starting point tends to be close to a solution for many of the problems.

The benchmark set $\mathcal{P}$ has 53 different problems. No problem is overrepresented in $\mathcal{P}$ in the sense that no function k_p appears more than six times. Moreover, no pair (k_p, n_p) appears more than twice. In all cases,

$$2 \le n_p \le 12, \quad 2 \le m_p \le 65, \qquad p = 1, \ldots, 53,$$

with $n_p \le m_p$. The distribution of the dimensions n_p among all 53 problems is shown in Table 4.1, the median dimension being 7.

Users interested in the precise specification of the benchmark problems in $\mathcal{P}$ will find the source code for evaluating the problems in $\mathcal{P}$ at www.mcs.anl.gov/~more/dfo. This site also contains source code for obtaining the standard starting points x_s and, a file dfo.dat that provides the integers (k_p, n_p, m_p, s_p).

We use the benchmark set $\mathcal{P}$ defined above to specify benchmark sets for three problem classes: Smooth, piecewise smooth, and noisy problems. The *smooth* problems $\mathcal{P}_S$ are defined by

$$f(x) = \sum_{k=1}^{m} f_k(x)^2. \tag{4.1}$$

These functions are twice continuously differentiable on the level set associated with x_0. Only two functions ($k_p = 7, 16$) have local minimizers that are not global minimizers, but the problems defined by these functions appear only three times in $\mathcal{P}_S$.

The second class of problems mimics simulations that are defined by an iterative process, for example, solving to a specified accuracy a differential equation where the differential equation or the data depends on several parameters. These simulations are not stochastic, but do tend to produce results that are generally considered noisy. We believe the noise in this type of simulation is better modeled by a function with both high-frequency and low-frequency oscillations. We thus defined the *noisy* problems

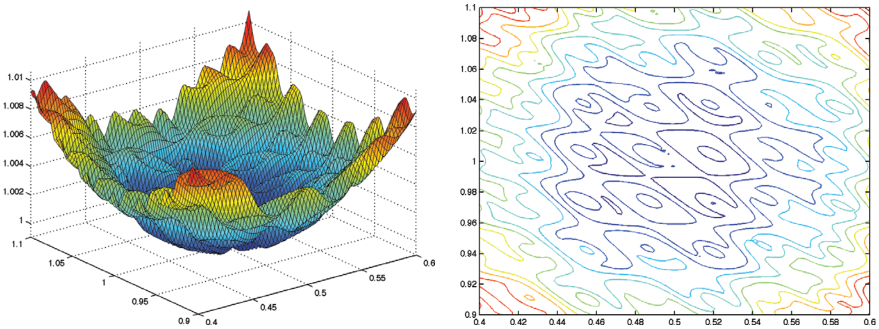


FIG. 4.1. *Plots of the noisy quadratic* (4.5) *on the box* $[0.4, 0.6] \times [0.9, 1.1]$. *Surface plots (left) and level sets (right) show the oscillatory nature of* f.

$\mathcal{P}_N$ by

$$f(x) = (1 + \varepsilon_f \phi(x)) \sum_{k=1}^{m} f_k(x)^2, \tag{4.2}$$

where ε_f is the relative noise level and the noise function $\phi : \mathbb{R}^n \mapsto [-1, 1]$ is defined in terms of the cubic Chebyshev polynomial T_3 by

$$\phi(x) = T_3(\phi_0(x)), \qquad T_3(\alpha) = \alpha(4\alpha^2 - 3), \tag{4.3}$$

where

$$\phi_0(x) = 0.9 \sin(100\|x\|_1) \cos(100\|x\|_\infty) + 0.1 \cos(\|x\|_2). \tag{4.4}$$

The function ϕ_0 defined by (4.4) is continuous and piecewise continuously differentiable with $2^n n!$ regions where ϕ_0 is continuously differentiable. The composition of ϕ_0 with T_3 eliminates the periodicity properties of ϕ_0 and adds stationary points to ϕ at any point where ϕ_0 coincides with the stationary points $(\pm\frac{1}{2})$ of T_3.

Figure 4.1 illustrates the properties of the noisy function (4.2) when the underlying smooth function ($\varepsilon_f = 0$) is a quadratic function. In this case

$$f(x) = (1 + \tfrac{1}{2}\|x - x_0\|^2)(1 + \varepsilon_f \phi(x)), \tag{4.5}$$

where $x_0 = [\frac{1}{2}, 1]$, and noise level $\varepsilon_f = 10^{-3}$. The graph on the left shows f on the two-dimensional box around x_0 and sides of length $\frac{1}{2}$, while the graph on the right shows the contours of f. Both graphs show the oscillatory nature of f, and that f seems to have local minimizers near the global minimizer. Evaluation of f on a mesh shows that, as expected, the minimal value of f is 0.99906, that is, $1 - \varepsilon_f$ to high accuracy.

