Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 436415, 9 pages
http://dx.doi.org/10.1155/2014/436415

Research Article

A QP-Free Algorithm for Finite Minimax Problems

Daolan Han,[1] Jinbao Jian,[2] and Qinfeng Zhang[3]

[1] *College of Science, Guangxi University for Nationalities, Nanning, Guangxi 530006, China*
[2] *School of Mathematics and Information Science, Yulin Normal University, Yulin, Guangxi 537000, China*
[3] *Guangxi Economic Management Cadre College, Nanning, Guangxi 530007, China*

Correspondence should be addressed to Jinbao Jian; jianjb@gxu.edu.cn

Received 5 April 2014; Accepted 5 June 2014; Published 24 June 2014

Academic Editor: Chong Li

The nonlinear minimax problems without constraints are discussed. Due to the expensive computation for solving QP subproblems with inequality constraints of SQP algorithms, in this paper, a QP-free algorithm which is also called sequential systems of linear equations algorithm is presented. At each iteration, only two systems of linear equations with the same coefficient matrix need to be solved, and the dimension of each subproblem is not of full dimension. The proposed algorithm does not need any penalty parameters and barrier parameters, and it has small computation cost. In addition, the parameters in the proposed algorithm are few, and the stability of the algorithm is well. Convergence property is described and some numerical results are provided.

1. Introduction

We consider the nonlinear minimax problems of the form

$$\min_{x\in R^n} \max_{j\in I} f_j(x), \tag{1}$$

where $I = \{1, 2, \ldots, m\}$ and $f_j\ (j \in I) : R^n \to R$ are continuously differentiable. Denote

$$\begin{aligned} F(x) &= \max\left\{f_j(x), j \in I\right\}, \\ I(x) &= \left\{j \in I \mid f_j(x) = F(x)\right\}, \\ g_j(x) &= \nabla f_j(x), \quad j \in I. \end{aligned} \tag{2}$$

By introducing an auxiliary variable $s \in R$, the problem (1) can be represented as the following standard nonlinear program (see [1]):

$$\min_{x\in R^n, s\in R} \left\{s \mid f_j(x) - s \le 0, j \in I\right\}. \tag{3}$$

It is obvious that the KKT conditions of (3) are equivalent to

$$\begin{aligned} &\sum_{j\in I} \lambda_j g_j(x) = 0, \qquad \sum_{j\in I} \lambda_j = 1; \\ &\lambda_j\left[f_j(x) - F(x)\right] = 0, \quad \lambda_j \ge 0, \ j \in I. \end{aligned} \tag{4}$$

So, a point x is called the stationary point (see [2]) of (1), if there exists a vector $\lambda = (\lambda_j, j \in I)$ such that (4) holds, where λ is called the multiplier vector.

1.1. Related Work.

1.1. Related Work. Some algorithms have been proposed to solve the minimax problems (1) and can be grouped into three classes. The first one is the direct nonsmooth method. The problem (1) is viewed as an unconstrained nonsmooth optimization problem, which can be solved by some nonsmooth methods, such as bundle methods, gradient sampling methods, and cutting plane methods; see, for example, [3–5].

Secondly, a variety of regularization approaches have been used to obtain smooth approximations to the problem (1); see, for example, [6–10]. The main advantage of the smoothing techniques is that the minimax problems are converted into smooth unconstrained optimization problems that can be solved by a standard unconstrained minimization solver. However, when the approximation accuracy is high, the smooth approximating problems become significantly ill-conditioned. Hence, the unconstrained optimization solver may experience numerical difficulty and slow convergence. Consequently, the simple use of smoothing techniques is complicated by the need of trading-off accuracy of approximation against problem ill-conditioning.

The third one is based on solving the equivalent smooth nonlinear programming problem (3), such as the sequential quadratic programming (SQP) methods (see [11–16]), the trust-region strategies (see [17, 18]), sequential quadratically constrained quadratic programming (SQCQP) method (see [19]), gradient projection method (see [20]), and the interior point (IP) methods (see [21–23]). The advantage of this class method is that the nondifferentiable optimization problem (1) can be transformed into an equivalent smooth constrained nonlinear programming problem which can be solved by well-established methods.

