SIAM J. OPTIM.
Vol. 16, No. 4, pp. 1054–1075

A DERIVATIVE-FREE ALGORITHM FOR LINEARLY CONSTRAINED FINITE MINIMAX PROBLEMS*

G. LIUZZI†, S. LUCIDI†, AND M. SCIANDRONE‡

Abstract. In this paper we propose a new derivative-free algorithm for linearly constrained finite minimax problems. Due to the nonsmoothness of this class of problems, standard derivative-free algorithms can locate only points which satisfy weak necessary optimality conditions. In this work we define a new derivative-free algorithm which is globally convergent toward standard stationary points of the finite minimax problem. To this end, we convert the original problem into a smooth one by using a smoothing technique based on the exponential penalty function of Kort and Bertsekas. This technique depends on a smoothing parameter which controls the approximation to the finite minimax problem. The proposed method is based on a sampling of the smooth function along a suitable search direction and on a particular updating rule for the smoothing parameter that depends on the sampling stepsize. Numerical results on a set of standard minimax test problems are reported.

Key words. derivative-free optimization, linearly constrained finite minimax problems, nonsmooth optimization

AMS subject classifications. 90C56, 90C47, 65K05

DOI. 10.1137/040615821

1. Introduction.

Many problems of interest in real world applications can be modeled as finite minimax problems. This class of problems arises, for instance, in the solution of approximation problems, systems of nonlinear equations, nonlinear programming problems, and multiobjective problems. Many algorithms have been developed for the solution of finite minimax problems which require the knowledge of first- or second-order derivatives of the functions involved in the definition of the problem. Unfortunately, in some engineering applications, such as some of those arising in optimal design problems, the function values are obtained by direct measurements (which are often affected by numerical error or random noise) or are the result of complex simulation programs so that first-order derivatives cannot be explicitly calculated or approximated. Moreover, the nonsmoothness of the minimax problem does not allow us to employ some off-the-shelf derivative-free method, since most of these methods are based on a well-established convergence theory which, in order to guarantee convergence to a stationary point, requires first-order derivatives to be continuous, even though they cannot be computed. In particular, if the continuity assumption on the derivatives is relaxed, it is no longer possible to prove global convergence of the derivative-free method to a stationary point, but it is possible only to prove convergence towards a point where the (Clarke) generalized directional derivative is nonnegative with respect to every search direction explored by the algorithm

*Received by the editors September 27, 2004; accepted for publication (in revised form) August 12, 2005; published electronically February 3, 2006. This work was supported by MIUR-FIRB research project RBNE01WBBB "*Large-scale nonlinear optimization*," Rome, Italy.
http://www.siam.org/journals/siopt/16-4/61582.html

†Università di Roma "La Sapienza," Dipartimento di Informatica e Sistemistica "A. Ruberti," Via Buonarroti 12, 00185 Rome, Italy (liuzzi@dis.uniroma1.it, lucidi@dis.uniroma1.it).

‡Istituto di Analisi dei Sistemi ed Informatica del CNR, Viale Manzoni 30, 00185 Rome, Italy (sciandro@iasi.rm.cnr.it).

(see Appendix A for such a general result). Such points can be considered as weak stationary points in the sense that the (Clarke) generalized directional derivative can still be negative along some unexplored direction.

In this paper we consider a particular class of nonsmooth problems, namely, the problem of minimizing the maximum among a finite number of smooth functions. We recall that, for such a class of problems, the (Clarke) generalized directional derivative is proved to coincide with the directional derivative, but, also in this case, classical derivative-free codes can still be convergent toward weak stationary points (see [8] for a thorough discussion on this topic). Finite minimax problems have the valuable feature that they can be approximated by a smooth problem. This smooth approximation of the minimax problem can be achieved by using different techniques (see [19], [1], [2], [4], [7], [15], [17], [18], and [20]). In particular, we consider an approximation approach based on a so-called smoothing function which depends on a precision parameter (see [3], [16], and [11]). In order to define a solution method based on a smoothing technique, two different aspects, one computational and the other theoretical, must be considered. From a computational point of view, a trade-off should be found between the accuracy of the approximation and the problem of limiting the ill-conditioning due to the nonsmoothness of the minimax problem at the solutions. From a theoretical point of view, the algorithm should be guaranteed to converge a stationary point of the original minimax problem. In particular, a class of algorithms [16] for the solution of the minimax problem has been proposed, which takes into account the above two requirements. This is accomplished by using a feedback precision-adjustment rule which updates the precision parameter during the optimization process of the smoothing function. Roughly speaking, the idea behind the proposed updating rule is that of updating the parameter only when the minimization method has carried out a significant improvement. However, these updating rules are based upon the knowledge of the first derivatives of the problem.

In this paper we propose a derivative-free method which is based on a sampling of the smooth function along suitable search directions and on a particular updating rule for the smoothing parameter that depends on the sampling stepsize. We manage to prove convergence of the method to a stationary point of the minimax problem, while reducing the negative effects of the ill-conditioning that the smoothing approach incurs.

In section 2, we describe the minimax problem, its properties, and the smoothing function. In section 3, we report some convergence results for a general derivative-free approach to solve the minimax problem. In section 4, we report the proposed derivative-free algorithm and its convergence analysis. Finally, section 5 is devoted to some results of our method.

2. Problem, definition, and smooth approximation. In this paper we consider the solution of finite minimax problems where the variables are subject to linear inequality constraints. In particular, we consider problems of the form

$$\begin{aligned} &\min \quad f(x) \\ &s.t. \quad Ax \leq b, \end{aligned} \tag{1}$$

where $x \in \Re^n$, $b \in \Re^m$, $A \in \Re^{m \times n}$, and

$$f(x) = \max_{1 \leq i \leq q} f_i(x).$$

We indicate by $\mathcal{F}$ the *feasible set* of problem (1), namely,

$$\mathcal{F} = \{x \in \Re^n : Ax \leq b\}.$$

We require the following standard assumption to hold, which ensures that the level sets of $f(x)$ are compact.

ASSUMPTION 1. *The functions $f_i : \Re^n \to \Re$, $i = 1, \dots, q$, are twice continuously differentiable functions on $\Re^n$, and the function $f(x)$ is radially unbounded on the feasible set $\mathcal{F}$; that is, for every sequence $\{x_k\} \subset \mathcal{F}$ satisfying $\lim_{k\to\infty} \|x_k\| = +\infty$,*

$$\lim_{k\to\infty} f(x_k) = +\infty.$$

Note that, even though every function $f_i(x)$, $i = 1, \dots, q$, is twice continuously differentiable, we assume that their gradients can be neither calculated nor approximated explicitly.

We denote by $B(x)$ the following set of indices:

$$B(x) = \{i = 1, \dots, q : f_i(x) = f(x)\}. \tag{2}$$

For every feasible point $x \in \mathcal{F}$, we define the *set of indices of active constraints* by

$$I(x) = \{j = 1, \dots, m : a_j^T x = b_i\}, \tag{3}$$

and the *cone of feasible directions*

$$T(x) = \{d \in \Re^n : a_j^T d \leq 0,\ j \in I(x)\}, \tag{4}$$

where a_j^T, $j = 1, \dots, m$, denotes the jth row of the constraints matrix A. The directional derivative of the max function f at x in the direction $d \in \Re^n$ is given by (see, e.g., [3])

$$Df(x, d) = \max_{i \in B(x)} \{\nabla f_i(x)^T d\}.$$

We define $\bar{x} \in \mathcal{F}$ a stationary point of problem (1) if

$$Df(\bar{x}, d) \geq 0 \quad \forall d \in T(\bar{x}). \tag{5}$$

In particular, the following proposition shows a different characterization of the stationary points of problem (1).

PROPOSITION 1. *A point $\hat{x} \in \mathcal{F}$ is a stationary point of problem* (1) *if and only if there exist $\lambda_i \geq 0$, $i \in B(\hat{x})$, such that*

$$\sum_{i \in B(\hat{x})} \lambda_i = 1, \tag{6}$$

$$\left(\sum_{i \in B(\hat{x})} \lambda_i \nabla f_i(\hat{x}) \right)^T d \geq 0 \quad \forall d \in T(\hat{x}). \tag{7}$$

Proof. If $\hat{x} \in \mathcal{F}$ is a stationary point of problem (1), then there exists at least one index $j \in B(\hat{x})$ such that $\nabla f_j(\hat{x})^T d \geq 0$. Then conditions (6) and (7) hold with $\lambda_j = 1$ and $\lambda_i = 0$ for all $i \neq j$.

If $\hat{x} \in \mathcal{F}$ satisfies conditions (6) and (7), then for any $d \in T(\hat{x})$, we can write

$$0 \leq \left(\sum_{i \in B(\hat{x})} \lambda_i \nabla f_i(\hat{x}) \right)^T d \leq \max_{i \in B(\hat{x})} \nabla f_i(\hat{x})^T d,$$

which shows that $\bar{x}$ is a stationary point of problem (1). ◻

In order to find a stationary point of problem (1) we adopt a smoothing technique [3], [11], [16], [19] which consists of solving a sequence of smooth problems approximating the minimax problem in the limit. Let $\mu > 0$ be a smoothing parameter and define

$$f(x, \mu) = \mu \ln \sum_{i=1}^{q} \exp\left(\frac{f_i(x)}{\mu} \right),$$

which is sometimes referred to as an exponential penalty function [3]. An alternative expression for $f(x, \mu)$ is given by

$$f(x, \mu) = f(x) + \mu \ln \sum_{i=1}^{q} \exp\left(\frac{f_i(x) - f(x)}{\mu} \right).$$

We report some properties of $f(x, \mu)$ [19].