Our interest centers on smooth and noisy problems, but we also wanted to study the behavior of derivative-free solvers on piecewise-smooth problems. An advantage of the benchmark problems $\mathcal{P}$ is that a set of *piecewise-smooth* problems $\mathcal{P}_{PS}$ can be easily derived by setting

$$f(x) = \sum_{k=1}^{m} |f_k(x)|. \tag{4.6}$$

These problems are continuous, but the gradient does not exist when $f_k(x) = 0$ and $\text{grad}\, f_k(x) \neq 0$ for some index k. They are twice continuously differentiable in the regions where all the f_k do not change sign. There is no guarantee that the problems in $\mathcal{P}_{PS}$ have a unique minimizer, even if (4.1) has a unique minimizer. However, we found that all minimizers were global for all but six functions and that these six functions had global minimizers only, if the variables were restricted to the positive orthant. Hence, for these six functions ($k_p = 8, 9, 13, 16, 17, 18$) the piecewise-smooth problems are defined by

$$f(x) = \sum_{k=1}^{m} |f_k(x_+)|, \tag{4.7}$$

where $x_+ = \max(x, 0)$. This function is piecewise-smooth and agrees with the function f defined by (4.6) for $x \geq 0$.

5. Computational experiments.

We now present the results of computational experiments with the performance measures introduced in section 2. We used the solver set $\mathcal{S}$ consisting of the three algorithms detailed in section 3 and the three problem sets $\mathcal{P}_S$, $\mathcal{P}_N$, and $\mathcal{P}_{PS}$ that correspond, respectively, to the smooth, noisy, and piecewise-smooth benchmark sets of section 4.

The computational results center on the short-term behavior of derivative-free algorithms. We decided to investigate the behavior of the algorithms with a limit of 100 simplex gradients. Since the problems in our benchmark sets have at most 12 variables, we set $\mu_f = 1300$ so that all solvers can use at least 100 simplex gradients.

Data was obtained by recording, for each problem and solver $s \in \mathcal{S}$, the function values generated by the solver at each trial point. All termination tolerances were set as described in section 3 so that solvers effectively terminate only when the limit μ_f on the number of function evaluations is exceeded. In the exceptional cases where the solver terminates early after $k < \mu_f$ function evaluations, we set all successive function values to $f(x_k)$. This data is then processed to obtain a history vector $h_s \in \mathbb{R}^{\mu_f}$ by setting

$$h_s(x_k) = \min \left\{ f(x_j) : 0 \leq j \leq k \right\},$$

so that $h_s(x_k)$ is the best function value produced by solver s after k function evaluations. Each solver produces one history vector for each problem, and these history vectors are gathered into a history array H, one column for each problem. For each problem, $p \in \mathcal{P}$, f_L was taken to be the best function value achieved by any solver within μ_f function evaluations, $f_L = \min_{s \in \mathcal{S}} h_s(x_{\mu_f})$.

We present the data profiles for $\tau = 10^{-k}$ with $k \in \{1, 3, 5, 7\}$ because we are interested in the short-term behavior of the algorithms as the accuracy level changes. We present performance profiles for only $\tau = 10^{-k}$ with $k \in \{1, 5\}$, but a comprehensive set of results is provided at www.mcs.anl.gov/˜more/dfo.

We comment only on the results for an accuracy level of $\tau = 10^{-5}$ and use the other plots to indicate how the results change as τ changes. This accuracy level is mild compared to classical convergence tests based on the gradient. We support this claim by noting that (2.8) implies that if x satisfies the convergence test (2.2) near a minimizer x^*, then

$$\|\nabla f(x)\|_* \leq 0.45 \cdot 10^{-2} \left(f(x_0) - f(x^*) \right)^{1/2}$$