It is well known that the SQP method is one of the efficient methods for solving the smooth nonlinear program due to its fast convergence rate. Jian et al. [14], Hu et al. [15], and Zhu et al. [16] present some new SQP type algorithms for unconstrained minimax problems, respectively. For an iterative point x^k, a new quadratic program (QP) subproblem is given by

$$\begin{aligned} \min \quad & z + \frac{1}{2} d^T H_k d \\ \text{s.t.} \quad & f_j\left(x^k\right) + g_j\left(x^k\right)^T d - F\left(x^k\right) \le z, \qquad j \in I_k, \end{aligned} \tag{5}$$

where H_k is a symmetric positive matrix and I_k is the ε active constraints set. The descent direction of $F(x)$ can be obtained by solving (5), and the algorithms have global and superlinear convergence by introducing another correction direction.

In these SQP algorithms or SQCQP method, QP subproblems or quadratically constrained quadratic programming (QCQP) subproblems with inequality constraints are also required to be solved which are computationally expensive compared with system of linear equations. In addition, the IP methods and the second class smoothing techniques do not need to solve QP and QCQP subproblems, but they need the penalty parameter or barrier parameter which is difficult to deal with and causes numerical difficulty when the penalty parameter or barrier parameter is too large. Therefore, it is necessary to construct a new algorithm without solving any QP and QCQP subproblems and using penalty parameters and barrier parameters.

1.2. Division of the Systems of Linear Equations. In this paper, we intend to replace the QP subproblem (5) used in [14–16] by two systems of linear equations with the same matrix so that the computation effort per iteration is much less and propose a QP-free algorithm, which does not need any penalty parameters and barrier parameters.

For (5), in order to speed up the rate of convergence and construct the systems of linear equations conveniently, we select an index $j_k \in I(x^k)$ and consider the following QP subproblem by introducing more parameters η_j^k, $j \in I_k \setminus \{j_k\}$, associated with the iterate x^k:

$$\begin{aligned} \min \quad & z + \frac{1}{2} d^T H_k d \\ \text{s.t.} \quad & g_{j_k}\left(x^k\right)^T d \le z, \\ & f_j\left(x^k\right) + g_j\left(x^k\right)^T d - F\left(x^k\right) \le \eta_j^k z, \\ & j \in \widetilde{I}_k \triangleq I_k \setminus \{j_k\}. \end{aligned} \tag{6}$$

Since (6) is a convex program with linear constraints, its optimal solution is the KKT point; that is, (6) is equivalent to the following KKT system with the multiplier vector $(\mu, \lambda_j, j \in \widetilde{I}_k)$:

$$\begin{gathered} \mu + \sum_{j \in \widetilde{I}_k} \eta_j^k \lambda_j = 1, \\ H_k d + \mu g_{j_k}\left(x^k\right) + \sum_{j \in \widetilde{I}_k} \lambda_j g_j\left(x^k\right) = 0, \\ 0 \le \mu \perp \left(-g_{j_k}\left(x^k\right)^T d + z\right) \ge 0, \\ 0 \le \lambda_j \perp \left(-f_j\left(x^k\right) + F\left(x^k\right) - g_j\left(x^k\right)^T d + \eta_j^k z\right) \ge 0, \\ j \in \widetilde{I}_k. \end{gathered} \tag{7}$$

Motivated by the KKT conditions above, we present the following system of linear equations (SLE):

$$\begin{gathered} \mu + \sum_{j \in \widetilde{I}_k} \eta_j^k \lambda_j = 1, \\ H_k d + \mu g_{j_k}\left(x^k\right) + \sum_{j \in \widetilde{I}_k} \lambda_j g_j\left(x^k\right) = 0, \\ g_{j_k}\left(x^k\right)^T d - z = 0, \qquad g_j\left(x^k\right)^T d - \eta_j^k z = 0, \\ j \in \widetilde{I}_k. \end{gathered} \tag{8}$$

The equation "$g_{j_k}(x^k)^T d - z = 0$" comes from $\eta^k = (\eta_j^k, j \in \widetilde{I}_k) \to 0$ $(\Rightarrow \mu \to 1)$. Obviously, the system above is equivalent to

$$\begin{gathered} H_k d + \sum_{j \in \widetilde{I}_k} \left(g_j\left(x^k\right) - \eta_j^k g_{j_k}\left(x^k\right)\right) \lambda_j = -g_{j_k}\left(x^k\right), \\ \left(g_j\left(x^k\right) - \eta_j^k g_{j_k}\left(x^k\right)\right)^T d = 0, \quad j \in \widetilde{I}_k. \end{gathered} \tag{9}$$