PROPOSITION 2. *Suppose $f_i(x)$, $i = 1, \ldots, q$, are twice continuously differentiable functions. Then*

(i) *$f(x, \mu)$ is increasing with respect to μ, and*

$$f(x) \leq f(x, \mu) \leq f(x) + \mu \ln q; \tag{8}$$

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

$$\nabla_x f(x, \mu) = \sum_{i=1}^{q} \lambda_i(x, \mu) \nabla f_i(x), \tag{9}$$

$$\begin{aligned} \nabla_x^2 f(x, \mu) = & \sum_{i=1}^{q} \left(\lambda_i(x, \mu) \nabla^2 f_i(x) + \frac{1}{\mu} \lambda_i(x, \mu) \nabla f_i(x) \nabla f_i(x)^T \right) \\ & - \frac{1}{\mu} \left(\sum_{i=1}^{q} \lambda_i(x, \mu) \nabla f_i(x) \right) \left(\sum_{i=1}^{q} \lambda_i(x, \mu) \nabla f_i(x) \right)^T, \end{aligned} \tag{10}$$

where

$$\lambda_i(x, \mu) = \frac{\exp(f_i(x)/\mu)}{\sum_{j=1}^{q} \exp(f_j(x)/\mu)} \in (0, 1), \qquad \sum_{i=1}^{q} \lambda_i(x, \mu) = 1. \tag{11}$$

3. Derivative-free convergence conditions. A derivative-free algorithm for problem (1) must account for two difficulties. The first is the nonsmoothness of problem (1). The second is that stationary points of problem (1), as stated by (5) and Proposition 1, are characterized by first-order derivatives of the component functions $f_i(x)$, $i = 1, \ldots, q$, which are not available.

In order to treat the nonsmoothness of problem (1), we employ the smooth approximating problem,

$$\min_{x \in \mathcal{F}} f(x, \mu), \tag{12}$$

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

In order to tackle the second difficulty of unavailable first derivatives, we try to obtain first-order information by sampling the objective function along a suitable set of search directions. Specifically, we follow the approach proposed in [13], which uses a set of search directions that positively span an "ϵ-approximation" of the cone of feasible directions or, in other words, the cone of feasible directions with respect to the ϵ-active constraints.

Formally, for any $\epsilon > 0$ and $x \in \mathcal{F}$, we define the set of indices of ϵ-active constraints by

$$I(x;\epsilon) = \{j : a_j^T x \geq b_j - \epsilon\}$$

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

$$T(x;\epsilon) = \{d \in \Re^n : a_j^T d \leq 0 \quad \forall j \in I(x;\epsilon)\}.$$

The following proposition (see [13]) describes some properties of sets $I(x;\epsilon)$ and $T(x;\epsilon)$.

PROPOSITION 3. *Let $\{x_k\}$ be a sequence of iterates converging towards a point $\bar{x} \in \mathcal{F}$. Then there exists a value $\epsilon^* > 0$ (depending on $\bar{x}$ only) such that for every $\epsilon \in (0, \epsilon^*]$ there exists $\bar{k}_\epsilon$ such that*

$$I(x_k;\epsilon) = I(\bar{x}), \tag{13}$$

$$T(x_k;\epsilon) = T(\bar{x}) \tag{14}$$

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

Proof. See the proof of Proposition 1 in [13]. □

The first step toward defining a derivative-free method for the solution of problem (12) 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 2. *Let $\{x_k\}$ be a sequence of feasible points and $\{D_k\}$ be a sequence of sets of search directions. Then, for all k,*

$$D_k = \{d_k^i : \|d_k^i\| = 1, \quad i = 1, \dots, r_k\},$$

and, for some constant $\bar{\epsilon} > 0$,

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

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

The proposition that follows states a general convergence result. In particular, it identifies sufficient conditions on the sampling of the smoothing function along the directions d_k^i, $i = 1, \dots, r_k$, and on the updating of the smoothing parameter, which will guarantee global convergence of the method to a stationary point of the original minimax problem (1).

PROPOSITION 4. *Let $\{x_k\}$ be a sequence of feasible points and $\bar{x}$ be a limit point of a subsequence $\{x_k\}_K$ for some infinite set $K \subseteq \{0, 1, \dots\}$. Let $\{D_k\}$, with $D_k = \{d_k^1, \dots, d_k^{r_k}\}$, be a sequence of sets of directions which satisfy Assumption 2 and $J_k = \{i \in \{1, \dots, r_k\} : d_k^i \in T(x_k, \epsilon)\}$ with $\epsilon \in (0, \min\{\bar{\epsilon}, \epsilon^\star\}]$, where $\bar{\epsilon}$ and $\epsilon^\star$ are defined in Assumption 2 and Proposition 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*

$$y_k^i + \xi_k^i d_k^i \in \mathcal{F}; \tag{15}$$

$$f(y_k^i + \xi_k^i d_k^i, \mu_k) \geq f(y_k^i, \mu_k) - o(\xi_k^i); \tag{16}$$

(ii) *furthermore,*

$$\lim_{k\to\infty, k\in K} \mu_k = 0; \tag{17}$$

$$\lim_{k\to\infty, k\in K} \frac{\max_{i\in J_k}\{\xi_k^i, \|x_k - y_k^i\|\}}{\mu_k} = 0. \tag{18}$$

Then $\bar{x}$ is a stationary point of the minimax problem (1).

Proof. By applying the mean-value theorem to (16), we can write

$$-o(\xi_k^i) \leq f(y_k^i + \xi_k^i d_k^i, \mu_k) - f(y_k^i, \mu_k) = \xi_k^i \nabla_x f(u_k^i, \mu_k)^T d_k^i, \qquad i \in J_k, \tag{19}$$

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

$$\begin{aligned} \xi_k^i \nabla_x f(u_k^i, \mu_k)^T d_k^i &= \xi_k^i \nabla_x f(x_k, \mu_k)^T d_k^i + \xi_k^i (u_k^i - x_k)^T \nabla_x^2 f(\tilde{u}_k^i, \mu_k) d_k^i \\ &\leq \xi_k^i \nabla_x f(x_k, \mu_k)^T d_k^i + \xi_k^i \|u_k^i - x_k\| \|\nabla_x^2 f(\tilde{u}_k^i, \mu_k) d_k^i\|, \end{aligned}$$

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 (10) of $\nabla_x^2 f(\tilde{u}_k^i, \mu_k)$ and the triangle inequality, we get that

$$\begin{aligned} &\xi_k^i \nabla_x f(u_k^i, \mu_k)^T d_k^i \leq \xi_k^i \nabla_x f(x_k, \mu_k)^T d_k^i \\ &+ \xi_k^i \|u_k^i - x_k\| \left\{ \left\| \sum_{j=1}^q \lambda_j(\tilde{u}_k^i, \mu_k) \nabla^2 f_j(u_k^i) d_k^i \right\| + \frac{1}{\mu_k} \left\| \sum_{j=1}^q \lambda_j(\tilde{u}_k^i, \mu_k) \nabla f_j(\tilde{u}_k^i) \nabla f_j(\tilde{u}_k^i)^T d_k^i \right. \right. \\ &- \left. \left. \left(\sum_{j=1}^q \lambda_j(\tilde{u}_k^i, \mu_k) \nabla f_j(\tilde{u}_k^i) \right) \left(\sum_{j=1}^q \lambda_j(\tilde{u}_k^i, \mu_k) \nabla f_j(\tilde{u}_k^i) \right)^T d_k^i \right\| \right\}. \end{aligned}$$

Since $\{x_k\}_K$ converges, it follows from Assumption 2 and (11) that, for all i and j, $\{x_k\}_K$, $\{\tilde{u}_k^i\}$, $\{\lambda_j(\tilde{u}_k^i, \mu_k)\}$, and $\{d_k^i\}$ are bounded sequences. Therefore, by Assumption 1, we can find constants c_1 and c_2 such that

(20)

$$-o(\xi_k^i) \leq \xi_k^i \nabla_x f(u_k^i, \mu_k)^T d_k^i \leq \xi_k^i \nabla_x f(x_k, \mu_k)^T d_k^i + \xi_k^i \|u_k^i - x_k\| \left(c_1 + \frac{1}{\mu_k} c_2 \right).$$

From (20) and (9), we obtain

$$\left(\sum_{j=1}^q \lambda_j(x_k, \mu_k) \nabla f_j(x_k) \right)^T d_k^i + \left(c_1 + \frac{1}{\mu_k} c_2 \right) \|u_k^i - x_k\| \geq -\frac{o(\xi_k^i)}{\xi_k^i}. \tag{21}$$

Since $\cup_{k\in K} D_k$ is a finite set by Assumption 2 and recalling the boundedness of each sequence $\{\lambda_j(x_k,\, \mu_k)\}$, $j = 1,\dots,q$, there exist an infinite set $\bar K \subseteq K$ and, given the fact that r_k is bounded, a finite set $J \subseteq \{1,2,\dots\}$ and $\bar d^j \in \Re^n$, $j \in J$, such that

$$\lim_{\substack{k\to\infty\\ k\in\bar K}} x_k = \bar x, \tag{22}$$

$$\lim_{\substack{k\to\infty\\ k\in\bar K}} \lambda_j(x_k,\mu_k) = \bar\lambda_j, \qquad j = 1,\dots,q, \tag{23}$$

$$J_k = J \quad \forall k \in \bar K, \tag{24}$$

$$d_k^j = \bar d^j \quad \forall j \in J \text{ and } \; k \in \bar K. \tag{25}$$

