A BARRIER FUNCTION METHOD FOR MINIMAX PROBLEMS

by

E. Polak, J. E. Higgins, and D. Q. Mayne

Memorandum No. UCB/ERL M88/64

20 October 1988
(Revised May 30, 1989)

$$\beta_i - \beta_{i_2} = \sum_{k=i_2}^{i-1} (\beta_{k+1} - \beta_k) \le \sum_{k=i_2}^{\infty} \eta_k \le \frac{\varepsilon}{2} . \tag{3.3}$$

Hence $\beta_i \le \hat{\beta} - \frac{\varepsilon}{2}$ for all i sufficiently large, which contradicts the definition of $\hat{\beta}$. It follows that $\lim_{i \to \infty} \beta_i = \hat{\beta}$. ■

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 $i \to \infty$.

Proof: Let $\beta_i \triangleq \max\{ \gamma_i , \gamma_{i-1} \}$. Clearly,

$$\gamma_{i+1} \le \max\{ \gamma_i , \gamma_{i-1} \} + \eta_i = \beta_i + \eta_i , \tag{3.4}$$

which shows that $\beta_{i+1} \le \beta_i + \eta_i$. Making use of Lemma 3.1, we conclude that either $\beta_i \to -\infty$, or else $\hat{\beta} \triangleq \lim_{i \to \infty} \beta_i$ exists. If $\beta_i \to -\infty$, then so does the sequence $\{ \gamma_i \}_{i=-1}^{\infty}$. We will show by contradiction that $\lim_{i \to \infty} \gamma_i = \hat{\beta}$. Let $\varepsilon > 0$ be arbitrary and suppose that there is no i_0 such that $\gamma_i > \hat{\beta} - \varepsilon$ for all $i \ge i_0$. Clearly, there exists an i_1 such that $\sum_{k=i}^{\infty} \eta_k < \varepsilon / 8$, and $|\beta_i - \hat{\beta}| < \varepsilon / 8$, for all $i \ge i_1$. By assumption, there exists an $i \ge i_1$, such that $\gamma_i \le \hat{\beta} - \varepsilon$. Hence, by definition of β_i, we must have that $\gamma_{i-1} = \beta_i$. Hence we obtain that

$$\gamma_{i+1} \le \tfrac{1}{2}(\gamma_i + \gamma_{i-1}) + \eta_i \le \tfrac{1}{2}(\hat{\beta} - \varepsilon + \hat{\beta} + \frac{\varepsilon}{8}) + \frac{\varepsilon}{8} \le \hat{\beta} - \frac{5}{16}\varepsilon. \tag{3.5}$$

Since $\gamma_i \le \hat{\beta} - \varepsilon$, it follows from (3.4) that $\beta_{i+1} \le \hat{\beta} - \frac{5}{16}\varepsilon$. Next, for any $j > i+1$,

$$\beta_j - \beta_{i+1} = \sum_{k=i+1}^{j-1} (\beta_{k+1} - \beta_k) \le \sum_{k=i+1}^{\infty} \eta_k \le \frac{\varepsilon}{8} . \tag{3.6}$$

Combining (3.5) and (3.6), we obtain that for all $j > i+1$,

$$\beta_j \le \hat{\beta} - \frac{3}{16}\varepsilon , \tag{3.7}$$

which contradicts the definition of $\hat{\beta}$. It follows that $\lim_{i \to \infty} \gamma_i = \hat{\beta}$. ■

A BARRIER FUNCTION METHOD FOR MINIMAX PROBLEMS

by

E. Polak, J. E. Higgins, and D. Q. Mayne

Memorandum No. UCB/ERL M88/64

20 October 1988
(Revised May 30, 1989)

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

A BARRIER FUNCTION METHOD FOR MINIMAX PROBLEMS

by

E. Polak, J. E. Higgins, and D. Q. Mayne

Memorandum No. UCB/ERL M88/64

20 October 1988
(Revised May 30, 1989)

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

A BARRIER FUNCTION METHOD FOR MINIMAX PROBLEMS*

E. Polak†, J. E. Higgins† and D. Q. Mayne‡

ABSTRACT

This paper presents an algorithm based on barrier functions, for solving semi-infinite minimax problems which arise in an engineering design setting. The algorithm bears a resemblance to some of the current interior penalty function methods used to solve constrained minimization problems. Global convergence is proven, and numerical results are reported which show that the algorithm is exceptionally robust, and that its peformance is comparable, while its structure is simpler than that of current first-order minimax algorithms.

KEY WORDS

Barrier function methods, interior penalty methods, minimax algorithms, engineering design, nondifferentiable optimization.

---

* 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, and the United Kingdom Science and Engineering Research Council.

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

‡ Department of Electrical Engineering, Imperial College, London, SW7-2BT, ENGLAND.

1. INTRODUCTION

Following Karmarkar's spectacular success in utilizing a barrier function technique in his linear programming algorithm [Kar.1], there has been a flutter of activity reevaluating homotopy and barrier function methods for both linear and nonlinear programming (see, e.g., [Gol.1, Gon.1, Jar.1, Jar.2, Son.1, Son.2, Ye.1]). Barrier function methods have considerable potential for solving minimax problems arising in engineering design (see, e.g., [Pol.1] for a discussion of these problems). These problems are often semi-infinite; in addition, because of their complexity, computation of the gradients of the component functions (of the max function) is expensive, while computation of Hessians is often impractical. Frequently, only feasibility is required, in which case the advantages of higher order methods over first-order methods are substantially reduced. In computer-aided-design applications, run times of over 100 hours are not infrequent; hence algorithms which may fail to converge, even to a local solution, are deemed undesirable. On-line applications are increasingly more common, for example, optimization for control of batch processes (see, e.g., [Pol.2]). In such applications, 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 massive subroutines.

Very few algorithms successfully address these engineering problems. We present in this paper a barrier function method which constructs solutions to semi-infinite minimax problems under hypotheses which are much less restrictive than those required by by nearly all existing algorithms. The new algorithm has a simple structure and it requires small memory (it does not utilize, for example, linear or quadratic programming subroutines). Furthermore, it has very strong theoretical and experimental convergence properties. Hence it meets the major criteria for engineering applications. The numerical performance of the new algorithm is, in most of the examples studied, superior or comparable to that of the only other first-order algorithm (Algorithm 5.2 in [Pol.2]) which can solve semi-infinite minimax problems of the same generality. Significantly, it is less affected by ill-conditioning than Algorithm 5.2 in [Pol.2]: it computes solutions for problems on which Algorithm 5.2 in [Pol.2] fails because of ill-conditioning. When applied to finite minimax problems, it is again distinguished by exceptional robustness: it does not fail on many problems which cause several excellent competing algorithms to fail. On more benign problems its performance is either comparable or not significantly inferior to that of other first-order minimax algorithms.

We address the minimax problem:

$$\mathbf{MMP}: \quad \min_{x \in \mathbb{R}^n} \psi(x) , \tag{1.1a}$$

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

$$\psi(x) \triangleq \max \left[f^1(x), \ldots f^m(x) , \max_{t \in [0,1]} \phi^1(x , t), \ldots , \max_{t \in [0,1]} \phi^l(x , t) \right] , \tag{1.1b}$$

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.

Engineering design problems often have at least one max-function ($\phi^j(\cdot , \cdot)$), arising from, for example, constraints on time or frequency responses. In addition, in many cases the presence of measurement and modeling inaccuracies often means that it is not worthwhile obtaining accurate solutions.

The essential features of the barrier function algorithm can be explained by considering the case of MMP where $\psi(\cdot)$ is defined in terms of a single function, i.e.,

$$\psi(x) \triangleq \max_{t \in [0,1]} \phi(x , t) \tag{1.2a}$$

The barrier function employed in our algorithm is defined by

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

where $\alpha > \psi(x)$. 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 shown later that $p(\cdot , \alpha)$ is a barrier function for the set $C(\alpha)$, so that, if $\{\alpha_i\}$ is a monotone decreasing sequence which converges to $\min_{x \in \mathbb{R}^n} \psi(x)$, then the sets $\operatorname{argmin}_{x \in \mathbb{R}^n} p(x , \alpha_i)$ must converge to $\operatorname{argmin}_{x \in \mathbb{R}^n} \psi(x)$. The conceptual algorithm which, at iteration i, sets $\alpha_i = \psi(x_i)$ and selects x_i as any element in $\operatorname{argmin}_{x \in \mathbb{R}^n} p(x , \alpha_i)$ generates such a sequence. Our algorithm is a practical version of the conceptual algorithm, in which, inter alia, exact minimization of $p(x , \alpha_i)$ is replaced by approximate minimization. Under mild continuity assumptions, *all* accumulation points generated by the algorithm satisfy first order optimality conditions.

