SIAM J. NUMER. ANAL.
Vol. 15, No. 1, February 1978

AN EFFICIENT METHOD TO SOLVE THE MINIMAX PROBLEM DIRECTLY*

C. CHARALAMBOUS† AND A. R. CONN‡

Abstract. Over the past few years the circuit and system designers have shown great interest in minimax algorithms. The purpose of this paper is to present a new algorithm to solve the nonlinear minimax problem. The minimax optimization problem can be stated as:

$$\underset{x}{\text{minimize}}\ M_f(x)$$

where

$$M_f(x)=\max_{1\leqq i\leqq m} f_i(x)$$

and $x=[x_1, x_2, \cdots, x_n]^T$.

The above objective function has discontinuous first partial derivatives at points where two or more of the functions f_i are equal to M_f even if $f_i(x)$, $1\leqq i\leqq m$ have continuous first partial derivatives. Thus we cannot use directly the well known gradient methods to minimize $M_f(x)$. Unlike the work by Bandler and Charalambous where they tackle the minimax problem as a limiting case of the least pth problem (so as to overcome the difficulty of discontinuous first partial derivatives) our approach is direct. We use two distinct search directions in the algorithm. The first, the horizontal direction, attempts to reduce $M_f(x)$ whilst, at the same time, keeping those functions whose values are close to $M_f(x)$, approximately equal. The second, the vertical direction, amounts to attempting to decrease the error to within which those functions are equal to $M_f(x)$ by means of linearization.

A linear search follows after the horizontal direction has been calculated. The linear search incorporates several simple features of the algorithm and numerical results to date suggest the resulting algorithm is very efficient.

1. Introduction.

The minimax optimization problem can be stated as

$$\textbf{P}\quad \underset{x}{\text{minimize}}\ M_f(x)=\max_{i\in[M]} f_i(x) \tag{1.1}$$

where

$$[M]=[1, 2, \cdots, m], \tag{1.2}$$

$$x=[x_1\ \ x_2\cdots x_n]^T \tag{1.3}$$

and $f_1(x), f_2(x), \cdots, f_m(x)$ are in general nonlinear functions with respect to the variables, $x_1, x_2, \cdots, x_n$.

The objective function $M_f(x)$ has discontinuous first partial derivatives at points where two or more of the functions $f_i(x)$ are equal to M_f even if $f_i(x)$, $1\leqq i\leqq m$ have continuous first partial derivatives. As an illustration, Fig. 1 shows the contours for $M_f(x)$ for Example 1 viz.

Example 1.

$$f_1(x)=x_1^2+x_2^4,$$

$$f_2(x)=(2-x_1)^2+(2-x_2)^2,$$

$$f_3(x)=2\exp(-x_1+x_2).$$

Sharp corners denote points of discontinuous first partial derivatives [indicated by dotted lines in Fig. 1]. Because of these discontinuities we cannot use directly the well known gradient methods to minimize $M_f(x)$.

* Received by the editors June 20, 1975, and in revised form November 22, 1976.

† Department of Systems Design, University of Waterloo, Waterloo, Ontario, Canada.

‡ Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ontario, Canada.

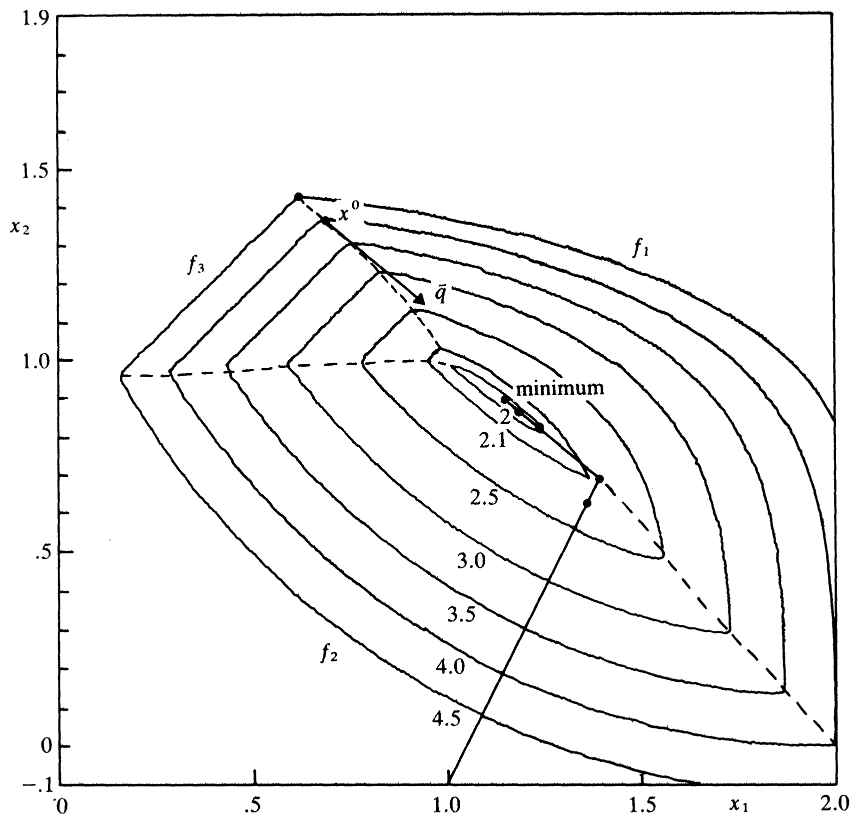

FIG. 1. *Contours of* $M_f(x)$ *for Example* 1.

The minimax problem is equivalent to the following nonlinear programming problem (see for example [19]).

$$\text{(1.4)} \qquad \mathbf{P}_1 \quad \text{minimize } z \text{ (a new independent variable)}$$

subject to the conditions

$$\text{(1.5)} \qquad \phi_i(z, x) = z - f_i(x) \geqq 0, \quad i = 1, 2, \cdots, m.$$

Now the nonlinear programming problem can be solved by any well known nonlinear programming algorithm, thus obtaining the optimum minimax solution.

Various other approaches have been proposed for solving the minimax problem, some of the most relevant of which are due to Warren, Lasdon and Suchman [19], Osborne and Watson [16], Bandler, Srinivasan and Charalambous [5], Bandler and Charalambous [3], [7], and Zangwill [20].

The first method [19] transforms the nonlinear minimax optimization problem into a nonlinear programming problem and solves it by well-established methods. The second method deals with minimax formulations by following two steps—a linear programming part which provides a given step in the parameter space, followed by a linear search along the search direction (similar algorithms have been proposed by

Ishizaki and Watanabe [14] and by Madsen [15]). The third method uses gradient information of one or more of the functions to get a downhill direction by solving a suitable linear programming problem. A linear search follows to find the minimum in that direction, and the procedure is repeated. The penultimate method [3], [7] is a generalization of the Polya algorithm [9]. A pth norm-like function is formed which has the property that if $p=\infty$, the function is equal to the maximum of the set of functions which we want to minimize.

Finally, the method of Zangwill [20] is closely related to the algorithm proposed in this paper and will be discussed in more detail below.

Consider Fig. 1 again. Let us suppose we are at the point x^0 where $f_1(x)=f_3(x)=M_f(x)$. An obvious direction to choose is direction $\bar{q}$ shown, which tries to keep both functions $f_1(x)$ and $f_3(x)$ equal. (Note that this would be exact if the functions $f_1(x)$ and $f_3(x)$ were linear). One way to obtain this direction is by using projection matrices. Consider the following nonlinear programming problem at the point x^0.

$$\text{minimize } z \text{ (a new independent variable)}$$

subject to

$$\phi_1(z,x)=z-f_1(x)\geqq 0,$$
$$\phi_3(z,x)=z-f_3(x)\geqq 0.$$

Let

$$N=\begin{bmatrix}\nabla\phi_1^T(z,x^0)\\ \nabla\phi_3^T(z,x^0)\end{bmatrix},\quad \text{where } \nabla=\left[\frac{\partial}{\partial z}\,\frac{\partial}{\partial x_1}\cdots\frac{\partial}{x_n}\right]^T.$$

Define

$$P=I-N^T[N\,N^T]^{-1}N.$$

The direction

$$q=P\nabla(-z)$$

is orthogonal to $\nabla\phi_1(z,x)$ and $\nabla\phi_3(z,x)$ and decreases z. (Note that P and q are *independent* of the value of z.) Let $\bar{q}$ be the vector obtained from q by deleting its first component. Then $\bar{q}$ has the property that it will decrease f_1 and f_2 and at the same time will try to keep them equal. Therefore $\bar{q}$ is the required direction. Now we have $\bar{q}$ we can proceed to do our minimization on the minimax function directly. It is important to note that we are not solving the nonlinear programming problem given by (1.4)–(1.5). Instead we use a similar formulation to get the direction $\bar{q}$.

In our algorithm each iteration consists of two directions. The first, the horizontal direction which tries to keep locally the same set of the functions near active (two or more functions are considered near active if they are equal to the present maximum up to a specified tolerance) and at the same time to decrease the value of M_f; the direction $\bar{q}$ in the previous example. The second, the vertical direction, amounts to attempting to satisfy the near active functions exactly by means of linearization. A linear search follows after the horizontal component has been calculated. The linear search incorporates several simple features of the algorithm and numerical results to date suggest the resulting algorithm is very efficient.