Moreover, recalling that $u_k^i = y_k^i + t_k^i \xi_k^i d_k^i$, with $t_k^i \in (0,1)$, we have that

$$\left(c_1 + \frac{1}{\mu_k} c_2\right) \|u_k^j - x_k\| \le \left(c_1 + \frac{1}{\mu_k} c_2\right) (\|y_k^j - x_k\| + \xi_k^j) \quad \forall j \in J,$$

which, by using (18), implies that

$$\lim_{\substack{k\to\infty\\ k\in\bar K}} \left(c_1 + \frac{1}{\mu_k} c_2\right) \|u_k^j - x_k\| = 0 \quad \forall j \in J. \tag{26}$$

We note that expression (11) can be rewritten as

$$\lambda_j(x,\mu) = \frac{\exp((f_j(x) - f(x))/\mu)}{\sum_{l=1}^q \exp((f_l(x) - f(x))/\mu)}, \qquad j = 1,\dots,q,$$

so that it is easily seen that

$$\begin{aligned} \bar\lambda_j \ge 0 \quad &\forall j, \\ \bar\lambda_j = 0 \quad &\forall j \notin B(\bar x). \end{aligned} \tag{27}$$

Furthermore, since $\sum_{j=1}^q \lambda_j(x_k,\mu_k) = 1$ for all k, then

$$\sum_{j=1}^q \bar\lambda_j = 1. \tag{28}$$

Now, recalling (26) and taking limits in (21) as $k \to \infty$, $k \in \bar K$, we obtain

$$\left(\sum_{j=1}^q \bar\lambda_j \nabla f_j(\bar x)\right)^T \bar d^i \ge 0 \quad \forall i \in J. \tag{29}$$

Now, Proposition 3 and Assumption 2 imply that, for $k \in K$,

$$T(\bar x) = T(x_k;\epsilon) = cone\{D_k \cap T(x_k;\epsilon)\} = cone\{d_k^i\}_{i\in J_k}. \tag{30}$$

Hence, by (30), (24), and (25) we have that

$$T(\bar x) = cone\{\bar d^i\}_{i\in J}, \tag{31}$$

so that, for every $d \in T(\bar{x})$, there exist $\beta_i \geq 0$, for all $i \in J$, such that

$$d = \sum_{i \in J} \beta_i \bar{d}^i. \tag{32}$$

Thus, we obtain from (29) and (32) that, for every $d \in T(\bar{x})$,

$$\left(\sum_{j=1}^{q} \bar{\lambda}_j \nabla f_j(\bar{x}) \right)^T d = \sum_{i \in J} \beta_i \left(\sum_{j=1}^{q} \bar{\lambda}_j \nabla f_j(\bar{x}) \right)^T \bar{d}^i \geq 0,$$

which, along with (27) and (28), proves the proposition (see Proposition 1). ∎

The above proposition is a nontrivial extension of similar results established in the context of derivative-free methods for smooth optimization (see, for instance, [13]). The major novelty of Proposition 4 is (18), which relates the convergence rate of the smoothing parameter with that of the sampling stepsizes. Indeed, Proposition 4 has two crucial aspects:

1. When $x_k \to \bar{x}$ and $\mu_k \to 0$, eventually,

$$\nabla_x f(x_k, \mu_k)^T d_k^i = \left(\sum_{j=1}^{q} \lambda_j(x_k, \mu_k) \nabla f_j(x_k) \right)^T d_k^i \geq 0 \quad \forall\, i \in J_k.$$

2. The bounded sequence $\{(\lambda_1(x_k, \mu_k), \dots, \lambda_q(x_k, \mu_k))\}$ has an accumulation point. This allows us to overcome the difficulty tied to the indefiniteness of $\nabla_x^2 f(x_k, \mu_k)$ in the limit.

The sampling of the smooth objective function along the directions d_k^i, $i \in J_k$, introduces a further difficulty, namely, that $\nabla_x f(x_k, \mu_k)^T d_k^i$ is approximated by the quantity

$$\nabla_x f(u_k^i, \mu_k)^T d_k^i = \left(\sum_{j=1}^{q} \lambda_j(u_k^i, \mu_k) \nabla f_j(u_k^i) \right)^T d_k^i,$$

where, for every index $i \in J_k$, we have different bounded sequences $\{(\lambda_1(u_k^i, \mu_k), \dots, \lambda_q(u_k^i, \mu_k))\}$. This raises the problem that each of these sequences converges to its own limit while the optimality condition (29) requires them to have the same limit point. In order to guarantee the existence of a unique limit point of the sequences $\{(\lambda_1(u_k^i, \mu_k), \dots, \lambda_q(u_k^i, \mu_k))\}$, for all $i \in J_k$, it is necessary that $\|u_k^i - x_k\|$, $i \in J_k$, tends to zero faster than μ_k, where $\|u_k^i - x_k\|$ can be viewed as a measure of the degree of approximation of first-order derivatives and μ_k gives a measure of the degree of approximation of the original minimax problem.

To conclude, we note that, since Proposition 4 poses only an upper bound on the convergence rate of μ_k towards zero, it allows us to choose an updating rule for the smoothing parameter which conciliates global convergence with the problem of avoiding the ill-conditioning of the smooth approximating problem.

4. A derivative-free method and global convergence result. In this section we define an algorithm for the solution of problem (1). The proposed method stems from the union of a derivative-free approach for smooth and linearly constrained optimization with a suitable handling of the smoothing parameter μ. In particular, the derivative-free method samples the smoothing function value along a finite set of search directions and decreases the sampling stepsize and the smoothing parameter if a sufficiently improved objective function value is not attained. The sampling strategy and the updating rule for the smoothing parameter are guided by the convergence conditions of Proposition 4. The derivative-free technique for sampling the smoothing function is based on the *feasible descent method* 2 proposed in [13] for a class of smooth optimization problems, including those with linear constraints. The formal description of the algorithm is reported below.

Algorithm DF

Data. $x_0 \in \mathcal{F}$, $init\_step_0 > 0$, $\mu_0 > 0$, $\gamma > 0$, $\theta \in (0,1)$, $\bar{\epsilon} > 0$.

Step 0. Set $k = 0$.

Step 1. *(Computation of search directions)*
Choose a set of directions $D_k = \{d_k^1, \dots, d_k^{r_k}\}$ satisfying Assumption 2.

Step 2. *(Minimization on the cone$\{D_k\}$)*

Step 2.1. *(Initialization)*
Set $i = 1$, $y_k^i = x_k$, $\tilde{\alpha}_k^i = init\_step_k$.

Step 2.2. *(Computation of the initial stepsize)*
Compute the maximum steplength $\bar{\alpha}_k^i$ such that $y_k^i + \bar{\alpha}_k^i d_k^i \in \mathcal{F}$ and set $\hat{\alpha}_k^i = \min\{\bar{\alpha}_k^i, \tilde{\alpha}_k^i\}$.

Step 2.3. *(Test on the search direction)*
If $\hat{\alpha}_k^i > 0$ and $f(y_k^i + \hat{\alpha}_k^i d_k^i, \mu_k) \le f(y_k^i, \mu_k) - \gamma (\hat{\alpha}_k^i)^2$,
compute α_k^i by the *Expansion Step*$(\bar{\alpha}_k^i, \hat{\alpha}_k^i, y_k^i, d_k^i; \alpha_k^i)$
and set $\tilde{\alpha}_k^{i+1} = \alpha_k^i$;
otherwise set $\alpha_k^i = 0$ and $\tilde{\alpha}_k^{i+1} = \theta \tilde{\alpha}_k^i$.

Step 2.4. *(New point)*
Set $y_k^{i+1} = y_k^i + \alpha_k^i d_k^i$.

Step 2.5 *(Test on the minimization on the cone*$\{D_k\}$)
If $i = r_k$, go to Step 3;
otherwise set $i = i + 1$ and go to Step 2.2.

Step 3. *(Main iteration)*
Find $x_{k+1} \in \mathcal{F}$ such that $f(x_{k+1}, \mu_k) \le f(y_k^{i+1}, \mu_k)$;
otherwise, set $x_{k+1} = y_k^{i+1}$.
Set $init\_step_{k+1} = \tilde{\alpha}_k^{i+1}$,
choose $\mu_{k+1} = \min\left\{\mu_k, \max_{i=1,\dots,r_k} \{(\tilde{\alpha}_k^i)^{1/2}, (\alpha_k^i)^{1/2}\}\right\}$,
set $k = k + 1$, and go to Step 1.

Expansion Step $(\bar{\alpha}_k^i, \hat{\alpha}_k^i, y_k^i, d_k^i; \alpha_k^i)$.

Data. $\gamma > 0$, $\delta \in (0,1)$.

Step 1. Set $\alpha = \hat{\alpha}_k^i$.

Step 2. Let $\tilde{\alpha} = \min\{\bar{\alpha}_k^i, (\alpha/\delta)\}$.

Step 3. If $\alpha = \bar{\alpha}_k^i$ or $f(y_k^i + \tilde{\alpha} d_k^i, \mu_k) > f(y_k^i, \mu_k) - \gamma \tilde{\alpha}^2$, set $\alpha_k^i = \alpha$ and return.

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

At Step 1 a suitable set of search directions $d_k^1, \dots, d_k^{r_k}$ is determined. At Step 2 the behavior of the objective function is evaluated along each search direction. In particular, if the search direction is feasible and is of sufficient decrease, the behavior of the objective function along this direction is further investigated by executing an *Expansion Step* until a suitable decrease is no longer obtained or the trial point reaches the boundary of the feasible region.