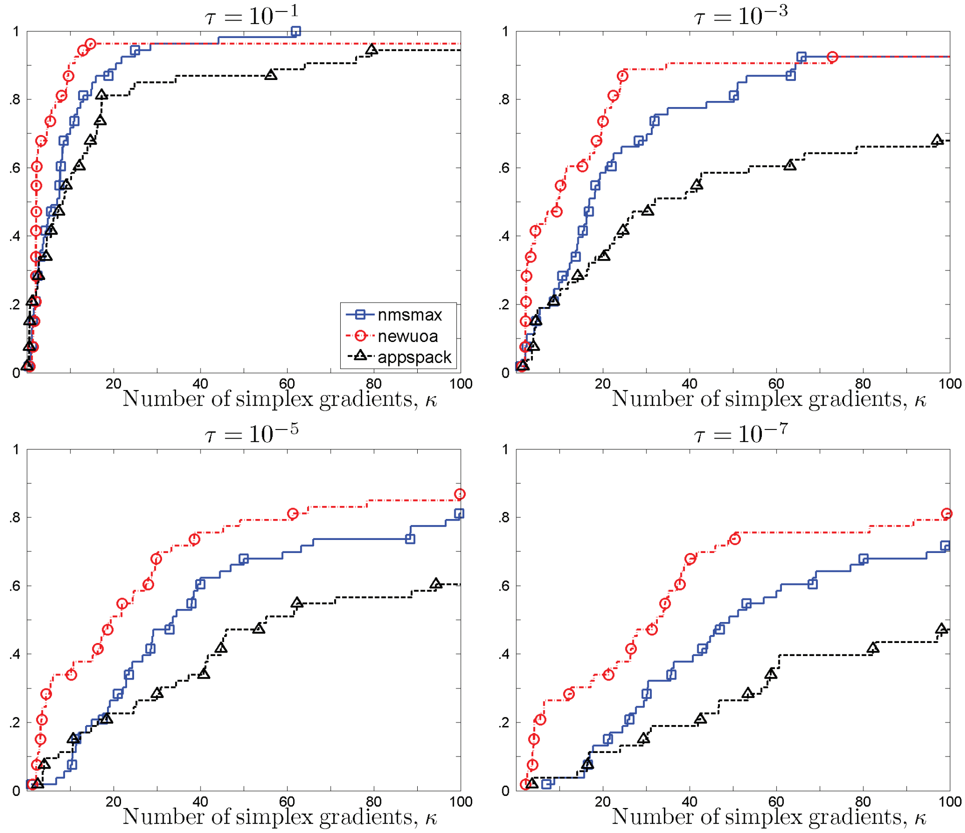


FIG. 5.1. *Data profiles $d_S(\kappa)$ for the smooth problems $\mathcal{P}_S$ show the percentage of problems solved as a function of a computational budget of simplex gradients.*

for $\tau = 10^{-5}$ and for the norm $\|\cdot\|_*$ defined in Theorem 2.1. If the problem is scaled so that $f(x^*) = 0$ and $f(x_0) = 1$, then

$$\|\nabla f(x)\|_* \leq 0.45 \cdot 10^{-2}.$$

This test is not comparable to a gradient test that uses an unscaled norm. It suggests, however, that for well-scaled problems, the accuracy level $\tau = 10^{-5}$ is mild compared to that of classical convergence tests.

5.1. Smooth problems. The data profiles in Figure 5.1 show that NEWUOA solves the largest percentage of problems for all sizes of the computational budget and levels of accuracy τ. This result is perhaps not surprising because NEWUOA is a model-based method based on a quadratic approximation of the function, and thus could be expected to perform well on smooth problems. However, the performance differences are noteworthy.

Performance differences between the solvers tend to be larger when the computational budget is small. For example, with a budget of 10 simplex gradients and $\tau = 10^{-5}$, NEWUOA solves almost 35% of the problems, while both NMSMAX and APPSPACK solve roughly 10% of the problems. Performance differences between NEWUOA and NMSMAX tend to be smaller for larger computational budgets. For example, with a budget of 100 simplex gradients, the performance difference between

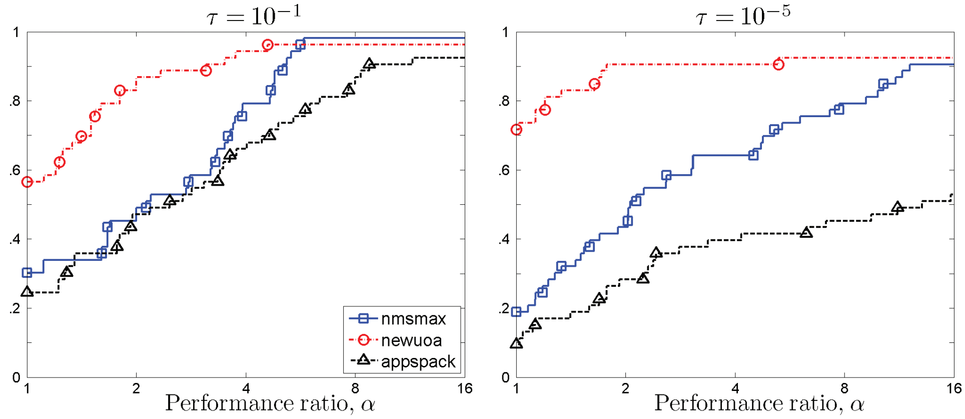


FIG. 5.2. *Performance profiles $\rho_s(\alpha)$ (logarithmic scale) for the smooth problems $\mathcal{P}_S$.*