2. Notation. When we want to find the direction of search at a point x^k we set $z=z_k=M_f(x^k)$. By doing so a near active function in the minimax problem is a near active constraint in the corresponding nonlinear programming problem. More specifically,

$$E(x^k, \varepsilon) = \{i \in [M] | M_f(x^k) - f_i(x^k) < \varepsilon\}$$ (active functions at the point x^k for the minimax problem)

$$= \{i \in [M] | z_k - f_i(x^k) < \varepsilon\}$$ (active constraints at the point x^k for the corresponding nonlinear programming problem).

(Note $E(x^k, \varepsilon)$ can be considered dependent only on x^k since $z_k = M_f(x^k)$.)

$$I(x^k, \varepsilon) = [M] \backslash E(x^k, \varepsilon)$$ (inactive functions at the point x^k for the minimax problem and inactive constraints at the point x^k for the corresponding nonlinear programming problem).

$$\nabla \phi_j(z, x) = \left[\frac{\partial \phi_j(z, x)}{\partial z}, \frac{\partial \phi_j(z, x)}{\partial x_1}, \ldots, \frac{\partial \phi_j(z, x)}{\partial x_n}\right]^T = [1, -\nabla f^T(x)]^T$$

(note that $\nabla \phi_j(z, x)$ is independent of z)

$$e = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \text{ an } n+1\text{-dimensional unit vector.}$$

Obviously $\nabla z = e$.

At this point let us accept the following convention: if, in a particular context, there will be no ambiguity, we may denote functional expressions dependent upon x and sometimes ε more simply by abandoning one, or often both, of its arguments. Thus $A(x, \varepsilon)$, $E(x, \varepsilon)$, $q(x, \varepsilon)$, $S(x, \varepsilon)$, etc., will sometimes be denoted by $A(x)$, $E(x)$, $q(x)$, $S(x)$, etc., or sometimes as simply as A, E, q or S, etc.

3. The algorithm.

Step 0. Set Label $= 0$, $k = 0$, $VS = 0$, $x = x^0$, the starting point, and a value of ε, and εstop. Set $\tau_{\max}$; Note $\tau_{\max}$ is used in the linear search algorithm. It denotes an upper bound on the admissible stepsize.

(Note that Label is used to indicate whether a vertical direction should be taken (Label $= 6$) or not (otherwise). Similarly VS indicates whether the vertical step was successful ($VS = 1$) or not ($VS = 0$)).

Step 1. Set $z_k = M_f(x^k)$. At the point x^k determine the active functions within the specified tolerance. In other words, determine $E(x^k, \varepsilon)$.

Step 2. *Determine the projection matrix.* (This step involves inner iterations.) Set

$$j = 0,$$

$$p^{(0)} = I \quad \text{(the } (n+1) \times (n+1) \text{ unit matrix)},$$

$$A_0 = \varnothing \quad \text{(the empty set)},$$

and go to (3.3) below to calculate the direction of search. For an arbitrary integer $j > 0$

define

$$N^{(j)}=\begin{bmatrix}\nabla\phi_{i_1}^T(z,x^k)\\ \vdots\\ \nabla\phi_{i_j}^T(z,x^k)\end{bmatrix}, \tag{3.1}$$

i.e., $N^{(j)}$ is a $j\times(n+1)$ matrix whose rows are the gradients of some or all of the active constraints.

Also, the set $A_j=\{i_1,i_2,\cdots,i_j\}$ has been defined at the $(j-1)$ step. (We shall basically consider A_j as an ordered set. However, sometimes we shall treat it as an unordered set carefully avoiding ambiguities). Let

$$P^{(j)}=I-(N^{(j)})^T[N^{(j)}(N^{(j)})^T]^{-1}N^{(j)}; \tag{3.2}$$

$P^{(j)}$ is an $(n+1)\times(n+1)$ matrix which is a projector onto the space orthogonal to the space spanned by the vectors $\nabla\phi_{i_1}(z,x^k),\cdots,\nabla\phi_{i_j}(z,x^k)$. It is important also to note that the way the projection matrix is built up guarantees that the gradient vectors $\nabla\phi_{i_1}(z,x^k),\cdots,\nabla\phi_{i_j}(z,x^k)$ are linearly independent and therefore the matrix $N^{(j)}(N^{(j)})^T$ is nonsingular (n.b., in practice, of course, we do not in fact actually compute matrix inverses but use the iterative formulae of Rosen [18] for the nonlinear f's and methods similar to [6] for the linear functions). Set

$$q^{(j)}=P^{(j)}e \tag{3.3}$$

(note that $\nabla z=e$ and therefore the last n components of $q^{(j)}$ is an uphill direction for $M_f(x)$ at the point x^k). Set

$$i_{j+1}=\left\{\text{the } i \text{ that maximizes } \frac{q^{(j)T}\nabla\phi_i(z,x^k)}{\|q^{(j)}\|\,\|\nabla\phi_i(z,x^k)\|}\,\middle|\, i\in E(x^k,\varepsilon)\backslash A_j\right\} \tag{3.4}$$

(by $\|x\|$ we mean $(x^Tx)^{1/2}$). If

$$q^{(j)T}\nabla\phi_{i_{j+1}}(z,x^k)\leqq 0$$

go to (3.6) below. If

$$q^{(j)T}\nabla\phi_{i_{j+1}}(z,x^k)>0 \tag{3.5}$$

set

$$A_{j+1}=A_j\cup\{i_{j+1}\}$$
$$j\leftarrow j+1.$$

If the cardinality of $A_j=n+1$ go to Step 7 below, otherwise go back to (3.1)–(3.3) (that is, calculate the new projection matrix $P^{(j)}$ and the new direction $q^{(j)}$ which is now going to be orthogonal also to $\nabla\phi_{i_j}$. Then calculate i_{j+1} and continue the procedure until either (3.5) is not satisfied or the cardinality of A_j reaches $n+1$).

$$\begin{aligned}A(x^k,\varepsilon)&=A,\\ P(x^k,\varepsilon)&=P^{(j)},\\ q(x^k,\varepsilon)&=q^{(j)}.\end{aligned} \tag{3.6}$$

(Recall that q is a function of x and ε only.)

Step 3. *Check if optimum is reached.* If $\|q\|<\varepsilon$stop and Label $\neq 6$ or if $\varepsilon<\varepsilon$stop stop.

Step 4. *Linear search.* The linear search is done *directly* on the minimax function. Let $\bar{q}$ be the direction q of Step 2 with the first component deleted (remember that the first component of q corresponds to the z and $\bar{q}$ corresponds to $x \in R^n$). Then $-\bar{q}$ has the property that it will try to decrease the subset of the active functions at the point x^k, $f_{i_1}(x), f_{i_2}(x), \cdots, f_{i_j}(x)$, (i.e., those we put into the projection) by the same amount (thus decreasing $M_f(x)$).

Also, by construction, the remaining active functions at the point x^k will locally decrease along $-\bar{q}$ since this is the basis on which the projection matrix was determined, i.e., at the point x^k we move downhill along the valley defined by the minimax functions $f_{i_1}(x), f_{i_2}(x), \cdots, f_{i_j}(x)$. Note that if we consider only one function in A, say the function $f_{i_1}(x)$, then $-\bar{q} = -\nabla f_{i_1}(x)$ which is the steepest descent direction for the function $f_{i_1}(x)$. Determine $\tau > 0$ such that

$$\max f_i(x^k - \tau\bar{q}), \qquad i \in [M],$$

is minimized.

For details of how this is done and whether exact minimization is required, see the sub-algorithm "The Linear Search Algorithm" below. Put

$$x^k \leftarrow x^k - \tau\bar{q}.$$

Step 5. *Decision as to whether to do the vertical step or not.* The vertical direction amounts to attempting to make the near active functions exactly equal, and by doing so an effort is made to get *exactly* on the line of the discontinuous derivatives, which is very desirable when we are close to the solution. We expect that we will be close to the solution either when the active functions remain the same at each iteration or if we are trying to consider more than n functions in the projection. [Actually, even if we are not near the solution, should either of these two phenomena occur we might need to "reach the valley floor" in order to move away from the current situation.] Algorithmically, if the number of active functions have not changed in 3 consecutive iterations, and $\|q\| < .1$, or if Label $= 6$ (see Step 7) go to Step 6. Otherwise, set

$$x^{k+1} = x^k,$$

$$k \leftarrow k+1$$

and go to Step 1.

Step 6. *The vertical step.* Set $z_k = M_f(x^k)$. Determine $E(x^k, \varepsilon)$. (Note that x^k is the point obtained from the horizontal step.)

$$N = N^{(j)} = \begin{bmatrix} \nabla^T \phi_{i_1}(z, x^k) \\ \vdots \\ \nabla^T \phi_{i_j}(z, x^k) \end{bmatrix}.$$

N is a $j \times (n+1)$ matrix whose rows are the gradients of the active constraints which form a basis for the space spanned by the gradients of all active constraints.

