AN ε-ACTIVE BARRIER FUNCTION METHOD FOR SOLVING MINIMAX PROBLEMS

by

J. E. Higgins and E. Polak

Memorandum No. UCB/ERL M89/94

4 August 1989

AN ε-ACTIVE BARRIER FUNCTION METHOD FOR SOLVING MINIMAX PROBLEMS

by

J. E. Higgins and E. Polak

Memorandum No. UCB/ERL M89/94

4 August 1989

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

AN ε-ACTIVE BARRIER FUNCTION METHOD FOR SOLVING MINIMAX PROBLEMS

by

J. E. Higgins and E. Polak

Memorandum No. UCB/ERL M89/94

4 August 1989

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

AN ε-ACTIVE BARRIER FUNCTION METHOD FOR SOLVING MINIMAX PROBLEMS*

J. E. Higgins† and E. Polak†

ABSTRACT

A modification of an existing barrier function method is presented. The modified algorithm may be used to solve semi-infinite minimax problems arising in engineering design. The modification preserves the global convergence properties, simple structure and numerical robustness of the original algorithm, while substantially reducing the computational cost.

KEY WORDS

Barrier function methods, interior penalty methods, minimax algorithms, engineering design, nondifferentiable optimization, semi-infinite optimization, active set methods.

---

* This research was supported by the National Science Foundation grant ECS-8517362, the Air Force Office Scientific Research grant 86-0116, the California State MICRO program,

† Department of Electrical Engineering and Computer Sciences and the Electronics Research Laboratory, University of California, Berkeley, CA 94720, U.S.A.

1. INTRODUCTION

We are currently witnessing a considerable revival of interest in the use of barrier function techniques in the construction of both linear and nonlinear programming algorithms (see, e.g., [Kar.1, Gol.1, Gon.1, Jar.1, Jar.2, Son.1, Son.2, Ye.1]). This revival is due to the the following discoveries. First, when used as part of a method of centers and combined with an efficient homotopy procedure, barrier functions do not lead to the severe ill-conditioning previously associated with interior penalty function methods. Second, they result in algorithms of very simple structure, and third, in many cases, these algorithms have been found to compete favorably with existing algorithms.

In [Pol.2], we find the first barrier-function-based algorithm for solving semi-infinite minimax problems. This algorithm offers several advantages over other semi-infinite minimax algorithms: (i) it converges under weaker assumptions than other semi-infinite minimax algorithms, with the exception of Algorithm 5.2 in [Pol.1], (ii) it has a simple structure and it requires small memory (it does not utilize, for example, linear or quadratic programming subroutines), (iii) its numerical performance is, in most of the examples studied, superior or comparable to that of the only other first-order algorithm (Algorithm 5.2 in [Pol.1]) which can solve semi-infinite minimax problems of the same generality, (iv) it is exceptionally robust and does not fail on ill-conditioned problems on which Algorithm 5.2 in [Pol.1] fails. These qualities make the barrier-function-based algorithm in [Pol.2] an attractive candidate for solving semi-infinite minimax problems arising in engineering design (see, e.g., [Pol.1] for a discussion of these problems). In these problems, the computation of gradients is expensive, and the computation of Hessians is often impractical. Frequently only feasibility is required, in which case higher order methods may not offer significant advantages over first-order methods. In computer-aided-design applications, algorithm robustness is very important, because failure to converge, even to a local solution, may cause the loss of tens, and sometimes of hundreds of hours of computing time. In on-line applications, such as in moving-horizon control of feedback systems (see e.g., [May.1]) and certainty-equivalent adaptive control (see, e.g., [Pol.4]), robustness is of paramount importance; furthermore, optimization algorithms must be implemented using microprocessors or dedicated VLSI chips, and hence there is a premium on algorithms that are simple and that do not call large subroutines.

The essential features of the barrier function algorithm in [Pol.2] can be explained by considering the simple minimax problem

$$\min_{x \in \mathbb{R}^n} \max_{t \in [0,1]} \phi(x, t), \tag{1.1}$$

where $\phi : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ is continuously differentiable. The barrier function used in [Pol.2] is defined by

$$p(x, \alpha) \triangleq \int_{[0,1]} \frac{1}{(\alpha - \phi(x, t))} dt, \tag{1.2}$$

where $\alpha > \psi(x) \triangleq \max_{t \in [0,1]} \phi(x, t)$. For such α, the function $p(\cdot, \alpha)$ is continuously differentiable on the set

$$C(\alpha) \triangleq \{ x \in \mathbb{R}^n \mid \psi(x) < \alpha \}. \tag{1.3}$$

It is straightforward to show that $p(\cdot, \alpha)$ is a barrier function for the set $C(\alpha)$, i.e., that if $\{ x_i \}_{i=0}^{\infty} \subset C(\alpha)$ is such that $\psi(x_i) \to \alpha$, then $p(x_i, \alpha) \to \infty$. Furthermore, if $\{\alpha_i\}$ is a monotone decreasing sequence which converges to $\min_{x \in \mathbb{R}^n} \psi(x)$, then the minimizers of $p(\cdot, \alpha_i)$ must converge to minimizers of $\psi(x)$. The algorithm in [Pol.2], will be described in detail later; essentially, at iteration i, it sets $\alpha_i = ½(\psi(x_i) + \psi(x_{i-1}))$, and it computes x_i as an approximate minimizer of $p(\cdot, \alpha_i)$.

It is easy to see that we may define many other barrier functions. As an example, let $\tau : [0, \infty) \to [0, \infty)$ be any non-decreasing, continuously differentiable function such that (i) $\tau(\omega) = 0$ if and only if $\omega = 0$, and (ii) $\tau'(0) > 0$. Then

$$p(x, \alpha) \triangleq \int_{[0,1]} \frac{1}{\tau(\alpha - \phi(x, t))} dt, \tag{1.4}$$

may be shown to be a barrier function. When substituted for $p(x, \alpha)$ in the algorithm in [Pol.2] the convergence properties remain unaltered. In this paper we exploit the wealth of barrier functions to construct an efficient, first order semi-infinite minimax algorithm which requires considerably fewer gradient evaluations than the algorithm of [Pol.2].

In [Pol.1], we find semi-infinite minimax algorithms, related to the Method of Linearizations [Psh.1], which mimic methods of feasible directions by using only ε-active gradients in the search direction calculation[1]. In one of these minimax algorithms, the value of ε is driven to zero as a solution point is approached. Other active set type strategies, such as those in [Cha.1, Mur.1, Wom.1], keep the number of gradient evaluations reasonably small by only considering the ε-active gradients at each step. In this paper we introduce a semi-infinite minimax algorithm which uses a new set of barrier functions to achieve an ε-active strategy, in which ε is driven to zero as a solution is approached. This algorithm converges under the same mild assumptions as the algorithm in [Pol.2], but, as our limited numerical examples show, it is computationally more efficient because it uses fewer gradient evaluations. In the next section we describe the new barrier function method. In Section 3, we show that it is globally convergent. Numerical experience is reported in Section 4, and conclusions are drawn in Section 5.

2. THE ALGORITHM

In this section we describe a new set of barrier functions, and present a new semi-infinite minimax algorithm based on these barrier functions, together with an ε-update strategy. This algorithm solves the problem:

$$\mathbf{MMP}: \quad \min_{x \in \mathbb{R}^n} \psi(x)\,, \tag{2.1}$$

where the function $\psi : \mathbb{R}^n \to \mathbb{R}$ is defined by

$$\psi(x) \triangleq \max\left[f^1(x)\,, \cdots, f^m(x)\,, \max_{t \in [0,1]} \phi^1(x\,,t)\,, \cdots, \max_{t \in [0,1]} \phi^l(x\,,t) \right], \tag{2.2}$$

and the component functions $f^i : \mathbb{R}^n \to \mathbb{R}$ and $\phi^j : \mathbb{R}^n \times [0,1] \to \mathbb{R}$ satisfy certain continuity hypotheses.

To simplify the exposition we note that, without loss of generality, we may assume that all the functions in (2.2) are max-functions, and that $\psi(\cdot)$ is given by:

$$\psi(x) \triangleq \max\left[\max_{t \in [0,1]} \phi^1(x\,,t)\,, \cdots, \max_{t \in [0,1]} \phi^l(x\,,t) \right]. \tag{2.3}$$

This follows since any ordinary function $f : \mathbb{R}^n \to \mathbb{R}$ may be trivially converted into a max-function by

[1] Given the max-function $\psi(x) \triangleq \max_{t \in [0,1]} \phi(x\,,t)$, a function $\phi(\cdot\,,t)$ (and its corresponding gradient $\nabla_x \phi(\cdot\,,t)$) is considered to be ε-active at x if $\phi(x\,,t) \geq \psi(x) - \varepsilon$.

defining $\phi : \mathbb{R}^n \times [0,1] \to \mathbb{R}$ to be $\phi(x, t) \triangleq f(x)$.

We will make the following mild smoothness hypothesis. For ordinary functions, this hypothesis is equivalent to requiring continuous differentiability. We use the notation $\underline{l} \triangleq \{ 1, \cdots, l \}$.

Assumption 2.1: For each $k \in \underline{l}$, the function $\phi^k : \mathbb{R}^n \times [0,1] \to \mathbb{R}$ is continuous, and has a continuous first derivative $\nabla_x \phi^k(\cdot, \cdot)$. In addition, for each compact $S \subset \mathbb{R}^n$, there exists a finite L_S such that for each $x \in S$, the function $\phi^k(x, \cdot)$ is Lipschitz continuous on $[0, 1]$, with constant L_S. ■

We begin by recalling that in [Pol.2], the barrier function used for solving Problem **MMP** was:

$$p(x, \alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{1}{(\alpha - \phi^k(x, t))} dt, \tag{2.4}$$

with $\alpha > \psi(x)$ (where $\psi(\cdot)$ is defined by (2.3)). As in [Pol.2], we define the sets $C(\alpha)$ and C by

$$C \triangleq \{ (x, \alpha) \in \mathbb{R}^{n+1} \mid \psi(x) < \alpha \}, \tag{2.5}$$

$$C(\alpha) \triangleq \{ x \in \mathbb{R}^n \mid \psi(x) < \alpha \}, \tag{2.6}$$

It should be clear that under the above assumptions, the function $p(\cdot, \alpha)$ is continuously differentiable on $C(\alpha)$ for any α such that $C(\alpha) \neq \phi$. For any such α, the derivative of $p(\cdot, \alpha)$, is given by

$$\nabla_x p(x, \alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{\nabla_x \phi^k(x, t)}{(\alpha - \phi^k(x, t))^2} dt. \tag{2.7}$$

The algorithm of [Pol.2] is reproduced here for convenience.