The early literature on semi-infinite optimization was devoted to linear problems. A conceptual algorithm for solving nonlinaer semi-infinite optimization problems is presented in [Oet.1]. The first implementable algorithm for solving nonlinear, nonconvex semi-infinite programs, such as $\min\{ f(x) \mid \psi(x) \leq w \}$, (with $\psi(\cdot)$ defined as above, and $w = 0$), appears in [Pol.4], where a first order algorithm is described and global convergence established (all accumulation points satisfy first order conditions of optimality). An improved version of the algorithm [Gon.1] has been extensively used in complex control design problems. The algorithm can be used to solve minimax problems by replacing $f(x)$ by w, i.e., determining (x, w) to solve $\min\{ w \mid \psi(x) \leq w \}$. However, in our experience, probably because the transcribed problem is not as well conditioned, it often takes more time to solve a minimax problem in transcribed form than in original form.

A basic assumption, in all the above implementable algorithms, is that the sets $T^j \triangleq \{ t \mid \phi^j(x^*, t) = \psi(x^*) \}$, $j \in \underline{l} \triangleq \{ 1, 2, \ldots, l \}$, are finite at any local solution x^* to MMP [1]. With additional assumptions $((\phi_{tt}(x^*, t) > 0$ for all $t \in T^j$, all $j \in \underline{l})$, it is possible to obtain quadratically convergent algorithms [Het.1, May.1, Pol.5]. Similar results have been independently obtained by Coope and Watson [Coo.1], Conn and Gould [Con.1] and others. The only algorithms which dispense with the assumption that the sets T^j (of maximizers of $\phi^j(x^*, \cdot)$) are finite for all $j \in \underline{l}$ appear to be Algorithm 5.2 in [Pol.1] (a first order method which uses a proximity algorithm to generate search directions) and the algorithm presented in this paper. Global convergence has been established for these algorithms.

The literature dealing with the finite minimax problem $(\psi(x) \triangleq \min_{j \in \underline{m}} f^j(x))$ is, of course, much more extensive - see, for example, [Psh.1, Cha.1, Mur.1, Han.1, Hal.1, Wom.1, Pol.1] and the references contained therein. Both first and second order algorithms have been proposed; some of the first order algorithms achieve superlinear convergence of the Haar condition is satisfied. However, it is not clear how often this condition is satisfied in practice. Finally, by transcription into the form

[1] This assumption is not always valid for engineering design problems with constraints on dynamic responses. For example, it does not hold in the design of obstacle avoidance paths for robotic manipulators.

$$\min\{\, w \mid f^j(x) - w \le 0 \,\},\tag{1.4a}$$

one can solve the finite minimax problem using many nonlinear programming algorithms. One can generate a barrier function method for solving (1.4a), which is quite close to our minimax algorithm for the finite minimax problem, by combining the parameter free Fiacco-McCormick penalty function [Fia.1]

$$\tilde{p}(x\,,w\,,\alpha) \triangleq \frac{1}{\alpha - w} + \sum_{j=1}^{m} \frac{1}{w - f^j(x)} \tag{1.4b}$$

with the Mifflin truncation rule [Mif.1]

$$\|\nabla \tilde{p}(x_k\,,w_k\,,\alpha_{k-1})\| \le K\,,\tag{1.4b}$$

and with the Tremolieres penalty adjustment rule [Tre.1]

$$\alpha_k = \alpha_{k-1} + \tfrac{1}{2}[w_{k-1} - \alpha_{k-1}]\,.\tag{1.4c}$$

However, our numerical results in Section 5 show that this method computes considerably slower than our direct minimax algorithm.

As we have already mentioned, the performance of our algorithm on finite minimax problems is mainly distinguished by robustness: it is to be used on problems that cause other algorithms to fail. Nevertheless, we expect that the main use for our algorithm will be in solving semi-infinite engineering optimization problems, where it offers advantages of reliability, speed, and ease of on-line implementation.

In the next section we describe the conceptual and implementable algorithms and prove that the conceptual algorithm constructs a minimizing sequence. In Section 3, we establish global convergence of the implementable algorithm. The results of our numerical experiments, on both finite and semi-infinite minimax problems, are presented in Section 4. We draw a few conclusions in Section 5.

2. THE ALGORITHM

We associate with the problem MMP the barrier function

$$p(x,\alpha) \triangleq \sum_{j \in \underline{m}} \frac{1}{(\alpha - f^j(x))} + \sum_{k \in \underline{l}} \int_{[0,1]} \frac{1}{(\alpha - \phi^k(x,t))} dt , \tag{2.1}$$

where $\alpha > \psi(x)$. However, to simplify the exposition we note that, without loss of generality, we may assume that all the functions in (1.1b) are max-functions, and that $\psi(\cdot)$ is given by:

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

This follows since any ordinary function $f : \mathbb{R}^n \to \mathbb{R}$ may be trivially converted into a max-function by defining $\phi : \mathbb{R}^n \times [0,1] \to \mathbb{R}$ to be $\phi(x,t) \triangleq f(x)$. In this case, the barrier function simplifies out to the form

$$p(x,\alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{1}{(\alpha - \phi^k(x,t))} dt. \tag{2.3a}$$

We will assume that the following hypothesis holds:

Assumption 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. ■

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

$$\nabla_x p(x,\alpha) \triangleq \sum_{k \in \underline{l}} \int_{[0,1]} \frac{\nabla_x \phi^k(x,t)}{(\alpha - \phi^k(x,t))^2} dt. \tag{2.3b}$$

For ordinary functions, Assumption 1 is equivalent to requiring continuous differentiability.

It is straightforward to see, in the finite case, that the function (2.1) is a barrier function for the set $C(\alpha)$. The fact that the function (2.1) is also a barrier function follows from the following lemma. Some additional notation is necessary at this point. Define the set C by

$$C \triangleq \{ (x,\alpha) \in \mathbb{R}^{n+1} \mid \psi(x) < \alpha \}, \tag{2.4}$$

and let the ε-active sets $A_\varepsilon^k(x) \subset [0,1]$, $k \in \underline{l}$, be defined by:

$$A_\varepsilon^k(x) \triangleq \{ t \in [0,1] \mid \phi^k(x,t) \geq \psi(x) - \varepsilon \} \ , \tag{2.5}$$

Lemma 2.1: Suppose that Assumption 1 holds, and that C_o is a bounded subset of C. Then there exists a constant $L > 0$, such that for all $(x, \alpha) \in C_o$,

$$p(x,\alpha) \geq \frac{1}{L} \log \left[1 + \frac{L/2}{(\alpha - \psi(x))} \right]^2 . \tag{2.6}$$

Proof: Since C_o is bounded, so is the projection Π of C_o onto $\mathbb{R}^n$. Hence by assumption there exists a Lipschitz constant, $L < \infty$, such that each $\phi^k(x, \cdot)$ is uniformly Lipschitz in t on $[0,1]$, for all $x \in \Pi$. Without loss of generality, we may assume (since C_o is bounded) that $L \geq \alpha - \psi(x)$, for all $(x, \alpha) \in C_o$. 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

$$\phi^k(x,t) \geq \phi^k(x,t_x) - L|t - t_x| = \psi(x) - L|t - t_x| \ . \tag{2.7}$$

Consequently, we have:

$$p(x,\alpha) \geq \int_{[0,1]} \frac{1}{(\alpha - \phi^k(x,t))} \, dt \tag{2.8}$$

$$\geq \int_{[0,1]} \frac{1}{(\alpha - \psi(x) + L|t - t_x|)} \, dt \tag{2.9}$$

$$\geq \frac{1}{L} \log \left[\frac{(\alpha - \psi(x) + Lt_x)(\alpha - \psi(x) + L(1 - t_x))}{(\alpha - \psi(x))^2} \right] \tag{2.10}$$

$$\geq \frac{1}{L} \log \left[1 + \frac{L/2}{(\alpha - \psi(x))} \right]^2 , \tag{2.11}$$

where (2.11) is obtained by minimizing (2.10) with respect to t_x. ■

Consequently, if $\alpha \in \mathbb{R}$ is such that $C(\alpha) \neq \phi$ 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 $p(x_i, \alpha) \to \infty$ as $i \to \infty$, i.e., $p(\cdot, \alpha)$ is indeed a barrier function for $C(\alpha)$.

We now establish that our conjecture, made in Section 1, that the following *conceptual* algorithm constructs a minimizing sequence.

Algorithm 1 (Minimizes $\psi(\cdot)$).

Data: $x_0 \in \mathbb{R}^n$.

Step 0: Set $i = 0$.

Step 1: Set $\alpha_i \triangleq \psi(x_i)$.