Put $v(x^k, \varepsilon) = -N^T(N\,N^T)^{-1}\phi$ where $\phi^T = (\phi_{i_1}, \cdots, \phi_{i_j})$. Put

$$\begin{aligned} x_{\text{temp}} &= x^k + v \quad \text{(with 1st component missing)} \\ &= x^k + \bar{v}. \end{aligned}$$

If

$$\max_{i \in [M]} f_i(x_{\text{temp}}) < \max_{i \in [M]} f_i(x^k),$$

set $VS = 1$,

$$x^{k+1} = x_{\text{temp}}, \qquad k \leftarrow k+1,$$
$$\text{Label} = 0$$

and go to Step 1. Otherwise, set $VS = 0$,

$$x^{k+1} = x^k, \qquad k \leftarrow k+1,$$

and go to Step 1. (Note that $f_{i_l}(x^k) + \nabla f_{i_l}^T(x^k)\bar{v} = M_f(x^k) + v_1 + \varepsilon_2$ where $|\varepsilon_2| < \varepsilon$, $l = 1, 2, \cdots, j$, where v_1 is the first component of $v(x^k, \varepsilon)$. Therefore the linearized active minimax functions at x^k will be within ε at the point $x_{\text{temp}} = x^k + \bar{v}$).

Step 7. Trying to put $(n+1)$ constraints in the projection, i.e.,

$$|A_j| = n+1.$$

$|A_j| = n+1$ means one of two possibilities, either (a) we have reached the neighborhood of the optimum, or (b) we have constraints considered active that actually are not. This situation is handled in two ways. Firstly, by ensuring that we take a vertical step and secondly, by reducing ε. Algorithmically, set Label $= 6$ put $\varepsilon = \varepsilon/10$ and go to Step 6.

THE LINEAR SEARCH ALGORITHM.

Step 0. If vertical step was successful (i.e., if $VS = 1$), go to Step 4.

Step 1. Estimate any new function to become active. We consider all inactive constraints ϕ_j and estimate, in turn, the stepsize to make each ϕ_j zero. Hence we calculate

$$\tau_j = \frac{\phi_j(z_k, x^k)}{\nabla \phi_j^T(z, x^k)q}, \qquad j \in I(x^k, \varepsilon). \tag{3.7}$$

In other words, if ϕ_j was linear we calculate τ_j so that $\phi_j((z_k, x^k) - \tau_j q) = 0$ and in general we linearize the ϕ's. Let $f_m(x^k) = M_f(x^k)$ for some m, $f_j(x)$ an inactive function at the point x^k and τ_j as calculated by (3.7). Then we want the linear approximation of the functions f_m and f_j about the point x^k to be intersected at the point $x^k - \tau_j\bar{q}$. By construction q is orthogonal to $\nabla(z - f_m(x))$ at the point x^k, i.e.,

$$\nabla f_m(x^k)^T\bar{q} = q_1. \tag{3.8}$$

where

$$\nabla f_m(x) = \left[\frac{\partial f_m(x)}{\partial x_1}, \cdots, \frac{\partial f_m(x)}{\partial x_n}\right],$$

Also, by construction of τ_j,

$$z_k - f_j(x^k) - \tau_j[1, -\nabla f_j(x^k)^T]\begin{bmatrix} q_1 \\ \cdots \\ \bar{q} \end{bmatrix} = 0.$$

By using the fact that $z_k = M_f(x^k)$ we have

$$\begin{aligned} f_j(x^k) - \tau_j \nabla f_j(x^k)^T\bar{q} &= z_k - \tau_j q_1 \\ &= M_f(x^k) - \tau_j q_1 \\ &= f_m(x^k) - \tau_j q_1. \end{aligned} \tag{3.9}$$

From (3.8)

$$f_m(x^k) - \tau_j \nabla f_m(x^k)^T\bar{q} = f_m(x^k) - \tau_j q_1. \tag{3.10}$$

Comparing (3.9) and (3.10) we can see that we have the desired result.

Step 2. Omitting unlikely values of τ_j, estimate the optimum, τ, by linearizing the minimax function about x^k. For $j \in I(x^k, \varepsilon)$ do the following: If $\tau_j < 0$ or $\tau_j > \tau_{\max}$ neglect it as inadmissible. Otherwise calculate

$$\hat{f}_{ij} \triangleq f_i(x^k) - \tau_j \nabla f_i(x^k)^T \bar{q}, \qquad i \in [M],$$

where

$$\nabla f_i(x) = \left[\frac{\partial}{\partial x_i}\nabla f_i(x), \cdots, \frac{\partial}{\partial x_i}\nabla f_i(x)\right]^T.$$

Put

$$\hat{F}_j = \max_{i \in [M]} \hat{f}_{ij}.$$

Now, determine l such that

$$F_l = \min \hat{F}_j,$$
$$j \in I(x^k, \varepsilon) \backslash \{j \mid \tau_j < 0 \text{ or } \tau_j > \tau \max\}.$$

Put $\tau_{\text{opt}} = \tau_l$.

Step 3. Determine if τ_{opt} is acceptable. Calculate the true minimax value at

$$\hat{x}^k = x^k - \tau_{\text{opt}}\bar{q}.$$

If this new value is an improvement over the old value, set $x^k = \hat{x}^k$. Otherwise go to Step 4.

Step 4. Use cubic line search on the maximum of the functions taking τ_{opt} as an upper bound, to obtain the new x^k if τ_{opt} is available from Step 3 [i.e., if $VS = 0$].

Some remarks on the two algorithms. (i) One useful way of looking at the above algorithm is as follows. Firstly, the direction of search is obtained by formulating the minimax problem as a nonlinear programming problem in the standard way (see for example [19]). Whereupon the horizontal and vertical directions are obtained analogous to Conn [11] and Conn and Pietrzykowski [12]. Secondly, however, instead of using the determined horizontal direction to minimize a penalty function (as in [11] and [12]) we proceed to do our minimization on the min max function *directly*.

(ii) In practice we do not use an exact cubic linear search but merely ask for sufficient improvement in the minimax value.

(iii) If in Step 5 of the main algorithm we decided to do the vertical step and it was successful, we dispense with the estimation of τ_{opt} as above and merely do the cubic search. The motivation for this is as follows. The estimates for τ_i are based on the surmise that some new function will become active whereas the vertical step is based on the assumption that this will not be the case.

(iv) Although much of the mechanics of the Zangwill [20] algorithm are in effect equivalent to that of the horizontal direction of the above algorithm, the vertical direction is dispensed with at the cost of having to reduce the activity tolerance to zero [cf. the algorithms of Conn [11] and Conn and Pietrzykowski [12] for a similar result]. Furthermore the numerical aspects of the Zangwill algorithm are not considered by him. In particular, no linear search suitable for minimizing $M_f(x)$ is given. Furthermore, Zangwill introduces separate cases dependent on whether a certain matrix is of full or almost full ranks. Such a differentiation of cases is numerically undesirable and does not occur in the algorithm of this paper.

4. Theoretical results. We now proceed to prove the theoretical results of this paper.

In the light of the first remark above in the previous section it is not surprising that these results parallel those of Conn and Pietrzykowski in [12] with some simplifications because of the absence of the penalty function and its accompanying parameter.

We begin by proving several propositions. Define

$$h(x) = -\overset{*}{\tau}\bar{q}(x) \tag{4.1}$$

where, for theoretical purposes, we assume the linear search locates the *exact* minimum.

We will need some additional notation and two assumptions.

Assumption 1. A minimax optimum at x_0 can be characterized, as is well known (see for example [20]) by

$$\sum_{i \in E(x_0, 0)} u_i \nabla f_i(x_0) = 0,$$

$$\sum_{i \in E(x_0, 0)} u_i = 1,$$

$$u_i \geqq 0, \qquad i \in E(x_0, 0).$$

The above is equivalent to the well-known Kuhn–Tucker conditions of nonlinear programming. We shall assume that the u_i's, $i \in E(x_0, 0)$, are strictly positive. We shall refer to this as the strict complementarity assumption.

Assumption 2. Throughout this work we assume that the minimax solution is unique. We denote the corresponding x by x_0.

Additional notation: define

$$S(x, \varepsilon) = \{j \in E(x, \varepsilon) | q^T(x) \nabla \phi_j(z, x) = 0\},$$

$$B(x, \delta) = \{y \in R^n | \|y - x\| < \delta\},$$

$$D(x_0, \delta, \varepsilon) = B(x_0, \delta) \cap C(A^0, \varepsilon),$$

where

$$C(A, \varepsilon) = \{x \in R^n | E(x, \varepsilon) = A\}$$

and

$$A^0 = A(x_0, 0),$$

i.e., $D(x_0, \delta, \varepsilon)$ is the intersection of two sets—one of which is an open neighborhood of x_0 and the other of which is the set of all x such that the near active functions coincide with those that define $M_f(x_0)$.