Algorithm 2.1 ([Pol.2]).

Data: x_{-1}, $x_0 \in \mathbb{R}^n$, $K \geq 0$, $\{ \eta_k \}_{k=0}^{\infty}$ such that $\eta_k > 0$, and $\sum_{k=0}^{\infty} \eta_k < \infty$.

Step 0: Set $i = 0$.

Step 1: Set

$$\alpha_i \triangleq \begin{cases} \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) & \text{if } \psi(x_{i-1}) \neq \psi(x_i), \\ \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) + \eta_i & \text{if } \psi(x_{i-1}) = \psi(x_i), \end{cases} \tag{2.8}$$

$$y_i \triangleq \begin{cases} x_i & \text{if } \psi(x_{i-1}) \geq \psi(x_i), \\ x_{i-1} & \text{if } \psi(x_{i-1}) < \psi(x_i). \end{cases} \tag{2.9}$$

Step 2: Using y_i as an initial point, use *any* method to generate a $x_{i+1} \in C(\alpha_i)$ satisfying

$$|\nabla_x p(x_{i+1}, \alpha_i)| \leq K . \tag{2.10}$$

Step 3: Replace i by i+1 and go to Step 1. ■

The following result was established in [Pol.2].

Theorem 2.1: Suppose that Assumption 2.1 holds. If $\{ x_i \}_{i=1}^{\infty}$ is a sequence produced by Algorithm 2.1, when applied to Problem MMP, then any accumulation point $\hat{x}$, of $\{ x_i \}_{i=1}^{\infty}$, satisfies $0 \in \partial\psi(\hat{x})$. ■

We will develop the new algorithm in two steps. First we will introduce a new family of barrier functions that can be used with an ε-active set strategy. Then we will extend Algorithm 2.1 by combining this family of functions with an ε-update rule.

Clearly, (2.4) may be rewritten as

$$p(x, \alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{1}{\tau(\alpha - \phi^k(x, t))} dt. \tag{2.11}$$

where $\tau(\cdot)$ is the function defined by $\tau(\omega) \triangleq \omega$. Given this new form (2.11), we may write the derivative of the barrier function as

$$\nabla_x p(x, \alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{\tau'(\alpha - \phi^k(x, t))}{\tau(\alpha - \phi^k(x, t))^2} \nabla_x \phi^k(x, t)\, dt . \tag{2.12}$$

Since in this case $\tau'(\omega) = 1$, we note that even if for some k and t, the function $\phi^k(\cdot, t)$ is strongly inactive (i.e. $\phi^k(x, t) << \psi(x)$), its gradient must be evaluated in order to compute $\nabla_x p(x, \alpha)$. To see how this evaluation may be avoided, let $\varepsilon > 0$ be *fixed*, and let $\tau_\varepsilon : [0, \infty) \to [0, \infty)$ be the non-decreasing, continuously differentiable function whose graph is given in Figure 1. Now suppose that the function $\tau(\cdot)$ in (2.11) and (2.12) is replaced by $\tau_\varepsilon(\cdot)$. Then if $\alpha - \phi^k(x, t) \geq \varepsilon$, there is no longer any need to evaluate the corresponding gradient $\nabla_x \phi^k(x, t)$ since $\tau_\varepsilon'(\alpha - \phi^k(x, t)) = 0$. This can result in considerable savings in terms of gradient evaluations. Referring to (2.11) and (2.12), we see that to ensure that the modified function is still a barrier function, the replacement function, $\tau_\varepsilon(\cdot)$ must retain the critical properties of *both* $\tau(\cdot)$ and the ratio $\tau'(\omega)/\tau(\omega)^2$. These properties can be stated as follows:

Assumption 2.2:

(i) $\tau_\varepsilon : [0\,,\infty) \rightarrow [0\,,\infty)$ is continuously differentiable, non-decreasing and $\tau_\varepsilon(\omega) = 0$ if and only if $\omega = 0$.

(ii) $\lim\limits_{\omega \downarrow 0}\ \omega^2 \dfrac{\tau_\varepsilon'(\omega)}{\tau_\varepsilon(\omega)^2} > 0$.

(iii)[2] For all $\omega \geq \varepsilon$, $\tau_\varepsilon'(\omega) = 0$. ■

A simple example of a function satisfying Assumption 2.2 (shown in Figure 1) is given by:

$$\tau_\varepsilon(\omega) \triangleq \begin{cases} \omega - \dfrac{\omega^2}{\varepsilon} + \dfrac{\omega^3}{3\varepsilon^2} & \text{if } \omega < \varepsilon \\ \dfrac{\varepsilon}{3} & \text{if } \omega \geq \varepsilon \end{cases} . \tag{2.13}$$

Given a function $\tau_\varepsilon(\cdot)$ satisfying Assumption 2.2, define the new barrier function $p_\varepsilon(x\,,\alpha)$.

$$p_\varepsilon(x\,,\alpha) \triangleq \sum_{k \in L} \int_{[0,1]} \frac{1}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))} dt. \tag{2.14}$$

A straightforward modification of the proof in [Pol.2] shows that the conclusion of Theorem 2.1 remains valid if we replace $p(\cdot\,,\cdot)$ in Algorithm 2.1 by $p_\varepsilon(\cdot\,,\cdot)$. However, as mentioned above, using the new barrier function (2.14) can result in considerable savings in terms of gradient evaluations.

Next we explore the possibility of developing a "greedy" algorithm which reduces ε as a solution is approached. In the ε-active scheme described in [Pol.1, pp. 66-79], the ε parameter is driven to zero as a stationary point is approached. The net effect of this scheme is to progressively lighten the computational burden as a solution is approached. A consideration affecting the development of a similar scheme for a barrier function minimax algorithm is the fact that if $\alpha - \psi(x) \geq \varepsilon$ then $\nabla_x p_\varepsilon(x\,,\alpha) = 0$. Figure 2 illustrates this effect in terms of the level sets of $\psi(\cdot)$. As ε is reduced, the region in which this gradient is zero grows. A possible consequence of this is that if ε is too small when far away from the solution, the algorithm may make slow progress. In fact, if ε is driven to zero too quickly, the algorithm may jam at a non-stationary point. To avoid this, we must ensure that ε is small but not *too* small. To make any progress, we at least require that $\varepsilon > \alpha - \psi(x)$. This involves specifying a suitable

[2] Condition (iii) is necessary to create an *efficient* algorithm, but is not necessary in order to prove convergence.

scheme for updating ε and the imposition of some uniformity conditions on the family of functions $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$. The conditions we impose are given in the following assumption which may be viewed as an extension of Assumption 2.2. As will be demonstrated in Lemma 3.1, Condition (ii) ensures that the new barrier function (2.14) is indeed a barrier function. Condition (iii) ensures that certain multiplier functions arising in the convergence proof remain sufficiently positive, and Condition (iv) ensures that inactive functions make sufficiently small contributions to the gradient of (2.14).

Assumption 2.3: The family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ satisfies the following conditions:

(i) For each $\varepsilon > 0$, $\tau_\varepsilon : [0, \infty) \to [0, \infty)$ is continuously differentiable, non-decreasing and $\tau_\varepsilon(\omega) = 0$ if and only if $\omega = 0$.

(ii) There exists a $K_\tau > 0$ such that $\displaystyle \lim_{\omega \downarrow 0} \inf_{\varepsilon \geq K_\tau \omega} \omega^2 \frac{\tau_\varepsilon'(\omega)}{\tau_\varepsilon(\omega)^2} > 0$.

(iii) For any δ, Δ such that $\Delta \geq \delta > 0$, $\displaystyle \sup_{\substack{\Delta \geq \omega \geq \delta \\ \varepsilon > 0}} \frac{\tau_\varepsilon'(\omega)}{\tau_\varepsilon(\omega)^2} < \infty$.

(iv)[3] For all $\omega \geq \varepsilon$, $\tau_\varepsilon'(\omega) = 0$. ■

Given a family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ satisfying Assumption 2.3, we associate with it a family of barrier functions $\{ p_\varepsilon(\cdot, \cdot) \}_{\varepsilon > 0}$, with each $p_\varepsilon(\cdot, \cdot)$ defined by (2.14).

The main differences between Algorithm 2.1 and the algorithm below are (i) the algorithm below uses a family of barrier functions, rather than a fixed barrier function, and (ii) the algorithm below has a procedure for adjusting ε. As we have already mentioned, to prevent a barrier function minimax algorithm from jamming at a nonstationary point, the ε adjustment rule must not drive ε to zero too rapidly, so that the functions $\phi^k(x, \cdot)$ which attain the maximum $\psi(x)$ remain "sufficiently active". To simplify the exposition, we write Step 3 of the algorithm as a separate subprocedure.

Algorithm 2.2 (Greedy algorithm).

Data: x_{-1}, $x_0 \in \mathbb{R}^n$, $\delta_0 > 0$, $K \geq 0$, $\{ \eta_k \}_{k=0}^{\infty}$ such that $\eta_k > 0$, $\sum_{k=0}^{\infty} \eta_k < \infty$, $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ and $K' > K_\tau$.

[3] As before, Condition (iv) is necessary to create an *efficient* algorithm, but is not necessary in order to prove convergence.

Step 0: Set $i = 0$.

Step 1: Set

$$\alpha_i \triangleq \begin{cases} \frac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) & \text{if } \psi(x_{i-1}) \neq \psi(x_i) , \\ \frac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) + \eta_i & \text{if } \psi(x_{i-1}) = \psi(x_i) , \end{cases} \tag{2.15}$$

$$y_i \triangleq \begin{cases} x_i & \text{if } \psi(x_{i-1}) \geq \psi(x_i) , \\ x_{i-1} & \text{if } \psi(x_{i-1}) < \psi(x_i) . \end{cases} \tag{2.16}$$

Step 2: If $i > 0$, set $\delta_i = K'(\alpha_{i-1} - \psi(x_i))$.