Step 2: Compute $x_{i+1} \in \arg \min_{x \in C(\alpha_i)} p(x, \alpha_i)$.

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

Let $\hat{G} \triangleq \arg \min_{x \in \mathbb{R}^n} \psi(x)$. We see that Algorithm 1 defines the iteration function $A : \mathbb{R}^n \to 2^{\mathbb{R}^n}$, defined on the complement of $\hat{G}$ as follows:

$$A(x) = \arg \min_{x' \in C(\psi(x))} p(x', \psi(x)) , \quad \text{if } x \notin \hat{G} . \tag{2.12}$$

To complete the definition of $A(\cdot)$, we set

$$A(x) = \hat{G} , \text{ if } x \in \hat{G} . \tag{2.13}$$

Now suppose that the set $C(\psi(x_0))$ is bounded, and consider the sequence $\{x_i\}_{i=0}^{\infty}$ constructed by Algorithm 1. Then, there is an infinite subset $K \subset \mathbb{N}$ and vectors x^* , x^{**}, such that $x_i \overset{K}{\to} x^*$ and $x_{i+1} \overset{K}{\to} x^{**}$ as $i \to \infty$. Since the sequence $\{\psi(x_i)\}_{i=0}^{\infty}$ is monotone decreasing, and since $\psi(\cdot)$ is continuous, we must have that $\psi(x_i) \to \psi(x^*)$, as $i \to \infty$, and hence that $\psi(x^*) = \psi(x^{**})$. For the sake of contradiction, suppose that $x^* \notin \hat{G}$. Then the set $C(\psi(x^*))$ is nonempty, and hence for any $x' \in A(x^*)$, $\psi(x') < \psi(x^*)$ must hold. Let $x' \in A(x^*)$, then it follows that $x' \in C(\psi(x_i))$ for all i. By construction, $p(x_{i+1} , \psi(x_i)) \le p(x' , \psi(x_i))$. Continuity implies that $p(x' , \psi(x_i)) \to p(x' , \psi(x^*))$. However, Lemma 2.1 implies that $p(x_{i+1} , \psi(x_i)) \to \infty$, which yields a contradiction. Hence we must have that $x^* \in \hat{G}$.

To convert Algorithm 1 into an implementable form, we introduce two modifications. First, we relax Step 2, which requires that $x_{i+1} \in \arg \min_{x \in C(\alpha_i)} p(x , \alpha_i)$, by accepting *any* $x_{i+1} \in C(\alpha_i)$ which satisfies $|\nabla_x p(x_{i+1} , \alpha_i)| \le K$, for some fixed $K > 0$. Clearly, such an x_{i+1} can be computed in a finite number of operations by means of any number of descent algorithms. Next, because $x_i \notin C(\alpha_i)$ and since an initial point in $C(\alpha_i)$ is needed for the computation of x_{i+1} by means of a descent method, we replace the construction in Step 1 by setting $\alpha_i \triangleq \frac{1}{2}(\psi(x_{i-1})+\psi(x_i))$, and using either x_i or x_{i-1}

(whichever has the smaller $\psi(\cdot)$ value) as an initial point for the routine that computes x_{i+1}. It is still possible that $\psi(x_{i-1}) = \psi(x_i)$, in which case we increase α_i by a suitably small amount. The resulting implementable algorithm is as follows:

Algorithm 2 (Minimizes $\psi(\cdot)$).

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

$$\alpha_i \triangleq \begin{cases} \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) & \text{if } \psi(x_{i-1}) \neq \psi(x_i) , \\ \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) + \eta_i & \text{if } \psi(x_{i-1}) = \psi(x_i) , \end{cases} \tag{2.14}$$

$$y_i \triangleq \begin{cases} x_i & \text{if } \psi(x_{i-1}) \geq \psi(x_i) , \\ x_{i-1} & \text{if } \psi(x_{i-1}) < \psi(x_i) . \end{cases} \tag{2.15}$$

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

$$|\nabla_x p(x_{i+1} , \alpha_i)| \leq K . \tag{2.16}$$

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

Before proceeding with the proofs of convergence, we make the following observations.

(i) Step 2, of Algorithm 2, may require a few iterations of a descent method. In an effort to simplify the algorithm, one is tempted to hypothesize that a single iteration of the method of steepest descent (with exact line search on $C(\alpha_i)$), applied to $p(\cdot , \alpha_i)$, instead of the full Step 2, might suffice. Unfortunately, it is possible to show that this strategy does not work, by applying this "one-inner-iteration" algorithm to the simple problem $\min_{x \in \mathbb{R}^2} \max\{ 2x^1 + x^2 , -x^1 \}$, with $x_{-1} = (1 , 1)^T$, $x_0 = (0 , 0)^T$. The "simplified" algorithm constructs a sequence $\{x_i\}_{i=0}^{\infty}$ which converges to a non-stationary point.

(ii) The convergence properties of Algorithm 2 are unaffected when the term $\tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i))$ in (2.14) is replaced by the term $((1 - \rho)\psi(x_{i-1}) + \rho\psi(x_i))$ for any fixed $\rho \in (0 , 1)$. However, the proofs of convergence are slightly simpler for the case $\rho = \tfrac{1}{2}$ and hence we use that value in our analysis.

(iii) Because the evaluation of the barrier function may be quite expensive, conjugate gradient methods with an exact line search seem to be unsatisfactory for computing an x_{i+1} satisfying (2.16). Hence, a reasonable approach seems to be to use an algorithm of the Gauss-Newton type, which does not require second order derivatives. This algorithm is described in the Appendix.

(iv) It is possible, due to the nature of the function $p(\cdot\,,\cdot)$, that as the iterates $(x_i\,,\alpha_{i-1})$ approach a solution, the number of inner steps required to satisfy (2.16) grow rapidly. By briefly examining the proof of convergence in Section 3, it can be seen that the test (2.16) in Step 2 can be replaced by the following, much less restrictive test:

Step 2': Use any method to generate an $x_{i+1} \in C(\alpha_i)$ satisfying

$$\|\nabla_x p(x_{i+1}\,,\alpha_i)\| \le K \max\{\,1\,, \frac{1}{(\alpha_i - \psi(x_{i+1}))^{\delta}}\,\}\ , \tag{2.17}$$

where $\delta \in [0\,,1)$ (in fact, if only ordinary functions are present, this condition may be relaxed to requiring that $\delta \in [0,2)$).

(v) The sequence $\{\,\eta_k\,\}_{k=0}^{\infty}$ need not be specified completely at the outset. An essential requirement is that whenever $\psi(x_{i-1}) = \psi(x_i)$, we have $\eta_i > 0$, so that a usable starting point for Step 2 is generated. With this in mind, the sequence $\{\,\eta_k\,\}_{k=0}^{\infty}$ may be generated in the following manner. Initially choose some $\nu_0 > 0$. In iteration i, if $\psi(x_{i-1}) \neq \psi(x_i)$, then set $\eta_i = 0$ and $\nu_{i+1} = \nu_i$, otherwise set $\eta_i = \nu_i$ and $\nu_{i+1} = \nu_i / 1.1$. Clearly, the resulting sequence $\{\,\eta_k\,\}_{k=0}^{\infty}$ has a convergent sum.

(vi) Rather than using y_i as an initial point in Step 2, a variant of a homotopy method can be used to generate an alternative starting point. Ideally, we would like to compute a new starting point $x' \in C(\alpha_i)$ such that

$$\|\nabla_x p(x'\,,\alpha_i)\| \le K\ . \tag{2.18}$$

In this case, Step 2 would require zero iterations to satisfy (2.16). Since at the previous iteration, we have computed a pair $(x_i\,,\alpha_{i-1})$ satisfying

$$\|\nabla_x p(x_i\,,\alpha_{i-1})\| \le K\ . \tag{2.19}$$

a *sufficient* condition which guarantees satisfaction of (2.18) is that x' satisfies

$$\nabla_x p(x'\,,\alpha_i) = \nabla_x p(x_i\,,\alpha_{i-1})\ . \tag{2.20}$$

Using Taylor's Theorem (under suitable differentiability hypotheses and ignoring higher order terms) we expand the function $\nabla_x p(\cdot\,,\cdot)$ about the point $(x_i\,,\alpha_{i-1})$ to get

$$\nabla_x p(x'\,,\alpha_i) \cong \nabla_x p(x_i\,,\alpha_{i-1}) + p_{xx}(x_i\,,\alpha_{i-1})(x' - x_i) + p_{\alpha x}(x_i\,,\alpha_{i-1})(\alpha_i - \alpha_{i-1})\,. \tag{2.21}$$