Let $(d^{k0}, \lambda_{\widetilde{I}_k}^{k0})$ be the solution of (9), and d^{k0} is taken as the first direction of our algorithm, which is not entirely suitable as the main search direction although it is a descent direction of $F(x)$. In fact, the global convergence cannot be guaranteed, since the properties of the nonnegativity and complementarity of multiplier vector cannot be guaranteed at the point x^k with d^{k0} and at the accumulation points of the infinite sequence $\{x^k\}$ generated by the proposed algorithm. Thus, we consider computing the second direction d^k by another linear system:

$$\begin{gathered} H_k d + \sum_{j \in \widetilde{I}_k} \left(g_j\left(x^k\right) - \eta_j^k g_{j_k}\left(x^k\right) \right) \lambda_j = -g_{j_k}\left(x^k\right), \\ \left(g_j\left(x^k\right) - \eta_j^k g_{j_k}\left(x^k\right) \right)^T d = v_j^k, \quad j \in \widetilde{I}_k, \end{gathered} \tag{10}$$

where the right-hand parameters v_j^k, $j \in \widetilde{I}_k$, are yielded by $\lambda_{\widetilde{I}_k}^{k0}$ and $F(x^k) - f_j(x^k)$, $j \in \widetilde{I}_k$ as follows:

$$v_j^k = \begin{cases} \lambda_j^{k0}, & \text{if } \lambda_j^{k0} < 0; \\ \lambda_j^{k0} \left[F\left(x^k\right) - f_j\left(x^k\right) \right], & \text{if } \lambda_j^{k0} \geq 0. \end{cases} \tag{11}$$

Considering that the linear systems (10) and (9) have the same decomposed coefficient matrix, the computational cost is typically low. Lemma 8 shows that d^k is still a descent direction of $F(x)$, so d^k can be taken as the main search direction to design the algorithm.

The parameters η_j^k, $j \in \widetilde{I}_k$, need to be devised deliberately to guarantee the nonsingularity of the coefficient matrix and global convergence. It is a difficult work throughout the whole research.

1.3. Properties of Our Algorithm. The proposed algorithm in this paper possesses the following properties.

(i) Only the constraints indexed by some subset I_k of I are considered which reduces the scale and computation cost of the subproblems to some extent.

(ii) At each iteration, only the solutions of two linear systems with the same coefficient matrix are required; that is, the new algorithm is completely QP-free.

(iii) It does not need any penalty parameters and barrier parameters. Therefore, the difficulty of choosing some suitable penalty parameters and barrier parameters is avoided.

(iv) It needs few parameters which are adjusted easily, and the algorithm is robust.

(v) It has weakly global convergence under some suitable assumptions.

We conclude this section by giving some notation which is used throughout this paper. The symbol $\|\cdot\|$ refers to the Euclidean norm. In addition, we denote by $\emptyset$ an empty set, the cardinality of any finite set J by $|J|$, and by det (M) the determinant of the matrix M. Furthermore, the directional derivation of F at the point x along with the direction d is denoted by $F'(x; d)$. It is easy to know that

$$F'(x; d) = \max\left\{ g_j(x)^T d, j \in I(x) \right\}. \tag{12}$$

2. Description of Algorithm

The new algorithm is based on the following assumption.

(A1) The vectors $\{g_j(x), j \in I(x) \setminus \{l\}\}$ are linearly independent for any $l \in I(x)$ and each point $x \in R^n$.

Lemma 1. *Suppose that the vectors $\{g_j(x), j \in J\}$ are linearly independent for $x \in R^n$ and index set $J \subseteq I$. Then, for any given $l \in I$, there exists a constant $\rho_J(x) > 0$ such that $\{g_j(x) - \zeta \|g_j(x)\| g_l(x), j \in J\}$ are linearly independent for each $\zeta \in [0, \rho_J(x)]$.*

For a point $x \in R^n$, an index $l \in I$, and a given index set J such that $\{g_j(x), j \in J\}$ are linearly independent, we introduce the following technique similar to [24] to generate the parameter $\rho_J(x)$ in Lemma 1. Define

$$\begin{gathered} f_J(x) = \left(f_j(x), j \in J\right), \qquad g_J(x) = \left(g_j(x), j \in J\right), \\ n_j(x) = \frac{g_j(x)}{\left\|g_j(x)\right\|}, \qquad N_J(x) = \left(n_j(x), j \in J\right), \end{gathered} \tag{13}$$

$$\rho_J(x) = \frac{\det\left[N_J(x)^T N_J(x)\right]}{e\,|J|\, \|g_l(x)\| + 1}, \tag{14}$$

$$a_j(x, \epsilon) = n_j(x) - \epsilon \rho_J(x)\, g_l(x), \tag{15}$$

where e is Napierian base and the parameter $\epsilon \geq 0$.