We indicate by $init\_step_k$ the initial stepsize at iteration k, and, for every direction d_k^i, with $i = 1, \dots, r_k$, we denote

- by $\tilde{\alpha}_k^i$ the candidate initial stepsize;
- by $\bar{\alpha}_k^i$ the maximum feasible stepsize;
- by $\hat{\alpha}_k^i$ the initial stepsize;
- by α_k^i the stepsize actually taken.

At Step 3 the new point x_{k+1} can be the point y_k^{i+1} produced by Steps 1–2 or any point where the objective function is improved with respect to $f(y_k^{r_k}, \mu_k)$. This fact allows us to adopt any approximation scheme for the objective function to produce a new better point. This flexibility can be particularly useful when the evaluation of objective function is computationally expensive, in which case the objective function values produced in previous iterations can be used to build an inexpensive model of $f(x)$ to be minimized with the aim of producing a potentially good point x_{k+1}. However, we note that this option can be discarded simply by setting $x_{k+1} = y_k^{i+1}$.

Then the smoothing parameter μ_k is reduced whenever $\max_{i=1,\dots,r_k}\{(\tilde{\alpha}_k^i)^{1/2}, (\alpha_k^i)^{1/2}\}$ gets smaller than the current smoothing value μ_k. We recall that $\max_{i=1,\dots,r_k}\{(\tilde{\alpha}_k^i)^{1/2}, (\alpha_k^i)^{1/2}\}$ can be viewed as a stationarity measure of the current iterate (see [9], for example). Thus, according to the updating rule, the smoothing parameter is reduced whenever a more precise approximation of a stationary point of the smoothing function is obtained.

The following proposition describes some key properties of certain sequences generated by Algorithm DF.

PROPOSITION 5. *Let $\{x_k\}$, $\{\mu_k\}$ be the sequences generated by Algorithm DF. Then*

(i) *$\{x_k\}$ is well-defined;*
(ii) *the sequence $\{f(x_k, \mu_k)\}$ is monotonically nonincreasing;*
(iii) *the sequence $\{x_k\}$ is bounded;*
(iv) *every cluster point of $\{x_k\}$ belongs to $\mathcal{F}$;*
(v) *the sequences $\{f(x_{k+1}, \mu_k)\}$ and $\{f(x_k, \mu_k)\}$ are both convergent and have the same limit.*

Proof. To prove assertion (i), it suffices to show that the Expansion Step, when performed along a direction d_k^i from y_k^i, for $i \in \{1, \dots, r_k\}$, terminates in a finite number $\bar{\jmath}$ of steps either because $\delta^{-\bar{\jmath}}\hat{\alpha}_k^i \geq \bar{\alpha}_k^i$ or because $f(y_k^i + \delta^{-\bar{\jmath}}\hat{\alpha}_k^i d_k^i, \mu_k) > f(y_k^i, \mu_k) - \gamma\big(\delta^{-\bar{\jmath}}\hat{\alpha}_k^i\big)^2$.

If this were not true, we would have for some k and $i \in \{1, \dots, r_k\}$ that $\hat{\alpha}_k^i > 0$ and

$$\delta^{-j}\hat{\alpha}_k^i < \bar{\alpha}_k^i, \qquad y_k^i + \delta^{-j}\hat{\alpha}_k^i d_k^i \in \mathcal{F},$$
$$f(y_k^i + \delta^{-j}\hat{\alpha}_k^i d_k^i, \mu_k) \leq f(y_k^i, \mu_k) - \gamma\big(\delta^{-j}\hat{\alpha}_k^i\big)^2$$

for all $j = 0, 1, \dots$. But by (i) of Proposition 2,

$$f(y_k^i + \delta^{-j}\hat{\alpha}_k^i d_k^i) \leq f(y_k^i + \delta^{-j}\hat{\alpha}_k^i d_k^i, \mu_k) \leq f(y_k^i, \mu_k) - \gamma\big(\delta^{-j}\hat{\alpha}_k^i\big)^2,$$

for all $j = 0, 1, \dots$, which, since δ^{-j} is unbounded, violates Assumption 1.

To prove assertion (ii), we note that the instructions of the algorithm imply that

$$f(x_{k+1}, \mu_k) \leq f(x_k, \mu_k).$$

Since $\mu_{k+1} \leq \mu_k$ and $f(x, \mu)$ is increasing with respect to μ (see (i) of Proposition 2), we have

$$f(x_{k+1}, \mu_{k+1}) \leq f(x_{k+1}, \mu_k) \leq f(x_k, \mu_k), \tag{33}$$

so that assertion (ii) is proved.

By assertion (ii) we have for all k that $f(x_k, \mu_k) \leq f(x_0, \mu_0)$, and hence

$$x_k \in \{x \mid f(x, \mu_k) \leq f(x_0, \mu_0)\}.$$

Then for any x satisfying

$$f(x, \mu_k) \leq f(x_0, \mu_0)$$

we have from (i) of Proposition 2 that

$$f(x) \leq f(x_0, \mu_0).$$

Thus we can write

$$x_k \in \{x \mid f(x, \mu_k) \leq f(x_0, \mu_0)\} \subseteq \{x \mid f(x) \leq f(x_0, \mu_0)\}.$$

It follows from Assumption 1 that the set $\{x \mid f(x) \leq f(x_0, \mu_0)\}$ is bounded, which proves assertion (iii).

To prove assertion (iv), we note that the instructions of Algorithm DF imply that $x_k \in \mathcal{F}$ for all k. Since $\mathcal{F}$ is a closed set, the assertion follows.

To prove point (v), we note that, by Assumption 1, $f(x)$ is bounded from below on the feasible set $\mathcal{F}$. Therefore, by recalling (8), we have that $\{f(x_k, \mu_k)\}$ is also bounded below, and hence, by point (ii), convergent. From (33), we have that $\{f(x_{k+1}, \mu_k)\}$ converges to the same limit of $\{f(x_k, \mu_k)\}$, which proves assertion (v). ∎

The proposition that follows establishes some results concerning the adopted sampling technique. In particular, point (i) guarantees that the sampling points tend to cluster more and more. Point (ii) ensures the existence of sufficiently large stepsizes providing feasible points along the search directions.

PROPOSITION 6. *Let $\{x_k\}$ be the sequence produced by Algorithm DF. Then we have*

(i)

$$\lim_{k\to\infty} \max_{1\leq i\leq r_k} \{\alpha_k^i\} = 0, \tag{34}$$

$$\lim_{k\to\infty} \max_{1\leq i\leq r_k} \{\tilde{\alpha}_k^i\} = 0, \tag{35}$$

$$\lim_{k\to\infty} \max_{1\leq i\leq r_k} \|x_k - y_k^i\| = 0. \tag{36}$$

(ii) $\bar{\alpha}_k^i \geq \epsilon/c - \|x_k - y_k^i\|$ *whenever* $d_k^i \in T(x_k, \epsilon)$ *and* $\epsilon > 0$*, where* $c = \max_{j=1,\dots,m} \|a_j\|$.

Proof. To prove assertion (i), we note from the construction of α_k^i and y_k^{i+1} in Step 2.3 that

$$f(y_k^{i+1}, \mu_k) \leq f(y_k^i, \mu_k) - \gamma(\alpha_k^i)^2 \tag{37}$$

and from the construction of $\tilde{\alpha}_k^{i+1}$ that for each k and each $i \in \{1, \ldots, r_k\}$, one of the following holds:

$$\tilde{\alpha}_k^{i+1} = \alpha_k^i, \tag{38}$$

$$\tilde{\alpha}_k^{i+1} = \theta\tilde{\alpha}_k^i. \tag{39}$$

Summing (37) for $i = 1, \ldots, r_k$ and using the construction of x_{k+1} in Step 3 yields

$$f(x_{k+1}, \mu_k) \leq f(x_k, \mu_k) - \gamma \sum_{i=1}^{r_k} (\alpha_k^i)^2.$$

Recalling point (v) of Proposition 5, $\{f(x_k, \mu_k)\}$ and $\{f(x_{k+1}, \mu_k)\}$ are both convergent and have the same limit, and $\{\sum_{i=1}^{r_k}(\alpha_k^i)^2\} \to 0$, thus proving (34).

For all k we have

$$\tilde{\alpha}_k^i = (\theta)^{p_k^i}\ \alpha_{m_k^i}^{l_k^i}, \tag{40}$$

where $m_k^i \leq k$ and $l_k^i \leq r_{m_k^i}$ are, respectively, the largest iteration index and the largest direction index such that (38) holds, and the exponent p_k^i is given by

$$p_k^i = \begin{cases} i - l_k^i & \text{if } m_k^i = k, \\ i + r_{k-1} + r_{k-2} + \cdots + r_{m_k^i} - l_k^i & \text{otherwise.} \end{cases} \tag{41}$$

Then let i be an arbitrary integer such that the set $K^i = \{k \in \{0, 1, \ldots\} : r_k \geq i\}$ has infinitely many elements. If $m_k^i \to \infty$, as $k \to \infty$ with $k \in K^i$, then, by (40) and (34), we get (35).

On the other hand, suppose that m_k^i is bounded above. In this case, for all $k \in K^i$ sufficiently large, $m_k^i < k$, so that p_k^i is given by the second part of (41). Since $r_{m_k^i} \geq l_k^i$ and $r_l \geq 1$ for $l = m_k^i + 1, \ldots, k-1$, this then implies that $p_k^i \geq i + (k - 1 - m_k^i)$, so that $p_k^i \to \infty$ as $k \to \infty$, $k \in K^i$. Hence, by (40) and $\theta \in (0, 1)$ we get (35).