This suggests that

$$x' \triangleq x_i - (p_{xx}(x_i\,,\alpha_{i-1}))^{-1} p_{\alpha x}(x_i\,,\alpha_{i-1})(\alpha_i - \alpha_{i-1}) \tag{2.22}$$

might be a reasonable starting point for Step 2 (assuming that $x' \in C(\alpha_i)$). A straightforward computation shows that (under smoothness assumptions) the Hessian (with respect to x) of the barrier function is given by

$$p_{xx}(x\,,\alpha) \triangleq \sum_{k \in \underline{k}} \int_{[0,1]} \left[\frac{\nabla_x \phi(x\,,t)\,\nabla_x \phi(x\,,t)^T}{(\alpha - \phi(x\,,t))^3} + \frac{\phi_{xx}(x\,,t)}{(\alpha - \phi(x\,,t))^2} \right] dt\,, \tag{2.23}$$

Since we wish to avoid computing Hessian information (and making an additional smoothness hypothesis), we approximate the Hessian $p_{xx}(x_i\,,\alpha_{i-1})$ in expression (2.22) by the positive definite matrix

$$\tilde{H}(x\,,\alpha) \triangleq \sum_{k \in \underline{k}} \int_{[0,1]} \left[\frac{\nabla_x \phi(x\,,t)\,\nabla_x \phi(x\,,t)^T}{(\alpha - \phi(x\,,t))^3} + \frac{\sigma I}{(\alpha - \phi(x\,,t))^2} \right] dt\,, \tag{2.24}$$

where $\sigma > 0$ is some fixed constant. This yields the following formula

$$x' \triangleq x_i - (\tilde{H}(x_i\,,\alpha_{i-1}))^{-1} p_{\alpha x}(x_i\,,\alpha_{i-1})(\alpha_i - \alpha_{i-1}) \tag{2.25}$$

Before using the x' estimated by this calculation, we must, of course, verify that $x' \in C(\alpha_i)$. A similar type of initialization may be obtained by replacing (2.20) by

$$\nabla_x p(x'\,,\alpha_i) \cong \lambda\, \nabla_x p(x_i\,,\alpha_{i-1})\,, \tag{2.26}$$

where $\lambda \in [0,1)$, and repeating the above expansions.

Before concluding this section, we note that we have tacitly assumed that $\psi(\cdot)$, $p(\cdot\,,\cdot)$ and $\nabla_x p(\cdot\,,\cdot)$ can be evaluated exactly. Consequently, Algorithm 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

The main theorem of this section shows that any accumulation point $\hat{x}$, of a sequence produced by Algorithm 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 $\overline{G}\psi : \mathbb{R}^n \to 2^{\mathbb{R}^{n+1}}$.

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

It is straightforward to show that $\overline{G}\psi(\cdot)$ 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 $\overline{G}\psi(\cdot)$: (i) $\overline{G}\psi(\cdot)$ is upper semi-continuous (See [Ber.1]), and (ii) $0 \in \overline{G}\psi(\hat{x})$ if and only if $0 \in \partial\psi(\hat{x})$.

The proof of convergence depends on the following two technical lemmas, which generalize the fact that a decreasing sequence either converges or diverges properly to $-\infty$.

Lemma 3.1: Suppose that the sequences of real numbers $\{ \beta_i \}_{i=0}^{\infty}$ and $\{ \eta_i \}_{i=0}^{\infty}$ satisfy the following conditions: (i) $\eta_i \geq 0$ for all $i \in \mathbb{N}$, (ii) $\sum_{i=0}^{\infty} \eta_i < \infty$, and, (iii) $\beta_{i+1} \leq \beta_i + \eta_i$, for all $i \in \mathbb{N}$. Then either the sequence $\{ \beta_i \}_{i=0}^{\infty}$ converges, or $\beta_i \to -\infty$ as $i \to \infty$.

Proof: It is clear from the assumptions that the following holds:

$$\beta_n - \beta_0 = \sum_{i=0}^{n-1} (\beta_{i+1} - \beta_i) \leq \sum_{i=0}^{\infty} \eta_i . \tag{3.2}$$

Hence, β_i is bounded from above, and therefore $\hat{\beta} \triangleq \overline{\lim}_{i \to \infty} \beta_i < \infty$ Obviously, if $\hat{\beta} = -\infty$, then $\beta_i \to -\infty$ as $i \to \infty$.

Now suppose that $\hat{\beta} > -\infty$. To prove convergence of the sequence $\{ \beta_i \}_{i=0}^{\infty}$, we will show by contradiction that $\underline{\lim}_{i \to \infty} \beta_i \geq \hat{\beta}$. Thus, let $\varepsilon > 0$ be arbitrary, and suppose that there is no i_0 such that $\beta_i > \hat{\beta} - \varepsilon$ for all $i > i_0$. Clearly, there exists an i_1 such that $\sum_{k=i}^{\infty} \eta_k < \varepsilon / 2$ for all $i \geq i_1$. It follows from our hypothesis that there exists an $i_2 \geq i_1$, such that $\beta_{i_2} \leq \hat{\beta} - \varepsilon$. It follows from (3.2) that for $i > i_2$,

$$\beta_i - \beta_{i_2} = \sum_{k=i_2}^{i-1} (\beta_{k+1} - \beta_k) \leq \sum_{k=i_2}^{\infty} \eta_k \leq \frac{\varepsilon}{2} . \tag{3.3}$$

Hence $\beta_i \leq \hat{\beta} - \frac{\varepsilon}{2}$ for all i sufficiently large, which contradicts the definition of $\hat{\beta}$. It follows that $\lim_{i \to \infty} \beta_i = \hat{\beta}$. ■

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) $\eta_i \geq 0$, for all $i \in \mathbb{N}$, (ii) $\sum_{i=0}^{\infty} \eta_i < \infty$, and (iii) $\gamma_{i+1} \leq \frac{1}{2}(\gamma_i + \gamma_{i-1}) + \eta_i$ for all $i \in \mathbb{N}$. Then either $\{ \gamma_i \}_{i=-1}^{\infty}$ converges, or $\gamma_i \to -\infty$ as $i \to \infty$.

Proof: Let $\beta_i \triangleq \max\{ \gamma_i , \gamma_{i-1} \}$. Clearly,

$$\gamma_{i+1} \leq \max\{ \gamma_i , \gamma_{i-1} \} + \eta_i = \beta_i + \eta_i , \tag{3.4}$$

which shows that $\beta_{i+1} \leq \beta_i + \eta_i$. Making use of Lemma 3.1, we conclude that either $\beta_i \to -\infty$, or else $\hat{\beta} \triangleq \lim_{i \to \infty} \beta_i$ exists. If $\beta_i \to -\infty$, then so does the sequence $\{ \gamma_i \}_{i=-1}^{\infty}$. We will show by contradiction that $\lim_{i \to \infty} \gamma_i = \hat{\beta}$. Let $\varepsilon > 0$ be arbitrary and suppose that there is no i_0 such that $\gamma_i > \hat{\beta} - \varepsilon$ for all $i \geq i_0$. Clearly, there exists an i_1 such that $\sum_{k=i}^{\infty} \eta_k < \varepsilon / 8$, and $|\beta_i - \hat{\beta}| < \varepsilon / 8$, for all $i \geq i_1$. By assumption, there exists an $i \geq i_1$, such that $\gamma_i \leq \hat{\beta} - \varepsilon$. Hence, by definition of β_i, we must have that $\gamma_{i-1} = \beta_i$. Hence we obtain that

$$\gamma_{i+1} \leq \frac{1}{2}(\gamma_i + \gamma_{i-1}) + \eta_i \leq \frac{1}{2}(\hat{\beta} - \varepsilon + \hat{\beta} + \frac{\varepsilon}{8}) + \frac{\varepsilon}{8} \leq \hat{\beta} - \frac{5}{16}\varepsilon. \tag{3.5}$$

Since $\gamma_i \leq \hat{\beta} - \varepsilon$, it follows from (3.4) that $\beta_{i+1} \leq \hat{\beta} - \frac{5}{16}\varepsilon$. Next, for any $j > i+1$,

$$\beta_j - \beta_{i+1} = \sum_{k=i+1}^{j-1} (\beta_{k+1} - \beta_k) \leq \sum_{k=i+1}^{\infty} \eta_k \leq \frac{\varepsilon}{8} . \tag{3.6}$$

Combining (3.5) and (3.6), we obtain that for all $j > i+1$,

$$\beta_j \leq \hat{\beta} - \frac{3}{16}\varepsilon , \tag{3.7}$$

which contradicts the definition of $\hat{\beta}$. It follows that $\lim_{i \to \infty} \gamma_i = \hat{\beta}$. ■

The following lemmas derive inequalities which are used in Theorem 3.5.