Step 3: Call Subprocedure 2.3 with parameters $\beta = \alpha_i$, $z_0 = y_i$ and $\gamma_0 = \delta_i$. Set $\varepsilon_{i+1} = \gamma_k$, $x_{i+1} = z_{k+1}$, where (γ_k , z_{k+1}) are the returned values.

Step 4: Replace i by i+1 and go to Step 1. ■

Note that Step 3 of Algorithm 2.2 generates a pair $(\varepsilon_{i+1} , x_{i+1})$ satisfying

$$|\nabla_x p_{\varepsilon_{i+1}}(x_{i+1} , \alpha_i)| \leq K , \tag{2.17}$$

$$\varepsilon_{i+1} \geq K'(\alpha_i - \psi(x_{i+1})) . \tag{2.18}$$

The proof of convergence in Section 3 only requires that Step 3 produce a pair $(\varepsilon_{i+1} , x_{i+1})$ satisfying (2.17)-(2.18). However, as will be shown in Proposition 3.5, using the following subprocedure ensures that the ε parameter is driven to zero as a solution is approached.

Subprocedure 2.3 (Solves Step 3 of Algorithm 2.2).

Parameters: $\beta \in \mathbb{R}$, $z_0 \in C(\beta)$, $\gamma_0 > 0$.

Data: $\{ \tau_\gamma(\cdot) \}_{\gamma > 0}$, $K' > K_\tau$, $K > 0$.

Step 0: Set $k = 0$.

Step 1: Using z_k as an initial point, use *any* method to Compute $z_{k+1} \in C(\beta)$ such that $|\nabla_x p_{\gamma_k}(z_{k+1} , \beta)| \leq K$.

Step 2: If $\gamma_k < K'(\beta - \psi(z_{k+1}))$, set $\gamma_{k+1} = 2\gamma_{k+1}$, replace i by i+1 and go to Step 1.

Step 3: Return the pair (γ_k , z_{k+1}). ■

Note that the constant K' must be *strictly* greater than the constant K_τ of the family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$. This fact is used in the proof of Lemma 3.2. Furthermore, it should be noted that K' depends *only* on this family. To show that Algorithm 2.2 is well defined, we must show that Subprocedure 2.3 terminates in a *finite* number of iterations. The proof of this fact requires the following technical lemma.

Lemma 2.2: Suppose that $U \subset \mathbb{R}^n$ is open and non-empty, and that $f: U \to \mathbb{R}$ is continuously differentiable and bounded from below on U. Furthermore, suppose that for any sequence $\{ x_i \}_{i=0}^{\infty} \subset U$ such that $x_i \to x \in \partial U$ as $i \to \infty$ (where ∂U denotes the boundary of U), $f(x_i) \to \infty$ as $i \to \infty$. Then for each $\varepsilon > 0$ there exists a $x \in U$ such that $|\nabla f(x)| \le \varepsilon$.

Proof: Suppose, for the sake of contradiction, that there exists an $\bar{\varepsilon} > 0$ such that $|\nabla f(x)| > \bar{\varepsilon}$ for all $x \in U$. Apply the gradient algorithm of [Arm.1] (or [Pol.5]) to the function $f(\cdot)$ with initial point x_0. The algorithm must be modified slightly to ensure that the iterates remain in U. Since U is open, the algorithm remains well defined, and generates a sequence of non-negative step sizes $\{ \lambda_i \}_{i=0}^{\infty}$ and iterates $\{ x_i \}_{i=0}^{\infty}$ such that

$$f(x_{i+1}) - f(x_i) \le -\lambda_i |\nabla f(x_i)|^2 \le 0 \ , \tag{2.19}$$

$$x_{i+1} = x_i - \lambda_i \nabla f(x_i) \ . \tag{2.20}$$

Summing both sides of (2.19) yields $f(x_{n+1}) - f(x_0) \le -\sum_{i=0}^{n} \lambda_i |\nabla f(x_i)|^2 \le 0$. Since $f(\cdot)$ is bounded from below, this implies that $\sum_{i=0}^{\infty} \lambda_i |\nabla f(x_i)|^2 < \infty$. By assumption, $|\nabla f(x_i)| > \bar{\varepsilon}$, from which it follows that $\sum_{i=0}^{\infty} \lambda_i |\nabla f(x_i)|^2 > \bar{\varepsilon} \sum_{i=0}^{\infty} \lambda_i |\nabla f(x_i)|$. It then follows from (2.20) that $\{ x_i \}_{i=0}^{\infty}$ is a Cauchy sequence, which must converge to some $\hat{x} \in \mathbb{R}^n$. Since $x_i \in U$ we must have that $\hat{x} \in \overline{U}$, the closure of U. However, since $f(x_i) \le f(x_0)$ for all i, it follows that, in fact, $\hat{x} \in U$. As a consequence we have $\nabla f(\hat{x}) = 0$ (See [Arm.1] or [Pol.5]), which is a contradiction. ■

Proposition 2.3: Suppose that Assumption 2.1 holds, and that the family $\{ \tau_\gamma(\cdot) \}_{\gamma > 0}$ satisfies Assumption 2.3. In addition, suppose that $\inf_{x \in \mathbb{R}^n} \psi(x) > -\infty$. Then Subprocedure 2.3 terminates in a finite number of iterations.

Proof: Suppose that $\beta \in \mathbb{R}$, $z_0 \in C(\beta)$ and $\gamma_0 > 0$. Since for any $x \in C(\beta)$, $\gamma > 0$, we have $p_\gamma(x, \beta) > 0$, it follows from Lemma 2.2 (applied to $p_\gamma(\cdot, \beta)$ on the set $C(\beta)$) that Step 1 is well defined. For the sake of contradiction, suppose that the algorithm does *not* terminate in a finite number of iterations. From Step 2 we have, for each $k \geq 0$, $2^k \gamma_0 < K'(\beta - \psi(z_{k+1}))$. However, by hypothesis the right hand side is bounded, which yields a contradiction for k sufficiently large. ■

As in [Pol.2], the computation in Step 1 of Subprocedure 2.3 may be carried out using either the Armijo gradient algorithm [Arm.1], or a Gauss-Newton type algorithm applied to the function $p_\varepsilon(\cdot\,, \alpha_i)$. A description of a suitable Gauss-Newton type algorithm is given in the appendix of [Pol.1]. In practice, we have found that Step 2 usually requires only one iteration of the Gauss-Newton type algorithm. In either case, an appropriate modification of Lemma 2.2 shows that Step 1 of Subprocedure 2.3 will be computed in a finite number of iterations.

Before presenting the convergence proof for Algorithm 2.2, we make a few remarks.

(i) It is straightforward to show that the family of functions given by (2.13) satisfies Assumption 2.3 (choose any $K_\tau > 1$). However, from a numerical standpoint we have found the following family to offer superior performance on the problems tested. In our experiments, we have set the parameter $\theta = 0.9$, however any $\theta \in (0,1)$ will do.

$$\tau_\varepsilon(\omega) \triangleq \begin{cases} \dfrac{2}{1+\theta}\omega & \text{if } \omega < \theta\varepsilon\,, \\ \dfrac{\varepsilon t^4 - 2(1+\theta)\varepsilon^2 t^3 + 6\theta\varepsilon^3 t^2 + (2-6\theta)\varepsilon^4 t + (2\theta^3 - \theta^4)\varepsilon^5}{(1-\theta)^3(1+\theta)\varepsilon^4} & \text{if } \theta\varepsilon \leq \omega < \varepsilon\,, \\ \varepsilon & \text{if } \omega \geq \varepsilon\,. \end{cases} \tag{2.21}$$

A straightforward, but tedious calculation shows that this family also satisfies Assumption 2.3 (again choose any $K_\tau > 1$).

(ii) A brief examination of the proof of convergence in Section 3 shows that the ε requirement (2.18) may be relaxed to the following:

$$\varepsilon_{i+1} \geq \min\{\, \varepsilon_0\,, K'(\alpha_i - \psi(x_{i+1}))\,\}\,, \tag{2.22}$$

where $\varepsilon_0 > 0$. This may be advantageous in the initial iterations, particularly when significant progress is made (i.e. when $\alpha_i - \psi(x_{i+1})$ is large).

(iii) From a numerical standpoint, it is useful to add a few more conditions on the family $\{\, \tau_\varepsilon(\cdot)\,\}_{\varepsilon > 0}$.

Assumption 2.4: The family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ satisfies the following conditions:

(i) For each $\varepsilon > 0$, $\tau_\varepsilon : [0,\infty) \to [0,\infty)$ is *twice* continuously differentiable.

(ii) For each $\varepsilon > 0$, and for each $\omega > 0$, $\quad 2\dfrac{\tau_\varepsilon'(\omega)^2}{\tau_\varepsilon(\omega)} - \tau_\varepsilon''(\omega) \geq 0.$ ■

Assumption 2.4 gives sufficient conditions which ensure that if the component functions of (2.3) are convex, then so is the barrier function $p_\varepsilon(\cdot\,, \alpha)$. In addition, it allows us to approximate the Hessian of the barrier function in a reasonable way. To illustrate this, we note that under suitable smoothness hypotheses, the Hessian is given by:

$$\frac{\partial^2 p_\varepsilon(x\,,\alpha)}{\partial x^2} \triangleq \sum_{k \in \underline{k}} \int_{[0,1]} \left[2\frac{\nabla_x\phi^k(x\,,t)\,\nabla_x\phi^k(x\,,t)^T}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))^2} \left[2\frac{\tau_\varepsilon'(\alpha - \phi^k(x\,,t))^2}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))} - \tau_\varepsilon''(\alpha - \phi^k(x\,,t)) \right] \right.$$
$$\left. + \frac{\phi^k_{xx}(x\,,t)}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))^2}\tau_\varepsilon'(\alpha - \phi^k(x\,,t)) \right] dt\,, \tag{2.23}$$

To avoid computing Hessians (and making the additional smoothness hypothesis), we approximate (2.23) by the positive definite matrix

$$H_\varepsilon(x\,,\alpha) \triangleq \sum_{k \in \underline{k}} \int_{[0,1]} \left[2\frac{\nabla_x\phi^k(x\,,t)\,\nabla_x\phi^k(x\,,t)^T}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))^2} \left[2\frac{\tau_\varepsilon'(\alpha - \phi^k(x\,,t))^2}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))} - \tau_\varepsilon''(\alpha - \phi^k(x\,,t)) \right] \right.$$
$$\left. + \frac{\sigma I}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))^2}\tau_\varepsilon'(\alpha - \phi^k(x\,,t)) \right] dt\,, \tag{2.24}$$

where $\sigma > 0$ is some fixed constant. The families of functions (2.13) and (2.21) both satisfy Assumption 2.4. The approximate Hessian may be used either in a Gauss-Newton type algorithm, or in a homotopy type initialization scheme which computes an alternative initial point (y_i) to that computed by Step 1 of Algorithm 2.2. Details may be found in [Pol.2].