PROPOSITION 1. *For $\varepsilon > 0$ and arbitrary compact $W \subseteq R^n$ there exists $\gamma_1 > 0$ such that*

$$M_f(x + h(x)) - M_f(x) \leqq -\gamma_1 \|\bar{q}(x)\|^2. \tag{4.2}$$

Proof.

$$\begin{aligned} M_f(x - \tau\bar{q}) &= \max_{i \in [M]} f_i(x - \tau\bar{q}) \\ &= \max_{i \in [M]} [f_i(x) - \tau\bar{q}^T \nabla f_i(x) + \tau^2 a_i(\bar{q}, x)] \end{aligned} \tag{4.3}$$

where $a_i(\cdot)$ is continuous, $\tau \in R^1$.

Since $j \in I$, $k \in E$ implies that $f_j(x) < f_k(x)$, it is clear that by choosing τ sufficiently small we may neglect all $j \in I$, and replace $[M]$ in the above by E. Therefore for τ

sufficiently small (4.3) can be replaced by

$$
\begin{aligned}
M_f(x-\tau\bar{q}) &= \max_{i\in E}\,[f_i(x)-\tau\bar{q}^T\nabla f_i(x)+\tau^2 a_i(\bar{q},x)] \\
&\leqq \max_{i\in E} f_i(x)+\max_{i\in E}\,[-\tau\bar{q}^T\nabla f_i(x)+\tau^2 a_i(\bar{q},x)].
\end{aligned}
\tag{4.4}
$$

Consider $-q$ orthogonal to the space spanned by $\nabla\phi_i$'s where $i\in A\cup S$. Then by property of projection

$$
\begin{aligned}
-q^T\nabla z &= (-P\nabla z)^T\nabla z \\
&= -\nabla z^T P^T P\nabla z \\
&= -(P\nabla z)^T P\nabla z \\
&= -\|q\|^2,
\end{aligned}
\tag{4.5}
$$

i.e., $q_1=\|q\|^2$, where q_1 is the 1st component of q. Also,

$$q^T\nabla(z-f_i)=0, \qquad i\in A\cup S,$$

i.e.

$$\bar{q}^T\nabla f_i = q_1 = \|q\|^2 \geqq \|\bar{q}\|^2. \tag{4.6}$$

Similarly for $i\in E\backslash(A\cup S)$, $q^T\nabla(z-f_i(x))<0$, and therefore it follows that (4.6) holds. Consequently

$$\max_{i\in E}\{-\tau\bar{q}^T\nabla f_i+\tau^2 a_i(\bar{q},x)\}\leqq -\gamma_1\|\bar{q}\|^2, \tag{4.7}$$

where γ_1 is chosen accordingly. But by definition (4.1)

$$M_f(x+h(x))\leqq M_f(x-\tau\bar{q}) \quad \forall\tau>0$$

and the proposition is proved.

PROPOSITION 2.

$$A(x_0,0)=E(x_0,0). \tag{4.8}$$

Proof. From the Kuhn–Tucker conditions it follows that for some $u_j>0$, $j\in[M]$,

$$\sum_{j=1}^{k} u_j\nabla f_{i_j}=0, \tag{4.9a}$$

$$\sum_{j=1}^{k} u_j = 1, \tag{4.9b}$$

where $\{i_1,\cdots,i_k\}\subseteq E(x_0,0)$.

However, the strict complementarity assumption guarantees that

$$\{i_1,\cdots,i_k\}=E(x_0,0). \tag{4.10}$$

We now note that in the proof of Theorem 1, below, we have that for an arbitrary (fixed for the rest of the proof) compact $W\subseteq R^n$,

$$\{i_1,\cdots,i_k\}\subseteq A(x_0,0),$$

[see formulae (4.53)–(4.58).]

Also, by definition $A(x_0,0)\subseteq E(x_0,0)$. Therefore

$$A(x_0,0)=E(x_0,0).$$

In what follows we shall denote $A(x_0, 0)$ by A^0.

PROPOSITION 3. *For sufficiently small positive δ, ε,*

$$A(x, \varepsilon) = A^0 \quad \textit{for } x \in D(x_0, \delta, \varepsilon). \tag{4.11}$$

[In other words for sufficiently small positive δ and ε the constraints in the projection remain the same in the neighborhood of the optimum point.]

Proof. Let us assume otherwise. Taking $\varepsilon > 0$ such that $E(x_0, \varepsilon) = A^0$ (the existence of such an ε is guaranteed by Proposition 2 and the continuity of the ϕ_i) there exists a sequence $\{(x_i, \delta_i)\}_{i \geqq 1}$ such that

$$x_i \in B(x_0, \delta_i), \quad \delta_i > 0, \quad \lim_{i \to \infty} \delta_i = 0, \tag{4.12}$$

$$A(x_i) \neq A^0. \tag{4.13}$$

Statement (4.13) can be satisfied since there is only a finite number of elements in any $A(x_i, \varepsilon)$.

From (4.12) it obviously follows that

$$\lim_{i \to \infty} x_i = x_0. \tag{4.14}$$

Statement (4.13) implies that there exists $i_l \in A^0$ such that for $j < l$, $i_j \in A^0$, $i_j \in A(x_i)$ and $i_l \notin A(x_i)$. This follows from the fact that in view of (4.14), $A(x_i) \subseteq E(x_0, \varepsilon) = A^0$ for sufficiently large i.

However, in view of the algorithm of § 3, above,

$$q_1^{(l-1)} - [\bar{q}^{(l-1)}(x_i)]^T \nabla f_{i_l}(x_i) \geqq 0. \tag{4.15}$$

Consequently, in view of the continuity of $q^{(l-1)}$ on $\{x_i\}_{i \geqq 1} \cup \{x_0\}$ we have that, using (4.14),

$$q_1^{(l-1)} - [\bar{q}^{(l-1)}(x_0)]^T \nabla f_{i_l}(x_0) \geqq 0 \tag{4.16}$$

contradicting the assumption that $i_l \in A^0$.

COROLLARY 3.1. *Under the assumption of Proposition 3, Φ, N and q are continuous on $D(x_0, \delta, \varepsilon)$.*

Now defining

$$|\Phi(z, x)| = \sum_{i \in A^0} \Phi_i(z, x), \quad \text{where } z = M_f(x),$$

we now state the following proposition.

PROPOSITION 4. *For sufficiently small positive δ, ε satisfying Proposition 3, there exists $\xi > 0$ such that*

$$\|v(x, \varepsilon)\| \leqq \xi |\Phi(z, x)| \quad \textit{for all } x \in D(x_0, \delta, \varepsilon).$$

Proof. The proof is essentially that given in [12].

PROPOSITION 5. *For sufficiently small δ, ε, there exists $\beta > 0$ such that*

$$M_f(x + \bar{v}) - M_f(x) \leqq -\beta |\Phi(z, x)|$$

for all

$$x \in D(x_0, \delta, \varepsilon).$$

Proof. By Taylor's theorem and the fact that the ϕ_j's are assumed twice continuously differentiable in the neighborhood of the optimum we deduce that there exists a continuous $a_j(x)$ such that

$$\phi_j(z+v_1, x+\bar{v}) = \phi_j(z, x) + v^T\nabla\phi_j(z, x) + \bar{v}^T a_j(x)\bar{v}.$$

Furthermore, from the definition of the vertical step,

$$\phi_j(z+v_1, x+\bar{v}) = \bar{v}^T a_j(x)\bar{v}, \qquad j\in E(x, \varepsilon). \tag{4.17}$$

Noting, from Proposition 2, that $A(x_0, 0)=E(x_0, 0)$ it follows that for sufficiently small ε, and $\delta>0$,

$$A^0 = E(x, \varepsilon) \tag{4.18}$$

for all $x\in D(x_0, \delta, \varepsilon)$. From the Kuhn–Tucker conditions (Assumption 1)

$$\sum_{i\in A^0} u_i\nabla\phi_i(z_0, x_0) = \begin{bmatrix} 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix}, \qquad u_i>0. \tag{4.19}$$

Again, using Taylor's theorem we have that

$$\nabla\phi_i(z_0, x_0) = \nabla\phi_i(z, x) + \begin{bmatrix} 0 \\ C_i(x)(x_0-x) \end{bmatrix} \tag{4.20}$$

where $C_i(x)$ is continuous, and $z_0=M_f(x_0)$. Now, substituting (4.20) into (4.19) we obtain

$$\sum_{i\in A^0} u_i\left[\nabla\phi_i(z, x) + \begin{bmatrix} 0 \\ C_i(x)(x_0-x) \end{bmatrix}\right] = \begin{bmatrix} 1 \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix}.$$

Hence,

$$\sum_{i\in A^0} u_i[\nabla\phi_i^T(z, x)v + (x_0-x)^T C_i(x)\bar{v}] = v_1. \tag{4.21}$$

Using the definition of v and (4.18) we see that (4.21) simplifies to

$$v_1 = -\sum_{i\in A^0} u_i\phi_i(z, x) + \sum_{i\in A^0} u_i(x_0-x)^T C_i(x)\bar{v}.$$

In addition, using the definition of ϕ_i, we see that (4.17) becomes

$$f_j(x+\bar{v}) - z = -\sum_{i\in A^0} u_i\phi_i(z, x) + \sum_{i\in A^0} u_i(x_0-x)^T C_i(x)\bar{v} - \bar{v}^T a_j(x)\bar{v} \tag{4.22}$$

for all $j\in E(x, \varepsilon)$.