NEWUOA and NMSMAX is less than 10%. On the other hand, the difference between NEWUOA and APPSPACK is more than 25%.

A benefit of the data profiles is that they can be useful for allocating a computational budget. For example, if a user is interested in getting an accuracy level of $\tau = 10^{-5}$ on at least 50% of problems, the data profiles show that NEWUOA, NMSMAX, and APPSPACK would require 20, 35, and 55 simplex gradients, respectively. This kind of information is not available from performance profiles because they rely on performance ratios.

The performance profiles in Figure 5.2 are for the smooth problems with a logarithmic scale. Performance differences are also of interest in this case. In particular, we note that both of these plots show that NEWUOA is the fastest solver in at least 55% of the problems, while NMSMAX and APPSPACK are each the fastest solvers on fewer than 30% of the problems.

Both plots in Figure 5.2 show that the performance difference between solvers decreases as the performance ratio increases. Since these figures are on a logarithmic scale, however, the decrease is slow. For example, both plots show a performance difference between NEWUOA and NMSMAX of at least 40% when the performance ratio is two. This implies that for at least 40% of the problems NMSMAX takes at least twice as many function evaluations to solve these problems. When $\tau = 10^{-5}$, the performance difference between NEWUOA and APPSPACK is larger, at least 50%.

5.2. Noisy problems. We now present the computational results for the noisy problems $\mathcal{P}_N$ as defined in section 4. We used the noise level $\varepsilon_F = 10^{-3}$ with the nonstochastic noise function ϕ defined by (4.3, 4.4). We consider this level of noise to be about right for simulations controlled by iterative solvers because tolerances in these solvers are likely to be on the order of 10^{-3} or smaller. Smaller noise levels are also of interest. For example, a noise level of 10^{-7} is appropriate for single-precision computations.

Arguments for a nonstochastic noise function were presented in section 4, but here we add that a significant advantage of using a nonstochastic noise function in benchmarking is that this guarantees that the computational results are reproducible up to the precision of the computations. We also note that the results obtained with a noise function ϕ defined by a random number generator are similar to those