(iv) The slope requirement of Step 2 (2.17) may be relaxed as in [Pol.2, Section 2, Remark (iv)], and the sequence $\{ \eta_k \}_{k=0}^{\infty}$ may be chosen as in Remark (v) of same.

(v) In Proposition 3.5 we will prove that if the sequence of costs $\{ \psi(x_i) \}_{i=1}^{\infty}$ converges to $\inf_{x \in \mathbb{R}^n}\psi(x)$ (assumed finite), then ε is driven to zero. By modifying Subprocedure 2.3, it is possible to ensure that ε is *always* driven to zero. The resulting modification is given as follows. The essential difference is that β, instead of γ, is varied.

Subprocedure 2.4 (Solves Step 3 of Algorithm 2.2).

Parameters: $\beta_0 \in \mathbb{R}$, $z_0 \in C(\beta_0)$, $\gamma > 0$.

Data: $\{ \tau_\gamma(\cdot) \}_{\gamma > 0}$, $K' > K_\tau$, $K > 0$.

Step 0: Set $k = 0$.

Step 1: Using z_k as an initial point, use *any* method to Compute $z_{k+1} \in C(\beta_k)$ such that $\| \nabla_x p_\gamma(z_{k+1}, \beta_k) \| \le K$.

Step 2: If $\gamma < K'(\beta_k - \psi(z_{k+1}))$, set $\beta_{k+1} = \frac{1}{2}(\beta_k + \psi(z_{k+1}))$, replace k by $k+1$ and go to Step 1.

Step 3: Return the pair (γ, z_{k+1}). ■

However, limited numerical experience shows that this procedure performs poorly when compared with Subprocedure 2.3. Furthermore, in all of our experiments (using Subprocedure 2.3), ε was driven to zero. Consequently, we have used Subprocedure 2.3.

(vi) Finally, we note that the family of functions defined by $\tau_\varepsilon(\omega) \triangleq \omega$ satisfies Assumptions 2.3 and 2.4, with the exception of Condition (iv) of Assumption 2.3. This yields the original algorithm presented in [Pol.2], and so we may view our scheme as a generalization of the original method.

Before concluding this section, we note that we have tacitly assumed that $\psi(\cdot)$, $p_\varepsilon(\cdot,\cdot)$ and $\nabla_x p_\varepsilon(\cdot,\cdot)$ can be evaluated exactly. Consequently, Algorithm 2.2 should be viewed as a conceptual algorithm. An implementable algorithm may be developed in a manner similar to that presented in [Kle.1], by adopting a suitable discretization scheme for the interval [0,1].

3. PROOF OF CONVERGENCE

This section contains two main results. The first shows that for a fixed $\varepsilon > 0$, the function $p_\varepsilon(\cdot, \alpha)$ is a barrier function for the set $C(\alpha)$. This ensures that if we use a descent method to compute the next point (Step 3), the iterates will never stray outside of the set $C(\alpha)$. The second result shows that any accumulation point $\hat{x}$, of a sequence produced by Algorithm 2.2 satisfies $0 \in \partial\psi(\hat{x})$ (where $\partial\psi(\hat{x})$ denotes the Clarke generalized gradient [Cla.1] of $\psi(\cdot)$ at $\hat{x}$). Our proof requires the following definition of the set valued function $\overline{G}\psi : \mathbb{R}^n \to 2^{\mathbb{R}^{n+1}}$.

$$\bar{G}\psi(x) \triangleq \underset{\substack{k \in \underline{l} \\ t \in [0,1]}}{co} \left\{ \begin{bmatrix} \psi(x) - \phi^k(x\,,t) \\ \nabla_x \phi(x\,,t) \end{bmatrix} \right\}. \tag{3.1}$$

It is straightforward to show that $\bar{G}\psi(\cdot)$ is an augmented convergent direction finding (a.c.d.f.) map for $\psi(\cdot)$ (see [Pol.1], Definition 5.1). In particular, we use two properties of $\bar{G}\psi(\cdot)$: (i) $\bar{G}\psi(\cdot)$ is upper semi-continuous (See [Ber.1]), and (ii) $0 \in \bar{G}\psi(\hat{x})$ if and only if $0 \in \partial\psi(\hat{x})$. Some additional notation is necessary at this point. Let the ε-active index sets $A_\varepsilon^k(x) \subset [0,1]$, $k \in \underline{l}$ be defined by:

$$A_\varepsilon^k(x) \triangleq \{\, t \in [0,1] \mid \phi^k(x\,,t) \geq \psi(x) - \varepsilon \,\}\ , \tag{3.2}$$

The fact that the function (2.14) is a barrier function follows from the following Theorem.

Theorem 3.1: Suppose that Assumption 2.1 holds, and that the family $\{\, \tau_\varepsilon(\cdot) \,\}_{\varepsilon > 0}$ satisfies Assumption 2.3. Then for each $\varepsilon > 0$ and for each bounded set $C_o \subset C$ there exists a constants $\lambda > 0$ and $\kappa > 0$ such that if $(x\,,\alpha) \in C_o$ and $\alpha - \psi(x) \leq \lambda$ then

$$p_\varepsilon(x\,,\alpha) \geq \frac{1}{\kappa}\log\left[1 + \frac{\lambda}{\alpha - \psi(x)}\right]. \tag{3.3}$$

Proof: Let Π be the projection of C_o onto $\mathbb{R}^n$. Since C_o is bounded, so is the projection Π. By Assumption 2.1 there exists a Lipschitz constant, $L < \infty$, such that each $\phi^k(x\,,\cdot)$ is uniformly Lipschitz in t on $[0,1]$, for all $x \in \Pi$. We may also assume (since C_o is bounded) that $L \geq \alpha - \psi(x)$, for all $(x\,,\alpha) \in C_o$. Choose any $k \in \underline{l}$ such that $A_0^k(x)$ is nonempty, and let $(x\,,\alpha) \in C_o$. Let $t_x \in A_0^k(x)$ and let $t \in [0\,,1]$. Then we have

$$\phi^k(x\,,t) \geq \phi^k(x\,,t_x) - L|t - t_x| = \psi(x) - L|t - t_x|\,. \tag{3.4}$$

Since each member of the family $\{\, \tau_\varepsilon(\cdot) \,\}_{\varepsilon > 0}$ is differentiable at zero, for a given $\varepsilon > 0$ there exists a $K > 0$ and $\varpi > 0$ such that $\tau_\varepsilon(\omega) \leq K\omega$ whenever $\omega \leq \varpi$. Without loss of generality, we may choose $\varpi \leq 4L$. Let $\lambda \triangleq \varpi/2$. Then if $\alpha - \psi(x) \leq \lambda$ and $t \in [t_x - \varpi/(2L)\,,\, t_x + \varpi/(2L)] \cap [0\,,1]$, we have

$$\alpha - \phi^k(x\,,t) \leq \varpi. \tag{3.5}$$

Since we have chosen $\varpi \leq 4L$, either $t \in [t_x\,,\, t_x + \varpi/(2L)] \subset [0\,,1]$ (if $t_x < ½$), or $t \in [t_x - \varpi/(2L)\,,\, t_x] \subset [0\,,1]$ (if $t_x \geq ½$). In the following analysis we have assumed, for brevity, that $[t_x\,,\, t_x + \varpi/(2L)] \subset [0,1]$. The analysis remains the same if this interval is replaced by $[t_x - \varpi/(2L)\,,\, t_x]$.

$$p_\varepsilon(x\,,\alpha) \ge \int_{[0,1]} \frac{1}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))}\,dt$$

$$\ge \int_{[t_x\,,\,t_x + \varpi/(2L)]} \frac{1}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))}\,dt$$

$$\ge \frac{1}{K} \int_{[t_x\,,\,t_x + \varpi/(2L)]} \frac{1}{\alpha - \phi^k(x\,,t)}\,dt$$

$$\ge \frac{1}{K} \int_{[t_x\,,\,t_x + \varpi/(2L)]} \frac{1}{\alpha - \psi(x) + L(t - t_x)}\,dt$$

$$= \frac{1}{KL} \log\left[1 + \frac{\lambda}{\alpha - \psi(x)}\right]. \tag{3.10}$$

Let $\kappa \triangleq KL$ to obtain the required result. ■

Consequently, if $\alpha \in \mathbb{R}$ is such that $C(\alpha) \ne \phi$ and $\{x_i\}_{i=0}^{\infty} \subset C(\alpha)$ is a bounded sequence such that $\psi(x_i) \to \alpha$ as $i \to \infty$, then $p_\varepsilon(x_i\,,\alpha) \to \infty$ as $i \to \infty$, i.e., $p_\varepsilon(\cdot\,,\alpha)$ is a barrier function for $C(\alpha)$.

The proof of convergence depends on the following technical lemma, which generalizes the fact that a decreasing sequence either converges or diverges properly to $-\infty$. The proof of the lemma is identical to that of Lemma 3.1 in [Pol.2] and is omitted.

Lemma 3.2: Suppose that the sequences of real numbers $\{\gamma_i\}_{i=-1}^{\infty}$ and $\{\eta_i\}_{i=0}^{\infty}$ satisfy the following conditions: (i) $\eta_i \ge 0$, for all $i \in \mathbb{N}$, (ii) $\sum_{i=0}^{\infty} \eta_i < \infty$, and (iii) $\gamma_{i+1} \le \frac{1}{2}(\gamma_i + \gamma_{i-1}) + \eta_i$ for all $i \in \mathbb{N}$. Then either $\{\gamma_i\}_{i=-1}^{\infty}$ converges, or $\gamma_i \to -\infty$ as $i \to \infty$. ■

The following lemma derives an inequality to be used in Theorem 3.4.