Lemma 2 (see [24]). *Suppose that the vectors $\{g_j(x), j \in J\}$ are linearly independent. Then the vectors $\{a_j(x, \epsilon), j \in J\}$ are also linearly independent for each $\epsilon \in [0, 1]$.*

Let x^k be a given iteration point. We denote $j_k = \min\{j \mid j \in I(x^k)\}$ and use the following pivoting operation to generate the index set $\widetilde{I}_k$ such that $R_k \triangleq g_{\widetilde{I}_k}(x^k)$ has full column rank, so vectors $\{g_j(x^k), j \in \widetilde{I}_k\}$ are linearly independent.

Pivoting Operation (POP)

Step (i). Select an initial parameter $\varepsilon = \varepsilon_{k-1}$.

Step (ii). Generate ε active constraint subset $I(x^k, \varepsilon)$ and matrix G_k by

$$\begin{gathered} I\left(x^k, \varepsilon\right) = \left\{ j \in I \mid -\varepsilon \leq f_j\left(x^k\right) - F\left(x^k\right) \leq 0 \right\}, \\ G_k = \left(g_j\left(x^k\right), j \in I\left(x^k, \varepsilon\right) \setminus \{j_k\} \right). \end{gathered} \tag{16}$$

Step (iii). If $I(x^k, \varepsilon) \setminus \{j_k\} = \emptyset$ or $\det(G_k^T G_k) \geq \varepsilon$, then set $\widetilde{I}_k := I(x^k, \varepsilon) \setminus \{j_k\}$, $\varepsilon_k = \varepsilon$, and stop; otherwise, set $\varepsilon := \varepsilon/2$ and repeat Step (ii).

For simplicity, denote

$$\rho_k \triangleq \rho_{\widetilde{I}_k}\left(x^k\right),$$
$$A_J\left(x^k,\zeta\right) = \left(g_j\left(x^k\right) - \zeta\left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right), j \in J\right), \tag{17}$$
$$\zeta \in [0,\rho_k].$$

From the POP and Lemma 2 above, we know that $(a_j(x^k,\epsilon), j \in \widetilde{I}_k) = (n_j(x^k) - \epsilon\rho_{\widetilde{I}_k}(x^k) g_{j_k}(x^k), j \in \widetilde{I}_k)$ has full column rank for each $\epsilon \in [0,1]$. Furthermore, it is easy to get the following lemma.

Lemma 3. *Suppose that (A1) holds. Then the matrix $A_J(x^k,\zeta)$ has full column rank for each $J \subseteq \widetilde{I}_k$ and $\zeta \in [0,\rho_k]$.*

To describe some beneficial properties of the POP above, which is helpful for discussing the convergence of our algorithm, we have the following results.

Lemma 4. *Suppose that (A1) holds, and let $x^k \in R^n$.*

(i) *The parameter ε_k can be obtained in a finite number of steps in the POP.*

(ii) *If a sequence $\{x^k\}$ is bounded, then there exist two constants $\bar{\varepsilon}, \widetilde{\varepsilon} > 0$ such that*

$$\inf\{\varepsilon_k\} = \bar{\varepsilon}, \qquad \det\left(R_k^T R_k\right) \geq \bar{\varepsilon}, \quad \forall k, \tag{18}$$

$$\det\left[A_{J_k}\left(x^k,\zeta\right)^T A_{J_k}\left(x^k,\zeta\right)\right] \geq \widetilde{\varepsilon}, \quad \forall k,\ \forall \zeta \in [0,\rho_k], \tag{19}$$
$$\forall J_k \subseteq \widetilde{I}_k.$$

Proof. Based on the assumption (A1), it is easy to get that (i) and (18) hold, so it is omitted here.