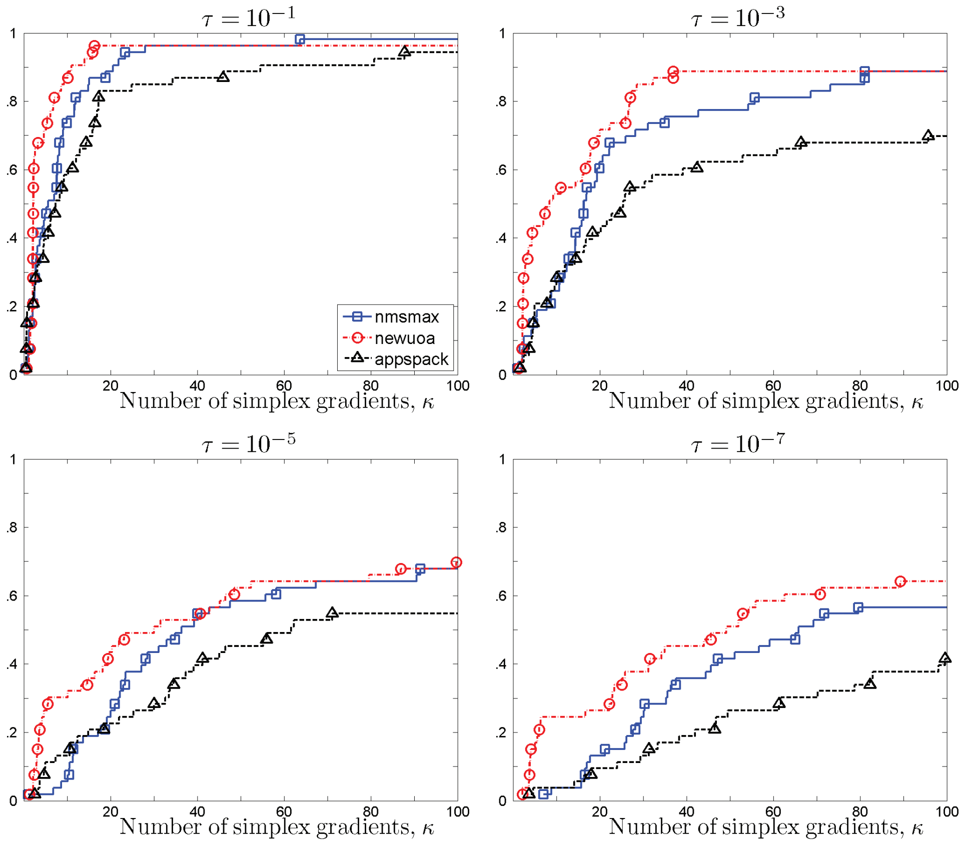


FIG. 5.3. *Data profiles $d_s(\kappa)$ for the noisy problems $\mathcal{P}_N$ show the percentage of problems solved as a function of a computational budget of simplex gradients.*

obtained by the ϕ defined by (4.3, 4.4); results for the stochastic case can be found at www.mcs.anl.gov/~more/dfo.

The data profiles for the noisy problems, shown in Figure 5.3, are surprisingly similar to those obtained for the smooth problems. The degree of similarity between Figures 5.1 and 5.3 is much higher for small computational budgets and the smaller values of τ. This similarity is to be expected for direct search algorithms because the behavior of these algorithms depends only on logical comparisons between function values, and not on the actual function values. On the other hand, the behavior of NEWUOA is affected by noise because the model is determined by interpolating points and is hence sensitive to changes in the function values. Since NEWUOA depends on consistent function values, a performance drop can be expected for stochastic noise of magnitudes near a demanded accuracy level.

An interesting difference between the data profiles for the smooth and noisy problems is that solver performances for large computational budgets tend to be closer than in the smooth case. However, NEWUOA still manages to solve the largest percentage of problems for virtually all sizes of the computational budget and levels of accuracy τ.

Little similarity exists between the performance profiles for the noisy problems $\mathcal{P}_N$ when $\tau = 10^{-5}$, shown in Figure 5.4, and those for the smooth problems. In general

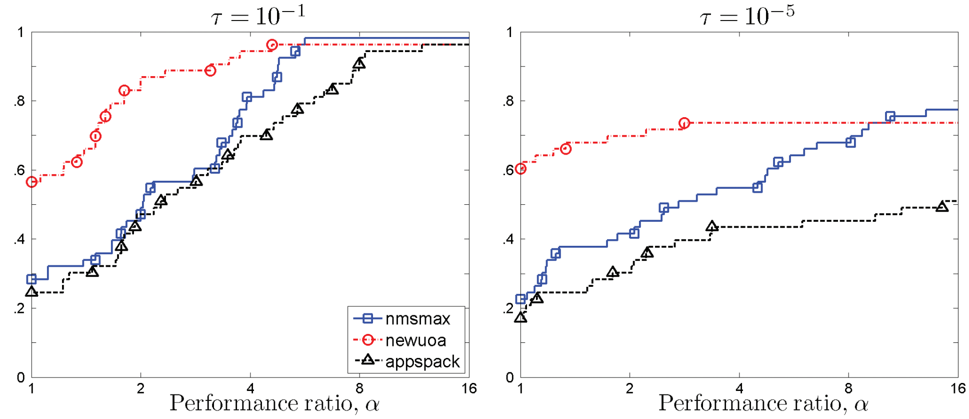


FIG. 5.4. *Performance profiles $\rho_s(\alpha)$ (logarithmic scale) for the noisy problems $\mathcal{P}_N$.*

these plots show that, as expected, noisy problems are harder to solve. For $\tau = 10^{-5}$, NEWUOA is the fastest solver on about 60% of the noisy problems, while it was the fastest solver on about 70% of the smooth problems. However, the performance differences between the solvers are about the same. In particular, both plots in Figure 5.4 show a performance difference between NEWUOA and NMSMAX of about 30% when the performance ratio is two. As we pointed out earlier, performance differences are an estimate of the gains that can be obtained when choosing a different solver.

5.3. Piecewise-smooth problems. The computational experiments for the piecewise-smooth problems $\mathcal{P}_{PS}$ measure how the solvers perform in the presence of nondifferentiable kinks. There is no guarantee of convergence for the tested methods in this case. We note that recent work has focused on relaxing the assumptions of differentiability [2].

