Mathematical Programming 17 (1979) 270–297.
North-Holland Publishing Company

ACCELERATION OF THE LEAST *p*th ALGORITHM FOR MINIMAX OPTIMIZATION WITH ENGINEERING APPLICATIONS*

Christakis CHARALAMBOUS

University of Waterloo, Waterloo, Ontario, Canada

Received 14 July 1977
Revised manuscript received 17 February 1979

Over the past few years a number of researchers in mathematical programming and engineering became very interested in both the theoretical and practical applications of minimax optimization. The purpose of the present paper is to present a new method of solving the minimax optimization problem and at the same time to apply it to nonlinear programming and to three practical engineering problems. The original problem is defined as a modified least pth objective function which under certain conditions has the same optimum as the original problem. The advantages of the present approach over the Bandler-Charalambous least pth approach are similar to the advantages of the augmented Lagrangians approach for nonlinear programming over the standard penalty methods.

Key words: Minimax Optimization, Nondifferentiable Optimization, Computer-Aided Circuit Design, Least pth Optimization.

1. Introduction

The minimax optimization problem can be stated as:

$$\underset{x}{\text{minimize}} \quad F(x) = \max_{i \in I} f_i(x) \tag{1}$$

where $I \triangleq \{1, 2, \dots, m\}$ and $f_i(x)$ are linear or nonlinear functions of the n-dimensional column vector $x \triangleq [x_1 x_2 \cdots x_n]^{\mathrm{T}}$.

One main characteristic of the above objective function is that it has discontinuous first partial derivatives at points where two or more of the functions are equal to $F(x)$ even if the $f_i(x)$ for $i \in I$ have continuous first partial derivatives. Bandler and Charalambous [2, 8] overcame the above difficulty by smoothing the functions at the points at which the function is not differentiable. This was done by using the generalized least pth objective function. Doing so we generate a sequence of unconstrained optimization problems whose limit will be the minimax solution. The main drawback of their approach is that their

* This work was supported by the National Research Council of Canada under Grant A4414, and from the University of Waterloo.

objective function becomes more and more ill-conditioned as we get closer to the minimax solution. The purpose of the present paper is to present an approach which overcomes the ill-conditioning problem. The present approach will be used to design a digital filter, a 3-section transmission-line transformer, and the design of group delay approximation or compensation. Also the algorithm will be used to solve a number of nonlinear programming problems. I should not leave this section without mentioning the excellent textbook by Dem'yanov and Malozemov [11] on minimax optimization. Some other important papers in this area are the ones given in references [5], [9] and [15].

2. Notations and assumptions

2.1. Notations

x^* will denote the optimum point of $F(x)$,

m^* will denote the number of functions which are equal to $F(x^*)$,

$B^* \triangleq \{i \mid f_i(x^*) = F(x^*),\ i = 1, 2, \dots, m\}$,

$\nabla = [\partial/\partial x_i]$ $(i = 1, 2, \dots, n)$,

$\nabla^2 = [\partial^2/(\partial x_i \partial x_j)]$ $(i, j = 1, 2, \dots, n)$,

$\nabla f_i^* = \nabla f_i(x^*)$, $f_i^* = f_i(x^*)$,

$N^* = [\nabla f_{i_1}^* \nabla f_{i_2}^* \cdots \nabla f_{i_{m^*}}^*]$ where $i_j \in B^*$, $j = 1, 2, \dots, m^*$ (the matrix whose columns are the gradients of the functions belonging in B^*)

e will denote the m^* dimensional column vector with all elements equal to unity

$$N_t^* = \begin{bmatrix} -1 & -1 & \cdots & -1 \\ \nabla f_{i_1}^* & \nabla f_{i_2}^* & \cdots & \nabla f_{i_{m^*}}^* \end{bmatrix} = \begin{bmatrix} -e^{\mathrm{T}} \\ \hline N^* \end{bmatrix},$$

u^* will denote the Kuhn–Tucker multiplier vector,

$L(x, u) = \sum_{i=1}^{m} u_i f_i(x)$ is the minimax Lagrangian.

2.2. Assumptions

(a) The functions $f_i(x)$, $i = 1, 2, \dots, m$ are twice-continuously differentiable.

(b) The $(n+1)$ dimensional column vectors

$$\left\{ \begin{bmatrix} -1 \\ \nabla f_i^* \end{bmatrix} \right\}, \quad i \in B^*$$

are linearly independent; i.e. the matrix N_t^* is of full rank.

(c) Strict complementarity holds; i.e., $u_i^* > 0$ for $i \in B^*$.

(d) The second order sufficiency conditions for a local minimum of the minimax function hold at x^*, with associated multiplier vector u^*. In other words, we assume that the point (x^*, u^*) satisfies

$$u_i^*(F(x^*) - f_i(x^*)) = 0 \quad (i = 1, \dots, m),$$
$$u_i^* \geq 0 \quad (i = 1, \dots, m),$$
$$\nabla L(x^*, u^*) = \sum_{i=1}^{m} u_i^* \nabla f_i(x^*) = 0,$$

and further,

$$\sum_{i=1}^{m} u_i^* = 1,$$
$$y^{\mathrm{T}} \nabla^2 L(x^*, u^*) y \geq \lambda \|y\|^2, \quad \lambda > 0$$

for all $y \neq 0$ such that

$$y^{\mathrm{T}} \nabla f_i(x^*) = 0 \quad \text{for all } i \in B^*.$$

(The above conditions can be easily derived by transforming the minimax problem into its equivalent nonlinear programming formulation.)

(e) $\nabla^2 f_i(x)$ for $i \in B^*$ satisfy the Lipshitz condition in an ϵ-neighbourhood of x^*; i.e.,

$$\|\nabla^2 f_i(x_1) - \nabla^2 f_i(x_2)\| < L\|x_1 - x_2\|.$$

Assumption (b) implies that the Kuhn–Tucker multipliers u_i^* are *unique.* Also

$$\|N_i^* y\| > \mu\|y\|, \qquad \mu > 0 \quad \text{for all } y \in \mathbf{R}^{m^*} \tag{2}$$

or

$$\|(N_i^{*\mathrm{T}} N_i^*)^{-1}\| \leq 1/\mu^2 \tag{3}$$

3. Bandler–Charalambous least pth objective function [2]

One way to overcome the difficulty of nondifferentiability of the minimax function is to use the following least pth objective function whose purpose is to smooth the minimax function.

$$U(x, p, \xi) = \begin{cases} M(x, \xi)\left(\sum_{i \in S(x, \xi)} \left(\dfrac{f_i(x) - \xi}{M(x, \xi)} \right)^q \right)^{1/q} \\ \text{for } M(x, \xi) \neq 0, \\ 0 \quad \text{for } M(x, \xi) = 0 \end{cases} \tag{4}$$

where

$$M(x, \xi) = \max_{i \in I} (f_i(x) - \xi) = F(x) - \xi, \tag{5}$$

$$q = p \times \operatorname{sgn}(M(x, \xi)), \quad p > 1, \tag{6}$$

$$S(x, \xi) = \begin{cases} \{i \mid f_i(x) - \xi > 0, i \in I\} & \text{if } M(x, \xi) > 0, \\ I & \text{if } M(x, \xi) < 0. \end{cases} \tag{7}$$

(p and ξ are fixed when minimizing U with respect to x and their significance will become apparent later.)

In building up the above objective function when $M(x, \xi) < 0$ we consider all the functions $(f_i(x) - \xi, i \in I)$ and we set $q = -p$, and when $M(x, \xi) > 0$, we consider only the functions which are positive and we set $q = p$.

The objective function (4) has the property that under the stated assumptions it is continuous with continuous first partial derivatives except when $M(x, \xi) = 0$ and two or more of $(f_i(x) - \xi, i \in I)$ are equal to zero in which case U is continuous but the gradients are discontinuous.

Note that if $p = \infty$, $U(x, \infty, \xi) = M(x, \xi)$. Therefore, if p is a very large positive number the optimum point of U (with respect to x) will be close to the optimum point of M. Due to this fact the role of p for the minimax problem is similar to that of the controlling parameter for the penalty functions in nonlinear programming. In other words, if we want to find the optimum of problem (1) (or (5)) for a specific value of ξ we minimize U with respect to x for a strictly increasing sequence of values of p tending to infinity (this actually is the Polya algorithm when all the functions are positive). Another method is to keep p constant and change the value of ξ at each optimum point of U such that $|F(x^*) - \xi| \to 0$. This was originally suggested by Charalambous and Bandler [8]. As in the case of the standard penalty functions, for the above two methods, the objective function becomes more and more ill-conditioned as we get closer to the minimax solution, because the least pth objective function tends to have discontinuous first partial derivatives at least at the minimax optimum. The following question arises: Is it possible by minor modification of (4) to overcome the above ill-conditioning problem? The answer is yes and the modified function is given in the next section.