Lemma 3.3: Suppose that Assumption 1 holds, that $\varepsilon > 0$, and that $(x, \alpha) \in C$. Then for each $t \notin A_\varepsilon^k(x)$, with $t \in [0,1]$ and $k \in \underline{L}$

$$\frac{\alpha - \psi(x)}{(\alpha - \phi^k(x, t))^2} \le (\alpha - \psi(x)) \frac{1}{\varepsilon^2} . \tag{3.8}$$

Proof: Since $t \notin A_\varepsilon^k(x)$, we have that $\phi^k(x, t) < \psi(x) - \varepsilon$. Hence,

$$\alpha - \phi^k(x, t) > \alpha - \psi(x) + \varepsilon > \varepsilon , \tag{3.9}$$

from which the desired inequality follows. ∎

Lemma 3.4: Suppose that Assumption 1 holds, and that C_o is a bounded subset of C. Then there exists a constant $\lambda > 0$, such that for all $(x, \alpha) \in C_o$,

$$(\alpha - \psi(x)) \sum_{k \in \underline{L}} \int_{[0,1]} \frac{1}{(\alpha - \phi^k(x, t))^2} dt \ge \lambda > 0 . \tag{3.10}$$

Proof: Since C_o is bounded, so is the projection Π of C_o onto $\mathbb{R}^n$. Hence by assumption there exists a Lipschitz constant, $L < \infty$, such that each $\phi^k(x, \cdot)$ is uniformly Lipschitz in t on $[0,1]$, for all $x \in \Pi$. Without loss of generality, we may assume (since C_o is bounded) that $L \ge \alpha - \psi(x)$, for all $(x, \alpha) \in C_o$. 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

$$\phi^k(x, t) \ge \phi^k(x, t_x) - L|t - t_x| = \psi(x) - L|t - t_x| . \tag{3.11}$$

Now suppose that $0 < \varepsilon \le L$. Then $\{ t \in [0,1] \mid |t - t_x| \le \frac{\varepsilon}{L} \} \subset A_\varepsilon^k(x)$, and hence $m(A_\varepsilon^k(x)) \ge \frac{\varepsilon}{L}$, where $m(\cdot)$ denotes the Lebesque measure on $\mathbb{R}$. Hence we conclude that

$$\sum_{k \in \underline{L}} \int_{[0,1]} \frac{\alpha - \psi(x)}{(\alpha - \phi(x, t))^2} dt \ge \int_{A_\varepsilon^k(x)} \frac{\alpha - \psi(x)}{(\alpha - \phi^k(x, t))^2} dt \ge \frac{\varepsilon}{L} \frac{\alpha - \psi(x)}{(\alpha - \psi(x) + \varepsilon)^2} . \tag{3.12}$$

Setting $\varepsilon = \alpha - \psi(x)$, and $\lambda = 1 / 4L$, we obtain the desired result. ∎

The essence of the proof of Theorem 3.5 is to show that, if $x_i \xrightarrow{s} \hat{x}$, there exist elements $\xi_i \in \bar{G}\psi(x_i)$ such that $\xi_i \xrightarrow{s} 0$. Upper semi-continuity of $\bar{G}\psi(\cdot)$ allows us to conclude that $0 \in \bar{G}\psi(\hat{x})$, which is equivalent to $0 \in \partial\psi(\hat{x})$.

Theorem 3.5: Suppose that Assumption 1 holds. If $\{ x_i \}_{i=1}^{\infty}$ is any sequence produced by Algorithm 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

$$\psi(x_{i+1}) < \alpha_i \leq \tfrac{1}{2}(\psi(x_{i-1}) + \psi(x_i)) + \eta_i. \tag{3.13}$$

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 $i \in \mathbb{N}$, $i > 0$,

$$\| \nabla_x p(x_i , \alpha_{i-1}) \| \leq K . \tag{3.14}$$

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

$$\lim_{i \to \infty} (\alpha_{i-1} - \psi(x_i)) \nabla_x p(x_i , \alpha_{i-1}) = 0 . \tag{3.15}$$

For each $k \in \underline{L}$ define $\rho_i^k : [0,1] \to \mathbb{R}$ by

$$\rho_i^k(t) \triangleq \frac{\alpha_{i-1} - \psi(x_i)}{(\alpha_{i-1} - \phi^k(x_i , t))^2}. \tag{3.16}$$

Since $\{ (\alpha_{i-1} , x_i) \}_{i \in S}$ is contained in some bounded subset of C, we conclude from Lemma 3.4 that there exists a $\lambda > 0$ such that for all $i \in S$,

$$\nu_i \triangleq \sum_{k \in \underline{L}} \int_{[0,1]} \rho_i^k(t) \, dt \geq \lambda. \tag{3.17}$$

It follows from (3.15) and (3.17) that

$$\frac{1}{\nu_i} \sum_{k \in \underline{L}} \int_{[0,1]} \rho_i^k(t) \nabla_x \phi^k(x , t) \, dt \xrightarrow{S} 0. \tag{3.18}$$

Furthermore, since $\{ x_i \}_{i \in S}$ is bounded, there exists some constant B such that $\psi(x_i) - \phi^k(x_i , t) \leq B$, for all $t \in [0,1]$, and for all $i \in S$. Consequently, for any $\varepsilon > 0$,

$$\sum_{k \in \underline{l}} \int_{[0,1]} \rho_i^k(t) \, (\psi(x_i) - \phi^k(x_i, t)) \, dt \tag{3.19}$$

$$= \sum_{k \in \underline{l}} \left[\int_{A_\varepsilon^k(x_i)} \rho_i^k(t) \, (\psi(x_i) - \phi^k(x_i, t)) \, dt + \int_{A_\varepsilon^k(x_i)^C} \rho_i^k(t) \, (\psi(x_i) - \phi^k(x_i, t)) \, dt \right] \tag{3.20}$$

$$\leq \varepsilon \nu_i + (\alpha_{i-1} - \psi(x_i)) \frac{1}{\varepsilon^2} l B. \tag{3.21}$$

It follows from (3.19)-(3.21) that

$$\frac{1}{\nu_i} \sum_{k \in \underline{l}} \int_{[0,1]} \rho_i^k(t) \, (\psi(x_i) - \phi^k(x_i, t)) \, dt \xrightarrow{S} 0. \tag{3.22}$$

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

$$\xi_i \triangleq \frac{1}{\nu_i} \sum_{k \in \underline{l}} \int_{[0,1]} \rho_i^k(t) \begin{bmatrix} \psi(x_i) - \phi^k(x_i, t) \\ \nabla_x \phi^k(x_i, t) \end{bmatrix} dt \in \overline{G}\psi(x_i). \tag{3.23}$$

Since (3.18) and (3.22) imply $\xi_i \xrightarrow{S} 0$, upper semi-continuity of $\overline{G}\psi(\cdot)$ implies $0 \in \overline{G}\psi(\hat{x})$. This completes the proof. ■

4. NUMERICAL RESULTS

We will now present a number of numerical examples which illustrate the performance of Algorithm 2. Since there is a scarcity of semi-infinite minimax test problems in the literature, 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 took 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.

In our experiments, the computations in Step 2 of Algorithm 2 were carried out using Algorithm A (presented in the Appendix). To improve performance, we used the homotopy type initialization described in observation (vi) of Section 2. All the computations were performed in double precision on a Sun 3 microcomputer with a floating point accelerator. For each of the problems below, the Armijo step size parameters in Algorithm A were set to $\alpha = 0.0001$, $\beta = 0.1$. The parameters σ (of Algorithm A) and K were chosen in an ad-hoc fashion. The algorithm performance was reasonably insensitive to moderate changes of these parameters. The heuristic used for choosing these parameters was as follows: (i) Choose K so that initially, the number of inner iterations is approximately one or two, and (ii) choose σ to be small, with the proviso that the Armijo procedure (Step 3) of Algorithm A should not repeatedly choose points in $C(\alpha)^C$.

A potential numerical problem with Algorithm 2 arrises 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 with a modified version of Algorithm 5.2 in [Pol.1] (see Example 5.1 and Corollary 5.1 in [Pol.1] for details), since it seems to be the only other first-order minimax algorithm in the literature which can be proved to be globally convergent under equally weak assumptions. 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: $x_{-1} = x_0 = (1,1,1)^T$. ■

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: $x_{-1} = x_0 = (0,0,0)^T$. ■

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: $x_{-1} = x_0 = (1 , 0.5 , 0)^T$. ■

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[2] 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

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

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:

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

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

---

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

■

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

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

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 with that of the linearization method, Algorithm 5.2 in [Pol.1]. On Problems TFI1 and TFI3, both algorithms performed similarly, while the linearization method [Pol.1] fails to achieve the required accuracy on Problem TFI2. The linearization method [Pol.1] fails to obtain a solution to Problem MODNYQ1 in 200 iterations (and over 5 hours !). In this case, Algorithm 2 obtained a solution reasonably quickly. Figure 1 plots the function $\psi(\cdot)$ versus iteration number for both of these algorithms when applied to Problem MODNYQ1. The difference in performance of the two algorithms on Problem MODNYQ2 is not as dramatic, but is nonetheless substantial. For the purposes of illustration, Figure 2 shows the semi-infinite function (4.10) at various iterations for Problem MODNYQ2.