The data profiles for the piecewise-smooth problems, shown in Figure 5.5, show that these problems are more difficult to solve than the noisy problems $\mathcal{P}_N$ and the smooth problems $\mathcal{P}_S$. In particular, we note that no solver is able to solve more than 40% of the problems with a computational budget of 100 simplex gradients and $\tau = 10^{-5}$. By contrast, almost 70% of the noisy problems and 90% of the smooth problems can be solved with this budget and level of accuracy. Differences in performance are also smaller for the piecewise smooth problems. NEWUOA solves the most problems in almost all cases, but the performance difference between NEWUOA and the other solvers is smaller than in the noisy or smooth problems.

Another interesting observation on the data profiles is that APPSPACK solves more problems than NMSMAX with $\tau = 10^{-5}$ for all sizes of the computational budget. This is in contrast to the results for smooth and noisy problem where NMSMAX solved more problems than APPSPACK.

The performance profiles for the piecewise-smooth problems $\mathcal{P}_{PS}$ appear in Figure 5.6. The results for $\tau = 10^{-5}$ show that NEWUOA, NMSMAX, and APPSPACK are the fastest solvers on roughly 50%, 30%, and 20% of the problems, respectively. This performance difference is maintained until the performance ratio is near $r = 2$. The same behavior can be seen in the performance profile with $\tau = 10^{-1}$, but now the initial difference in performance is larger, more than 40%. Also note that for $\tau = 10^{-5}$ NEWUOA either solves the problem quickly or does not solve the problem within μ_f evaluations. On the other hand, the reliability of both NMSMAX and APPSPACK in-

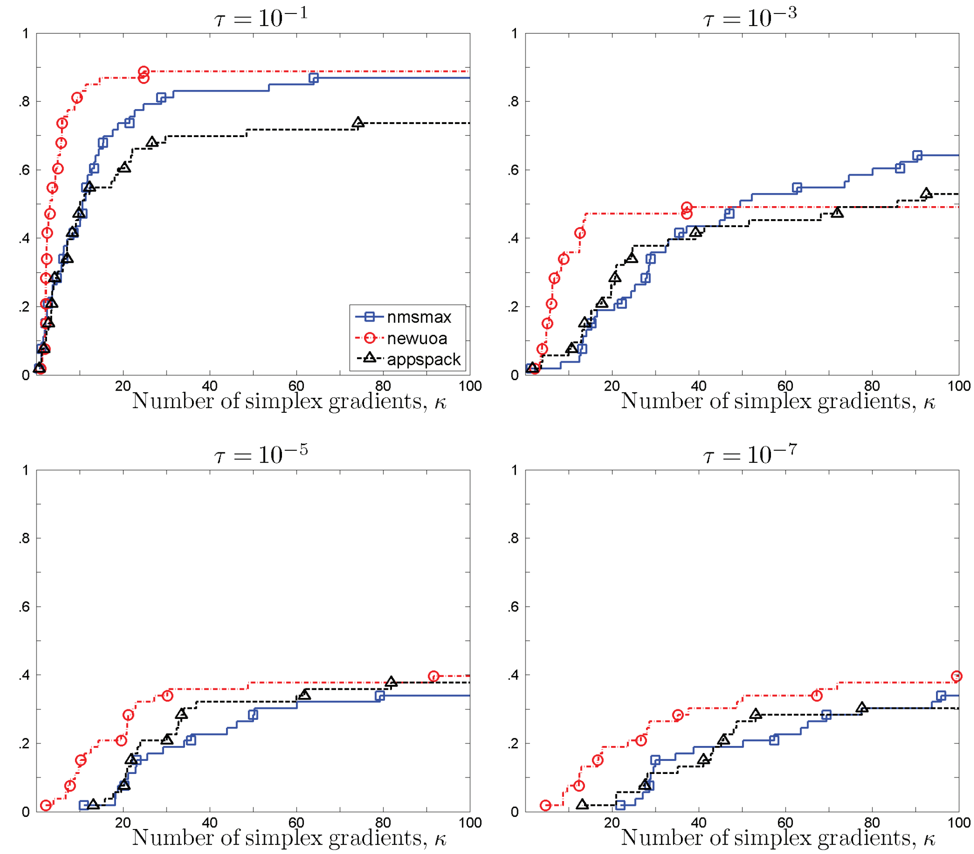


FIG. 5.5. *Data profiles $d_s(\kappa)$ for the piecewise-smooth problems $\mathcal{P}_{PS}$ show the percentage of problems solved as a function of a computational budget of simplex gradients.*