4. The modified least pth objective function

The modified objective function is:

$$U(x, u, p, \xi) = \begin{cases} M(x, \xi)\left(\sum_{i \in S(x,\xi)} u_i \left(\dfrac{f_i(x) - \xi}{M(x, \xi)} \right)^q \right)^{1/q} \\ \quad \text{for } M(x, \xi) \neq 0, \\ 0 \quad \text{for } M(x, \xi) = 0 \end{cases} \tag{8}$$

where

$$u_i \geq 0 \quad \text{and} \quad \sum_{i \in I} u_i = 1.$$

The only difference between the functions (4) and (8) is the u_i's appearing in (8), but as one will see this is a *very* important difference.

4.1. An illustrative example and intuitive ideas

Consider the following minimax function [8]:

$$F(x) = \max_{1 \le i \le 3} f_i(x)$$

where

$$f_1(x) = x_1^4 + x_2^2,$$
$$f_2(x) = (2 - x_1)^2 + (2 - x_2)^2,$$
$$f_3(x) = 2\exp(-x_1 + x_2).$$

The geometry of the problem is shown in Fig. 1. The optimum point occurs at the point $x^* = [1\ 1]^{\mathrm{T}}$ and $F(x^*) = 2$.

The Kuhn–Tucker multipliers for this problem are:

$$u_1^* = \tfrac{1}{3}, \qquad u_2^* = \tfrac{1}{2}, \qquad u_3^* = \tfrac{1}{6}.$$

As we will show later, if the vector u which is employed in the modified least pth objective function is such that $u = u^*$ then under the stated assumptions x^* is a stationary point of the modified least pth objective function for any value of $p > 1$ and any value of ξ.

Figs. 2 and 3 show contours for the modified least pth objective function with $\xi = 0$, $u = u^*$, $p = 2$ and $p = 10$, respectively. From these figures we can see that the optimum of (8) is the same as that of the minimax problem. In practice we do not know the value of the vector u^* in advance but if $u = u^r$ is close to u^* it may be expected that x^r (where x^r is the optimum point of (8) with $u = u^r$) will

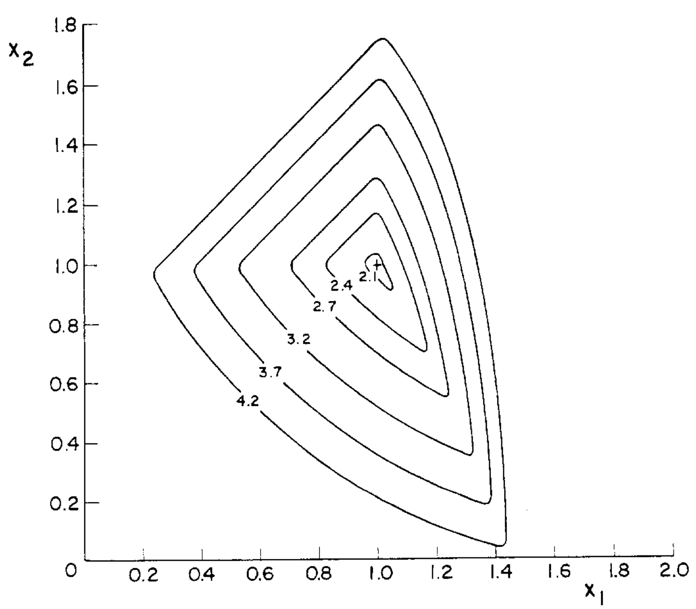

Fig. 1. Geometry of the illustrative example.

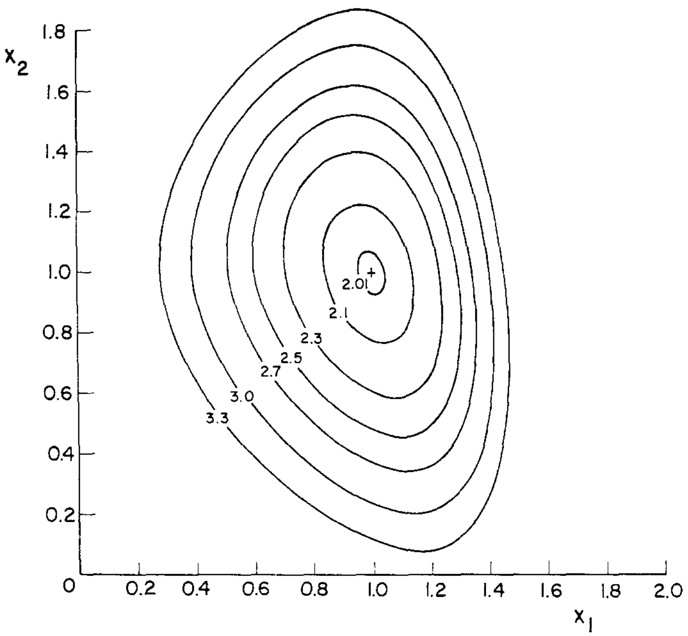

Fig. 2. Contours of the objective function U with $p = 2$ and $u = u^*$.

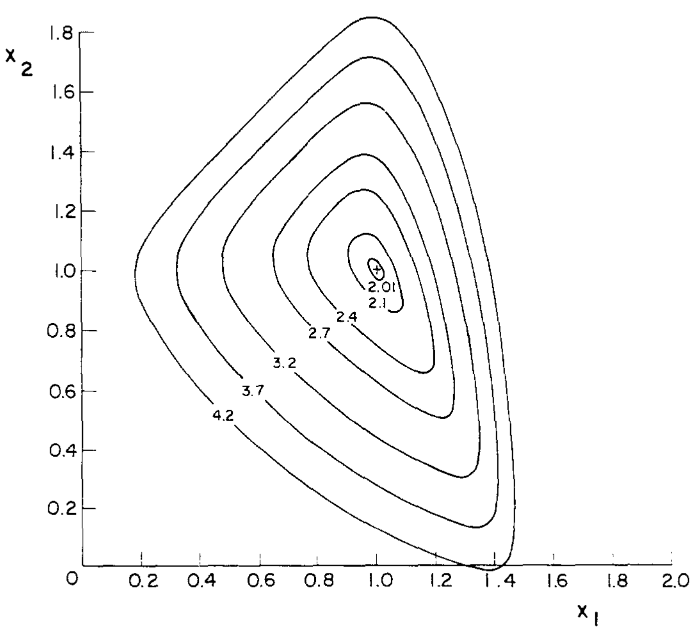

Fig. 3. Contours of the objective function U with $p = 10$ and $u = u^*$.

be close to x^* even for not too large p or not too small $|F(x^*)-\xi|$. In other words, knowledge of the approximations for the Kuhn–Tucker mutlipliers enables the result obtained by minimizing U to be improved.

Let

$$v_i = \begin{cases} u_i^r\left(\dfrac{f_i(x^r)-\xi}{M(x^r,\xi)}\right)^{q-1} & \text{if } i \in S(x^r,\xi), \\ 0 & \text{if } i \notin S(x^r,\xi). \end{cases} \tag{9}$$

At the point x^r we have

$$\sum_{i=1}^{m} v_i \nabla f_i(x^r) = 0$$

which when compared with the Kuhn–Tucker conditions, suggests

$$v/\sum_{j=1}^{m} v_j \quad (v = [v_1, v_2, \dots, v_m]^{\mathrm{T}}).$$

is an approximation to u^*.

4.2. *A possible algorithm*

(1) Set any value of $p > 1$, $\xi^{(1)} = 0$, $u_i^{(1)} = 1$, $i \in I$, $x = x^0$, and $r = 1$.
(2) Minimize (8) with respect to x.
(3) Let x^r denote the optimum point of U at the rth optimization. Set

$$u_i^{r+1} = \frac{v_i^{r+1}}{\sum_{j \in I} v_j^{r+1}}, \quad i \in I$$

where

$$v_i^{r+1} = \begin{cases} u_i^r\left(\dfrac{(f_i(x^r)-\xi)}{M(x^r,\xi)}\right)^{q-1} & \text{for } i \in S(x^r,\xi), \\ 0 & \text{for } i \notin I - S(x^r,\xi). \end{cases}$$

(4) Set $p = cp$, $c \geq 1$ and go to 2.