Then we note from the updating formula for y_k^i in Step 2.4 that

$$x_k - y_k^i = -\sum_{l=1}^{i-1} \alpha_k^l d_k^l.$$

Then, using (34), $\|d_k^l\| = 1$ for $1 \leq l \leq r_k$, $i \leq r_k$, and the assumption that $\{r_k\}$ is bounded, we obtain (36).

To prove assertion (ii), we note that, by the fact that $d_k^i \in T(x_k; \epsilon)$ and by the definition of $\bar{\alpha}_k^i$ in Step 2.2, either $\bar{\alpha}_k^i = +\infty$ (in which case, the result is proved) or an index $\bar{\jmath} \notin I(x_k; \epsilon)$ exists such that

$$a_{\bar{\jmath}}^T(y_k^i + \bar{\alpha}_k^i d_k^i) = b_{\bar{\jmath}}.$$

In the latter case, solving for $\bar{\alpha}_k^i$ and using $0 < a_{\bar{\jmath}}^T d_k^i \le c$ (where $c = \max_{j=1,\dots,m} \|a_j\|$) yields

$$\begin{aligned}
\bar{\alpha}_k^i &= \left(b_{\bar{\jmath}} - a_{\bar{\jmath}}^T y_k^i\right)/(a_{\bar{\jmath}}^T d_k^i) \\
&\ge \left(b_{\bar{\jmath}} - a_{\bar{\jmath}}^T y_k^i\right)/c \\
&= \left(b_{\bar{\jmath}} - a_{\bar{\jmath}}^T x_k + a_{\bar{\jmath}}^T (x_k - y_k^i)\right)/c \\
&\ge \left(\epsilon + a_{\bar{\jmath}}^T (x_k - y_k^i)\right)/c \\
&\ge \left(\epsilon - \|x_k - y_k^i\| c\right)/c,
\end{aligned}$$

where the second inequality follows from $\bar{\jmath} \notin I(x_k;\epsilon)$ and the definition of $I(x_k;\epsilon)$, so that the assertion is proved. ◻

The next proposition establishes the convergence properties of Algorithm DF.

THEOREM 1. *Let $\{x_k\}$ be the sequence generated by Algorithm DF. Then a limit point of the sequence $\{x_k\}$ exists which is a stationary point of the minimax problem* (1).

Proof. By applying the results of Proposition 6 to Step 3 of the algorithm, we get that

$$\lim_{k\to\infty} \mu_k = 0. \tag{42}$$

Let $\{x_k\}_K$ be the subsequence corresponding to the subset of indices

$$K = \{k : \mu_{k+1} < \mu_k\}, \tag{43}$$

which, due to (42), has infinitely many elements.

Now let $\bar{x}$ be an accumulation point of the subsequence $\{x_k\}_K$, and let $\epsilon \in (0, \min\{\bar{\epsilon},\ \epsilon^\star\}]$, where $\bar{\epsilon}$ and $\epsilon^\star$ are defined in Algorithm DF and Proposition 3, respectively. Let

$$J_k = \{i \in \{1,\dots,r_k\} : d_k^i \in D_k \cap T(x_k,\epsilon)\}.$$

Then Proposition 3 and Assumption 2 imply that, for $k \in K$,

$$T(\bar{x}) = T(x_k;\epsilon) = cone\{D_k \cap T(x_k;\epsilon)\} = cone\{d_k^i\}_{i\in J_k}. \tag{44}$$

For all $i \in J_k$, by definition, $d_k^i \in T(x_k;\epsilon)$ so that from point (ii) of Proposition 6 we get

$$\bar{\alpha}_k^i \ge \epsilon/c - \left\|x_k - y_k^i\right\|,$$

which, by point (i) of Proposition 6, implies that there exists an index $\bar{k}$ such that, for all $k \ge \bar{k}$ and $k \in K$,

$$\alpha_k^i/\delta < \bar{\alpha}_k^i \quad \text{and} \quad \hat{\alpha}_k^i = \min\{\bar{\alpha}_k^i, \tilde{\alpha}_k^i\} = \tilde{\alpha}_k^i < \bar{\alpha}_k^i. \tag{45}$$

Then the construction of α_k^i in Step 2.3 implies that, for each $i \in J_k$, either

$$y_k^i + \frac{\alpha_k^i}{\delta} d_k^i \in \mathcal{F}, \qquad f\left(y_k^i + \frac{\alpha_k^i}{\delta} d_k^i, \mu_k\right) > f(y_k^i, \mu_k) - \gamma \left(\frac{\alpha_k^i}{\delta}\right)^2,$$

if an Expansion Step is performed, or

$$y_k^i + \hat{\alpha}_k^i d_k^i \in \mathcal{F}, \qquad f(y_k^i + \hat{\alpha}_k^i d_k^i, \mu_k) > f(y_k^i, \mu_k) - \gamma(\hat{\alpha}_k^i)^2.$$

By letting $\xi_k^i = \alpha_k^i/\delta$ in the first case and $\xi_k^i = \hat{\alpha}_k^i$ in the second case, we have

$$f(y_k^i + \xi_k^i d_k^i, \mu_k) > f(y_k^i, \mu_k) - \gamma(\xi_k^i)^2. \tag{46}$$

From the updating formula for y_k^i in Step 2.4 of Algorithm DF, we note that

$$\|y_k^i - x_k\| \le \sum_{l=1}^{i-1} \alpha_k^l \le \delta \sum_{l=1}^{i-1} \xi_k^l \le \delta r_k \max_{j \in J_k}\{\xi_k^j\}, \tag{47}$$

from which we get

$$\max_{i \in J_k}\{\xi_k^i, \|x_k - y_k^i\|\} \le \max\{1, \delta r_k\} \max_{i \in J_k}\{\xi_k^i\}. \tag{48}$$

Since $r_k \ge 1$, $\delta \in (0,1)$, and, by definition of ξ_k^i, $\max_{i \in J_k}\{\xi_k^i\} \le \max_{i \in J_k}\{\tilde{\alpha}_k^i, \alpha_k^i/\delta\}$, we have

$$\max\{1, \delta r_k\} \max_{i \in J_k}\{\xi_k^i\} \le \frac{r_k}{\delta} \max_{i \in J_k}\{\tilde{\alpha}_k^i, \alpha_k^i\}. \tag{49}$$

Recalling the definition of K (see (43)), it follows from Step 3 of Algorithm DF that

$$\mu_k^2 > \max_{j=1,\dots,r_k} \left\{\tilde{\alpha}_k^j, \alpha_k^j\right\} = \mu_{k+1}^2, \tag{50}$$

so that, by (48), (49), and (50), we obtain $\max_{i \in J_k}\{\xi_k^i, \|x_k - y_k^i\|\} < \frac{r_k}{\delta}\mu_k^2$, from which we get

$$\lim_{k \to \infty, k \in K} \frac{\max_{i \in J_k}\{\xi_k^i, \|x_k - y_k^i\|\}}{\mu_k} = 0. \tag{51}$$

Finally, (42), (46), (51), and the result of Proposition 4 conclude the proof. □

COROLLARY 1. *Let $\{x_k\}$ be the sequence produced by Algorithm DF, and let $\{x_k\}_K$ be the subsequence corresponding to the subset of indices K such that*

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

Then every accumulation point of $\{x_k\}_K$ is a stationary point of the minimax problem (1).

5. Numerical results. The aim of the computational experiments is to investigate the ability of the proposed algorithm to locate a good approximation to a solution of the finite minimax problem (1). We report numerical results obtained by Algorithm DF both on a set of 33 unconstrained minimax problems with $n \in [1, 200]$ and $q \in [2, 501]$ (see [6] and [16] for a description of these problems) and on a set of 5 linearly constrained minimax problems with $n \in [2, 20]$, $q \in [3, 14]$, and $m \in [1, 4]$ (see [14] for a description of these test problems). We used as starting points those reported in [6], [16], and [14].

Parameter values used in the algorithm were chosen as follows:

$$\begin{aligned} \mathit{init\_step}_0 = 1.0, \mu_0 = 1.0, \quad \gamma = 10^{-6}, \\ \theta = 0.5, \delta = 0.5, \quad \bar{\epsilon} = 1.0\,. \end{aligned}$$

As for the search directions, in the linearly constrained setting we use the computation strategy proposed in [10], whereas, in the unconstrained case, we use $D_k = D = \{\pm e_1, \dots, \pm e_n\}$. In the latter case, we further exploit the fact that D_k is constant. First, we modify Step 2 by adopting the stepsize updating strategy proposed in [12], in which each search direction e_i, $i = 1, \dots, n$, has its own associated stepsize. Furthermore, in Step 3 a point $\hat{x}$ is computed by performing a linesearch along an additional direction described at Step 4 of Algorithm 3 in [12]. Then $x_{k+1} = \hat{x}$, provided that $f(\hat{x}, \mu_k) \le f(y_k^{i+1}, \mu_k)$; otherwise, we set $x_{k+1} = y_k^{i+1}$. We note that in the linearly constrained case we always set $x_{k+1} = y_k^{i+1}$.

For the stopping condition, we choose to stop the algorithm when $\mathit{init\_step}_k \le 10^{-4}$ in the constrained case and when $\max_{i=1,\dots,n} \tilde{\alpha}_k^i \le 10^{-4}$ in the unconstrained case. Furthermore, we also stop the computation whenever the code reaches a total of 50000 function evaluations.