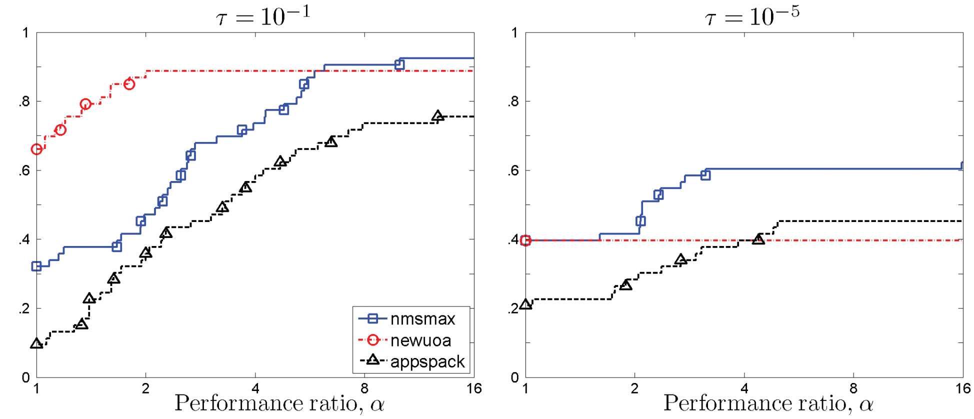


FIG. 5.6. *Performance profiles $\rho_s(\alpha)$ (logarithmic scale) for the piecewise-smooth problems $\mathcal{P}_{PS}$.*

creases with the performance ratio, and NMSMAX eventually solves more problems than NEWUOA.

Finally, note that the performance profiles with $\tau = 10^{-5}$ show that NMSMAX solves more problems than APPSPACK, while the data profiles in Figure 5.5 show that APPSPACK solves more problems than NMSMAX for a computational budget of k simplex gradients where $k \in [25, 100]$. As explained in section 2, this reversal of solver preference can happen when there is a constraint on the computational budget.

6. Concluding remarks.

Our interest in derivative-free methods is motivated in large part by the computationally expensive optimization problems that arise in DOE's SciDAC initiative. These applications give rise to the noisy optimization problems that have been the focus of this work.

We have used the convergence test (2.2) to define performance and data profiles for benchmarking unconstrained derivative-free optimization solvers. This convergence test relies only on the function values obtained by the solver and caters to users with an interest in the short-term behavior of the solver. Data profiles provide crucial information for users who are constrained by a computational budget and complement the measures of relative performance shown by performance plots.

Our computational experiments show that the performance of the three solvers considered varied from problem class to problem class, with the worst performance on the set of piecewise-smooth problems $\mathcal{P}_{PS}$. While NEWUOA generally outperformed the NMSMAX and APPSPACK implementations in our benchmarking environment, the latter two solvers may perform better in other environments. For example, our results did not take into account APPSPACK's ability to work in a parallel processing environment where concurrent function evaluations are possible. Similarly, since our test problems were unconstrained, our results do not readily extend to problems containing *hidden constraints*.

This work can be extended in several directions. For example, data profiles can also be used to benchmark solvers that use derivative information. In this setting we could use a gradient-based convergence test or the convergence test (2.2). Below we outline four other possible future research directions.

Performance on larger problems. The computational experiments in section 5 used problems with at most $n_p = 12$ variables. Performance of derivative-free solvers for larger problems is of interest, but this would require a different set of benchmark problems.

Performance on application problems. Our choice of noisy problems is a first step toward mimicking simulations that are defined by an iterative process, for example, solving a set of differential equations to a specified accuracy. We plan to validate this claim in future work. Performance of derivative-free solvers on other classes of simulations is also of interest.

Performance of other derivative-free solvers. As mentioned before, our emphasis is on the benchmarking process, and thus no attempt was made to assemble a large collection of solvers. Users interested in the performance of other solvers can find additional results at www.mcs.anl.gov/˜more/dfo. Results for additional solvers can be added easily.

Performance with respect to input and algorithmic parameters. Our computational experiments used default input and algorithmic parameters, but we are aware that performance can change for other choices.