The above algorithm will converge at the optimum point without $p \to \infty$ as long as (x, u) is in the neighbourhood of (x^*, u^*). (Note the similarity of this approach and the augmented Lagrangian approach due to Hestenes [14] and Powell [17]).

5. Local results

From now on we will assume that $p \geq 2$ and $\xi \neq F(x^*)$. Furthermore, extensive use is going to be made of Polyak and Tret'yakov's work [16].

Lemma 1. *There exists a* $\bar{\sigma} \geq 0$ *such that the matrix*

$$\nabla^2 L(x^*, u^*) + \sigma N^* U^* N^{*\mathrm{T}}$$

is positive definite for any $\sigma \geq \bar{\sigma}$, *where* U^* *is an* $m^* \times m^*$ *diagonal matrix, given by* $U^* = \mathrm{diag}(u_i^*)$ $(i \in B^*)$.

Proof. Pre- and post-multiplying by any vector $y \in \mathbf{R}^n$ we have

$$\begin{aligned} y^T\nabla^2 L(x^*, u^*) + \sigma y^T N^* U^* N^{*T} &= y^T \nabla^2 L(x^*, u^*) y \\ &\quad + \sigma \sum_{i \in B^*} u_i^* [y^T \nabla f_i(x^*)]^2 \\ &\geq y^T \nabla^2 L(x^*, u^*) y \\ &\quad + \sigma \sum_{i \in C^*} u_i^* [y^T \nabla f_i(x^*)]^2 \\ &> 0 \quad \text{for sufficiently large value of } \sigma \end{aligned}$$

(here C^* is an index set containing a basis for the space spanned by the vectors $\{\nabla f_i(x^*)\}$, $i \in B^*$).
(From Lemma 1 of Polyak and Tret'yakov [16].)

Theorem 1. *If*

$$\frac{q-1}{F(x^*) - \xi} \geq \bar{\sigma} \quad (\textit{the same } \bar{\sigma} \textit{ as in Lemma } 1),$$

then x^* *is a strong local minimum of* $U(x, u^*, p, \xi)$.

Proof. The gradient vector of $U(x, u, p, \xi)$ with respect to x is

$$\nabla U(x, u, p, \xi) = \mu(x, u, p, \xi) \sum_{i \in S(x,\xi)} u_i \left(\frac{f_i(x) - \xi}{M(x, \xi)}\right)^{q-1} \nabla f_i(x)$$

where

$$\mu(x, u, p, \xi) = \left(\sum_{i \in S(x,\xi)} u_i \left(\frac{f_i(x) - \xi}{M(x, \xi)}\right)^q\right)^{(1/q)-1}.$$

Let $x = x^*$ and $u = u^*$. Then $u_i^* = 0$ for all i such that

$$f_i(x^*) - \xi \neq F(x^*) - \xi$$

and for the other i's,

$$(f_i(x^*) - \xi) = (F(x^*) - \xi).$$

It then follows that

$$\nabla U(x^*, u^*, p, \xi) = \sum_{i \in B^*} u_i^* \nabla f_i(x^*) = 0$$

(from the Kuhn–Tucker necessary conditions for optimality).

Therefore, if $u = u^*$ the necessary conditions for x^* to be a local minimum of U are satisfied. The next thing we want to prove is that the Hessian matrix of U (with respect to x), $\nabla^2 U(x, u, p, \xi)$ is positive definite for $u = u^*$ and $x = x^*$.

$$\nabla^2 U(x^*, u^*, p, \xi) = \sum_{i \in B^*} u_i^* \nabla^2 f_i(x^*) + \frac{(q-1)}{M(x^*, \xi)} \sum_{i \in B^*} u_i^* \nabla f_i(x^*) \nabla^{\mathrm{T}} f_i(x^*)$$
$$= \nabla^2 L(x^*, u^*) + \frac{q-1}{F(x^*) - \xi} N^* U^* N^{*\mathrm{T}}$$

which by Lemma 1 it is positive definite. It should also be noted that the above result holds for any q and ξ when the problem is convex, because $\nabla^2 L(x^*, u^*)$ is positive definite.

In the sequel, we use the notation

$$\sigma = \frac{q-1}{F(x^*) - \xi}.$$

Let us suppose that σ has been chosen satisfying Theorem 1, and the interest is in determining optimum u parameters. In these cases the implicit dependence of U and x on σ can be dropped. Using the implicit function theorem with respect to $\nabla U(x, u) = 0$, and the particular solution (x^*, u^*) and the fact that $\nabla^2 U(x^*, u^*)$ is positive definite, we can conclude that in a neighbourhood of (x^*, u^*) there exists a unique once continuously differentiable function $x(u)$ satisfying $\nabla U(x(u), u) = 0$ with $x(u^*) = x^*$ and $\nabla^2 U(x(u), u)$ positive definite. Consequently, $x(u)$ is a local isolated minimum of $U(x, u)$.

Lemma 2. *The matrix*

$$H = \left[\begin{array}{c|c} \nabla^2 L(x^*, u^*) & N^* \\ \hline U^* N^{*\mathrm{T}} & -I/\sigma \end{array}\right]$$

is nonsingular for sufficiently large σ.

Proof. Let $y_1 \in \mathbf{R}^n$, $y_2 \in \mathbf{R}^{m^*}$, then we want to show that the only solution to

$$H \left[\begin{array}{c} y_1 \\ \hline y_2 \end{array}\right] = 0$$

is the zero solution. Equivalently,

$$\nabla^2 L(x^*, u^*) y_1 + N^* y_2 = 0 \tag{10a}$$

$$U^* N^{*\mathrm{T}} y_1 - (y_2/\sigma) = 0 \tag{10b}$$

From eq. (10b) we get

$$y_2 = \sigma U^* N^{*\mathrm{T}} y_1. \tag{10c}$$

Substituting into eq. (10a) we get

$$(\nabla^2 L(x^*, u^*) + \sigma N^* U^* N^{*\mathrm{T}}) y_1 = 0. \tag{11}$$

From Lemma 1 it follows that for sufficiently large values of σ the only solution to eq. (11) is the zero solution. Also using eq. (10c) the proof of the Lemma follows.

Let A^* be the $(n+1)\times(n+1)$ matrix whose first column and first row are the zero vectors and the remaining part is $\nabla^2 L(x^*, u^*)$, i.e.,

$$A^* = \left[\begin{array}{c|ccc} 0 & 0 & \cdots & 0 \\ \hline 0 & & & \\ \vdots & & \nabla^2 L(x^*, u^*) & \\ 0 & & & \end{array}\right] \tag{12}$$

Lemma 3. *The matrix*

$$A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}}$$

is positive definite for sufficiently large σ.

Proof. Let $y_1 \in \mathbf{R}^1$, $y \in \mathbf{R}^n$. Pre- and post-multiplying the above matrix by any vector $[y_1 \mid y^{\mathrm{T}}]^{\mathrm{T}}$ and using the definition of N_t^* we get

$$\begin{aligned}[y_1 \mid y^{\mathrm{T}}](A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})\left[\frac{y_1}{y}\right] &= y^{\mathrm{T}}\nabla^2 L(x^*, u^*)y + \sigma \sum_{i=1}^{m} u_i^*[y^{\mathrm{T}}\nabla f_i(x^*)]^2 + \sigma\|y_1\|^2 \\ &> 0 \quad \text{for sufficiently large } \sigma,\end{aligned}$$

by using Lemma 1.

Corollary [16, Lemma 1]. *$(A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})^{-1}$ exists for sufficiently large σ and $\|(A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})^{-1}\| \geq c_1$ (here and below the c_i are constants independent of σ, as long as σ is sufficiently large).*

Further, the proof of the next Lemma is similar to Lemmas 2, 3 and 4 in [16]. Let

$$B = \left[\begin{array}{c|c} A^* & N_t^* \\ \hline U^* N_t^{*\mathrm{T}} & -I/\sigma \end{array}\right]$$

Lemma 4. *For sufficiently large values of σ*

(a) $\|(A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})^{-1} N_t^*\| \leq c_2/\sigma$,

(b) $\|N_t^{*\mathrm{T}}(A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})^{-1}\| \leq c_2/\sigma$,

(c) $\|I - \sigma U^* N_t^{*\mathrm{T}}(A^* + \sigma N_t^* U^* N_t^{*\mathrm{T}})^{-1} N_t^*\| \leq c_3/\sigma$

(d) B^{-1} *exists and* $\|B^{-1}\| \leq c_4$.