Now, we will prove that (19) holds. In view of $\widetilde{I}_k$ and J_k being the subsets of the fixed and finite set I and the boundedness of $\{x^k\}$, we assume by contradiction without loss of generality that there exist an infinite index set K and $\zeta_k \in [0,\rho_k]$ such that

$$\widetilde{I}_k \equiv \widetilde{I}, \qquad J_k \equiv J \subseteq \widetilde{I}, \qquad j_k \equiv j_0, \qquad x^k \longrightarrow x^*,$$
$$\det\left[A_J\left(x^k,\zeta_k\right)^T A_J\left(x^k,\zeta_k\right)\right] \longrightarrow 0, \quad k \in K. \tag{20}$$

According to (14) and (A1), we can get $\rho_k \xrightarrow{k\in K} \rho_* = \rho_{\widetilde{I}}(x^*)$. So, $\{\zeta_k, k \in K\}$ is bounded from $\zeta_k \in [0,\rho_k]$; we may assume that there exists an infinite index set $K' \subseteq K$ such that $\zeta_k \xrightarrow{k\in K'} \zeta_* \in [0,\rho_*]$. On the other hand, from (18) one knows that vectors $\{g_j(x^*), j \in \widetilde{I}\}$ are linearly independent. Therefore, by Lemma 2, we know that $\{a_j(x^*,\epsilon), j \in \widetilde{I}\}$ are also linearly independent for each $\epsilon \in [0,1]$, which implies that vectors

$$\left\{\left\|g_j\left(x^*\right)\right\| a_j\left(x^*,\epsilon\right) = g_j\left(x^*\right) - \epsilon\rho_*\left\|g_j\left(x^*\right)\right\| g_{j_0}\left(x^*\right),\ j \in \widetilde{I}\right\} \tag{21}$$

are linearly independent for each $\epsilon \in [0,1]$. So, $\{g_j(x^*) - \zeta\|g_j(x^*)\|g_{j_0}(x^*), j \in J \subseteq \widetilde{I}\}$ are linearly independent for each $\zeta \in [0,\rho_*]$. Denote that $A_* = (g_j(x^*) - \zeta_*\|g_j(x^*)\|g_{j_0}(x^*), j \in J)$, A_*, has full column rank since $\zeta_* \in [0,\rho_*]$. However,

$$0 = \lim_{k\in K'} \det\left[A_J\left(x^k,\zeta_k\right)^T A_J\left(x^k,\zeta_k\right)\right] = \det\left(A_*^T A_*\right), \tag{22}$$

which contradicts the fact that A_* has full column rank. So (19) holds. □

A detailed description of the algorithm for solving (1) is given below.

Algorithm A 1. Parameters: $\alpha \in (0,1/2)$, $\beta \in (0,1)$, $\varepsilon_{-1} > 0$.

Step 0 (initialization). Consider $x^0 \in R^n$, an initial symmetric positive definite matrix H_0. Set $k := 0$.

Step 1 (generating ε active set). Set parameter $\varepsilon = \varepsilon_{k-1}$, generate the set $\widetilde{I}_k$ by the POP, and let ε_k be the corresponding termination parameter.

Step 2. Compute $\rho_k = \rho_{\widetilde{I}_k}(x^k)$ according to (14), and adjust the parameter ζ_k by

$$\zeta_k = \begin{cases} \rho_0, & \text{if } k = 0; \\ \min\left\{\rho_k, \left\|d^{k-1,0}\right\| + \left\|\nu^{k-1}\right\|, \zeta_{k-1}\right\}, & \text{if } k > 0. \end{cases} \tag{23}$$

Step 3. Compute the unique solution $(d^{k0}, \lambda^{k0}_{\widetilde{I}_k})$ of the following linear system:

$$\text{(LS1)} \quad M\left(x^k, H_k, \zeta_k\right)\begin{pmatrix} d \\ \lambda \end{pmatrix} = \begin{pmatrix} -g_{j_k}\left(x^k\right) \\ 0 \end{pmatrix}, \tag{24}$$

with

$$M\left(x^k, H_k, \zeta_k\right) = \begin{pmatrix} H_k & A_k \\ A_k^T & 0 \end{pmatrix}, \quad A_k \triangleq A_{\widetilde{I}_k}\left(x^k,\zeta_k\right). \tag{25}$$