Problem	Algorithm 2 (NF/NG)	[Pol.1] (NF/NG)
TFI1	70/37	147/27
TFI2	122/74	FAILS
TFI3	34/25	63/13
MODNYQ1	63/42	FAILS
MODNYQ2	6/6	55/14

Table 1. Performance on semi-infinite problems.

To illustrate the behavior of Algorithm 2 on finite minimax problems, we begin by applying it to the following particularly difficult problem with spiral level sets, as shown in Figure 3.

Problem SPIRAL: In this problem, $n = 2$, $m = 2$ and the functions $f^j : \mathbb{R}^2 \to \mathbb{R}$, $j = 1, 2$, are defined by:

$$f^1(x) \triangleq (x^1 - \sqrt{(x^1)^2 + (x^2)^2}\cos(\sqrt{(x^1)^2 + (x^2)^2}))^2 + 0.005((x^1)^2 + (x^2)^2) \ , \tag{4.11}$$

$$f^2(x) \triangleq (x^2 - \sqrt{(x^1)^2 + (x^2)^2}\sin(\sqrt{(x^1)^2 + (x^2)^2}))^2 + 0.005\,((x^1)^2 + (x^2)^2)\,. \tag{4.12}$$

The solution of this problem is $\hat{x} = (0\,,0)^T$, and we used the initial points $x_{-1} = x_0 = (1.41831\,, -4.79462)^T$. ■

Table 2 gives the number of iterations (I), the time in seconds (T), and the number of function evaluations (NF) and gradient evaluations (NG) required to satisfy $|x_i - \hat{x}|_2 \le \varepsilon$, for various values of ε. We present figures for Algorithm 2, the linearization method of [Pol.1] (Algorithm 5.2), and the exceptionally effective combined LP/Quasi-Newton method of [Hal.1].

ε	Algorithm 2			[Pol.1]			[Hal.1]	
	I	T	NF/NG	I	T	NF/NG	I	NF/NG
5	0	0	0/0	0	0	0/0	0	0/0
4	29	9.56	469/176	224	18.7	2306/224	1743	1743/1743
3	29	9.56	469/176	398	32.58	3904/398	3903	3903/3903
2	30	11.32	562/210	640	47.48	5592/640	5005	5005/5005
1	31	15.08	782/290	732	53.32	6247/732	5248	5248/5248
1e-02	32	16.82	890/321	811	57.98	6717/811	5325	5325/5325
1e-04	33	17.68	940/335	876	61.92	7069/876	5337	5337/5337
1e-07	33	17.68	940/335	894	63.58	7270/894	5344	5344/5344
1e-10	34	19	1032/356	912	65.18	7469/912	5357	5357/5357

Table 2. Performance on Problem SPIRAL.

The behavior of Algorithm 2 on Problem SPIRAL highlights a number of points. Algorithm 2 quickly achieves a reasonable level of accuracy with relatively few function and gradient evaluations. On this difficult problem, it outperforms the methods of [Pol.1] and [Hal.1], both in terms of evaluations and time (the method of [Hal.1] required significantly more time than the other algorithms). As would be expected, the number of inner iterations (required by Algorithm A to satisfy Step 2) grows as a solution is approached. Figure 4 illustrates this by plotting the number of inner iterations versus the iteration number. For reasonable levels of accuracy, however, this growth is not too severe. Furthermore, this may be alleviated somewhat by replacing the Step 2 acceptance criterion by that of remark (iv) of Section 2.

Next we applied Algorithm 2 to the finite minimax problems below. These problems are considerably less difficult than SPIRAL.

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

$$f^1(x) \triangleq \frac{1}{2}\left[x^1 + \frac{10x^1}{(x^1 + 0.1)} + 2(x^2)^2\right], \tag{4.13}$$

$$f^2(x) \triangleq \frac{1}{2}\left[-x^1 + \frac{10x^1}{(x^1 + 0.1)} + 2(x^2)^2\right], \tag{4.14}$$

$$f^3(x) \triangleq \frac{1}{2}\left[x^1 - \frac{10x^1}{(x^1 + 0.1)} + 2(x^2)^2\right]. \tag{4.15}$$

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

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

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

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

$$f^5(x) \triangleq \cos(x^2)\,, \qquad f^6(x) \triangleq -f^5(x). \tag{4.18}$$

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

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

$$f^1(x) \triangleq 10(x^2 - (x^1)^2)\,, \qquad f^2(x) \triangleq -f^1(x)\,, \tag{4.19}$$

$$f^3(x) \triangleq 1 - x^1\,, \qquad f^4(x) \triangleq -f^3(x). \tag{4.20}$$

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

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

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

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

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

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

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

$$f^1(x) \triangleq (x^1)^4 + (x^2)^2\,, \tag{4.24}$$

$$f^2(x) \triangleq (2 - x^1)^2 + (2 - x^2)^2\,, \tag{4.25}$$

$$f^3(x) \triangleq 2e^{-x^1 + x^2}. \tag{4.26}$$

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

The results obtained (and comparable results from the literature) are presented in Table 3. 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 and Algorithm 5.2 [Pol.1] until the first iteration which satisfied the test $\|x_i - \hat{x}\|_2 < 10^{-4}$. In addition, we have compared our algorithm with the Fiacco-McCormick-Mifflin-Tremolieres (FMMT) algorithm mentioned in the introduction (1.4a-c). To provide a fair comparison, we have augmented the FMMT scheme by adding a homotopy type initialization, similar to that presented in remark (vi) of Section 2. 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 (-).

Problem	Algorithm 2	[Pol.1]	[Mur.1]	[Wom.1]	[Cha.1]	[Hal.1]	FMMT
	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG	NF/NG
WF	25/25	FAILS		FAILS		FAILS	149/88
M	42/25	58/11	19/-			22/22	187/115
RB	87/41	56/10		37/29		21/21	249/145
CB2	24/14	150/25	6/-	12/7	21/-	11/11	109/72
CB3	33/21	40/8		6/10	8/-	9/9	181/114

Table 3. Evaluation count for finite minimax problems.

It is clear from our experimental results that Algorithm 2 is exceptionally 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. However, on severely ill-conditioned problems, such as SPIRAL and WF, it has considerable advantages since it computes results when other methods fail. Finally we note that the related constrained nonlinear programming algorithm FMMT performs poorly in comparison with Algorithm 2.

5. CONCLUSION

We have presented a first-order minimax algorithm based on barrier function techniques which resemble parameter-free interior penalty function methods used in constrained optimization. The use of an Armijo-Gauss-Newton type routine which is initialized by means of a primitive homotopy approach appears to mitigate the scaling difficulties caused by the barrier functions in the inner problem.

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 (2.1) to a suitable power. However, the problem of implementation of this new algorithm, as well as of any other currently known semi-inifinite minimax algorithm, is bound to become more severe.

Our new algorithm offers two major advantages. 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.

6. APPENDIX: A PROCEDURE FOR SOLVING THE INNER PROBLEM

The *inner problem*, defined by Step 2 of Algorithm 2, requires the computation of a point $x_{i+1} \in C(\alpha_i)$ which satisfies:

$$\|\nabla_x p(x_{i+1}, \alpha_i)\| \leq K \ , \qquad \text{(A.1)}$$

for a given $K > 0$ and a given initial point $y_i \in C(\alpha_i)$. Clearly, in general, such a point can be obtained in a finite number of iterations by means of any number of descent algorithms, including the Armijo gradient method [Arm.1].

However, as a solution of Problem (MMP) is approached, some of the terms in (2.1), become very large and hence the penalty function $p(\cdot\,,\alpha_i)$, defined by (2.1) tends to become badly scaled. After some experimentation with alternatives, we have concluded that an effective way to deal with this difficulty, as well as with the fact that the penalty function can be quite expensive to evaluate (which discourages complex step size calculations, as well as computation of second order derivatives), is to solve the inner problem by means of the following Armijo-Gauss-Newton variant. In this variant, we approximate the hessian of $p(\cdot\,,\alpha)$, defined by (2.23), by the expression (2.24). The resulting procedure is as follows:

Algorithm A (Solve Step 2 of Algorithm 2)

Data: $(y_0\,,\alpha) \in C, K \geq 0, \alpha\,,\ \beta \in (0\,,1)\,,\ S \triangleq \{\beta^k\}_{k=0}^{\infty},\ \sigma > 0.$

Step 0: Set $i = 0$.