Lemma 5. *Let σ be such that H and B are nonsingular. Let the unique solution $[y_1^*, y_2^*]$ ($y_1^* \in \mathbf{R}^n$, $y_2 \in \mathbf{R}^{m^*}$) of*

$$H\left[\frac{y_1}{y_2}\right] = \left[\frac{a_1}{a_2}\right], \qquad a_1 \in \mathbf{R}^n, \quad a_2 \in \mathbf{R}^{m^*} \tag{13a}$$

be such that the sum of the coordinates of y_2^* is 0. Then $[0, y_1^*, y_2^*]$ is the unique solution of

$$B\begin{bmatrix} y_0 \\ \hline y_1 \\ \hline y_2 \end{bmatrix} = \begin{bmatrix} 0 \\ \hline a_1 \\ \hline a_2 \end{bmatrix}. \tag{13b}$$

Proof. From the assumption that the solution of (13a) is such that the sum of the components of y_2^* is equal to zero and from the special structure of B it follows that $[0 \mid y_1^{*\mathrm{T}} \mid y_2^{*\mathrm{T}}]^{\mathrm{T}}$ is a solution of (13b). But since B is nonsingular for sufficiently large values of σ it follows that this is the only solution.

For Theorems 2 and 3 we will make the assumption that we use only the active functions.

Theorem 2. *For every $u \in S_\rho = \{u: \|u - u^*\| \le \rho\}$ a number $\sigma = \sigma(\rho)$ can be found such that for $\sigma \ge \sigma(\rho)$ we have*

$$\|x - x^*\| \le \frac{c_0}{\sigma}\|u - u^*\|, \qquad \|\eta\| \le \frac{c_0}{\sigma}\|u - u^*\|$$

where x is the optimum point of U and

$$\eta_i = \begin{cases} v_i \Big/ \sum_{j \in S(x,\xi)} v_j - u_i^* & \text{if } i \in S(x, \xi), \\ 0 & \text{if } i \notin S(x, \xi) \end{cases} \tag{14}$$

where

$$v_i = \begin{cases} u_i \left(\dfrac{f_i(x) - \xi}{M(x, \xi)}\right)^{q-1} & \text{if } i \in S(x, \xi), \\ 0 & \text{if } i \notin S(x, \xi). \end{cases} \tag{15}$$

Proof. Let

$$\Delta x = x - x^*, \qquad \Delta u = u - u^*, \qquad z = \begin{bmatrix} \Delta x \\ \eta \end{bmatrix}.$$

From assumption 2.2(e) we have

$$\nabla f_i(x) = \nabla f_i(x^*) + \nabla^2 f_i(x^*)\Delta x + r_1^i(\Delta x) \tag{16}$$

where

$$r_1^i(0) = 0, \qquad \|\nabla r_1^i(\Delta x)\| = \|\nabla^2 f_i(x) - \nabla^2 f_i(x^*)\| \le L\|\Delta x\|.$$

Using the fact that $\nabla U(x, u, p, \xi) = 0$, (14) and (16), we get that Δx and η satisfy the following set of equations.

$$\left[\sum_{i \in B^*} u_i^* \nabla^2 f_i(x^*)\right]\Delta x + \sum_{i \in B^*} \eta_i \nabla f_i(x^*) + r_2(z) = 0 \tag{17}$$

where

$$r_2(z) = \sum_{i \in B^*} \eta_i \nabla^2 f_i(x^*)\Delta x + \sum_{i \in B^*} (u_i^* + \eta_i) r_1^i(\Delta x). \tag{18}$$

From (18) we have that

$$r_2(0) = 0$$

and for small $\|z\|$,

$$\|\nabla r_2(z)\| \le L_1\|z\|.$$

Also, Δx and η satisfy the following set of equations coming from the linearization of η.

$$\eta_i(x, u) = \eta_i(x^*, u^*) + [\nabla_x^{\mathrm{T}}\eta_i(x^*, u^*), \nabla_u^{\mathrm{T}}\eta_i(x^*, u^*)]\begin{bmatrix}\Delta x\\ \Delta u\end{bmatrix} + r_3^i(\Delta x, \Delta u). \tag{19}$$

Using the fact that the Hessian matrix of the f_i's satisfy the Lipshitz condition in the ϵ neighbourhood of x^* and the denominator of (14) is bounded away from zero (from the assumption at the beginning of this section $\xi \neq F(x^*)$) it then follows that (note that when $z = 0$, $\Delta u = 0$)

$$\begin{aligned} &r_3^i(\Delta x, \Delta u) = 0 \quad \text{when } z = 0,\\ &\|\nabla_x r_3^i(\Delta x, \Delta u)\| = \|\nabla_x^2\eta_i(x, u) - \nabla_x^2\eta_i(x^*, u^*)\|\\ &\qquad \le L_2\|\Delta x\|. \end{aligned}$$

From (14) and (15) we have

$$\begin{aligned} &\eta_i(x^*, u^*) = 0,\\ &\nabla_x v_j^* = \sigma u_j^*\nabla f_j(x^*), \qquad \nabla_x\eta_j(x^*, u^*) = \sigma u_j^*\nabla f_j(x^*),\\ &\nabla_u v_j^* = e_j, \qquad \nabla_u\eta_i(x^*, u^*) = e_i - u_i e \end{aligned}$$

where e_j is an m^* dimensional unit vector with unity in its jth position, and e is an m^* dimensional column vector whose elements are equal to one.

Therefore,

$$\eta_i(x, u) = \sigma u_i^*\nabla^{\mathrm{T}} f_i(x^*)\Delta x + \Delta u_i - u_i^* \sum_{i\in B^*} \Delta u_i + r_3^i(\Delta x, \Delta u). \tag{20}$$

Using the fact that $\sum_{i\in B^*}\Delta u_i = 0$ and $\sum_{i\in B^*}\eta_i^* = 0$ and (19) and (20) we get

$$\left[\begin{array}{c|c} A^* & N_t^* \\ \hline U^*N_t^{*\mathrm{T}} & -I/\sigma \end{array}\right]\left[\begin{array}{c} 0\\ \hline \Delta x\\ \hline \eta\end{array}\right] = \left[\begin{array}{c} 0\\ \hline 0\\ \hline \dfrac{u^*-u}{\sigma}\end{array}\right] + \left[\begin{array}{c} 0\\ \hline -r_2(\Delta x, \eta)\\ \hline -\dfrac{r_3(\Delta x, \Delta u)}{\sigma}\end{array}\right], \tag{21}$$

$$B\bar{z} = a + r(\bar{z}) \tag{22}$$

where

$$\bar{z} = \left[\begin{array}{c}0\\ \hline z\end{array}\right], \qquad a = \left[\begin{array}{c} 0\\ \hline 0\\ \hline (u^*-u)/\sigma\end{array}\right] \qquad r(z) = \left[\begin{array}{c} 0\\ \hline -r_2(\Delta x, \eta)\\ \hline -(r_3(\Delta x, \Delta u)/\sigma)\end{array}\right]. \tag{23}$$

Also, $r(0) = 0$, and for small z there exists a constant L_3 such that $\|\nabla_z r(z)\| \le$

$L_3\|z\|$. Accordingly, from Polyak and Tret'yakov [16] for sufficiently large σ and $u \in S_\rho$ the conditions of the theorem are satisfied.

Let us suppose that $u^0 \in S_\rho$ for sufficiently large σ and that we have identified the active functions. Consider the algorithm where we set $u^{r+1} = \eta(x(u^r), u^r) + u^*$ (where η is given by eq. (14)), then it can be easily shown that

$$\|u^r - u^*\| \leq \left(\prod_{i=0}^{r-1} \frac{c_0}{\sigma_i}\right)\|u^0 - u^*\| \tag{24a}$$

and

$$\|x^r - x^*\| \leq \left(\prod_{i=0}^{r-1} \frac{c_0}{\sigma_i}\right)\|u^0 - u^*\|. \tag{24b}$$

Therefore, the sequence $\{u^r\}$ converges to u^* at least linearly with the convergence ratio inversely proportional to σ. If $\sigma_i \to \infty$, then $\{u^r\}$ converges to u^* superlinearly.