Table 1 shows the numerical results obtained by Algorithm DF. The table reports, for each problem, its name, number n of variables, number q of component functions, number m of linear constraints, and number nF of function evaluations required to satisfy the stopping condition. We denote by $f(\bar{x})$ the minimum value obtained by Algorithm DF, by $\bar{\mu}$ the value of the smoothing parameter when the stopping condition is met, and by $f^\star$ the value of the known solution. Furthermore, we denote by

$$\Delta = \frac{f(\bar{x}) - f^\star}{1 + |f^\star|}$$

the error at the solution obtained by Algorithm DF.

The results reported in Table 1 show that Algorithm DF is able to locate a good estimate of the minimum point of the minimax problem (1) (as reported in [14] and [16]) with a limited number of function evaluations, especially for problems with a reasonably small number of variables (e.g., fewer than 10). It is worth noting that for almost every problem, the final smoothing parameter value is of order 10^{-2} or less.

In order to better point out the efficiency of the proposed approach, we compare Algorithm DF with some reasonable modifications of it. First, it seems reasonable to test a modified version of Algorithm DF, which we call $\mathrm{DF}_{\mathtt{mod1}}$, that always uses the max function $f(x)$ instead of the smooth approximation $f(x, \mu)$. This helps us to evaluate the computational benefit of our method, with its first-order stationary result, versus a modification that possesses a much weaker convergence property, as shown in Appendix A. Second, in order to judge the effectiveness of the updating rule for the smoothing parameter, we choose to compare Algorithm DF with Algorithms $\mathrm{DF}_{\mathtt{mod2}}$ and $\mathrm{DF}_{\mathtt{mod3}}$, which can be obtained from Algorithm DF by dropping the updating rule for μ at Step 3 and choosing $\mu_0 = 1$ and $\mu_0 = 10^{-2}$, respectively.

The complete results obtained by the three modified versions of Algorithm DF ($\mathrm{DF}_{\mathtt{mod1}}$, $\mathrm{DF}_{\mathtt{mod2}}$, and $\mathrm{DF}_{\mathtt{mod3}}$) are reported in Appendix B. Here, for the sake of clarity, we report only a summary of the obtained results. For each algorithm, Table 2 indicates how many problems were solved to within the accuracy specified by the column labels, while Table 3 reports the number of function evaluations. In particular, for every pair of algorithms (DF, $\mathrm{DF}_{\mathtt{modi}}$, $i = 1, 2, 3$), we identify those problems solved

Table 1
Numerical performance of Algorithm DF.

PROBLEM	n	q	m	nF	$f(\bar{x})$	$\bar{\mu}$	$f^\star$	Δ
crescent	2	2	0	160	3.061E-03	1.105E-02	0.000E+00	3.061E-03
polak 1	2	2	0	106	2.718E+00	7.812E-03	2.718E+00	7.654E-09
lq	2	2	0	343	-1.411E+00	7.812E-03	-1.414E+00	1.158E-03
mifflin 1	2	2	0	65	-1.000E+00	1.210E-02	-1.000E+00	0.000E+00
mifflin 2	2	2	0	188	-9.980E-01	7.813E-03	-1.000E+00	1.009E-03
char.-conn 1	2	3	0	118	1.954E+00	9.882E-03	1.952E+00	4.631E-04
char.-conn 2	2	3	0	208	2.003E+00	1.105E-02	2.000E+00	1.153E-03
demy-malo	2	3	0	84	-3.000E+00	1.105E-02	-3.000E+00	0.000E+00
ql	2	3	0	132	7.203E+00	1.105E-02	7.200E+00	3.575E-04
hald-mad. 1	2	4	0	170	1.582E-02	1.105E-02	0.000E+00	1.582E-02
rosen	4	4	0	368	-4.394E+01	7.906E-03	-4.400E+01	1.347E-03
hald-mad. 2	5	42	0	471	6.177E-03	7.906E-03	1.220E-04	6.055E-03
polak 2	10	2	0	285	5.460E+01	7.813E-03	5.459E+01	1.134E-04
maxq	20	20	0	1858	0.000E+00	1.105E-02	0.000E+00	0.000E+00
maxl	20	40	0	891	0.000E+00	1.105E-02	0.000E+00	0.000E+00
goffin	50	50	0	2045	0.000E+00	7.813E-03	0.000E+00	0.000E+00
polak 6.1	2	3	0	131	1.954E+00	1.118E-02	1.952E+00	4.760E-04
polak 6.2	20	20	0	692	2.384E-09	1.105E-02	0.000E+00	2.384E-09
polak 6.3	4	50	0	2055	6.253E-03	7.813E-03	2.637E-03	3.607E-03
polak 6.4	4	102	0	1105	9.166E-03	7.813E-03	2.650E-03	6.499E-03
polak 6.5	4	202	0	1890	9.181E-03	7.813E-03	2.650E-03	6.515E-03
polak 6.6	3	50	0	374	6.531E-03	7.813E-03	4.500E-03	2.022E-03
polak 6.7	3	102	0	335	7.141E-03	7.813E-03	4.505E-03	2.624E-03
polak 6.8	3	202	0	369	7.263E-03	7.813E-03	4.505E-03	2.746E-03
polak 6.9	2	2	0	91	1.162E-01	7.812E-03	0.000E+00	1.162E-01
polak 6.10	1	25	0	129	1.784E-01	1.105E-02	1.782E-01	1.625E-04
polak 6.11	1	51	0	136	1.784E-01	1.105E-02	1.783E-01	6.206E-05
polak 6.12	1	101	0	153	1.784E-01	1.105E-02	1.784E-01	2.368E-05
polak 6.13	1	501	0	153	1.784E-01	1.105E-02	1.784E-01	1.464E-05
polak 6.14	100	100	0	3452	3.433E-09	1.105E-02	0.000E+00	3.433E-09
polak 6.15	200	200	0	6891	3.433E-09	1.105E-02	0.000E+00	3.433E-09
polak 6.16	100	50	0	3452	5.364E-09	1.105E-02	0.000E+00	5.364E-09
polak 6.17	200	50	0	7233	1.023E-08	7.812E-03	0.000E+00	1.023E-08
mad 1	2	3	1	43	-3.896E-01	1.747E-02	-3.897E-01	5.878E-05
mad 2	2	3	1	42	-3.304E-01	1.353E-02	-3.304E-01	-9.735E-10
mad 4	2	3	2	72	-4.489E-01	1.562E-02	-4.489E-01	4.601E-07
wong 2	10	6	3	236	2.522E+01	1.948E-02	2.431E+01	3.609E-02
wong 3	20	14	4	451	1.076E+02	2.545E-02	1.337E+02	-1.938E-01

Table 2
Comparison of methods: Number of problems solved to within a given accuracy.

	$\Delta < 10^{-3}$	$10^{-3} \leq \Delta < 10^{-1}$	$\Delta \geq 10^{-1}$
DF	23	14	1
$\mathrm{DF}_{\mathtt{mod1}}$	14	12	12
$\mathrm{DF}_{\mathtt{mod2}}$	16	16	6
$\mathrm{DF}_{\mathtt{mod3}}$	21	14	3

TABLE 3
Comparison of methods: Cumulative number of function evaluations to solve the same problems to within a given accuracy.

	$\Delta < 10^{-3}$	$10^{-3} \leq \Delta < 10^{-1}$	$\Delta \geq 10^{-1}$
DF	3649	947	91
$\text{DF}_{\texttt{mod1}}$	3645	703	88
DF	27662	8112	91
$\text{DF}_{\texttt{mod2}}$	27662	7736	91
DF	7586	7716	91
$\text{DF}_{\texttt{mod3}}$	22164	8252	88

with the same accuracy both by DF and $\text{DF}_{\texttt{modi}}$ and compare the sum of the required number of function evaluations.

From these results, it is clear that Algorithm DF outperformed Algorithms $\text{DF}_{\texttt{mod1}}$ and $\text{DF}_{\texttt{mod2}}$. In fact, DF solved to high accuracy ($\Delta < 10^{-3}$) a larger number of problems with a comparable number of function evaluations. Furthermore, the comparison between Algorithms DF and $\text{DF}_{\texttt{mod1}}$, in terms of number of failures ($\Delta \geq 10^{-1}$), shows the computational advantage of using an algorithm with stronger convergence properties. As for method $\text{DF}_{\texttt{mod3}}$, it has two failures ($\Delta \geq 10^{-1}$) more than DF, but it still performs well and seems to exhibit a behavior quite similar to that of DF. However, as seen in Table 3, the two algorithms perform quite differently in terms of function evaluations. This difference in performance properly points out the fundamental importance of the updating rule for the smoothing parameter μ, whose ultimate task is that of limiting the ill-conditioning of the approximating problem. Indeed, when we fix the smoothing parameter to 10^{-2}, the problem is too ill-conditioned from the beginning of the solution process. On the other hand, Algorithm DF limits the possible ill-conditioning by decreasing the smoothing parameter at a suitable rate.

6. Appendix A.

A function $f : \Re^n \to \Re$ is said to be locally *Lipschitz* near a point $x \in \Re^n$ if there exist $L > 0$ and $\delta > 0$ such that

$$|f(y_1) - f(y_2)| \leq L\|y_1 - y_2\|$$

for all y_1, y_2 belonging to the open ball $\{y \in \Re^n : \|y - x\| < \delta\}$. The (Clarke) *generalized directional derivative* [5] of f at x in the direction d is denoted by $f^\circ(x; d)$ and is defined as follows:

$$f^\circ(x; d) = \limsup_{y \to x, t \to 0^+} \frac{f(y + td) - f(y)}{t}. \tag{52}$$

Under the assumption that f is Lipschitz near x, $f^\circ(x; d)$ is well defined.

The following proposition extends to Lipschitz continuous functions analogous results reported in [12] and [13] concerning general convergence conditions for derivative-free methods.