Lemma 3.3: Suppose that Assumption 2.1 holds, and that the family $\{\tau_\varepsilon(\cdot)\}_{\varepsilon>0}$ satisfies Assumption 2.3. Let K_τ be the constant of Condition (iii) of Assumption 2.3. Then there exist constants $\varpi > 0$, $\lambda > 0$ such that for each bounded set $C_o \subset C$ and constant $K' > K_\tau$, there exists a constant $L > 0$, such that if $(x\,,\alpha) \in C_o$ and $\varepsilon > 0$ are such that $K'(\alpha - \psi(x)) \le \varpi$ and $K'(\alpha - \psi(x)) \le \varepsilon$, then

$$(\alpha - \psi(x)) \sum_{k \in \underline{k}} \int_{[0,1]} \frac{\tau_\varepsilon'(\alpha - \phi^k(x\,,t))}{\tau_\varepsilon(\alpha - \phi^k(x\,,t))^2}\,dt \;\ge\; \frac{\lambda}{L} \frac{(K' - K_\tau)/K_\tau}{(1 + (K' - K_\tau)/K_\tau)^2}. \tag{3.11}$$

Proof: By Condition (iii) of Assumption 2.3, there exists $\lambda > 0$ and $\varpi > 0$ such that

$$\omega^2 \frac{\tau_\varepsilon'(\omega)}{\tau_\varepsilon(\omega)^2} \geq \lambda > 0 , \tag{3.12}$$

whenever $\omega \leq \varpi$ and $\varepsilon \geq K_\tau \omega$.

Since C_o is bounded, so is the projection Π of C_o onto $\mathbb{R}^n$. Hence by Assumption 2.1 there exists a Lipschitz constant, $L < \infty$, such that each $\phi^k(x , \cdot)$ is uniformly Lipschitz in t on $[0,1]$, for all $x \in \Pi$. Let $k \in \underline{l}$ be such that $A_0^k(x)$ is nonempty, let $t_x \in A_0^k(x)$ be given and let $t \in [0 , 1]$. Then we have that

$$\phi^k(x , t) \geq \phi^k(x , t_x) - L|t - t_x| = \psi(x) - L|t - t_x| . \tag{3.13}$$

Now suppose that $K' > K_\tau$, $(x , \alpha) \in C_o$ $K'(\alpha - \psi(x)) \leq \varpi$, and $K'(\alpha - \psi(x)) \leq \varepsilon$. Without loss of generality, we may assume (since C_o is bounded) that $L \geq (\alpha - \psi(x))(K' - K_\tau)/K_\tau$, for all $(x , \alpha) \in C_o$. Let $\delta \triangleq (\alpha - \psi(x))(K' - K_\tau)/K_\tau$. Then for any $t \in A_\delta^k(x)$,

$$\begin{aligned} K_\tau(\alpha - \phi(x , t)) &\leq K_\tau(\alpha - \psi(x) + \delta) \\ &\leq K_\tau(\alpha - \psi(x)) + (K' - K_\tau)(\alpha - \psi(x)) \\ &= K'(\alpha - \psi(x)) \leq \varpi. \end{aligned} \tag{3.16}$$

Similarly, we have $K_\tau(\alpha - \phi(x , t)) \leq \varepsilon$ for any $t \in A_\delta^k(x)$. Consequently, we may use (3.12) to get the following estimate:

$$\begin{aligned} (\alpha - \psi(x)) \sum_{j \in \underline{l}} \int_{[0,1]} \frac{\tau_\varepsilon'(\alpha - \phi^j(x , t))}{\tau_\varepsilon(\alpha - \phi^j(x , t))^2} dt \; &\geq (\alpha - \psi(x)) \int_{A_\delta^k(x)} \frac{\tau_\varepsilon'(\alpha - \phi^k(x , t))}{\tau_\varepsilon(\alpha - \phi^k(x , t))^2} dt \\ &\geq (\alpha - \psi(x)) \int_{A_\delta^k(x)} \frac{\lambda}{(\alpha - \phi^k(x , t))^2} dt \\ &\geq \frac{(\alpha - \psi(x)) \lambda m(A_\delta^k(x))}{(\alpha - \psi(x) + \delta)^2} , \end{aligned} \tag{3.19}$$

where $m(\cdot)$ denotes the Lebesque measure on $\mathbb{R}$. However, (3.13) implies that $m(A_\delta^k(x)) \geq \frac{\delta}{L}$, and so we have

$$(\alpha - \psi(x)) \sum_{j \in \underline{l}} \int_{[0,1]} \frac{\tau_\varepsilon'(\alpha - \phi^j(x , t))}{\tau_\varepsilon(\alpha - \phi^j(x , t))^2} dt \; \geq \frac{\lambda}{L} \frac{(\alpha - \psi(x)) \delta}{(\alpha - \psi(x) + \delta)^2}$$

$$= \frac{\lambda}{L} \frac{(K' - K_\tau)/K_\tau}{(1 + (K' - K_\tau)/K_\tau)^2} \,, \tag{3.21}$$

as required. ■

The essence of the proof of Theorem 3.4 is to show that, if for some infinite index set $S \subset \mathbb{N}$, $x_i \xrightarrow{S} \hat{x}$, then there exist elements $\xi_i \in \overline{G}\psi(x_i)$ such that $\xi_i \xrightarrow{S} 0$. It then follows from the upper semi-continuity of $\overline{G}\psi(\cdot)$ that $0 \in \overline{G}\psi(\hat{x})$, which is equivalent to $0 \in \partial\psi(\hat{x})$.

Theorem 3.4: Suppose that Assumption 2.1 holds, and that the family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ satisfies Assumption 2.3. Let K_τ be the constant of Condition (iii) in Assumption 2.3, and suppose the constant K' of Algorithm 2.2 (2.18) is strictly greater than K_τ. If $\{ x_i \}_{i=1}^{\infty}$ is a sequence produced by Algorithm 2.2, when applied to Problem MMP, then any accumulation point $\hat{x}$, of $\{ x_i \}_{i=1}^{\infty}$, satisfies $0 \in \partial\psi(\hat{x})$.

Proof: Suppose that $x_i \xrightarrow{S} \hat{x}$, as $i \to \infty$ for some infinite subset $S \subset \mathbb{N}$. By construction, $x_{i+1} \in C(\alpha_i)$ for all $i \in \mathbb{N}$, and hence it follows that

$$\psi(x_{i+1}) < \alpha_i \le \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) + \eta_i . \tag{3.22}$$

Therefore the sequence $\{ \psi(x_i) \}_{i=1}^{\infty}$ satisfies the conditions of Lemma 3.2. Since $\psi(\cdot)$ is continuous, we must have that $\psi(x_i) \xrightarrow{S} \psi(\hat{x})$, and hence, because of Lemma 3.2, the whole sequence $\{ \psi(x_i) \}_{i=1}^{\infty}$ converges to $\psi(\hat{x})$. As a consequence, the sequence $\{ \alpha_i \}_{i=0}^{\infty}$ also converges to $\psi(\hat{x})$. By construction, we have for all $i \in \mathbb{N}$, $i > 0$,

$$\| \nabla_x p_{\varepsilon_i}(x_i , \alpha_{i-1}) \| \le K \,. \tag{3.23}$$

Since $(\alpha_{i-1} - \psi(x_i)) \to 0$ as $i \to \infty$, it follows that

$$\lim_{i \to \infty} (\alpha_{i-1} - \psi(x_i)) \, \nabla_x p_{\varepsilon_i}(x_i , \alpha_{i-1}) = 0 \,. \tag{3.24}$$

For each $j \in \underline{L}$ define the *multiplier* function $\rho_i^j : [0,1] \to \mathbb{R}$ by

$$\rho_i^j(t) \triangleq (\alpha_{i-1} - \psi(x_i)) \frac{\tau_{\varepsilon_i}'(\alpha_{i-1} - \phi^j(x_i , t))}{\tau_{\varepsilon_i}(\alpha_{i-1} - \phi^j(x_i , t))^2} \,. \tag{3.25}$$

Note that since $\tau_{\varepsilon_i}(\cdot)$ is non-decreasing, $\rho_i^j(t) \ge 0$. Clearly $\{ (\alpha_{i-1} , x_i) \}_{i \in S}$ is a bounded subset of C

and $K' > K_\tau$ by assumption. Furthermore, by construction we have that $K'(\alpha_{i-1} - \psi(x_i)) \le \varepsilon_i$. Consequently, we conclude from Lemma 3.3 that there exists a $\beta > 0$ such that for all $i \in S$ sufficiently large (so that $K'(\alpha_{i-1} - \psi(x_i)) \le \overline{\omega}$)

$$\nu_i \triangleq \sum_{j \in \underline{L}} \int_{[0,1]} \rho_i^j(t)\, dt \ge \beta. \tag{3.26}$$

It follows from (2.12), (3.24) and (3.26) that

$$\frac{1}{\nu_i} \sum_{j \in \underline{L}} \int_{[0,1]} \rho_i^j(t)\, \nabla_x \phi^j(x, t)\, dt \overset{S}{\to} 0. \tag{3.27}$$

Furthermore, since $\{ x_i \}_{i \in S}$ is bounded, there exists some constant Δ such that $\psi(x_i) - \phi^j(x_i, t) \le \Delta$, for all $t \in [0,1]$, for all $i \in S$ and for all $j \in \underline{L}$. Consequently, for any $\delta \in (0, \Delta)$, and $i \in S$ sufficiently large,

$$\begin{aligned}
&\sum_{j \in \underline{L}} \int_{[0,1]} \rho_i^j(t)\, (\psi(x_i) - \phi^j(x_i, t))\, dt \\
&\quad = \sum_{j \in \underline{L}} \left[\int_{A_\delta^j(x_i)} \rho_i^j(t)\, (\psi(x_i) - \phi^j(x_i, t))\, dt + \int_{A_\delta^j(x_i)^C} \rho_i^j(t)\, (\psi(x_i) - \phi^j(x_i, t))\, dt \right] \\
&\quad \le \delta \nu_i + (\alpha_{i-1} - \psi(x_i)) \Delta Z,
\end{aligned} \tag{3.30}$$

where $Z \triangleq \sup\limits_{\substack{\Delta \ge \omega \ge \delta \\ \varepsilon > 0}} \dfrac{\tau_\varepsilon'(\omega)}{\tau_\varepsilon(\omega)^2}$. Condition (iv) of Assumption 2.3 ensures that Z is finite. It follows from (3.28)-(3.30) that

$$\frac{1}{\nu_i} \sum_{j \in \underline{L}} \int_{[0,1]} \rho_i^j(t)\, (\psi(x_i) - \phi^j(x_i, t))\, dt \overset{S}{\to} 0. \tag{3.31}$$

Since $\rho_i^j(t) \ge 0$ for all i, j, t, convexity of $\overline{G}\psi(\cdot)$ implies

$$\xi_i \triangleq \frac{1}{\nu_i} \sum_{j \in \underline{L}} \int_{[0,1]} \rho_i^j(t) \begin{bmatrix} \psi(x_i) - \phi^j(x_i, t) \\ \nabla_x \phi^j(x_i, t) \end{bmatrix} dt \in \overline{G}\psi(x_i). \tag{3.32}$$

Since (3.27) and (3.31) imply $\xi_i \overset{S}{\to} 0$ as $i \to \infty$, if follows from the upper semi-continuity of $\overline{G}\psi(\cdot)$ that $0 \in \overline{G}\psi(\hat{x})$. This completes the proof. ∎

The following proposition shows that under suitable assumptions, ε is driven to zero.