Let us suppose that we consider only the active constraints and we are at the point $x_\gamma = x_\gamma(u, p, \xi)$ in the neighbourhood of x^*, and which

$$\|\nabla U(x_\gamma, u, p, \xi)\| \leq \frac{\gamma}{\sigma}\|\beta(x_\gamma, u, p, \xi) - u\|, \quad \gamma \geq 0 \tag{25a}$$

where

$$\beta_i(x_\gamma, u, p, \xi) = \frac{v_i(x_\gamma, u, p, \xi)}{\sum_{i \in B^*} v_i(x_\gamma, u, p, \xi)}, \quad i \in B^*, \tag{25b}$$

$$v_i(x_\gamma, u, p, \xi) = u_i\left(\frac{f_i(x_\gamma) - \xi}{M(x_\gamma, \xi)}\right)^{q-1}. \tag{25c}$$

Theorem 3. For every $u \in S_\rho$ and $\sigma \geq \bar{\sigma} = \sigma(\rho)$ the following estimates hold:

$$\|x - x^*\| \leq \frac{c_0(1+\gamma)\|u - u^*\|}{\sigma}, \qquad \|\eta_\gamma\| \leq \frac{c_0(1+\gamma)\|u - u^*\|}{\sigma}.$$

Proof. Note that

$$\frac{\eta_\gamma + u^* - u}{\sigma} = \frac{1}{\sigma}[\beta(x_\gamma, u, p, \xi) - u].$$

Therefore

$$\begin{aligned}\|\nabla U(x_\gamma, u, p, \xi)\| &\leq \left(\frac{\|u - u^*\|}{\sigma} + \frac{\|\eta_\gamma\|}{\sigma}\right)\gamma \\ &\leq \left(\frac{\|u - u^*\|}{\sigma} + \frac{\left\|\begin{matrix}\Delta x \\ \eta_\gamma\end{matrix}\right\|}{\sigma}\right)\gamma.\end{aligned}$$

The result follows from Polyak and Tret'yakov's theorem [16].

Now if at each stage of the iteration process we let $x^{r+1} = x_\gamma(u^r, p, \xi)$ then it follows that

$$\|x^r - x^*\| \le \left(\frac{c_0(1+\gamma)}{\sigma}\right)^r \|u^0 - u^*\|, \tag{26a}$$

$$\|u^r - u^*\| \le \left(\frac{c_0(1+\gamma)}{\sigma}\right)^r \|u^0 - u^*\|. \tag{26b}$$

This shows that as long as we satisfy (28) and the assumptions of Theorem 3 the algorithm will converge even if we do not find the exact minimum of $U(x, u, p, \xi)$ with respect to x at each iteration.

All results presented above are only local. By strictly increasing the value of p at each optimum point, or the value of ξ such that $\lim_{r\to\infty} p \to \infty$ or $\lim_{r\to\infty} |\xi - F(x^*)| \to 0$ we can get close to the solution where the above results can be used. For more details see [2], [7], [8].

6. Examples

A number of examples from nonlinear programming, an example of microwave network design, an example of digital filter design, and the design of group delay approximation or compensation will be considered so that to illustrate the usefulness of this approach to nonlinear minimax optimization. The algorithm presented in section 4.2 will be used. Fletcher's [13] unconstrained optimization algorithm is going to be used to minimize (8). For all the examples considered in this paper the initial value of p was equal to 10.

The nonlinear programming problem

$$\min_x f(x),$$

subject to

$$g_i(x) \ge 0, \quad i = 1, 2, \dots, m$$

will be transformed into the equivalent minimax problem

$$\min_x \max_{1\le i\le m} \{f(x), f(x) - \alpha_i g_i(x)\}$$

where $\alpha_i > 0$ are sufficiently large (see [2], [4], [21]).

6.1. Rosen–Suzuki problem

In the case

$$f(x) = x_1^2 + x_2^2 + 2x_3^2 + x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4;$$
$$g_1(x) = -x_1^2 - x_2^2 - x_3^2 - x_4^2 - x_1 + x_2 - x_3 + x_4 + 8;$$
$$g_2(x) = -x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 + x_1 + x_4 + 10;$$
$$g_3(x) = -2x_1^2 - x_2^2 - x_3^2 - 2x_1 + x_2 + x_4 + 5.$$

This function has a minimum value $f = -44$ at $x^* = [0\ 1\ 2\ -1]^T$. The constraints $g_1(x)$ and $g_3(x)$ are active. The components of the vector α were all set equal to 10. Table 1 shows the results for three different approaches.

Table 1
Rosen–Suzuki problem (Starting point: $x^{(0)} = [0\ 0\ 0\ 0]^T$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	18	−42.17427	18	−42.17427	18	−42.17427
2	17	−43.80592	13	−43.93040	18	−43.99221
3	23	−43.98043	12	−43.99841	15	−43.99999
4	25	−43.99804	11	−43.99943	15	−44.000000
5	26	−43.99981	11	−43.99978		
6	26	−43.99998	11	−43.99990		
7	27	−43.99999	10	−43.99995		
Total number of function evaluations	162		86		66	

Optimum solution obtained using algorithm 3:

$$x^{(4)} = [0.\ 1.\ 2.\ -1.],\quad f(x^{(4)}) = -44.000000.$$

Algorithm 1. Algorithm 4.2, but keeping all the u_i's equal to 1 and changing p. $c = 10$ and initial value of $p = 10$. This is the Bandler–Charalambous algorithm.

Algorithm 2. Algorithm 4.2, with $c = 1$, $p = 10$.

Algorithm 3. Algorithm 4.2, with $c = 10$ and initial value of $p = 10$. Algorithm 3 is a combination of algorithms 1 and 2. From the table we can see the superiority of algorithm 3 over the algorithm without acceleration (algorithm 1).

6.2. *Wong problem* 1 [7]

The components of the vector α were all set equal to 10. Table 2 shows the results for the three different cases. The same comments as for example 1 hold.

6.3. *Wong problem* 2 [7]

The components of the vector α were all set equal to 10. Table 3 shows the results for this problem.

Table 2
Wong problem 1 (Starting point: $x_1 = x_6 = x_7 = 1$, $x_2 = 2$, $x_3 = x_5 = 0$, $x_4 = 4$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	22	704.9796	22	704.9796	22	704.9796
2	29	683.0003	22	684.8607	30	681.1248
3	38	680.8648	18	681.9335	32	680.6331
4	44	680.6535	18	681.1633	23	680.6301
5	55	680.6324	16	680.8724		
6	64	680.6303	17	680.7449		
7	64	680.6301	17	680.6855		
Total number of function evaluations		316		130		107

Optimum solution obtained using algorithm 3:

$x^{(4)} = [2.330499\ 1.951372\ -0.4775413\ 4.365726\ -0.6244870\ 1.038131\ 1.594227]^{\mathrm{T}}$, $f(x^{(4)}) = 680.6301$.

Table 3
Wong problem 2 (Starting point: $x^0 = [2\ 3\ 5\ 5\ 1\ 2\ 7\ 3\ 6\ 10]^{\mathrm{T}}$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	56	26.15797	56	26.15797	56	26.15797
2	37	24.48127	29	24.35283	39	24.31056
3	48	24.32362	21	24.31608	25	24.30622
4	60	24.30795	20	24.30718		
5	80	24.30638	19	24.30628		
6	79	24.30623	10	24.30622		
Total number of function evaluations		360		155		120

Optimum solution obtained using algorithm 3:
$x^{(3)} = [2.171996\ 2.363683\ 8.773926\ 5.095985\ 0.9906548\ 1.430574\ 1.321644\ 9.828726\ 8.280092\ 8.375927]^{\mathrm{T}}$
$f(x^{(3)}) = 24.30621$.

6.4. *Wong problem* 3 [1]

In this case

$$
\begin{aligned}
f(x) = {} & x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 \\
& + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 \\
& + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2
\end{aligned}
$$

$$+(x_{11}-9)^2+10(x_{12}-1)^2+5(x_{13}-7)^2$$
$$+4(x_{14}-14)^2+27(x_{15}-1)^2+x_{16}^4+(x_{17}-2)^2$$
$$+13(x_{18}-2)^2+(x_{19}-3)^2+x_{20}^2+95.$$

The first 8 constraints are the same constraints as for Wong problem 2.

$$g_8(x)=8x_1-2x_2-5x_9+2x_{10}+12,$$
$$g_9(x)=-x_1-x_2-4x_{11}+21x_{12},$$
$$g_{10}(x)=-x_1^2-15x_{11}+8x_{12}+28,$$
$$g_{11}(x)=-4x_1-9x_2-5x_{13}^2+9x_{14}+87,$$
$$g_{12}(x)=-3x_1-4x_2-3(x_{13}-6)^2+14x_{14}+10,$$
$$g_{13}(x)=-14x_1^2-35x_{15}+79x_{16}+92,$$
$$g_{14}(x)=-15x_2^2-11x_{15}+61x_{16}+54,$$
$$g_{15}(x)=-5x_1^2-2x_2-9x_{17}^4+x_{18}+68,$$
$$g_{16}(x)=-x_1^2+x_2-19x_{19}+20x_{20}-19,$$
$$g_{17}(x)=-7x_1^2-5x_2^2-x_{19}^2+30x_{20}.$$

The solution is:

$x_1=2.175216$,	$x_2=2.352850$,	$x_3=8.766448$,
$x_4=5.066932$,	$x_5=0.9886672$,	$x_6=1.430999$,
$x_7=1.329483$,	$x_8=9.835926$,	$x_9=8.287277$,
$x_{10}=8.370178$,	$x_{11}=2.275828$,	$x_{12}=1.358623$,
$x_{13}=6.077186$,	$x_{14}=14.17083$,	$x_{15}=0.9962345$,
$x_{16}=0.6556911$,	$x_{17}=1.466590$,	$x_{18}=2.000361$,
$x_{19}=1.046588$,	$x_{20}=2.063194$,	$f=133.7283$.