Step 4. Compute the unique solution $(d^k, \lambda^k_{\widetilde{I}_k})$ of the following linear system:

$$\text{(LS2)} \quad M\left(x^k, H_k, \zeta_k\right)\begin{pmatrix} d \\ \lambda \end{pmatrix} = \begin{pmatrix} -g_{j_k}\left(x^k\right) \\ \nu^k \end{pmatrix}, \tag{26}$$

where $\nu^k = (\nu^k_j, j \in \widetilde{I}_k)$ is yielded by

$$\nu^k_j = \begin{cases} \lambda^{k0}_j, & \text{if } \lambda^{k0}_j < 0; \\ \lambda^{k0}_j\left[F\left(x^k\right) - f_j\left(x^k\right)\right], & \text{if } \lambda^{k0}_j \geq 0. \end{cases} \tag{27}$$

If $d^k = 0$ and

$$\lambda^k_{j_k} \triangleq 1-\zeta_k \sum_{j\in\tilde{I}_k}\lambda^k_j \left\|g_j\left(x^k\right)\right\| \ge 0, \tag{28}$$

then x^k is a stationary point of (1), and stop.

Step 5 (doing line search). Compute the step size t_k, the first number t of the sequence $\{1,\beta,\beta^2,\dots\}$ satisfying

$$F\left(x^k+td^k\right) \le F\left(x^k\right)+\alpha t F'\left(x^k;d^k\right). \tag{29}$$

Step 6. Set $x^{k+1} = x^k + t_k d^k$, and compute a new symmetric positive definite matrix H_{k+1}. Set $k := k+1$, and go back to Step 1.

Remark 5. If the case $d^k = 0$, $\lambda^k_{j_k} < 0$ arises; the algorithm will stop at iterate x^k which is not a stationary point of (1). To avoid this pitfall, we will reset $\zeta_k := (1/2)\zeta_k$ and solve (LS1) and (LS2) again.

For convenience of analysis in the rest of this paper, we give the equivalent forms of (LS1)-(LS2):

(LS1)

$$\Longleftrightarrow \begin{cases} H_k d^{k0} + \displaystyle\sum_{j\in\tilde{I}_k}\lambda^{k0}_j \left(g_j\left(x^k\right)-\zeta_k\left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)\right) \\ \quad = -g_{j_k}\left(x^k\right), \\ \left(g_j\left(x^k\right)-\zeta_k\left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)\right)^T d^{k0} = 0, \\ \qquad\qquad\qquad\qquad j\in\tilde{I}_k; \end{cases} \tag{30}$$

(LS2)

$$\Longleftrightarrow \begin{cases} H_k d^{k} + \displaystyle\sum_{j\in\tilde{I}_k}\lambda^{k}_j \left(g_j\left(x^k\right)-\zeta_k\left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)\right) \\ \quad = -g_{j_k}\left(x^k\right), \\ \left(g_j\left(x^k\right)-\zeta_k\left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)\right)^T d^{k} = v^k_j, \\ \qquad\qquad\qquad\qquad j\in\tilde{I}_k. \end{cases} \tag{31}$$

So, the parameters η^k_j, $j\in\tilde{I}_k$, in (9)-(10) are selected as $\eta^k_j = \zeta_k\|g_j(x^k)\|$. From Algorithm A and (A1) as well as Lemma 3, we can get the following lemmas immediately.

Lemma 6. *The matrix $M(x^k,H_k,\zeta)$ is nonsingular for each $\zeta\in[0,\rho_k]$. Therefore, the coefficient matrix $M(x^k,H_k,\zeta_k)$ in systems (LS1)-(LS2) is nonsingular.*

Taking into account the inverse matrix of $M_k \triangleq M(x^k,H_k,\zeta_k)$ that can be expressed as $M_k^{-1} = \begin{pmatrix} C_k & B_k \\ B_k^T & -D_k^{-1}\end{pmatrix}$, with

$$D_k = A_k^T H_k^{-1} A_k, \qquad B_k = H_k^{-1} A_k D_k^{-1}, \\ C_k = H_k^{-1} - H_k^{-1} A_k B_k^T, \tag{32}$$

from (LS1)-(LS2) and (32), we have the following relations:

$$d^{k0} = -C_k g_{j_k}\left(x^k\right), \qquad \lambda^{k0}_{\tilde{I}_k} = -B_k^T g_{j_k}\left(x^k\right); \tag{33}$$

$$d^k = d^{k0} + B_k v^k, \qquad \lambda^k_{\tilde{I}_k} = \lambda^{k0}_{\tilde{I}_k} - D_k^{-1} v^k. \tag{34}$$

Lemma 7. *If Algorithm A stops at an iterate x^k with $d^k = 0$ and $\lambda^k_{j_k} \ge 0$, then x^k is a stationary point of the problem* (1).