Proposition 3.5: Suppose that Assumption 2.1 holds, and that the family $\{ \tau_\varepsilon(\cdot) \}_{\varepsilon > 0}$ satisfies Assumption 2.3. Let K_τ be the constant in Assumption 2.3 (iii), and suppose the constant K' of Algorithm 2.2 (2.18) is strictly greater than K_τ. Suppose that $\{ x_i \}_{i=1}^{\infty}$ is a sequence produced by Algorithm 2.2, when applied to Problem **MMP**, and that $\psi(x_i) \to \inf_{x \in \mathbb{R}^n} \psi(x) > -\infty$. Then $\varepsilon_i \to 0$ as $i \to \infty$.

Proof: Let $\hat{\psi} \triangleq \inf_{x \in \mathbb{R}^n} \psi(x)$. Since we have assumed that $\psi(x_i) \to \hat{\psi}$, it follows that $\alpha_i \to \hat{\psi}$. Suppose that at iteration i, Subprocedure 2.3 requires a single iteration. Then $\varepsilon_{i+1} = K'(\alpha_{i-1} - \psi(x_i))$. Otherwise, the subprocedure iterates until it satisfies the test in Step 2. However, by assumption we have $K(\alpha_i - \psi(x)) \le K'(\alpha_i - \hat{\psi})$ for all $x \in \mathbf{R}^n$. Since ε is doubled at each iteration, we have $\varepsilon_{i+1} \le 2K'(\alpha_i - \hat{\psi}))$. Consequently, we have

$$\varepsilon_{i+1} \le K' \max\{ \alpha_{i-1} - \psi(x_i) , 2(\alpha_i - \hat{\psi}) \} , \tag{3.33}$$

from which it follows that $\varepsilon_i \to 0$ as $i \to \infty$. ■

4. NUMERICAL RESULTS

We now present some numerical examples which illustrate the reduction in gradient evaluations obtained by using Algorithm 2.2 instead of the method in [Pol.2]. We have used the same set of test problems used in [Pol.2]. Briefly, we have (i) constructed three semi-infinite minimax problems by converting three constrained problems in [Tan.1] into semi-infinite minimax problems using l_∞ exact penalty functions, and (ii) we have taken from the control literature two semi-infinite minimax problems which correspond to the very important task of constructing a stabilizing compensator for a multivariable linear feedback system. Finally, to determine if our algorithm has any advantages in solving finite dimensional minimax problems, we have applied it to a few problems of varying degree of difficulty and compared its performance to existing algorithms. Since the number of component functions of these finite problems is small, and the starting points are reasonably close to the solution, the reduction in the number of gradient evaluations will not be as significant as in the semi-infinite case.

In our experiments, the computations in Step 2 of Algorithm 2.2 were carried out using a Gauss-Newton type algorithm described in [Pol.2]. To improve performance, we used the homotopy type initialization mentioned in Remark (iv) of Section 2. All the computations were performed in double precision on a Sun 3 microcomputer with a floating point accelerator. All of the parameters of Algorithm 2.2 (except K') were chosen in the same manner as in [Pol.2]. The parameter K' was selected in an ad-hoc manner, typically between 3 and 10. In all of our experiments, Subprocedure 2.3 drove the ε parameter to zero.

A potential numerical problem with Algorithm 2.2 arises from the fact that if the sequence $\{ x_i \}_{i=-1}^{\infty}$ which it constructs converges, then $\lim_{i \to \infty} p(x_{i+1}, \alpha_i) = \infty$. In practice however, this did not create any difficulties.

For semi-infinite problems, we compared Algorithm 2.2 with the method of [Pol.2] and a modified version of Algorithm 5.2 in [Pol.1] (see Example 5.1 and Corollary 5.1 in [Pol.1] for details). These appear to be the only other minimax algorithms in the literature which can be proved to be globally convergent under equally weak assumptions. To obtain a quantity comparable to the number of gradient evaluations in [Pol.2], we define NG to be the total number of gradient evaluations divided by the number of functions (l). For the semi-infinite problems, we have used a fine discretization of the interval $[0, 1]$ to evaluate $p_\varepsilon(\cdot, \cdot)$, and computed NG in the manner described previously. Our test problems were as follows:

Problem TFI1: This is a modification of Problem 1 of [Tan.1]. In this problem, and in Problems TFI2, TFI3, the exact penalty has been adjusted so that the minimax problem has the same solution as the original problem. Here $\psi(x) = \max\{ f^1(x), \max_{t \in [0,1]} \phi^1(x, t) \}$, where $f^1(\cdot)$, $g(\cdot, \cdot)$ (defined in [Tan.1]) and $\phi^1(\cdot, \cdot)$ are given by:

$$f^1(x) \triangleq (x^1)^2 + (x^2)^2 + (x^3)^2, \tag{4.1}$$

$$g(x, t) \triangleq x^1 + x^2 e^{x^3 t} + e^{2t} - 2\sin(4t), \tag{4.2}$$

$$\phi^1(x, t) \triangleq f^1(x) + 100\, g(x, t), \tag{4.3}$$

Initial points: $x_{-1} = x_0 = (1,1,1)^T$, solution: $(-0.213313, -1.361450, 1.853547)^T$. ■

Problem TFI2: In this problem $\psi(x) = \max\{ f^1(x), \max_{t \in [0,1]} \phi^1(x, t) \}$, where $\phi^1(x, t)$ is defined as above (4.3), but using the functions $f^1(\cdot)$, and $g(\cdot, \cdot)$ of Problem 2(a) [Tan.1] defined by:

$$f^1(x) \triangleq x^1 + x^2/2 + x^3/3, \tag{4.4}$$

$$g(x\,,t) \triangleq \tan(t) - x^1 - (x^2)t - (x^3)t^2 \tag{4.5}$$

Initial points: $x_{-1} = x_0 = (0,0,0)^T$, solution: $(0.089096,0.423052,1.045260)^T$. ■

Problem TFI3: In this problem $\psi(x) = \max\{\, f^1(x)\,,\, \max_{t \in [0\,,1]} \phi^1(x\,,t)\,\}$, where $\phi^1(x\,,t)$ is defined as above (4.3), but using the functions $f^1(\cdot)$, and $g(\cdot\,,\cdot)$ of Problem 3 [Tan.1] defined by:

$$f^1(x) \triangleq e^{x^1} + e^{x^2} + e^{x^3}\,, \tag{4.6}$$

$$g(x\,,t) \triangleq \frac{1}{1+t^2} - x^1 - (x^2)t - (x^3)t^2 \tag{4.7}$$

Initial points: $x_{-1} = x_0 = (1\,,0.5\,,0)^T$, solution: $(1.006605,-0.126880,-0.379725)^T$. ■

In [Pol.3], we find a method for designing stabilizing compensators for linear multi-variable feedback systems via semi-infinite optimization. We use this method here to compute a parameter vector $x \in \mathbb{R}^{13}$ (with components denoted by superscripts) which results in all the eigenvalues of the following matrix[4] having strictly negative real parts:

$$A(x) \triangleq \begin{bmatrix} 0 & 0 & -x^1 & -2x^2-4x^1 & -3x^2-3x^1 \\ 0 & 0 & -x^3 & -2x^4-4x^3 & -3x^4-3x^3 \\ x^5 & x^6 & -3 & -4 & -2 \\ 0 & 0 & 1 & 0 & 0 \\ x^7 & x^8 & 0 & -2 & -4 \end{bmatrix}. \tag{4.8}$$

As is shown in [Pol.3], the eigenvalues of the matrix $A(x)$ have strictly negative real parts if $\psi(x) \leq 0$, where $\psi(x) = \max\{\, \max_{j \in \underline{5}} f^j(x)\,,\, \max_{\omega \in [0,1]} \phi^1(x\,,\omega)\,\}$, with

$$f^j(x) \triangleq -x^{j+8} + 0.001\,, \qquad j \in \underline{5}\,. \tag{4.9}$$

and

$$\phi^1(x\,,\omega) \triangleq 0.001 - \mathrm{Re}\left[\frac{\det(j(60\omega)I - A(x))}{((j60\omega)^2 + x^9(j60\omega) + x^{10})((j60\omega)^2 + x^{11}(j60\omega) + x^{12})((j60\omega) + x^{13})}\right], \tag{4.10}$$

Note that in this problem we do not need to find a minimizer, only a point x which makes $\psi(x)$ negative. We used two different initial points, as stated below:

[4] For the purpose of testing our algorithm, we deliberately overspecified the number of design variables to make the resulting minimization problem ill-conditioned.

Problem MODNYQ1: Determine $x \in \mathbb{R}^{13}$ such that $\psi(x) \leq 0$.

Initial points: $x_{-1} = x_0 = (10,9.9,9.8,9.7,-9.6,-9.5,-9.4,-9.3,1,1,3.7341,3.4561,37.642)^T$,

our solution: $(3.58466,6.38621,2.19436,6.11969,-5.21685,-5.59284,5.30464,4.03995,0.21026,1.5782,$ $10.0921,6.74129,38.0014)^T$. ■

Problem MODNYQ2: Determine $x \in \mathbb{R}^{13}$ such that $\psi(x) \leq 0$.