The components of the vector α were all set equal to 10. Table 4 shows the results for this problem. It is important to note that algorithm 2 started to converge to the solution and suddenly started to diverge. This is probably due to the fact that p is not sufficiently large.

6.5. *Colville's test problem* 2 [10]

The components of the vector α were all set equal to 200. Table 5 shows the results for this problem. The same comments as in the previous examples hold.

6.6. *Colville's test problem* 3 [10]

The components of the vector α were all set equal to 2000. Table 6 shows the results for this problem. Again the superiority of algorithm 3 over algorithm 1 is apparent.

Table 4
Wong problem 3 (Starting point: $x^0 = [2\ 3\ 5\ 5\ 1\ 2\ 7\ 3\ 6\ 10\ 2\ 2\ 6\ 15\ 1\ 2\ 1\ 2\ 1\ 3]^T$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^{(r)})$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	61	145.5159	61	145.5159	61	145.5159
2	63	134.8940	32	135.0639	57	133.8687
3	62	133.8446	31	134.5258	44	133.8570
4	71	133.7399	30	134.0929	119	133.7306
5	88	133.7294	30	133.8863	37	133.7283
6	91	133.7283	31	133.8023		
7			29	134.0538		
8			30	134.2458		
Total number of function evaluations		436				318

Optimum solution obtained using algorithm 3:
$x^{(5)} = 2.175216\ 2.352850\ 8.766448\ 5.066932\ 0.9886672\ 1.430999\ 1.329483\ 9.835926$
$8.287277\ 8.370178\ 2.275828\ 1.358623\ 6.077186\ 14.077183\ 0.9962345\ 0.6556911$
$1.466590\ 2.000361\ 1.046588\ 2.063194]^T$,
$f(x^{(5)}) = 133.72828.$

Table 5
Colville's test problem 2 (Starting point: $x_i = 0.0001$, $i \neq 7$, $i = 1, 2, \ldots, 15$, $x_7 = 60$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	288	33.51591	288	33.51591	288	35.51591
2	104	32.46044	46	32.36701	94	32.35035
3	127	32.35981	27	32.34872	31	32.348679
4	89	32.34979				
5	102	32.34879				
6	87	32.34869				
7	89	32.34868				
8	124	32.348679				
Total number of function evaluations		1010		361		413

Optimum solution obtained using algorithm 3:
$x^{(4)} = [0.\ 0.\ 5.174040\ 0.\ 3.061109\ 11.83955\ 0.\ 0.\ 0.1038961\ 0.\ 0.3000000$
$0.3334676\ 0.4000000\ 0.4283101\ 0.2239649]^T$,
$f(x^{(4)}) = 32.348679.$

Table 6
Colville's test problem 3 (Starting point: $x^0 = [78.62\ 33.44\ 31.07\ 44.18\ 35.32]^T$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	17	−29 420.46	17	−29 420.46	17	−29 420.46
2	29	−30 529.38	17	−30 324.73	26	−30 627.96
3	37	−30 652.31	12	−30 596.20	18	−30 664.78
4	40	−30 664.22	11	−30 633.37	16	−30 665.54
5	51	−30 665.41	11	−30 643.23		
6	49	−30 665.53	10	−30 650.01		
7			10	−30 654.79		
8			10	−30 658.08		
Total number of function evaluations	223		98		77	

Optimum solution obtained using algorithm 3:
$x^{(4)} = [78.00000, 33.00000, 29.99526, 45.00000, 36.77580]^T$,
$f(x^{(4)}) = -30\,665.54$.

6.7. *Digital filter example* [6]

In this case we want the amplitude of the digital filter

$$|H(x, \theta)| = A \prod_{k=1}^{K} \left(\frac{1 + a_k^2 + b_k^2 + 2b_k(2\cos^2\theta - 1) + 2a_k(1 + b_k)\cos\theta}{1 + c_k^2 + d_k^2 + 2d_k(2\cos^2\theta - 1) + 2c_k(1 + d_k)\cos\theta} \right)^{1/2}$$

to approximate in the minimax sense the function

$$S(\psi) = |1 - 2\psi|, \quad 0 \le \psi \le 1$$

where

$$\theta = \pi\psi,$$
$$x = [a_1 b_1 c_1 d_1 \cdots a_k b_k c_k d_k A],$$

i.e. we have $n = 4k + 1$.

We define the error function

$$e(x, \psi) = |H(x, \theta)| - S(\psi).$$

Therefore, we want to

$$\min_{x} \max_{0 \le \psi \le 1} |e(x, \psi)|.$$

It was decided to use 41 sample points of ψ, namely

$$\psi = 0.0, 0.05\ (0.01); \quad i = 1, \ldots, 6,$$
$$\psi = 0.07, 0.46\ (0.03); \quad i = 7, \ldots, 20,$$

$$\psi = 0.5, 0.54\,(0.04); \quad i = 21, 22,$$
$$\psi = 0.57, 0.67\,(0.1); \quad i = 23, 24,$$
$$\psi = 0.63, 0.93\,(0.03); \quad i = 25, 35,$$
$$\psi = 0.95, 1.0\,(0.01); \quad i = 36, 41.$$

Therefore the problem becomes

$$\min_x \max_{1 \le i \le 41} \{f_i(x)\}$$

where

$$f_i(x) = |e(x, \psi_i)|.$$

Table 7 shows the results for this problem. Fig. 4 shows the optimum error curve.

Table 7
Digital filter example (Starting point: $x^0 = [0\ 1\ 0\ -0.15\ 0\ -0.68\ 0\ -0.72\ 0.37]^T$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r) \times 10^2$	*Algorithm* 2 Function evaluations per optimization	$F(x^r) \times 10^2$	*Algorithm* 3 Function evaluations per optimization	$F(x^r) \times 10^2$
1	44	0.64674	44	0.64674	44	0.64674
2	56	0.61954	26	0.63931	48	0.62166
3	23	0.61861	20	0.62538	47	0.61857
4	41	0.61855	18	0.62375	18	0.61853
5	36	0.61853	18	0.62186		
6			17	0.62057		
7			15	0.62029		
Total number of function evaluations		200		158		157

Optimum solution obtained using algorithm 3:
$x^{(4)} = [0\ \ 0.980039\ \ 0.\ \ -0.165771\ \ 0.\ \ -0.735078\ \ 0.\ \ -0.767228\ \ 0.367900]^T$.

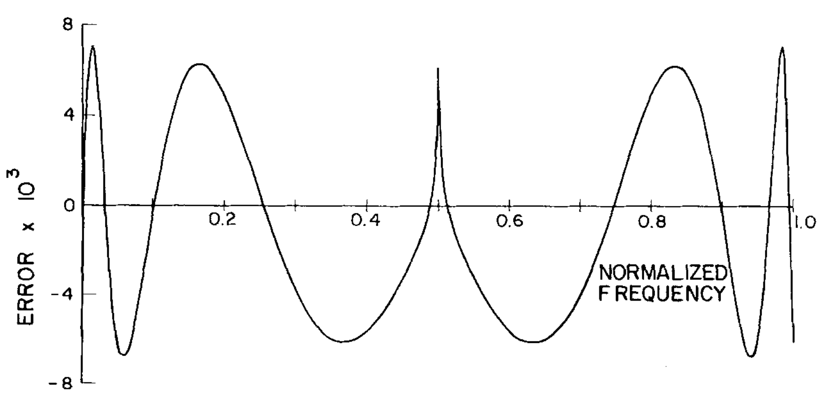

Fig. 4. Optimum error curve for digital filter example.