Proof. If Algorithm A stops at an iterate x^k with $d^k = 0$ and $\lambda^k_{j_k} \ge 0$, it follows from (31) that

$$\left(1-\zeta_k\sum_{j\in\tilde{I}_k}\lambda^k_j\left\|g_j\left(x^k\right)\right\|\right) g_{j_k}\left(x^k\right) + \sum_{j\in\tilde{I}_k}\lambda^k_j g_j\left(x^k\right) = 0, \\ v^k = 0. \tag{35}$$

By the definition of v^k, we have $\lambda^{k0}_{\tilde{I}_k} \ge 0$, $\lambda^{k0}_j[f_j(x^k)-F(x^k)] = 0$, $j\in\tilde{I}_k$, and hence, by (34),

$$\lambda^k_{\tilde{I}_k} = \lambda^{k0}_{\tilde{I}_k} \ge 0, \qquad \lambda^k_j\left[f_j\left(x^k\right)-F\left(x^k\right)\right] = 0, \; j\in\tilde{I}_k. \tag{36}$$

Since R_k has full column rank, from (35) it is easy to get $\lambda^k_{j_k} = 1-\zeta_k\sum_{j\in\tilde{I}_k}\lambda^k_j\|g_j(x^k)\| \ne 0$ and $\lambda^k_{j_k} > 0$. If one denotes $u^k_{I_k} = (u^k_j = \lambda^k_j/\sum_{j\in I_k}\lambda^k_j,\ j\in I_k)$, $u^k = (u^k_{I_k}, 0_{I\setminus I_k})$, then from (35)-(36) we can conclude that x^k is a stationary point of (1) with the multiplier vector u^k. □

Lemma 8. *If $d^k \ne 0$, then*

(i) $g_{j_k}(x^k)^T d^k = -(d^{k0})^T H_k d^{k0} - \sum_{\lambda^{k0}_j<0}(\lambda^{k0}_j)^2 + \sum_{\lambda^{k0}_j\ge 0}(\lambda^{k0}_j)^2[f_j(x^k)-F(x^k)] < 0$;

(ii) $g_j(x^k)^T d^k \le \zeta_k\|g_j(x^k)\| g_{j_k}(x^k)^T d^k < 0$, $j\in I(x^k)\setminus\{j_k\}$;

(iii) *d^k is a descent direction of $F(x)$ at the iterate x^k, so the proposed algorithm is well defined.*

Proof. (i) From (34), (30), (33), and (27), one gets

$$\begin{aligned} g_{j_k}\left(x^k\right)^T d^k &= g_{j_k}\left(x^k\right)^T\left(d^{k0}+B_k v^k\right) \\ &= -\left(d^{k0}\right)^T H_k d^{k0} - \left(\lambda^{k0}_{\tilde{I}_k}\right)^T A_k^T d^{k0} - \left(\lambda^{k0}_{\tilde{I}_k}\right)^T v^k \\ &= -\left(d^{k0}\right)^T H_k d^{k0} - \left(\lambda^{k0}_{\tilde{I}_k}\right)^T v^k \\ &= -\left(d^{k0}\right)^T H_k d^{k0} - \sum_{\lambda^{k0}_j<0}\left(\lambda^{k0}_j\right)^2 \\ &\quad + \sum_{\lambda^{k0}_j\ge 0}\left(\lambda^{k0}_j\right)^2\left[f_j\left(x^k\right)-F\left(x^k\right)\right]. \end{aligned} \tag{37}$$

Furthermore, since $d^k \neq 0$, from (34), we can get $(d^{k0}, v^k) \neq (0, 0)$. This implies that (i) holds.