Step 1: If $\|\nabla_x p(y_i\,,\alpha)\| \leq K$, set $x_{i+1} = y_i$ and stop.

Step 2: Set $h_i = -\tilde{H}(y_i\,,\alpha)^{-1}\,\nabla_x p(y_i\,,\alpha)$.

Step 3: Compute the step size:

$$\lambda_i \triangleq \max\{\,\lambda \in S \mid y_i + \lambda h_i \in C(\alpha)\ ,\ \ p(y_i + \lambda h_i\,,\alpha) - p(y_i\,,\alpha) \leq \alpha\lambda\langle h_i\,,\nabla_x p(y_i\,,\alpha)\rangle\,\}\ .$$

Step 4: Set $y_{i+1} = y_i + \lambda_i h_i$.

Step 5: Replace i by i+1 and go to Step 1. ■

It is straightforward to prove that Algorithm A either finds a point x_{i+1} satisfying (A.1) in a finite number of iterations, or $\|y_i\| \to \infty$ as $i \to \infty$.

7. REFERENCES

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

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

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

[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).

[Con.1] A. R. Conn and N. I. M. Gould, "An exact penalty function for semi-infinite programming", *Mathematical Programming*, Vol. 37, pp. 19-40 (1987).

[Coo.1] I.D. Coope and G. A. Watson, "A projected Lagrangian algorithm for semi-infinite programming", *Mathematical Programming*, Vol. 32, pp. 337-356 (1985).

[Fia.1] A. V. Fiacco and G. P. McCormick, "The Sequential Unconstrained Minimization Technique Without Parameters", Operations Research, Vol. 15, pp. 820-227 (1967).

[Gol.1] D. Goldfarb, S. Mehrotra, "Relaxed Variants of Karmarkars Algorithm for Linear Programs with Unknown Objective Value", *Mathematical Programming*, Vol. 40(2) pp. 183-195 (1988).

[Gon.1] C. C. Gonzaga, "An Algorithm for Solving Linear Programming Problems in $o(n^3L)$ Operations", Memo. No. UCB/ERL M87/10, Electronics Research Laboratory, University of California, Berkeley, California (5 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).

[Han.1] S. P. Han, "Variable Metric Methods for Minimizing a Class of Nondifferentiable functions", Mathematical Programming, Vol. 20, No. 1, pp. 1-13, 1981.

[Het.1] R. Hettich and W. Van Honstede, "On quadratically convergent methods for semi-infinite programming", in: R. Hettich, ed., Semi-infinite Programming, Lecture notes in control and information science 15, Springer-Verlag, New York, pp. 97-111, (1979).

[Jar.1] F. Jarre, "Convergence of the Method of Analytic Centers for Generalized Convex Programs", Report No. 67, *Schwerpunktprogramm der Deutschen Forschungsgemeinschaft - Anwendungsbezogene Optimierung und Steurerung, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg* (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* 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(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] D. Q. Mayne and E. Polak, "A Quadratically Convergent Algorithm for Solving Infinite Dimensional Inequalities", *Applied Mathematics and Optimization*, Vol. 9, pp. 25-40, 1982.

[Mif.1] R. Mifflin, "Rates of Convergence for a Method of Centers Algorithm", *Journal of Optimization Theory and Applications*, Vol. 18, No. 2, pp. 199-228, February 1976.

[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.

[Oet.1] W. Oettli, "The Method of Feasible Directions for Continuous Minimax Problems", in A. Prekopa, ed., *Survey of Mathematical Programming*, North Holland Publishing Company, Vol. 1, 1979.

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

[Pol.4] E. Polak and D. Q. Mayne, "An Algorithm for Optimization Problems with Functional Inequality Constraints", *IEEE Transactions on Automatic Control*, Vol. AC-21, pp. 184-193, April 1976.

[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, J. Stoer, "Global Ellipsoidal Approximations and Homotopy Methods for Solving Convex Analytic Programs", Report No. 40, *Schwerpunktprogramm der Deutschen Forschungsgemeinschaft - Anwendungsbezogene Optimierung und Steurerung, Institut fur Angewandte Mathematik und Statistik, Universitat Wurzburg* (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 pp. 311-327, Birkhauser Verlag (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).

[Tre.1] R. Tremolieres, "La Method des Centres a Troncature Variable", Ph. D. Thesis, *University of Paris*, 1968.

[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).

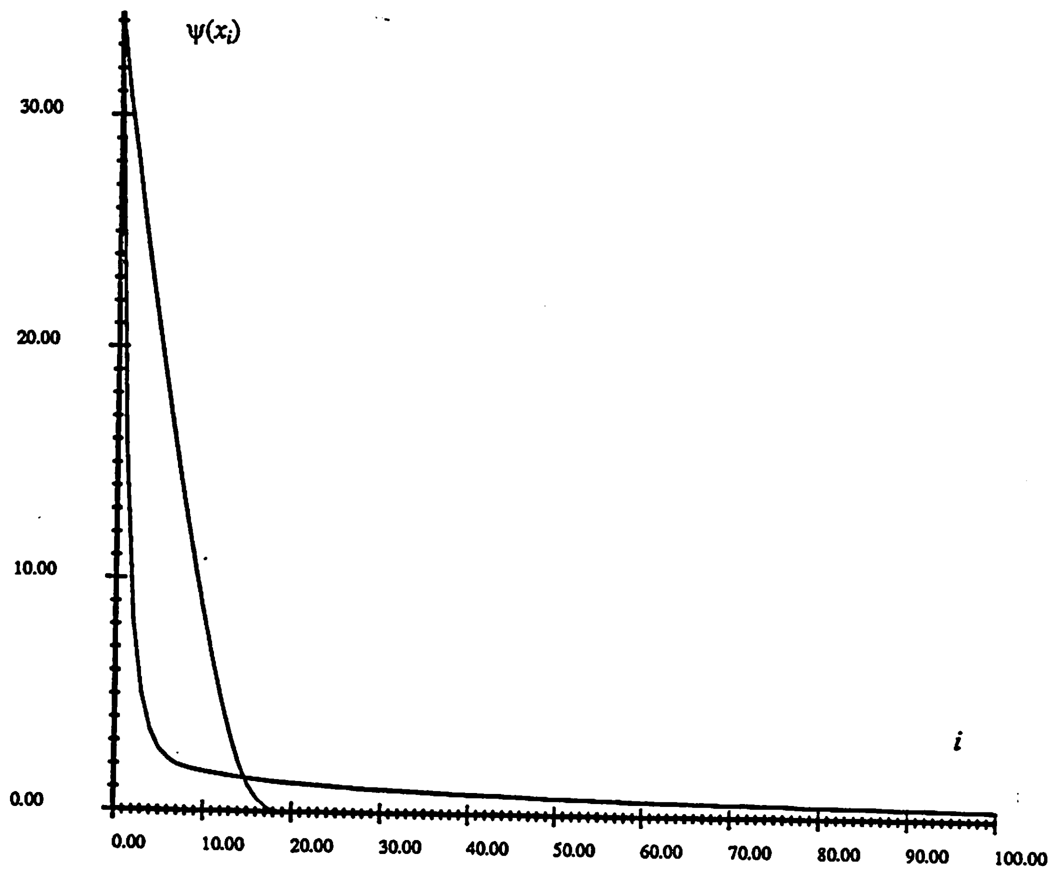

Figure 1. $\psi(x_i)$ versus iteration (i) for Problem MODNYQ1.