6.8. *3-Section transmission-line transformer* [2]

For the three section transformer we used the following 11 frequency points of ψ in gigahertz:

$$\{0.5, 0.6, 0.7, 0.77, 0.9, 1.0, 1.1, 1.23, 1.4, 1.5\}.$$

The variable vector x consists of the following variables

$$x = [Z_1, Z_2, Z_3, l_1/l_g, l_2/l_q, l_3/l_g]^{\mathrm{T}}.$$

Where Z_i is the characteristic impedance of the ith section, l_i is the length of the ith section, and l_q is the quarter-wavelength at center frequency. Table 8 shows the results for this problem.

Table 9 summarizes the results of the eight test problems. From table 9 it can be seen that the new approach is superior to that of the Bandler–Charalambous approach.

Table 8
Transmission-line transformer (Starting point: $x^0 = [1.5\ 3.0\ 6.0\ 0.8\ 1.2\ 0.8]^{\mathrm{T}}$)

r	*Algorithm* 1 Function evaluations per optimization	$F(x^r)$	*Algorithm* 2 Function evaluations per optimization	$F(x^r)$	*Algorithm* 3 Function evaluations per optimization	$F(x^r)$
1	25	0.206359	25	0.206359	25	0.206359
2	25	0.197807	21	0.20026	25	0.197686
3	24	0.197342	19	0.19825	28	0.1972906
4	18	0.197296	19	0.19759		
5	14	0.1972912	18	0.19738		
6			18	0.19732		
7			18	0.19730		
8			15	0.197292		
Total number of function evaluations		106		153		78

Optimum solution obtained using algorithm 3:
$x^3 = [1.634707\ \ 3.162277\ \ 6.117304\ \ 1.0\ \ 1.0\ \ 1.0]^{\mathrm{T}}$.

6.9. *Group delay approximation or compensation*

The design of networks with a specified amplitude and phase specifications can be achieved by first designing a network which approximates the given amplitude response and couple it with a constant amplitude phase corrector.

Correctors with constant amplitude can be obtained if their transfer functions have zeros symmetrical to the poles about the imaginary axis. This class of networks is called all-pass networks. The transfer function for the kth order

Table 9
Summary of the results

Problem		Rosen–Suzuki ($F(x^*) = -44$)	Wong problem 1 ($F(x^*) = -680.630$)	Wong problem 2 ($F(x^*) = 24.30621$)	Wong problems ($F(x^*) = 133.7283$)
New approach (algorithm 3)	F	−44.000000	−680.6301	24.30622	133.7283
	Function evaluations	66	107	120	318
Bandler–Charalambous approach (algorithms)	F	−43.99999	680.6301	24.30623	133.7283
	Function evaluations	162	316	360	436

Colville's test problem 2 ($F(x^*) = 32.34868$)	Colville's test problem 3 ($F(x^*) = -30\,665.54$)	Digital filter ($F(x^*) = 0.61853 \times 10^{-2}$)	Transmission-line transformer ($F(x^*) = 0.1972906$)
32.348679	−30 665.54	0.61853×10^{-2}	0.1972906
413	77	157	78
32.348679	−30 665.53	0.61853×10^{-2}	0.1972912
1010	223	200	106

network is given by

$$H(s)=\prod_{i=1}^{l}\frac{[s-(\alpha_i+j\beta_i)][s-(\alpha_i-j\beta_i)]}{[s-(-\alpha_i+j\beta_i)][s-(-\alpha_i-j\beta_i)]}\times H_1(s) \tag{27}$$

where

$$\left.\begin{aligned}&\text{for } k \text{ even} \quad l=\tfrac{1}{2}k \text{ and } H_1(s)=1\\&\text{for } k \text{ odd} \quad l=\tfrac{1}{2}(k-1) \text{ and } H_1(s)=\frac{s-\alpha_{l+1}}{s+\alpha_{l+1}}\end{aligned}\right\} \tag{28}$$

From (1) and (2) the group delay is given by

$$\left.\begin{aligned}\tau(\alpha,\beta,w)&=-\frac{\mathrm{d}}{\mathrm{d}w}[LH(jw)]=2\sum_{i=1}^{l}\left[\frac{\alpha_i}{\alpha_i^2+(w+\beta_i)^2}+\frac{\alpha_i}{\alpha_i^2+(w-\beta_i)^2}\right]\\&\begin{cases}0 & \text{if } k \text{ is even}\\ 2\dfrac{\alpha_{l+1}}{\alpha_{l+1}^2+w^2} & \text{if } k \text{ is odd}\end{cases}\end{aligned}\right\} \tag{29}$$

where α is a column vector whose elements are the α_i's, β is a column vector whose elements are the β_i's and w is the frequency in rad/sec.

The overall group delay is

$$\tau(\alpha,\beta,w)+\tau_0(w)$$

where $\tau_0(w)$ is the original filter group delay. Since we want the overall group delay to be constant, and this constant is not restricted to any specified value, we consider it as an additional variable x_0. Therefore the problem can be stated as

$$\underset{\alpha,\beta,x_0}{\text{minimize}}\ \max_{w\in[w_l,w_u]}|\tau(w,\alpha,\beta)+\tau_0(w)-x_0| \tag{30}$$

where $[w_l, w_u]$ is the frequency band. Usually we discretise the frequency axis into a sufficient number of frequency points. Let us consider m sample points and denote

$$x^{\mathrm{T}}=[\alpha^{\mathrm{T}},\beta^{\mathrm{T}},x_0].$$

Thus the discrete minimax problem to (30) is

$$\min_{x}\max_{1\le i\le m} f_i(x) \tag{31}$$

where

$$f_i(x)=|\tau(w_i,\alpha,\beta)+\tau_0(w_i)-x_0|. \tag{32}$$

6.10. Design example [20]

Here we considered the case where $k=4$, $w_l=0.1$ rad/sec, $w_u=1$ rad/sec, and the original filter is fourth order Chebychev with 0.5 db ripple. We used 31 uniformly distributed sample points in the specified frequency interval.

Starting point:

$$x^0 = [0.4, 0.3, 0.2, 0.6, 10]^T.$$

Optimal solution obtained:

$$x^* = [0.33551, 0.42136, 0.74146, 0.22470, 12.3215]^T.$$

The group delay of the original filter is shown in Fig. 5. The initial response of the filter plus an arbitrarily chosen corrector is shown in the same figure.

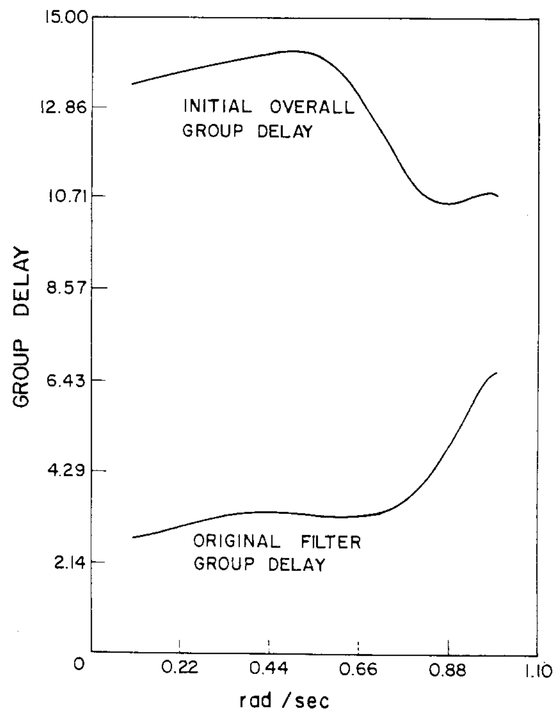

Fig. 5. Original filter group delay and initial overall group delay at the starting point.

Optimized results are shown in Fig. 6 and the final error is shown in Fig. 7. A comparison with the solution obtained by the expansion of the group delay into the series of Chebychev polynomials is shown in Fig. 8 [20].

It can be seen that a reduction of over 37% was achieved by the use of the minimax algorithm.

Although the example gives a low-pass filter compensation, the method is not restricted to it. Any group delay in any frequency band can be approximated or compensated in the minimax sense. This example was originally presented in [22].