Denoting

$$\gamma_i = \sup_{x\in D(x_0,\delta,\varepsilon)} \|C_i(x)\|,$$

$$\Gamma_i = \sup_{x\in D(x_0,\delta,\varepsilon)} \|a_i(x)\|,$$

and

$$u_{\min} = \min_{i \in A^0} (u_i) \quad [u_{\min} > 0 \text{ by Assumption 1}]$$

we see that (4.22) becomes

$$f_j(x+\bar{v}) - z \leqq -u_{\min}|\Phi(z,x)| + \|x_0 - x\| \Big(\sum_{i \in A^0} u_i \gamma_i \Big) \|\bar{v}\| + \|\bar{v}\|^2 \Gamma_j, \quad j \in E(x, \varepsilon).$$

We now make use of Proposition 4 to obtain

$$f_j(x+\bar{v}) - z \leqq \Big[-u_{\min} + \xi \|x_0 - x\| \Big(\sum_{i \in A^0} u_i \gamma_i \Big) \Big] |\Phi(z,x)| + \xi^2 |\phi(z,x)|^2 \Gamma_j, \quad j \in E(x, \varepsilon).$$

Furthermore, if we assume that

$$\delta < \frac{u_{\min}}{2\xi(\sum_{i \in A^0} u_i \gamma_i)},$$

it follows that, for $x \in D(x_0, \delta, \varepsilon)$,

$$f_j(x+\bar{v}) - z \leqq -\frac{u_{\min}}{2}|\Phi(z,x)| + \xi^2 |\Phi(z,x)|^2 \Gamma_j, \qquad j \in E(x, \varepsilon). \tag{4.23}$$

Since $|\Phi|$ is continuous and $|\Phi(z_0, x_0)| = 0$ it is clear that by choosing $\delta > 0$ sufficiently small we can assume that

$$|\Phi(z,x)| \leqq \frac{u_{\min}}{4\xi^2} \Gamma_j \quad \text{for } x \in B(x_0, \delta).$$

Consequently we have, using (4.23),

$$f_j(x+\bar{v}) - z \leqq -\frac{u_{\min}}{4}|\Phi(z,x)|, \qquad j \in E(x, \varepsilon).$$

Hence

$$M_f(x+\bar{v}) - M_f(x) \leqq -\beta |\Phi(z,x)|, \tag{4.24}$$

where $\beta = u_{\min}/4$.

PROPOSITION 6. *Let* $\{x_i\}_{i \geqq 1}$ *be a sequence with the following properties*:

$$x_i \in R^n, \tag{4.25a}$$

$$A(x_i) = \textit{constant set}, \quad A, \textit{say}, \tag{4.25b}$$

$$\lim_{i \to \infty} x_i = \bar{x}. \tag{4.25c}$$

Then

$$A \subseteq A'(\bar{x}, \varepsilon) = A(x, \varepsilon) \cup S(x, \varepsilon). \tag{4.26}$$

Proof. (See [12].)

COROLLARY 6.1. *Under the assumption of Proposition* 6 *we have that*

$$\lim_{i \to \infty} \|q(x_i)\| \geqq \|q(\bar{x})\|. \tag{4.27}$$

Proof. (The proof is essentially the same as [12].)

PROPOSITION 7. *For a compact $W \subseteq R^n$, sufficiently small ε and $x \in W$,*

$$q(x, \varepsilon) = 0 \quad \textit{implies} \quad x \in C(A^0, \varepsilon). \tag{4.28}$$

Proof. Assume (4.28) holds for no $\varepsilon > 0$. Hence, there exists a sequence $\{\langle x_i, \varepsilon_i \rangle\}_{i \geqq 1}$ such that

$$q(x_i, \varepsilon_i) = 0, \tag{4.29a}$$

$$\lim_{i \to \infty} x_i = \bar{x} \in W, \qquad x_i \in W, \tag{4.29b}$$

$$\lim_{i \to \infty} \varepsilon_i = 0, \qquad \varepsilon_i > 0, \tag{4.29c}$$

$$A(x_i, \varepsilon_i) = \text{const.} \tag{4.29d}$$

and

$$E(x_i, \varepsilon_i) \neq A_0. \tag{4.29e}$$

We first show that $\bar{x} = x_0$. For suppose otherwise. Choose $\varepsilon > 0$ such that

$$A(\bar{x}, \varepsilon) = A(\bar{x}, 0). \tag{4.30}$$

(It is always possible since there are only a finite number of functions, f_i, under consideration). From (4.29c) we have that

$$A(x_i, \varepsilon_i) \subseteq A(x_i, \varepsilon) \tag{4.31}$$

for sufficiently large i.

Thus (4.29a) implies that

$$q(x_i, \varepsilon) = 0 \tag{4.32}$$

which with Corollary 6.1 gives

$$q(\bar{x}, \varepsilon) = 0. \tag{4.33}$$

Furthermore, in view of (4.30),

$$q(\bar{x}, 0) = 0 \tag{4.34}$$

contradicting Theorem 1.

Now, recalling the definition of the q_i's and that q_i is the projection of e on the orthogonal complement of the space spanned by the $\nabla \phi_i$'s we have that

$$0 = \sum_{j \in A} \lambda_j \nabla f_j(x_i) \tag{4.35}$$

$$1 = \sum_{j \in A} \lambda_j \quad \text{and} \quad \lambda_j \geqq 0, \qquad j \in A. \tag{4.36}$$

However, Proposition 2 ensures that

$$A^0 = E(x_0, 0) \tag{4.37}$$

for sufficiently large i, and, using (4.29b) we see that

$$A \subseteq A^0. \tag{4.38}$$

Consequently, (4.35) and (4.36) may be replaced by

$$0 = \sum_{j \in A^0} \lambda_j \nabla f_j(x_i), \tag{4.39}$$

(4.40) $$1=\sum_{j\in A^0}\lambda_j,$$

where $\lambda_j=0, j\in A^0\backslash A, \lambda_j\geqq 0, j\in A$. But, then in view of the continuity of the ∇f_j's and (4.29b) we contradict our strict complementarity assumption, unless $A=A^0$, contradicting (4.29d) and (4.29e).

PROPOSITION 8. *Under the assumption of Proposition* 5, $h(\cdot)$ *is continuous in* x_0.

Proof. Let us assume otherwise. As a consequence of Theorem 1 below we have that $h(x_0)=0$. Therefore there exists $\{x_i\}_{i\geqq 1}$ and $\xi>0$ such that

(4.41) $$\lim_{i\to\infty} x_i = x_0,$$

(4.42) $$\|h(x_i)\|>\xi.$$

Since $h(x_i)$ is bounded we can additionally assume that

(4.43) $$\lim_{i\to\infty} h(x_i)=h.$$

But, in view of Proposition 1, (4.2), we have that

(4.44) $$M_f(x+h(x))-M_f(x)\leqq -\gamma_1\|\bar{q}(x)\|^2$$

which together with the continuity of M_f, (4.41) and (4.42) implies that

(4.45) $$M_f(x_0+h)-M_f(x_0)\leqq 0$$

contradicting the assumption that x_0 is a strong minimum of M_f. Hence our proposition is proved.

Our final proposition investigates the asymptotic properties of q.

PROPOSITION 9. *For arbitrary compact* $W\subset R^n$ *and sufficiently small positive* δ *and* ε,

(4.46) $$\inf_{x\in W\backslash D(x_0,\delta,\varepsilon)}\|q(x,\varepsilon)\|>0.$$

Proof. See [12].

THEOREM 1. *Let the functions* f_i, $i\in[M]$, *be continuously differentiable in a neighborhood of* x_0. *Furthermore, let Assumptions* 1 *and* 2 *above hold. Then, for* f_i *convex and for a compact* $W\subseteq R^n$ *a necessary and sufficient condition for*

(4.47) $$x\in W \quad and \quad P(x,0)e=0$$

is that

(4.48) $$x=x_0.$$

Proof of Theorem 1. We first prove necessity. If $x=x_0$, then, according to the Kuhn–Tucker conditions along with the strict complementarity assumptions we have that

(4.49) $$\sum_{j=1}^{k} u_j\nabla f_{i_j}(x_0)=0,$$

(4.50) $$\sum_{j=1}^{k} u_j=1,$$

(4.51) $$u_j>0, \qquad j=1,2,\cdots,k$$

where

$$E(x_0, 0) = \{i_1, i_2, \cdots, i_k\}. \tag{4.52}$$

We will now show that

$$E(x_0, 0) \subseteq A(x_0, 0). \tag{4.53}$$

Let us assume that (4.53) is false. So there is $l < m$ such that for $l \leqq j \leqq k$, $i_j \notin A(x_0, 0)$ and for $1 \leqq j < l$, $i_j \in A(x_0, 0)$. However, as a consequence of the algorithm we thereby know that

$$[\nabla \phi_{i_j}(z_0, x_0)]^T P^{l-1}(x_0) e \leqq 0, \qquad l \leqq j \leqq k. \tag{4.54}$$

Using (4.49) and (4.50) and remembering that $P^{l-1}(x_0)e$ is orthogonal to all $\nabla \phi_{i_j}(z_0, x_0)$ for $1 \leqq j < l$ we obtain

$$P^{(l-1)}(x_0)e = \sum_{j=l}^{k} u_j P^{(l-1)}(x_0) \nabla \phi_{i_j}. \tag{4.55}$$

Since $P^{(l-1)}(x_0)$ is an orthogonal projection

$$p^T P^{(l-1)}(x_0) q = [P^{(l-1)}(x_0) p]^T P^{(l-1)}(x_0) q \tag{4.56}$$

for any vectors p and q.