PROPOSITION 7. *Let $\{x_k\}$ be a sequence of feasible points, $\bar{x}$ be a limit point of a subsequence $\{x_k\}_K$ for some infinite set $K \subseteq \{0, 1, \ldots\}$, and $f(x)$ be locally Lipschitz near $\bar{x}$. Let $\{D_k\}$, with $D_k = \{d_k^1, \ldots, d_k^{r_k}\}$, be a sequence of sets of directions which satisfy Assumption* 2 *and $J_k = \{i \in \{1, \ldots, r_k\} : d_k^i \in T(x_k, \epsilon)\}$ with $\epsilon \in (0, \min\{\bar{\epsilon}, \epsilon^\star\}]$ (where $\bar{\epsilon}$ and $\epsilon^\star$ are defined in Assumption* 2 *and Proposition* 3, *respectively).*

Suppose that the following condition holds: for each $k \in K$ and $i \in J_k$, there exist y_k^i and scalars $\xi_k^i > 0$ such that

$$y_k^i + \xi_k^i d_k^i \in \mathcal{F}; \tag{53}$$

$$f(y_k^i + \xi_k^i d_k^i) \geq f(y_k^i) - o(\xi_k^i); \tag{54}$$

$$\lim_{k\to\infty, k\in K} \max_{i\in J_k}\{\xi_k^i\} = 0; \tag{55}$$

$$\lim_{k\to\infty} \max_{i\in J_k} \|x_k - y_k^i\| = 0. \tag{56}$$

Then

$$\lim_{k\to\infty, k\in K} \min_{i\in J_k} \left\{\min\{0, f^\circ(x_k; d_k^i)\}\right\} = 0. \tag{57}$$

Proof. Since $\bigcup_{k\in K} D_k$ is a finite set, there exist infinite subsets $K_1 \subseteq K$ and $J \subset \{1, 2, \dots\}$ and a positive integer r such that

$$J_k = J \quad \forall k \in K_1,$$

$$\{d_k^i\}_{i\in J_k} = \{\bar{d}^1, \dots, \bar{d}^r\}, \qquad \|\bar{d}^i\| = 1 \quad \forall k \in K_1.$$

By using condition (56) it follows that

$$\lim_{k\to\infty, k\in K_1} y_k^i = \bar{x}, \qquad i \in J. \tag{58}$$

Now, recalling condition (54), for all $k \in K_1$, we have

$$f(y_k^i + \xi_k^i \bar{d}^i) - f(y_k^i) \geq -o(\xi_k^i), \qquad i \in J, \tag{59}$$

from which we obtain

$$\limsup_{k\to\infty, k\in K_1} \frac{f(y_k^i + \xi_k^i \bar{d}^i) - f(y_k^i)}{\xi_k^i} \geq 0. \tag{60}$$

Since $f(x)$ is locally Lipschitz near $\bar{x}$, by using (52), (55), and (58) we can write

$$f^\circ(\bar{x}; \bar{d}^i) \geq \limsup_{k\to\infty, k\in K_1} \frac{f(y_k^i + \xi_k^i \bar{d}^i) - f(y_k^i)}{\xi_k^i}, \qquad i = 1, \dots, r,$$

so that, from (60), we obtain

$$f^\circ(\bar{x}; \bar{d}^i) \geq 0, \qquad i = 1, \dots, r, \tag{61}$$

which proves (57). $\square$

7. Appendix B. Here we report the complete results for the modified versions of Algorithm DF, namely $\text{DF}_{\texttt{mod1}}$, $\text{DF}_{\texttt{mod2}}$, and $\text{DF}_{\texttt{mod3}}$, in Tables 4, 5, and 6.

TABLE 4
Numerical performance of Algorithm $DF_{\texttt{mod1}}$.

PROBLEM	n	q	m	nF	$f(\bar{x})$	$\bar{\mu}$	$f^\star$	Δ
crescent	2	2	0	78	0.000E+00	1.105E-02	0.000E+00	0.000E+00
polak 1	2	2	0	106	2.718E+00	7.812E-03	2.718E+00	7.654E-09
lq	2	2	0	86	-1.395E+00	7.812E-03	-1.414E+00	7.771E-03
mifflin 1	2	2	0	185	-1.000E+00	1.210E-02	-1.000E+00	6.358E-08
mifflin 2	2	2	0	74	-1.000E+00	7.812E-03	-1.000E+00	0.000E+00
char.-con 1	2	3	0	80	2.000E+00	7.812E-03	1.952E+00	1.618E-02
char.-con 2	2	3	0	81	2.000E+00	1.105E-02	2.000E+00	0.000E+00
demy-malo	2	3	0	84	-3.000E+00	1.105E-02	-3.000E+00	0.000E+00
ql	2	3	0	92	7.812E+00	7.812E-03	7.200E+00	7.470E-02
hald-mad 1	2	4	0	122	1.767E-01	1.105E-02	0.000E+00	1.767E-01
rosen	4	4	0	259	-4.378E+01	7.906E-03	-4.400E+01	4.821E-03
hald-mad 2	5	42	0	194	3.126E-01	7.906E-03	1.220E-04	3.124E-01
polak 2	10	2	0	285	5.460E+01	7.813E-03	5.459E+01	1.134E-04
maxq	20	20	0	7190	8.713E-03	7.813E-03	0.000E+00	8.713E-03
maxl	20	40	0	12111	3.028E-03	7.813E-03	0.000E+00	3.028E-03
goffin	50	50	0	2045	0.000E+00	7.813E-03	0.000E+00	0.000E+00
polak 6.1	2	3	0	92	1.973E+00	7.906E-03	1.952E+00	7.087E-03
polak 6.2	20	20	0	5174	1.553E-03	9.244E-03	0.000E+00	1.553E-03
polak 6.3	4	50	0	138	5.467E-01	7.813E-03	2.637E-03	5.426E-01
polak 6.4	4	102	0	138	5.497E-01	7.813E-03	2.650E-03	5.456E-01
polak 6.5	4	202	0	139	5.495E-01	7.813E-03	2.650E-03	5.454E-01
polak 6.6	3	50	0	104	5.441E-01	7.813E-03	4.500E-03	5.372E-01
polak 6.7	3	102	0	104	5.441E-01	7.813E-03	4.505E-03	5.372E-01
polak 6.8	3	202	0	104	5.441E-01	7.813E-03	4.505E-03	5.372E-01
polak 6.9	2	2	0	88	1.161E-01	7.812E-03	0.000E+00	1.161E-01
polak 6.10	1	25	0	58	1.782E-01	1.105E-02	1.782E-01	6.121E-07
polak 6.11	1	51	0	60	1.783E-01	1.105E-02	1.783E-01	6.630E-08
polak 6.12	1	101	0	61	1.784E-01	1.105E-02	1.784E-01	5.382E-07
polak 6.13	1	501	0	59	1.784E-01	1.105E-02	1.784E-01	1.021E-07
polak 6.14	100	100	0	44694	3.337E-03	7.812E-03	0.000E+00	3.337E-03
polak 6.15	200	200	0	50001	1.210E-01	1.914E-02	0.000E+00	1.210E-01
polak 6.16	100	50	0	50002	1.621E-01	2.210E-02	0.000E+00	1.621E-01
polak 6.17	200	50	0	50003	1.782E+00	3.125E-02	0.000E+00	1.782E+00
mad 1	2	3	1	105	-3.879E-01	1.235E-02	-3.897E-01	1.246E-03
mad 2	2	3	1	42	-3.304E-01	1.353E-02	-3.304E-01	-9.735E-10
mad 4	2	3	2	201	-4.461E-01	1.105E-02	-4.489E-01	1.967E-03
wong 2	10	6	3	358	2.654E+01	1.377E-02	2.431E+01	8.830E-02
wong 3	20	14	4	660	1.019E+02	1.271E-02	1.337E+02	-2.364E-01

Table 5
Numerical performance of Algorithm $DF_{\mathtt{mod2}}$.