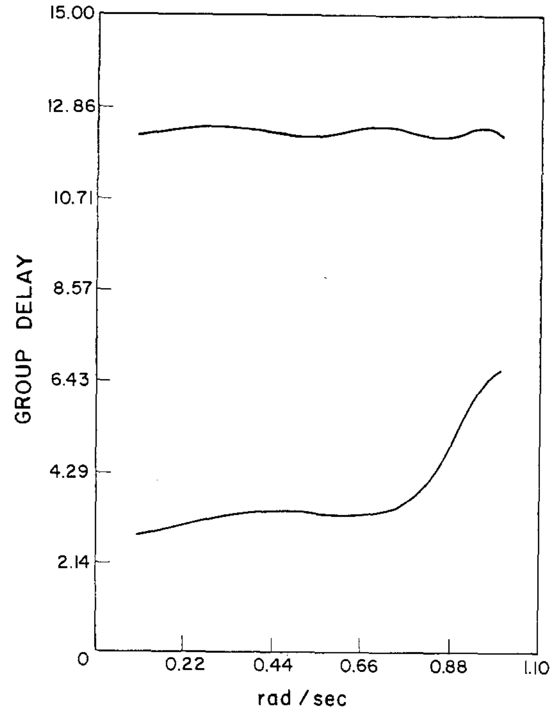

Fig. 6. Optimized group delay.

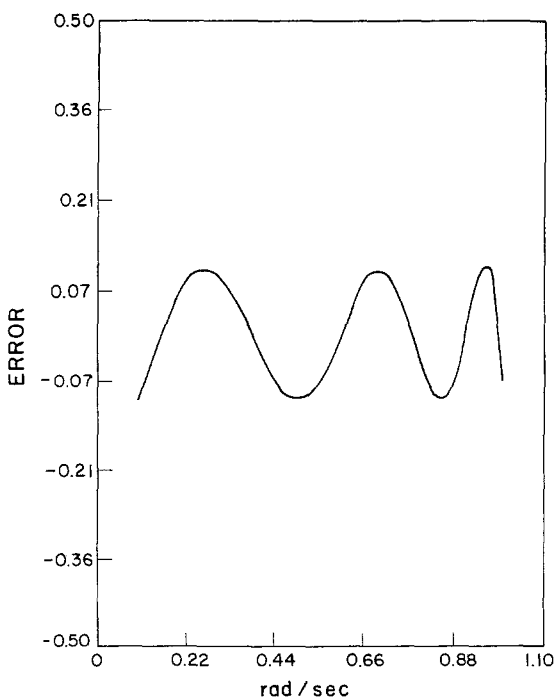

Fig. 7. Optimum error curve.

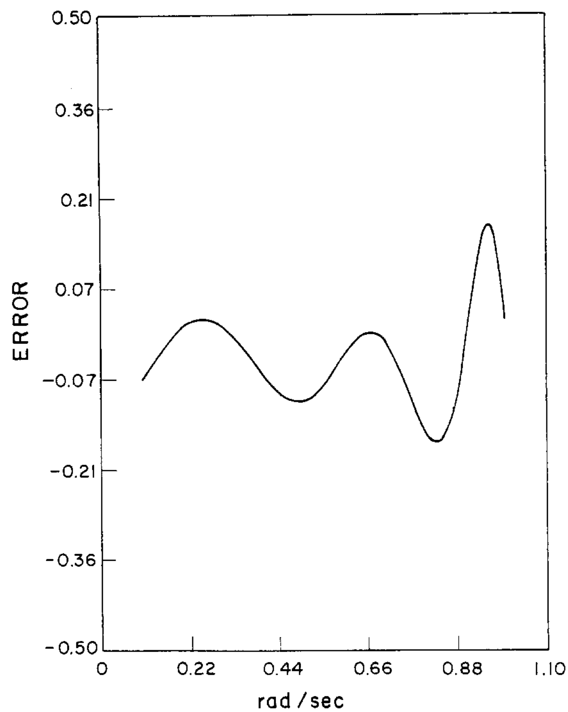

Fig. 8. Error curve obtained by expanding the group delay into the series of Chebychev polynomials [20].

7. Conclusions

From the theoretical and numerical results presented it can be seen that the present approach for nonlinear minimax optimization (and therefore for nonlinear programming) is very promising. The advantages of the present approach over the Bandler–Charalambous approach are similar to the advantages of the augmented Lagrangian approach for nonlinear programming over the standard penalty methods. In both cases, the present approach and the augmented Lagrangian approach we iterate on the estimates of the multipliers at each optimum solution. By transforming the minimax problem into a nonlinear programming problem and using the Rockafellar's approach [18, 19] to solve the problem, we will obtain an algorithm similar to the present approach. Therefore the present algorithm can be viewed either as an alternative to Rockafellar's approach, or as a smoothing of the minimax objective function. Other methods of updating the minimax multipliers is by performing a Newton (or a Quasi-Newton) step on the Kuhn–Tucker conditions for optimality. Other very recent work on the minimax problem can be found in [23, 24].

Acknowledgment

The author wishes to express his appreciation to Dr. A.R. Conn of the Department of Combinatorics and Optimization for his valuable comments. Furthermore, the author wishes to thank the Referee for his valuable comments.

References

[1] J. Asaadi, "A computational comparison of some non-linear programs", *Mathematical Programming* 4 (1973) 144–154.

[2] J.W. Bandler and C. Charalambous, "Practical least pth optimization of networks", *IEEE Transactions on Microwave Theory and Techniques (1972 Symposium Issue)* MTT-20 (1972) 834–840.

[3] J.W. Bandler and C. Charalambous, "A new approach to non-linear programming", in: *Proceeding of the Fifth Hawaii International Conference on Systems Sciences*, University of Hawaii, Honolulu, HI, January 1972.

[4] J.W. Bandler and C. Charalambous, "Nonlinear programming using minimax techniques", *Journal of Optimization Theory and Techniques* 13 (1974) 607–619.

[5] D.P. Bertsekas, "Nondifferentiable optimization via approximation", *Mathematical Programming* 3 (1975) 1–25.

[6] C. Charalambous, "Minimax design of recursive digital filters", *Computer Aided Design* 6 (1974) 73–81.

[7] C. Charalambous, "Nonlinear least pth optimization and nonlinear programming", *Mathematical Programming* 12 (1977) 195–225.

[8] C. Charalambous and J.W. Bandler, "Nonlinear minimax optimization as a sequence of least pth optimization with finite values of p", *International Journal of Systems Science* 7 (1976) 377–391.

[9] C. Charalambous and A.R. Conn, "An efficient method to solve the minimax problem directly", *SIAM Journal of Numerical Analysis* 15 (1978) 162–187.

[10] A.R. Colville, "A comparative study on nonlinear programming codes", IBM New York Scientific Center Report 320–2949 (1968).

[11] V.F. Dem'yanov and V.N. Malozemov, *Introduction to minimax* (Ketev Publishing House Jerusalem, Israel, 1974).

[12] A.V. Fiacco and G.P. McCormick, "*Nonlinear programming: sequential unconstrained minimization techniques*" (Wiley, New York, 1968).

[13] R. Fletcher, "A new approach to variable metric algorithms", *The Computer Journal* 13 (1970) 317–332.

[14] M.R. Hestenes, "Multiplier and gradient methods", *Journal of Optimization Theory and Applications* 4 (1969) 303–320.

[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, Howell Report H.L. 73/4558.

[16] V.T. Polyak and N.V. Tret'yakov, "The method of penalty estimates for conditional extremum problems", *Žurnal Vyčislitel noĭ Matematiki i Matematičeskoĭ Fiziki* 13 (1973) 34–36.

[17] M.J.D. Powell, "A method for non-linear constraints in minimization problems", in: R. Fletcher, ed., *Optimization* (Academic Press, 1969).

[18] R.T. Rockafellar, "A dual approach to solving nonlinear programming problems by unconstrained optimization", *Mathematical Programming* 5 (1973) 354–373.

[19] R.T. Rockafellar, "Augmented Lagrangian multiplier functions and duality in nonconvex programming", *SIAM Journal on Control* 12 (1974) 268–285.

[20] J. Vlach, *Computerized approximation and synthesis of linear networks* (Wiley, New York, 1969).

[21] W.I. Zangwill, "Nonlinear programming via penalty functions", *Management Science* 13 (1967) 344–358.

[22] F.M. El-Turky, C. Charalambous and J. Vlach, "Minimax approximation of group delay characteristics", 4th International Symposium on Computers, Electronics, and Control, Nov. 1978, Toronto, Ontario, Canada.

[23] C. Charalambous and O. Moharram, "Quasi-Newton methods for minimax optimization", Department of Systems Design, University of Waterloo, Waterloo, Ont., Canada, Technical Report No. 48 0-240179 (Jan. 1979).

[24] A.R. Conn, "An efficient second order method to solve the (constrained) minimax problem", Department of Combinatorics and Optimization, University of Waterloo, Waterloo, Ont., Canada (Jan. 1979).