Initial points: $x_{-1} = x_0 = (-1,0,0,-1,1,0,0,1,2,1,6.2055,9.1530,2)^T$,

our solution: $(-0.540925,-0.629175,0.874202,-0.577278,0.358501,0.87108,-0.757704,0.674695,$ $1.03326,1.63313,3.56932,9.40067,0.831283)^T$. ■

Table 1, below, summarizes the results in terms of the number of function evaluations (NF) and gradient evaluations (NG) required to achieve the specified accuracy. For the Problems TFI1-TFI3, we terminated computation at the first iterate x_i which satisfies the test $\|x_i - \hat{x}\|_2 < 10^{-4}$, where $\hat{x}$ is the corresponding solution. We compared the performance of Algorithm 2.2 with that of the barrier method of [Pol.2] and the linearization method in [Pol.1] (Algorithm 5.2). On Problem TFI1, all algorithms performed similarly. The linearization method [Pol.1] failed to achieve the required accuracy on Problem TFI2. A significant reduction in the number of gradient evaluations (over the method of [Pol.2]) was obtained on Problems TFI2, TFI3 and MODNYQ1. The linearization method [Pol.1] failed to obtain a solution to Problem MODNYQ1 in 200 iterations (and over 5 hours !). The difference in performance of the two barrier algorithms on Problem MODNYQ2 is not as dramatic.

Problem	Algorithm 2.2 (NF/NG)	[Pol.2] (NF/NG)	[Pol.1] (NF/NG)
TFI1	141/10	70/37	147/27
TFI2	78/42	122/74	FAILS
TFI3	33/7	34/25	63/13
MODNYQ1	43/9	63/42	FAILS
MODNYQ2	5/5	6/6	55/14

Table 1. Performance on semi-infinite problems.

Next we applied Algorithm 2.2 to the finite minimax problems below.

Problem WF: This is the example in [Wom.1] (p. 512) on which the algorithm of [Wom.1] fails to converge. Here $\psi(x) \triangleq \max_{j \in \mathbf{3}} f^j(x)$, where

$$f^1(x) \triangleq \frac{1}{2}\left[x^1 + \frac{10x^1}{(x^1+0.1)} + 2(x^2)^2\right], \tag{4.10}$$

$$f^2(x) \triangleq \frac{1}{2}\left[-x^1 + \frac{10x^1}{(x^1+0.1)} + 2(x^2)^2\right], \tag{4.11}$$

$$f^3(x) \triangleq \frac{1}{2}\left[x^1 - \frac{10x^1}{(x^1+0.1)} - 2(x^2)^2\right]. \tag{4.12}$$

Initial Points: $x_{-1} = x_0 = (3\ ,\ 1)^T$, solution: $(0\ ,\ 0)^T$. ■

Problem M: This is the second problem of [Mad.1]. Here $\psi(x) \triangleq \max_{j \in \underline{6}} f^j(x)$, where

$$f^1(x) \triangleq (x^1)^2 + (x^2)^2) + x^1x^2\ , \qquad f^2(x) \triangleq -f^1(x)\ , \tag{4.13}$$

$$f^3(x) \triangleq \sin(x^1)\ , \qquad f^4(x) \triangleq -f^3(x)\ , \tag{4.14}$$

$$f^5(x) \triangleq \cos(x^2)\ , \qquad f^6(x) \triangleq -f^5(x). \tag{4.15}$$

Initial Points: $x_{-1} = x_0 = (3\ ,\ 1)^T$, solution: $(0.453296\ ,\ -0.906592)^T$. ■

Problem RB: This is Example 1 from [Hal.1]. Here $\psi(x) \triangleq \max_{j \in \underline{4}} f^j(x)$, where

$$f^1(x) \triangleq 10(x^2 - (x^1)^2)\ , \qquad f^2(x) \triangleq -f^1(x)\ , \tag{4.16}$$

$$f^3(x) \triangleq 1 - x^1\ , \qquad f^4(x) \triangleq -f^3(x). \tag{4.17}$$

Initial Points: $x_{-1} = x_0 = (-1.2\ ,\ 1)^T$, solution: $(1\ ,\ 1)^T$. ■

Problem CB2: This is Problem CB2 of [Wom.1]. Here $\psi(x) \triangleq \max_{j \in \underline{3}} f^j(x)$, where

$$f^1(x) \triangleq (x^1)^2 + (x^2)^4\ , \tag{4.18}$$

$$f^2(x) \triangleq (2 - x^1)^2 + (2 - x^2)^2\ , \tag{4.19}$$

$$f^3(x) \triangleq 2e^{-x^1 + x^2}. \tag{4.20}$$

Initial Points: $x_{-1} = x_0 = (2\ ,\ 2)^T$, solution: $(1.139037652\ ,\ 0.89955384)^T$. ■

Problem CB3: This is Problem CB3 of [Wom.1]. Here $\psi(x) \triangleq \max_{j \in \underline{3}} f^j(x)$, where

$$f^1(x) \triangleq (x^1)^4 + (x^2)^2\ , \tag{4.21}$$

$$f^2(x) \triangleq (2 - x^1)^2 + (2 - x^2)^2\ , \tag{4.22}$$

$$f^3(x) \triangleq 2e^{-x^1 + x^2}. \tag{4.23}$$

Initial Points: $x_{-1} = x_0 = (2\ ,\ 2)^T$, solution: $(1\ ,\ 1)^T$. ■

The results obtained (and comparable results from the literature) are presented in Table 2. It should be pointed out that each algorithm has a different stopping criterion, and so care must be taken when interpreting the results. We executed both Algorithm 2.2, the barrier method of [Pol.2] and Algorithm 5.2 of [Pol.1] until the first iteration which satisfied the test $\| x_i - \hat{x} \|_2 < 10^{-4}$, where $\hat{x}$ is the solution. As in previous tables, NF refers to the number of function evaluations, and NG to the number of gradient evaluations. If these numbers are not explicitly given in the literature, we indicate this by (-). As expected, the difference in performance between Algorithm 2.2 and that of [Pol.2] is not significant. On Problems RB, WF and M, a slight reduction in NG is achieved. However, the cost of computing a solution increases on Problems CB2 and CB3. This increase can by mitigated somewhat by replacing the initialization of Step 2 (Algorithm 2.2) by

Step 2: If $i > 0$, set $\delta_i = F K'(\alpha_{i-1} - \psi(x_i))$.

where F is some number larger than 1. The effect of this change would be to essentially reduce the modified algorithm to the original method.

Problem	Algorithm 2.2	[Pol.2]	[Pol.1]	[Mur.1]	[Wom.1]	[Cha.1]	[Hal.1]
	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG
WF	27/18	25/25	FAILS		FAILS		FAILS
M	43/16	42/25	58/11	19/-			22/22
RB	50/32	87/41	56/10		37/29		21/21
CB2	35/25	24/14	150/25	6/-	12/7	21/-	11/11
CB3	36/30	33/21	40/8		6/10	8/-	9/9

Table 2. Evaluation count for finite minimax problems.

It is clear from our experimental results that Algorithm 2.2 is robust and that it is quite effective on semi-infinite minimax problems for which it was primarily intended. When applied to finite minimax problems, its performance is only fair on easy to moderately difficult problems.

5. CONCLUSION

We have presented a modification of a first-order minimax algorithm based on barrier functions [Pol.2]. The modification reduces the number of gradient evaluations required to solve a problem by adopting an ε-active technique. The technique used is new and has not appeared in the literature. Limited numerical results indicate that a substantial reduction in the number of gradient evaluations is possible, especially in the semi-infinite case.

In addition, our modification preserves two advantages of the original method [Pol.2]. The first is that no special purpose search direction routine is required (such as a quadratic program solver) which makes it particularly suitable for dedicated VLSI implementation in on-line applications where computing speed and component reliability are essential. The second advantage is that of robustness combined with reasonable speed: limited numerical experiments indicate that our algorithm does not fail when others do, that it converges linearly (with respect to the outer iterations), and that its computing times are comparable to those of other first-order minimax algorithms.

Theoretically, our algorithm can be generalized to solve problems where the "max-parameter" is an element of $[0,1]^k$ (for some integer $k > 1$) rather than $[0,1]$. This generalization requires raising the denominators of (2.14) to a suitable power. However, the problem of implementation of this new algorithm, as well as of any other currently known semi-infinite minimax algorithm, is bound to become more severe.

6. REFERENCES

[Arm.1] L. Armijo, "Minimization of Functions Having Continuous Partial Derivatives", *Pacific J. Math*, Vol. 16, pp. 1-3, 1966.

[Ber.1] C. Berge, *Topological Spaces*, Macmillan, New York, 1963.

[Cha.1] C. Charalambous and A. R. Conn, "An efficient method to solve the minimax problem directly", *SIAM Journal of Numerical Analysis*, Vol. 15, No. 1, 1978.

[Cla.1] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Wiley-Interscience, New York, 1983.

[Gol.1] D. Goldfarb, S. Mehrotra, "Relaxed Variants of Karmarkar's Algorithm for Linear Programs with Unknown Objective Value", *Mathematical Programming*, Vol. 40, No. 2, pp. 183-195, 1988.

[Gon.1] C. C. Gonzaga, "An Algorithm for Solving Linear Programming Problems in $O(n^3L)$ Operations", *University of California, Berkeley, Electronics Research Laboratory*, Memo. No. UCB/ERL M87/10, March 1987.

[Hal:1] J. Hald and K. Madsen, "Combined LP and Quasi-Newton methods for minimax optimization", *Mathematical Programming*, Vol. 20, pp. 49-62, 1981.

[Jar.1] F. Jarre, "Convergence of the Method of Analytic Centers for Generalized Convex Programs", *Schwerpunktprogramm der Deutschen Forschungsgemeinschaft - Anwendungsbezogene Optimierung und Steurerung, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg*, Report No. 67, 1988.

[Jar.2] F. Jarre, "An Implementation of the Method of Analytic Centers", *8th Conference on Analysis and Optimization of Systems*, INRIA, Antibes, France, June 1988.