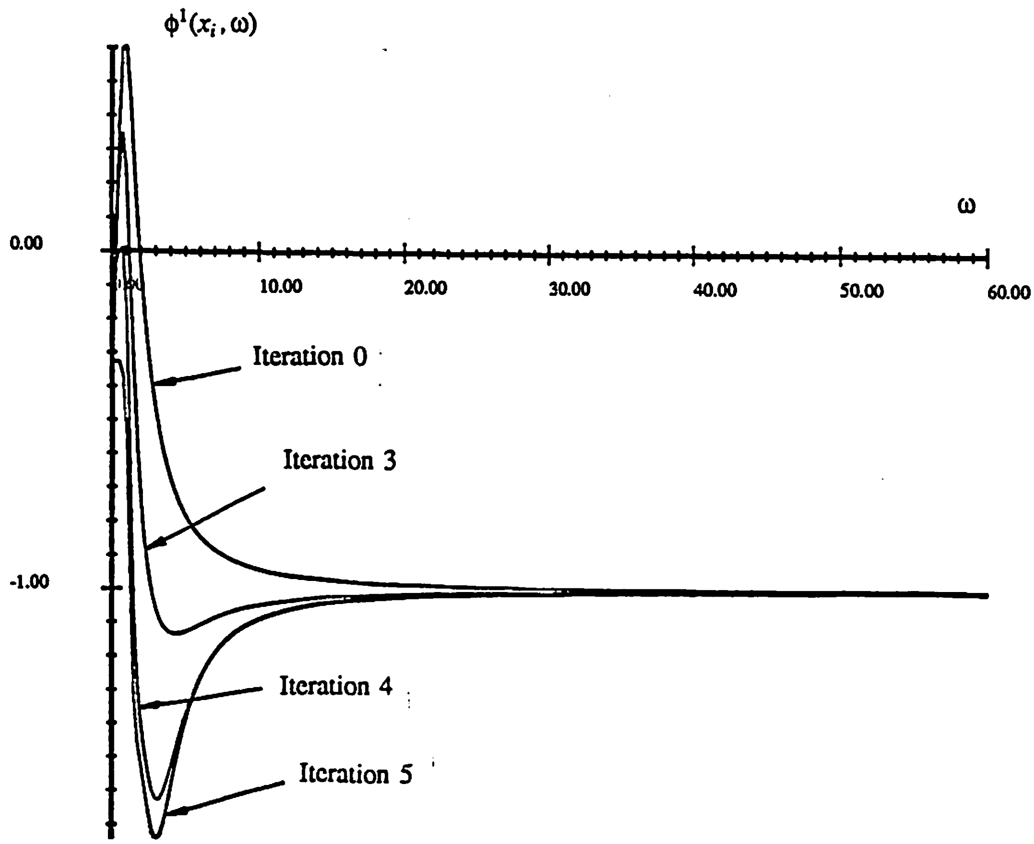

Figure 2. $\phi^1(x_i, \omega)$ for various iterations for Problem MODNYQ2.

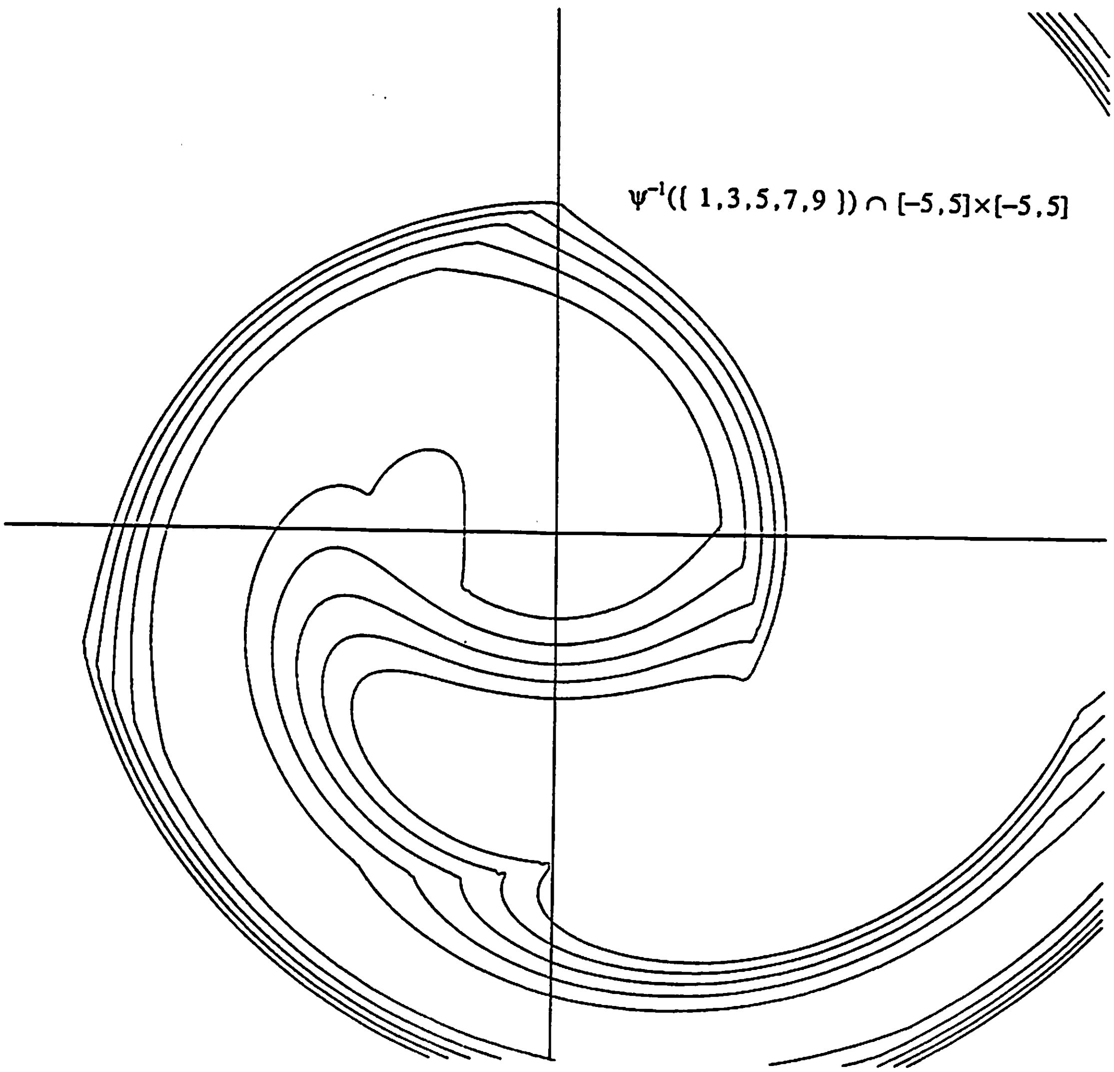

Figure 3. Contours of $\psi(\cdot)$.

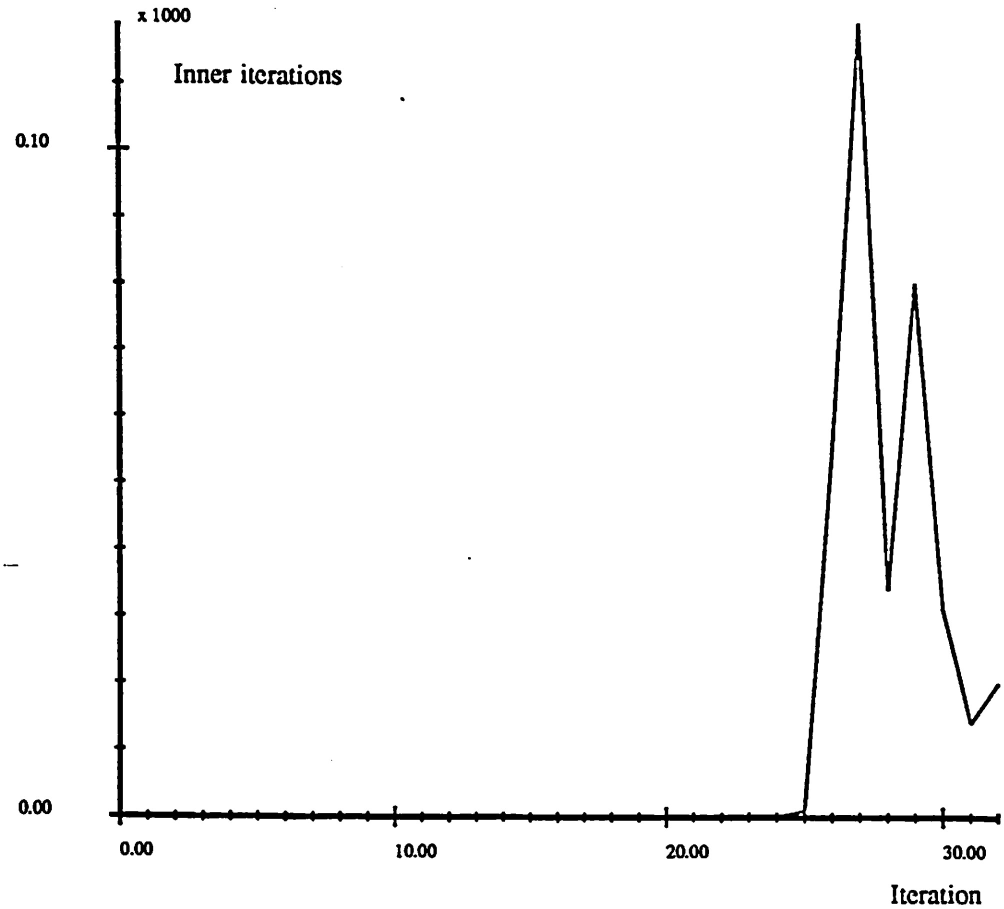

Figure 4. Inner iterations versus iteration for Problem SPIRAL.