Now, from (4.54), (4.55) and (4.56) we obtain

$$-\sum_{j_2=l}^{k} u_{j_2} \alpha_{j_1 j_2} \geqq 0 \quad \text{for } l \leqq j_1 \leqq k \tag{4.57}$$

where

$$\alpha_{j_1 j_2} = [P^{(l-1)}(x_0) \nabla \phi_{i_{j_1}}(z_0, x_0)]^T P^{(l-1)}(x_0) \nabla \phi_{i_{j_2}}. \tag{4.58}$$

But by our strict complementarity assumption $\nabla \phi_{ij} (l \leqq j \leqq k)$ are linearly independent so in view of (4.58) the matrix

$$\alpha = [\alpha_{j_1 j_2}], \qquad l \leqq j_1, j_2 \leqq k,$$

is positive definite [cf. Gram determinant] contradicting (4.51).

It now follows immediately from the definition of the algorithm that

$$P(x_0) = P^{(l-1)}(x_0)$$

is such that

$$P(x_0) \nabla \phi_{i_j} = 0 \quad \text{for all } i_j \in A(x_0, 0). \tag{4.59}$$

Consequently (4.49), (4.50) and (4.53) imply that

$$P(x_0)e = 0$$

which proves the necessity part. We will now prove sufficiency.

First let us notice that since Pe is the projection of e on to the hyperplane orthogonal to the one spanned by $\nabla \phi_{i_1}, \cdots, \nabla \phi_{i_k}$, (4.47) implies that

$$e = \lambda_1 \nabla \phi_{i_1} + \lambda_2 \nabla \phi_{i_2} + \cdots + \lambda_k \nabla \phi_{i_k} \tag{4.60}$$

where $A(x_0, 0) = \{i_1, \cdots, i_k\}$.

We shall now prove that all the λ's are positive.

Suppose otherwise; then there exists $l<k$ say, such that for $l \leqq j \leqq k$,

$$\lambda_j \leqq 0. \tag{4.61}$$

But from the algorithm it follows that

$$-[\nabla \phi_{i_j}]^T P^{(l-1)} e < 0, \qquad l \leqq j \leqq k, \tag{4.62}$$

and

$$P^{(l-1)} \nabla \phi_{i_j} = 0, \qquad j=1, \cdots, (l-1). \tag{4.63}$$

Now, using (4.51), (4.56), (4.60) and (4.63) we see that

$$-\sum_{j_2=l}^{k} \lambda_{j_2} \alpha_{j_1 j_2} = -[\nabla \phi_{i_{j_1}}]^T P^{l-1} e, \qquad l \leqq j_1 \leqq k. \tag{4.64}$$

But the (4.62) implies that

$$-\sum_{j_2=l}^{k} \lambda_{j_2} \alpha_{j_1, j_2} < 0 \tag{4.65}$$

which with (4.61) contradicts the fact that $[\alpha_{j_1 j_2}]$, $l \leqq j_1, j_2 \leqq k$, is positive definite. But (4.60) with λ's positive are sufficient conditions for a minimax optimum and so our theorem is proved.

We now state and prove Theorem 2.

THEOREM 2. *Assume f_i, $i \in [M]$, are convex functions that are twice differentiable and that x_0 is the strong minimum of $M_f(x)$ and that the strict complementarity assumption is satisfied. Then for any compact $W \subseteq R^n$ and sufficiently small positive ε a sequence $\{x_i(\varepsilon)\} \subseteq W$, generated by the algorithm above is convergent to x_0.*

Proof. Let us first notice that

$$M_f(x+w) \leqq M_f(x+h), \qquad x \in R^n, \tag{4.66a}$$

where

$$w = \begin{cases} \bar{v}+h & \text{if } M_f(x+v+h) < M_f(x+h), \\ h & \text{otherwise.} \end{cases} \tag{4.66b}$$

Let ε and δ' satisfy all the requirements of the propositions stated above and additionally let $\delta>0$ be small enough such that

$$x \in B(x_0, \delta) \quad \text{implies that} \quad x+h(x) \in B(x_0, \delta'). \tag{4.67}$$

That (4.67) is possible follows from Proposition 8.

We now show that under these circumstances

$$w(x) = h(x) + \bar{v}(x), \qquad x \in D(x_0, \delta, \varepsilon). \tag{4.68}$$

To verify (4.68) let us note that from Proposition 5 [(4.19)]

$$M_f(x+h+\bar{v}) - M_f(x+h) \leqq -\beta |\Phi(z+\tilde{v}_1, x+h)|$$

where

$$z + \tilde{v}_1 = M_f(x+h).$$

Consequently, in view of (4.66b), (4.68) holds.

Furthermore (4.68), Proposition 1 [(4.2)] and Proposition 5 [(4.19)] give the following:

$$M_f(x+w)-M_f(x)=M_f(x+h+\bar{v})-M_f(x+\bar{v})+M_f(x+\bar{v})-M_f(x) \tag{4.69}$$
$$\leqq -\hat{e}(x) \quad \text{for } x\in D(x_0,\delta,\varepsilon),$$

where $\hat{e}(x)=\gamma_1\|\bar{q}(x+\bar{v})\|^2+\beta|\Phi(z,x)|$.

The function $\hat{e}$ has the following properties for $x\in D(x_0,\delta,\varepsilon)$:

$$\hat{e}(x)\geqq 0, \tag{4.70a}$$

$$\hat{e}(x)=0 \quad \text{implies} \quad x=x_0, \tag{4.70b}$$

$$\hat{e}(\cdot) \quad \text{is continuous.} \tag{4.70c}$$

Property (4.70a) is obvious, (4.70b) follows since $\hat{e}(x)=0$ implies that $\phi(z,x)=0$ and $q(x+\bar{v})=0$ which with Theorem 1 gives $x+\bar{v}=x_0$ which from Proposition 5 and our assumption of a unique minimax solution gives $x=x_0$. The continuity of $\hat{e}$ follows directly from Corollary 3.1.

Now, let $W\subset R^n$ be a compact set, $a\in R^n$ and δ,ε satisfy the requirements of Proposition 9 and (4.69) with respect to W.

Let $\{x_i(\varepsilon)\}_{i\geqq 0}$ be the sequence generated by the above algorithm with $x_1(\varepsilon)=a$ and $x_i(\varepsilon)\in W$ for $i\geqq 1$.

We first note that

$$M_f(x_{i+1})\leqq M_f(x_i) \quad \text{for all } i\geqq 1. \tag{4.71}$$

Now, let us assume that our theorem is false. Thus, there exists a convergent subsequence $\{x_{i_j}\}_{i\geqq 1}$ such that

$$\lim_{j\to\infty} x_{i_j}=\bar{x}\neq x_0 \tag{4.72}$$

and

$$\{x_{i_j}\}_{j\geqq 1}\subseteq W\backslash D(x_0,\delta,\varepsilon) \tag{4.73a}$$

or

$$\{x_{i_j}\}_{j\geqq 1}\subseteq D(x_0,\delta,\varepsilon). \tag{4.73b}$$

Obviously, in view of (4.71),

$$M_f(x_{i_{j+1}})\leqq M_f(x_{i_j+1}). \tag{4.74}$$

If case (4.73a) holds it follows from Propositions 1 and 7 that for some $\xi_1>0$,

$$M_f(x_{i_{j+1}})-M_f(x_{i_j+1})\leqq -\xi_1. \tag{4.75}$$

In case (4.73b) holds, obviously, in view of (4.72),

$$\bar{x}\in D(x_0,\delta,\varepsilon).$$

Therefore (4.70b) and (4.72) imply that

$$\hat{e}(\bar{x})>0.$$

Using (4.70c) and (4.72) we deduce that there exists $\xi_2>0$ and j_0 such that

$$\hat{e}(x_{i_j})\geqq \xi_2 \quad \text{for } j\geqq j_0. \tag{4.76}$$

However, (4.76) combined with (4.69) gives

$$M_f(x_{i_{j+1}})-M_f(x_{i_j})\leqq -\xi_2. \tag{4.77}$$

Now, putting $\xi_3 = \min(\xi_1, \xi_2)$, we obtain from (4.74), (4.75) and (4.77),

$$M_f(x_{i_{j+1}}) - M_f(x_{i_j}) \leqq -\xi_3 \quad \text{for } j \geqq j_0.$$

In addition, summation of the above inequality leads to

$$M_f(x_{i_j}) - M_f(x_{i_{j_0}}) \leqq -(j - j_0)\xi_3$$

for $j \geqq j_0$, which in view of (4.53) implies that

$$M_f(\bar{x}) = -\infty$$

which contradicts the continuity of M_f and completes the proof.

This completes the theoretical results.