[Kar.1] N. Karmarkar, "A New Polynomial-Time Algorithm for Linear Programming", *Combinatorica*, Vol. 4, pp. 373-395, 1984.

[Kle.1] R. Klessig and E. Polak, "A Method of Feasible Directions Using Function Approximations, with Applications to Min Max Problems", *Journal of Mathematical Analysis and Applications*, Vol. 41, No. 3, pp. 583-602, March 1973.

[Mad.1] K. Madsen, "An Algorithm for Minimax Solution of Overdetermined Systems of Non-linear Equations", *Journal of the Institute of Mathematics and its Applications*, Vol. 16, pp. 321-328, 1975.

[May.1] Mayne, D. Q. and Michalska, H., "Receding Horizon Control of Non-Linear Systems", *Proceedings of the 27th IEEE Conference on Decision and Control*, Austin, Texas, Dec. 1988.

[Mur.1] W. Murray and M. L. Overton, "A Projected Lagrangian Algorithm for Nonlinear Minimax Optimization", *SIAM Journal on Scientific and Statistical Computing*, Vol. 1, No. 3, 1980.

[Pol.1] E. Polak, "On the Mathematical Foundations of Nondifferentiable Optimization in Engineering Design", *SIAM Review*, Vol. 29(1), pp. 21-89, March 1987.

[Pol.2] E. Polak, J. E. Higgins and D. Q. Mayne, "A Barrier Function Method for Minimax Problems", *University of California, Berkeley, Electronics Research Laboratory*, Memo. No. UCB/ERL M88/64, May 1989.

[Pol.3] E. Polak and T. S. Wuu, "On the Design of Stabilizing Compensators via Semi-Infinite Optimization", *IEEE Transactions on Automatic Control*, Vol. AC-34, No. 2, pp. 196-200, 1989.

[Pol.4] E. Polak, S. Salcudean and D. Q. Mayne, "Adaptive Control of ARMA Plants Using Worst Case Design by Semi-Infinite Optimization", *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 5, pp. 388-397, 1987.

[Pol.5] E. Polak, *Computational Methods in Optimization: A Unified Approach*, Academic Press, 1971.

[Psh.1] B. N. Pshenichnyi and Yu. M. Danilin, Numerical methods, in *extremal problems (Chislennye metody v ekstremal'nykh zadachakh)*, Nauka, Moscow, 1975.

[Son.1] G. Sonnevend and J. Stoer, "Global Ellipsoidal Approximations and Homotopy Methods for Solving Convex Analytic Programs", *Schwerpunktprogramm der Deutschen Forschungsgemeinschaft - Anwendungsbezogene Optimierung und Steurerung, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg*, Report No. 40, January 1988.

[Son.2] G. Sonnevend, "New Algorithms in Convex Programming Based on a Notion of Centre (for Systems of Analytic Inequalities) and on Rational Extrapolation", in K.-H. Hoffmann et al., eds., *Trends in Mathematical Optimization, ISNM*, Vol. 84, Birkhauser Verlag, pp. 311-327, April 1986.

[Tan.1] Y. Tanaka, M. Fukushima and T. Ibaraki, "A comparative study of several semi-infinite nonlinear programming algorithms", *European Journal of Operational Research*, Vol. 36, pp. 92-100, 1988.

[Wom.1] R. S. Womersley, and R. Fletcher, "An Algorithm for Composite Nonsmooth Optimization Problems", *Journal of optimization theory and applications*, Vol. 48, No. 3, 1986.

[Ye.1] Y. Ye, *Interior Algorithms for Linear, Quadratic, and Linearly Constrained Convex Programming*, Ph.D. Thesis, Department of Engineering-Economic Systems, Stanford University, California, 1987.

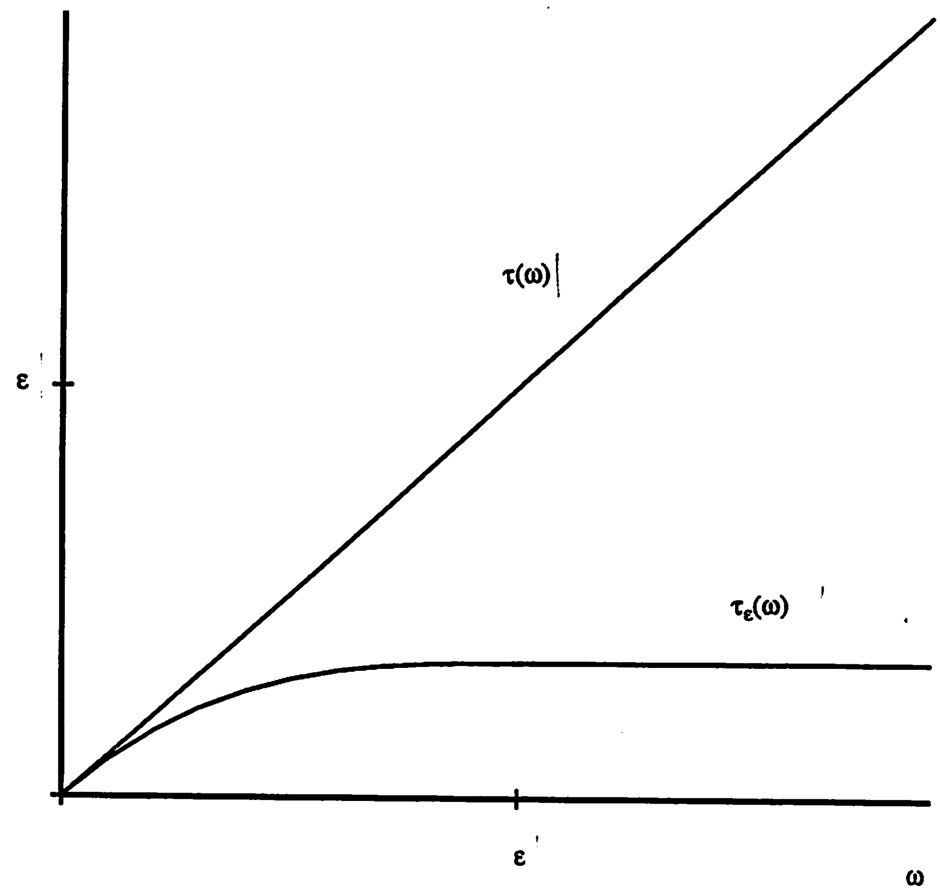

Figure 1. Graph of $\tau_\varepsilon(\cdot)$.

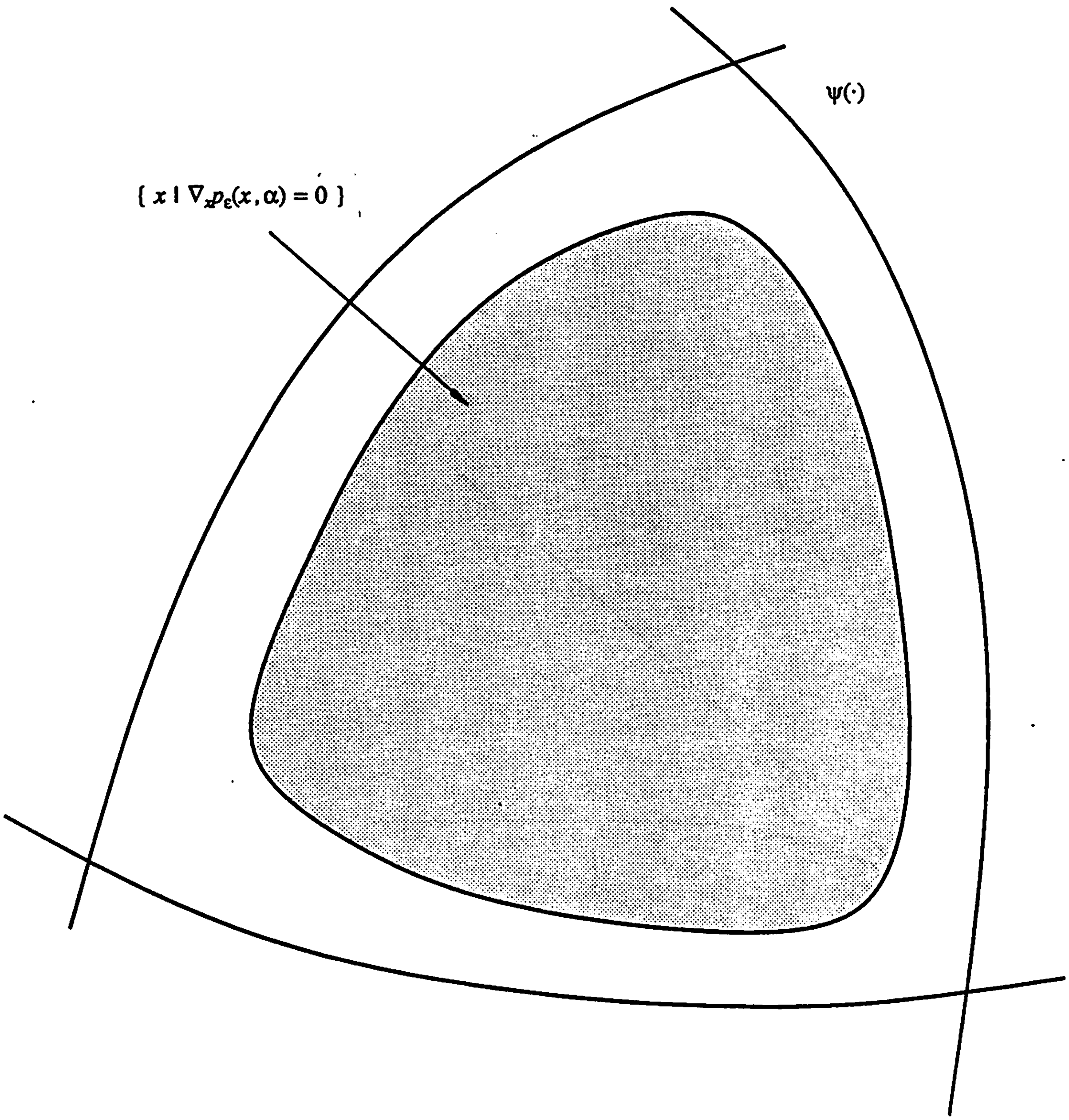

Figure 2. Illustration of the effect of a small ε.