PROBLEM	n	q	m	nF	$f(\bar{x})$	$\bar{\mu}$	$f^\star$	Δ
crescent	2	2	0	78	2.418E-01	1.000E+00	0.000E+00	2.418E-01
polak 1	2	2	0	106	2.718E+00	1.000E+00	2.718E+00	7.654E-09
lq	2	2	0	95	-1.274E+00	1.000E+00	-1.414E+00	5.796E-02
mifflin 1	2	2	0	65	-1.000E+00	1.000E+00	-1.000E+00	0.000E+00
mifflin 2	2	2	0	77	-8.193E-01	1.000E+00	-1.000E+00	9.033E-02
char.-con 1	2	3	0	94	2.041E+00	1.000E+00	1.952E+00	3.017E-02
char.-con 2	2	3	0	81	2.223E+00	1.000E+00	2.000E+00	7.435E-02
demy-malo	2	3	0	84	-3.000E+00	1.000E+00	-3.000E+00	0.000E+00
ql	2	3	0	156	7.473E+00	1.000E+00	7.200E+00	3.332E-02
hald-mad 1	2	4	0	292	8.496E-03	1.000E+00	0.000E+00	8.496E-03
rosen	4	4	0	515	-4.356E+01	1.000E+00	-4.400E+01	9.842E-03
hald-mad 2	5	42	0	299	9.496E-03	1.000E+00	1.220E-04	9.372E-03
polak 2	10	2	0	285	5.460E+01	1.000E+00	5.459E+01	1.134E-04
maxq	20	20	0	1858	0.000E+00	1.000E+00	0.000E+00	0.000E+00
maxl	20	40	0	891	0.000E+00	1.000E+00	0.000E+00	0.000E+00
goffin	50	50	0	2045	0.000E+00	1.000E+00	0.000E+00	0.000E+00
polak 6.1	2	3	0	106	2.041E+00	1.000E+00	1.952E+00	3.014E-02
polak 6.2	20	20	0	692	2.384E-09	1.000E+00	0.000E+00	2.384E-09
polak 6.3	4	50	0	1527	8.864E-03	1.000E+00	2.637E-03	6.211E-03
polak 6.4	4	102	0	2260	7.785E-03	1.000E+00	2.650E-03	5.122E-03
polak 6.5	4	202	0	1428	1.106E-02	1.000E+00	2.650E-03	8.388E-03
polak 6.6	3	50	0	262	6.592E-03	1.000E+00	4.500E-03	2.083E-03
polak 6.7	3	102	0	264	8.179E-03	1.000E+00	4.505E-03	3.657E-03
polak 6.8	3	202	0	400	8.545E-03	1.000E+00	4.505E-03	4.022E-03
polak 6.9	2	2	0	91	1.162E-01	1.000E+00	0.000E+00	1.162E-01
polak 6.10	1	25	0	52	1.038E+00	1.000E+00	1.782E-01	7.300E-01
polak 6.11	1	51	0	53	1.105E+00	1.000E+00	1.783E-01	7.866E-01
polak 6.12	1	101	0	51	1.139E+00	1.000E+00	1.784E-01	8.150E-01
polak 6.13	1	501	0	57	1.167E+00	1.000E+00	1.784E-01	8.389E-01
polak 6.14	100	100	0	3452	3.433E-09	1.000E+00	0.000E+00	3.433E-09
polak 6.15	200	200	0	6891	3.433E-09	1.000E+00	0.000E+00	3.433E-09
polak 6.16	100	50	0	3452	5.364E-09	1.000E+00	0.000E+00	5.364E-09
polak 6.17	200	50	0	7233	1.023E-08	1.000E+00	0.000E+00	1.023E-08
mad 1	2	3	1	43	-3.896E-01	1.000E+00	-3.897E-01	5.878E-05
mad 2	2	3	1	42	-3.304E-01	1.000E+00	-3.304E-01	-9.735E-10
mad 4	2	3	2	72	-4.489E-01	1.000E+00	-4.489E-01	4.601E-07
wong 2	10	6	3	236	2.522E+01	1.000E+00	2.431E+01	3.609E-02
wong 3	20	14	4	451	1.076E+02	1.000E+00	1.337E+02	-1.938E-01

TABLE 6
Numerical performance of Algorithm $DF_{\texttt{mod3}}$.

PROBLEM	n	q	m	nF	$f(\bar{x})$	$\bar{\mu}$	$f^{\star}$	Δ
crescent	2	2	0	79	2.693E-03	1.000E-02	0.000E+00	2.693E-03
polak 1	2	2	0	106	2.718E+00	1.000E-02	2.718E+00	7.654E-09
lq	2	2	0	142	-1.412E+00	1.000E-02	-1.414E+00	1.072E-03
mifflin 1	2	2	0	65	-1.000E+00	1.000E-02	-1.000E+00	0.000E+00
mifflin 2	2	2	0	74	-9.982E-01	1.000E-02	-1.000E+00	9.172E-04
char.-con 1	2	3	0	130	1.953E+00	1.000E-02	1.952E+00	4.080E-04
char.-con 2	2	3	0	91	2.003E+00	1.000E-02	2.000E+00	1.060E-03
demy-malo	2	3	0	84	-3.000E+00	1.000E-02	-3.000E+00	0.000E+00
ql	2	3	0	148	7.203E+00	1.000E-02	7.200E+00	3.656E-04
hald-mad 1	2	4	0	165	1.270E-03	1.000E-02	0.000E+00	1.270E-03
rosen	4	4	0	812	-4.399E+01	1.000E-02	-4.400E+01	3.083E-04
hald-mad 2	5	42	0	856	6.762E-03	1.000E-02	1.220E-04	6.639E-03
polak 2	10	2	0	285	5.460E+01	1.000E-02	5.459E+01	1.134E-04
maxq	20	20	0	7153	5.821E-11	1.000E-02	0.000E+00	5.821E-11
maxl	20	40	0	9663	5.913E-05	1.000E-02	0.000E+00	5.913E-05
goffin	50	50	0	2045	0.000E+00	1.000E-02	0.000E+00	0.000E+00
polak 6.1	2	3	0	329	1.953E+00	1.000E-02	1.952E+00	3.821E-04
polak 6.2	20	20	0	1305	2.384E-09	1.000E-02	0.000E+00	2.384E-09
polak 6.3	4	50	0	1990	8.010E-03	1.000E-02	2.637E-03	5.359E-03
polak 6.4	4	102	0	865	9.830E-03	1.000E-02	2.650E-03	7.162E-03
polak 6.5	4	202	0	2284	1.063E-02	1.000E-02	2.650E-03	7.963E-03
polak 6.6	3	50	0	590	6.429E-03	1.000E-02	4.500E-03	1.921E-03
polak 6.7	3	102	0	589	7.040E-03	1.000E-02	4.505E-03	2.524E-03
polak 6.8	3	202	0	365	7.446E-03	1.000E-02	4.505E-03	2.928E-03
polak 6.9	2	2	0	88	1.161E-01	1.000E-02	0.000E+00	1.161E-01
polak 6.10	1	25	0	62	1.784E-01	1.000E-02	1.782E-01	1.625E-04
polak 6.11	1	51	0	60	1.784E-01	1.000E-02	1.783E-01	5.924E-05
polak 6.12	1	101	0	61	1.784E-01	1.000E-02	1.784E-01	2.368E-05
polak 6.13	1	501	0	60	1.784E-01	1.000E-02	1.784E-01	1.464E-05
polak 6.14	100	100	0	50005	3.713E-02	1.000E-02	0.000E+00	3.713E-02
polak 6.15	200	200	0	50002	8.690E-02	1.000E-02	0.000E+00	8.690E-02
polak 6.16	100	50	0	50001	1.617E-01	1.000E-02	0.000E+00	1.617E-01
polak 6.17	200	50	0	50001	6.276E-01	1.000E-02	0.000E+00	6.276E-01
mad 1	2	3	1	43	-3.896E-01	1.000E-02	-3.897E-01	5.878E-05
mad 2	2	3	1	42	-3.304E-01	1.000E-02	-3.304E-01	-9.735E-10
mad 4	2	3	2	72	-4.489E-01	1.000E-02	-4.489E-01	4.601E-07
wong 2	10	6	3	236	2.522E+01	1.000E-02	2.431E+01	3.609E-02
wong 3	20	14	4	451	1.076E+02	1.000E-02	1.337E+02	-1.938E-01

Acknowledgment. The authors would like to thank two anonymous referees for their careful reading of the paper and helpful comments and suggestions which led to significant improvements in the paper.

REFERENCES

[1] J. W. Bandler and C. Charalambous, *Practical least pth optimization of networks*, IEEE Trans. Microwave Theory Tech., 20 (1972), pp. 834–840.

[2] J. W. Bandler and C. Charalambous, *Nonlinear minimax optimization as a sequence of least pth optimization with finite values of p*, Internat. J. Systems Sci., 7 (1976), pp. 377–391.

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

[4] C. Charalambous, *Acceleration of the least pth algorithm for minimax optimization with engineering applications*, Math. Programming, 17 (1979), pp. 270–297.

[5] F. H. Clarke, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York, 1983.

[6] G. Di Pillo, L. Grippo, and S. Lucidi, *A smooth method for the finite minimax problem*, Math. Programming, 60 (1993), pp. 187–214.

[7] C. Gígola and S. Gomez, *A regularization method for solving the finite convex min-max problem*, SIAM J. Numer. Anal., 27 (1990), pp. 1621–1634.

[8] 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.

[9] T. G. Kolda, R. M. Lewis, and V. Torczon, *Stationarity Results for Generating Set Search for Linearly Constrained Optimization*, Tech. report SAND2003-8550, Sandia National Laboratories, Livermore, CA, 2003, SIAM J. Optim., submitted.

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

[11] X. Li, *An entropy-based aggregate method for minimax optimization*, Eng. Optim., 18 (1992), pp. 277–285.

[12] S. Lucidi and M. Sciandrone, *On the global convergence of derivative-free methods for unconstrained optimization*, SIAM J. Optim., 13 (2002), pp. 97–116.

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

[14] L. Luksan and J. Vlcek, *Test Problems for Nonsmooth Unconstrained and Linearly Constrained Optimization*, Tech. report 798, Institute of Computer Science, Academy of Sciences of the Czech Republic, Prague, Czech Republic, 2000.

[15] D. Q. Mayne and E. Polak, *Nondifferentiable optimization via adaptive smoothing*, J. Optim. Theory Appl., 43 (1984), pp. 601–614.

[16] E. Polak, J. O. Royset, and R. S. Womersley, *Algorithms with adaptive smoothing for finite minimax problems*, J. Optim. Theory Appl., 119 (2003), pp. 459–484.

[17] R. A. Polyak, *Smooth optimization methods for minimax problems*, SIAM J. Control Optim., 26 (1988), pp. 1274–1286.

[18] F. Guerra Vazquez, H. Gunzel, and H. T. Jongen, *On logarithmic smoothing of the maximum function*, Ann. Oper. Res., 101 (2001), pp. 209–220.

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

[20] I. Zang, *A smoothing technique for min-max optimization*, Math. Programming, 19 (1980), pp. 61–77.