5. Examples and numerical results. Five examples are going to be considered so as to illustrate the usefulness of this approach to nonlinear minimax optimization. The value of ε was fixed at 0.1 with a facility for reduction if results indicated that ε was too large (see the algorithm above).

5.1. Example 1 (see [7]). This is the example given in the introduction. When $x_1 = x_2 = 1$, $f_1 = f_2 = f_3$, but this point is not a minimax optimum because the necessary conditions for a minimax optimum are not satisfied. The minimax optimum is defined by the functions f_1 and f_2 at $x_1 = 1.13904$, $x_2 = 0.89956$ where $f_1 = f_2 = 1.95222$ and $f_3 = 1.57408$. Figure 1 shows contours of $M_f(x)$ for this problem.

Starting at the point $(1., -0.1)$ the algorithm generated the sequence of points shown in Table 1. From this table one can see the usefulness of the vertical step and that of the linear search. In Fig. 1 we show the path taken by the algorithm.

TABLE 1

Results for Example 1 *starting at* $x_1 = 1, x_2 = -0.1$.

Number of iterations	Number of function evaluations	x_1	x_2	$M_f(x)$
0	1	1.	−0.1	5.41
1	2	1.344212	0.622844	2.326617
2	3	1.375465	0.688476	2.116580
3	6	1.222938	0.843783	2.002479
4	9	1.174607	0.884782	1.992538
5	12	1.158160	0.897937	1.991438
5	13	1.152752	0.888806	1.952899
6	15	1.136205	0.901962	1.952802
6	16	1.136107	0.901847	1.952247
7	18	1.138047	0.900336	1.952232
7	19	1.138047	0.900335	1.952227
8	21	1.138702	0.899822	1.952225

5.2. Example 2 (see [7]). Minimize the maximum of the following three functions:

$$f_1(x) = x_1^4 + x_2^2,$$

$$f_2(x) = (2 - x_1)^2 + (2 - x_2)^2,$$

$$f_3(x) = 2\exp(-x_1 + x_2).$$

The optimum minimax value of 2 occurs at $x_1 = x_2 = 1$. Figure 2 shows contours of $M_f(x)$ for this problem. Starting at the point $(1., -0.1)$ the algorithm generated the sequence of points shown in Table 2. It is important to note that the algorithm took 6 iterations and *only* 8 function evaluations to produce very accurate results. This means that the linear search algorithm works extremely well in this case. In Figure 2 we show the path taken by the algorithm from which it can be seen that the algorithm follows ridges.

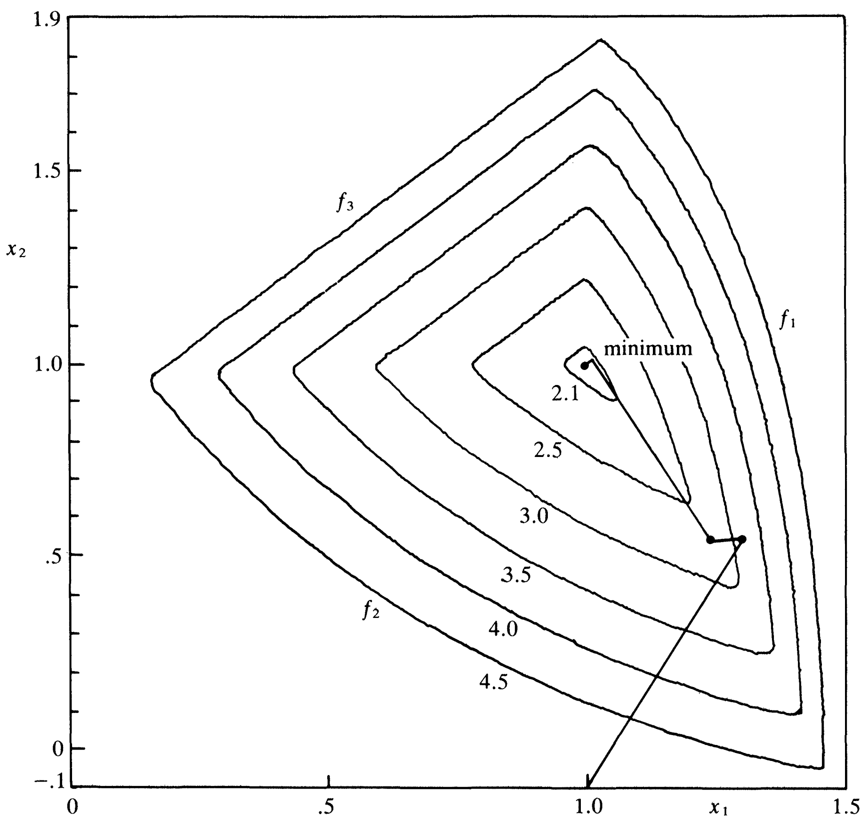

FIG. 2. *Contours of $M_f(x)$ for Example* 2.

TABLE 2
Results for Example 2 *starting at* $x_1 = 1, x_2 = -0.1$.

Number of iterations	Number of function evaluations	x_1	x_3	$M_f(x)$
0	1	1.	−0.1	5.41
1	2	1.305555	0.541666	3.198639
2	3	1.250841	0.535008	2.734218
3	4	1.046349	1.011812	2.222451
4	5	1.005593	0.993813	2.010226
4	6	1.003805	0.991570	2.009336
5	7	1.004877	0.992656	2.005017
6	8	1.000008	1.000019	2.000071

The number of function equations are cumulative, i.e. iteration 6, for example, took 1 function evaluation.

The rest of the examples are well known nonlinear programming problems which can be solved as minimax problems by using the Bandler and Charalambous approach [4].

Consider the following nonlinear programming problem:

$$\text{minimize } F(x)$$

subject to

$$g_i(x) \geqq 0, \qquad i = 2, 3, \cdots, m.$$

Consider the following minimax function:

$$M_f(x) = \max_{1 \leqq i \leqq m} f_i(x)$$

where

$$\begin{aligned} f_1(x) &= F(x), \\ f_i(x) &= F(\mathbf{x}) - \alpha_i g_i(x), \qquad && 2 \leqq i \leqq m, \\ \alpha_i &> 0, && 2 \leqq i \leqq m. \end{aligned}$$

Bandler and Charalambous [4] proved that for sufficiently large α_i the optimum of the minimax function coincides with that of the nonlinear programming problem. (See also Zangwill [21].)

5.3. Rosen–Suzuki problem (see [17]). For the Rosen–Suzuki problem

$$\begin{aligned} F(x) &= x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4, \\ g_2(x) &= -x_1^2 - x_2^2 - x_3^2 - x_4^2 - x_1 + x_2 - x_3 + x_4 + 8, \\ g_3(x) &= -x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 + x_1 + x_4 + 10, \\ g_4(x) &= -x_1^2 - x_2^2 - x_3^2 - 2x_1 + x_2 + x_4 + 5. \end{aligned}$$

The solution is

$$F = -44, \quad x_1 = 0, \quad x_2 = 1, \quad x_3 = 2, \quad x_4 = -1.$$

We used

$$\alpha_2 = \alpha_3 = \alpha_4 = 10.$$

Table 3 shows the progress of the algorithm. After 16 iterations our unknowns have the following values:

$$x_1 = 1.5883 \times 10^{-4}, \qquad x_2 = 0.99869,$$
$$x_3 = 2.00049, \qquad x_4 = -0.999677.$$

The same comments as in the previous examples hold.

5.4. Example 4 (see [1]). In this case

$$\begin{aligned} F(x) = {} & (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 \\ & + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7 \end{aligned}$$

TABLE 3
Results for Rosen–Suzuki problem starting at $x_1 = x_2 = x_3 = x_4 = 0$.

Number of iterations	Number of function evaluations	$M_f(x)$
0	1	0.
1	5	−24.5475
2	6	−31.7628
3	7	−40.6113
4	8	−41.1357
5	9	−41.3358
6	10	−43.6286
7	11	−43.7708
8	12	−43.8410
9	16	−43.9141
10	20	−43.9352
11	22	−43.9374
12	25	−43.9383
13	27	−43.93855
14	30	−43.9386
14	31	−43.99978
15	33	−43.99984
16	36	−43.99995
16	37	−43.99999

subject to

$$g_2(x) = -2x_1^2 - 3x_2^4 - x_3 - 4x_2^2 - 5x_5 + 127,$$

$$g_3(x) = -7x_1 - 3x_2 - 10x_2^3 - x_4 + x_5 + 282,$$

$$g_4(x) = -23x_1 - x_2^2 - 6x_6^2 + 8x_7 + 196,$$

$$g_5(x) = -4x_1^2 - x_2^2 + 3x_1x_2 - 2x_3^2 - 5x_6 + 11x_7.$$

We used

$$\alpha_2 = \alpha_3 = \alpha_4 = \alpha_5 = 10.$$

The solution is

$$F = 680.6301, \quad x_1 = 2.3305, \quad x_2 = 1.9514,$$
$$x_3 = -0.47754, \quad x_4 = 4.3657, \quad x_5 = -0.62449,$$
$$x_6 = 1.0381, \quad x_7 = 1.5942.$$