Acknowledgments. The authors are grateful to Christine Shoemaker for useful conversations throughout the course of this work and to Josh Griffin and Luís Vicente for valuable comments on an earlier draft. Lastly, we are grateful for the developers of the freely available solvers we tested for providing us with source code and installation assistance: Genetha Gray and Tammy Kolda (APPSPACK [11]), Nick Higham

(MDSMAX and NMSMAX [13]), Tim Kelley (nelder and imfil [15]), Michael Powell (NEWUOA [20] and UOBYQA [19]), and Ana Custódio and Luís Vicente (SID-PSM [4]).

REFERENCES

[1] M. M. Ali, C. Khompatraporn, and Z. B. Zabinsky, *A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems*, J. Global Optim., 31 (2005), pp. 635–672.

[2] C. Audet and J. E. Dennis, *Analysis of generalized pattern searches*, SIAM J. Optim., 13 (2002), pp. 889–903.

[3] A. R. Conn, K. Scheinberg, and P. L. Toint, *A derivative free optimization algorithm in practice*, in Proceedings of 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, 1998.

[4] A. L. Custódio and L. N. Vicente, *Using sampling and simplex derivatives in pattern search methods*, SIAM J. Optim., 18 (2007), pp. 537–555.

[5] E. D. Dolan and J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), pp. 201–213.

[6] C. Elster and A. Neumaier, *A grid algorithm for bound constrained optimization of noisy functions*, IMA J. Numer. Anal., 15 (1995), pp. 585–608.

[7] K. R. Fowler, J. P. Reese, C. E. Kees, J. E. Dennis, Jr., C. T. Kelley, C. T. Miller, C. Audet, A. J. Booker, G. Couture, R. W. Darwin, M. W. Farthing, D. E. Finkel, J. M. Gablonsky, G. Gray, and T. G. Kolda, *A comparison of derivative-free optimization methods for groundwater supply and hydraulic capture community problems*, Advances in Water Resources, 31 (2008), pp. 743–757.

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

[9] N. I. M. Gould, D. Orban, and P. L. Toint, *CUTEr and SifDec: A constrained and unconstrained testing environment, revisited*, ACM Trans. Math. Software, 29 (2003), pp. 373–394.

[10] G. A. Gray, T. G. Kolda, K. Sale, and M. M. Young, *Optimizing an empirical scoring function for transmembrane protein structure determination*, INFORMS J. Comput., 16 (2004), pp. 406–418.

[11] G. A. Gray and T. G. Kolda, *Algorithm* 856: *APPSPACK* 4.0: *Asynchronous parallel pattern search for derivative-free optimization*, ACM Trans. Math. Software, 32 (2006), pp. 485–507.

[12] J. D. Griffin and T. G. Kolda, *Nonlinearly-constrained optimization using asynchronous parallel generating set search*, Tech. Rep. SAND2007-3257, Sandia National Laboratories, Albuquerque, NM and Livermore, CA, May 2007, submitted.

[13] N. J. Higham, *The matrix computation toolbox*, www.ma.man.ac.uk/˜higham/mctoolbox.

[14] P. D. Hough, T. G. Kolda, and V. J. Torczon, *Asynchronous parallel pattern search for nonlinear optimization*, SIAM J. Sci. Comput., 23 (2001), pp. 134–156.

[15] C. T. Kelley, *Users guide for imfil version* 0.5, available at www4.ncsu.edu/˜ctk/imfil.html.

[16] M. Marazzi and J. Nocedal, *Wedge trust region methods for derivative free optimization*, Math. Program., 91 (2002), pp. 289–305.

[17] R. Oeuvray and M. Bierlaire, *A new derivative-free algorithm for the medical image registration problem*, Int. J. Modelling and Simulation, 27 (2007), pp. 115–124.

[18] R. Oeuvray, *Trust-region methods based on radial basis functions with application to biomedical imaging*, Ph.D. thesis, EPFL, Lausanne, Switzerland, 2005.

[19] M. J. D. Powell, *UOBYQA: Unconstrained optimization by quadratic approximation*, Math. Program., 92 (2002), pp. 555–582.

[20] M. J. D. Powell, *The NEWUOA software for unconstrained optimization without derivatives*, in Large Scale Nonlinear Optimization, G. Di Pillo and M. Roma, eds., Springer, Netherlands, 2006, pp. 255–297.

[21] M. J. D. Powell, *Developments of NEWUOA for unconstrained minimization without derivatives*, Preprint DAMTP 2007/NA05, University of Cambridge, Cambridge, England, 2007.

[22] R. G. Regis and C. A. Shoemaker, *A stochastic radial basis function method for the global optimization of expensive functions*, INFORMS J. Comput., 19 (2007), pp. 457–509.