Table 4 shows the progress of the algorithm. After 150 function evaluations our algorithm gave the solution:

$$x_1 = 2.3312, \quad x_2 = 1.95130, \quad x_3 = -0.47136,$$
$$x_4 = 4.3655, \quad x_5 = -0.62450, \quad x_6 = 1.0383,$$
$$x_7 = 1.5941.$$

TABLE 4
Results for Wong problem starting at
$x_1=x_2=x_6=3, x_3=x_7=0, x_4=5, x_5=1.$

Number of iterations	Number of function evaluations	$M_f(x)$
0	1	2,285.
1	2	1,552.62
2	3	1,142.24
3	4	1,066.90
4	6	1.063.56
5	10	768.026
6	13	739.710
7	14	733.512
8	15	728.472
14	30	687.677
20	39	681.760
26	58	680.796
38	103	680.636
48	134	680.6304
52	150	680.6301

5.5. Colville's Test Problem 2 (see [10]). In this case

$$F(x)=\sum_{i=1}^{10} b_i x_{5+i}-\sum_{i=1}^{5}\sum_{j=1}^{5} c_{ij}x_i x_j-2\sum_{j=1}^{5} d_j x_j^3$$

subject to the constraints

$$1\text{–}15, \qquad x_i \geqq 0, \qquad i=1,2,\cdots,15,$$

$$15\text{–}20, \qquad \sum_{i=1}^{10} a_{ij}x_{5+i} \leqq e_j+2\sum_{i=1}^{5} c_{ij}x_i+3d_j y_j^2, \qquad j=1,2,\cdots,5,$$

where a_{ij}, c_{ij}, b_i, d_j, e_j are given in Table 5. The solution is

$$F=32.34868 \quad \text{at } (0.3002, 0.3334, 0.4002,$$
$$0.4281, 0.2240, 0., 0., 5.1710, 0., 3.0616,$$
$$11.8348, 0., 0., 0.1030, 0.).$$

We used $\alpha_2=\alpha_3=\cdots, \alpha_{21}=10$ and we scaled F by 80. Table 6 shows the progress of the algorithm. After 281 function evaluations our algorithm gave the solution:

$$(0.3033, 0.3323, 0.4037, 0.4253, 0.2241,$$
$$0., 0., 5.1289, 0., 3.0682, 11.7685, 0., 0., 0.0911, 0.),$$

TABLE 5

	i/j	1	2	3	4	5	
e_j		−15	−27	−36	−18	−12	
c_{ij}	1	30	−20	−10	32	−10	
	2	−20	39	−6	−31	32	
	3	−10	−6	10	−6	−10	
	4	32	−31	−6	39	−20	
	5	−10	32	−10	−20	30	
d_j		4	8	10	6	2	b_i
a_{ij}	1	−16	2	0	1	0	−40
	2	0	−2	0	0.4	2	−2
	3	−3.5	0	2	0	0	−.25
	4	0	−2	0	−4	−1	−4
	5	0	−9	−2	1	−2.8	−4
	6	2	0	−4	0	0	−1
	7	−1	−1	−1	−1	−1	−40
	8	−1	−2	−3	−2	−1	−60
	9	1	2	3	4	5	5
	10	1	1	1	1	1	1

TABLE 6
Results for Colville's Test Problem 2 *starting at*
$x_i = 0.0001, i \neq 12, x_{12} = 60.0$.

Number of iterations	Number of function evaluations	$M_f(x)$
0	1	2,400.01
1	2	2,400.00
2	3	1,702.18
3	4	1,486.66
14	15	1,414.37
30	34	789.049
65	69	42.7118
98	104	36.4468
123	134	33.8278
148	167	32.7783
165	190	32.4524
187	226	32.3572
205	281	32.3490

6. Conclusions. It is the opinion of the authors that the results do indicate that the method above is both of theoretical and practical interest (see also [8]). The results show that the present algorithm is very efficient and it is probably one of the most efficient algorithms for nonlinear minimax optimization. It is important to note that if the functions are linear then the minimax function will be piecewise linear and the present algorithm will take consideration of this fact. Among the advantages is the well defined stopping criteria afforded by Theorem 1 and the way in which the linear search is carried out. To the authors knowledge it is the first time a special linear search was used to minimize minimax functions. Furthermore, by virtue of the way the search direction is determined, the method tends to follow ridges. Finally, no special treatment is required during the later stages of convergence, since due to the vertical component we are able to obtain a sequence of points that converge directly to the desired optimum.

It should be pointed out that our horizontal component is the same as the direction of search at each iteration of the algorithm proposed by Bandler, Srinivasan and Charalambous [5] and that of Zangwill [20], but in their papers they have not considered the vertical step which means that their algorithm will only reach the optimum within a supplied tolerance and final convergence in their case is very difficult. Furthermore, the former take the linear programming approach whereas we use orthogonal projections. The connections between these two approaches is well known. As is noted in [6] for the linear case, the addition of constraints to problem **P** can be handled in a natural way. This is an immediate consequence of the fact that we are using projections.

Finally, some further avenues of research suggested by this paper include the following.

Firstly, it would be desirable to prove some results about rates of convergence of the above algorithm. It is the feeling of the authors that a super-linear convergence rate can be proved and they hope to publish results along these lines in a subsequent paper.

Secondly, it is suggested that optimization problems of a similar nature, for example, standard nonlinear programming problems with an objective function that is continuous but has discontinuous derivatives, might be solved by an analogous method.

Acknowledgment. The authors are indebted to Professor Richard Bartels of The Johns Hopkins University for bringing to their attention the work of Zangwill referenced in this paper. In addition his lucid comments were largely responsible for erasing, we hope, many of the obscurities, notational inconsistencies and clumsiness of exposition that were in the original manuscript.

REFERENCES

[1] J. ASAADI, *A Computational Comparison of Some Non-linear Programs*, Math. Programming, 4 (1973), pp. 144–154.

[2] J. W. BANDLER, *Conditions for a Minimax Optimum*, IEEE Trans. Circuit Theory, CT-18 (1971), pp. 476–479.

[3] J. W. BANDLER AND C. CHARALAMBOUS, *Practical least pth optimization of networks*, IEEE Trans. Microwave Theory Tech., MTT-20 (1972), pp. 834–840.

[4] J. W. BANDLER AND C. CHARALAMBOUS, *Nonlinear programming using minimax techniques*, J. Optimization Theory Appl., 13 (1974), pp. 607–619.

[5] J. W. BANDLER, T. V. SRINIVASAN AND C. CHARALAMBOUS, *Minimax optimization of networks by Grazor search*, IEEE Trans. Microwave Theory Tech., MTT-20 (1972), pp. 596–604.

[6] R. H. BARTELS, A. R. CONN AND C. CHARALAMBOUS, *On Cline's direct method for solving overdetermined linear systems in the l_∞ sense*, this Journal, to appear.

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

[8] C. CHARALAMBOUS AND A. R. CONN, *Optimization of microwave networks*, IEEE Trans. Microwave Theory Tech., MTT-23 (1975), pp. 834–838.

[9] E. W. CHENEY, *Introduction to Approximation Theory*, McGraw-Hill, New York, 1966.

[10] A. R. COLVILLE, *A comparative study on non-linear programming codes*, Rep. 320–2949, New York Scientific Centre, IBM, Yorktown Heights, NY, 1968.

[11] A. R. CONN, *Constrained optimization using a non-differentiable penalty function*, this Journal, 10 (1973), pp. 760–784.

[12] A. R. CONN AND T. PIETRZYKOWSKI, *A penalty function method converging directly to a constrained optimum*, this Journal, 14 (1977), pp. 348–375.

[13] V. F. DEM'YANOV, *Sufficient conditions for minimax problems*, Zh. Vychisl. Mat. i Mat. Fiz., 10 (1970), pp. 1107–1115.

[14] Y. ISHIZAKI AND H. WATANABE, *An iterative Chebyshev approximation method for network design*, IEEE Trans. Circuit Theory, CT-15 (1968), pp. 326–336.

[15] K. MADSEN, *An algorithm for minimax solution of overdetermined systems of non-linear equations*, Theoretical Physics Division, U.K.A.E.A. Research Group, Atomic Energy Research Establishment, Harwell Report HL.73/4553, Harwell, England.

[16] M. R. OSBORNE AND G. A. WATSON, *An algorithm for minimax approximation in the non-linear case*, Comput. J., 12 (1969), pp. 63–68.

[17] J. B. ROSEN AND S. SUZUKI, *Construction of nonlinear programming test problems*, Comm. ACM, 8 (1965), p. 113.

[18] J. B. ROSEN, *The gradient projection method for nonlinear programming. Part I: Linear constraints*, J. Soc. Indust. Appl. Math., 8 (1960), pp. 181–217.

[19] A. D. WARREN, L. S. LASDON AND D. F. SUCHMAN, *Optimization in engineering design*, Proc. IEEE, 55 (1967), pp. 1885–1897.

[20] W. I. ZANGWILL, *An algorithm for the Chebyshev problem—with an application to concave programming*, Management Sci., 14 (1967), pp. 58–78.

[21] ———, *Non-linear programming via penalty functions*, Ibid., 13 (1967), pp. 344–358.