(ii) If $j \in I(x^k) \setminus \{j_k\}$, from $I(x^k) \setminus \{j_k\} \subseteq \widetilde{I}_k$ and (27), we have $v_j^k \leq 0$. From Remark 5, we can assume without loss of generality that $d^{k-1} \neq 0$. Furthermore, it follows from (34) and (23) that $\zeta_k > 0$. Therefore, from the second formula of (31) and assumption (A1), it is easy to get

$$\begin{aligned} g_j\left(x^k\right)^T d^k &= \zeta_k \left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)^T d^k + v_j^k \\ &\leq \zeta_k \left\|g_j\left(x^k\right)\right\| g_{j_k}\left(x^k\right)^T d^k < 0, \qquad (38) \\ & j \in I\left(x^k\right) \setminus \{j_k\}. \end{aligned}$$

(iii) From (12) and the above results (i)-(ii), one knows that $F'(x^k; d^k) < 0$, which implies that d^k is a descent direction of $F(x)$ at x^k. So the line search can be performed and the proposed algorithm is well defined. □

3. Global Convergence

In this part, under mild assumptions, we show that Algorithm A is weak globally convergent; that is, there exists at least one accumulation point of the iterates $\{x^k\}$ yielded by Algorithm A such that it is a stationary point of (1). To this end, in addition to (A1), the following two assumptions are necessary.

(A2) The sequence $\{x^k\}$ generated by Algorithm A is bounded.

(A3) There exist two constants $a, b > 0$ such that

$$a\|d\|^2 \leq d^T H_k d \leq b\|d\|^2, \quad \forall d \in R^n, \ \forall k. \qquad (39)$$

The following lemma establishes the boundedness of the associated sequences generated by the algorithm, and its proof is similar to Lemma 3.1 in [25], so it is omitted here.

Lemma 9. *Suppose that (A1)–(A3) hold. Then, the sequences* $\{\rho_k\}$, $\{\zeta_k\}$, $\{M_k^{-1}\}$, $\{d^{k0}\}$, $\{\lambda_{\widetilde{I}_k}^{k0}\}$, $\{d^k\}$, $\{\lambda_{\widetilde{I}_k}^{k}\}$, $\{D_k^{-1}\}$, *and* $\{B_k\}$ *are all bounded.*

Lemma 10. *Suppose that (A1)–(A3) hold. If an infinite index set K satisfies*

$$x^k \longrightarrow x^*, \qquad j_k \equiv j', \qquad g_{j'}\left(x^k\right)^T d^k \longrightarrow 0, \qquad k \in K, \qquad (40)$$

then x^* is a stationary point of (1).

Proof. From Lemma 9, we know there exists an infinite subset $K' \subseteq K$ such that

$$\widetilde{I}_k \equiv \widetilde{I}, \qquad d^{k0} \longrightarrow d^{*0}, \qquad \lambda_{\widetilde{I}}^{k0} \longrightarrow \lambda_{\widetilde{I}}^{*0}, \qquad k \in K'. \qquad (41)$$

Again, $\lim_{k\to\infty}\zeta_k = \zeta_*$ follows since $\{\zeta_k\}$ is monotone and bounded. In view of Lemma 8 (i) and (A3), we have

$$\begin{aligned} 0 \longleftarrow g_{j'}\left(x^k\right)^T d^k \leq -a\left\|d^{k0}\right\|^2 &- \sum_{\lambda_j^{k0}<0}\left(\lambda_j^{k0}\right)^2 \\ &+ \sum_{\lambda_j^{k0}\geq 0}\left(\lambda_j^{k0}\right)^2\left[f_j\left(x^k\right) - F\left(x^k\right)\right] < 0. \end{aligned} \qquad (42)$$

This along with (40)-(41) shows that

$$\begin{gathered} d^{k0} \longrightarrow 0, \qquad \lambda_{\widetilde{I}}^{*0} \geq 0, \qquad k \in K', \\ \left(\lambda_j^{*0}\right)^2\left[f_j\left(x^*\right) - F\left(x^*\right)\right] = 0, \quad j \in \widetilde{I}. \end{gathered} \qquad (43)$$

So, passing to the limit for $k \in K'$, $k \to \infty$, in (30), one gets

$$\begin{gathered} \left(1 - \zeta_* \sum_{j\in\widetilde{I}}\lambda_j^{*0}\left\|g_j\left(x^*\right)\right\|\right) g_{j'}\left(x^*\right) + \sum_{j\in\widetilde{I}}\lambda_j^{*0} g_j\left(x^*\right) = 0; \\ \lambda_j^{*0} \geq 0, \quad \lambda_j^{*0}\left[f_j\left(x^*\right) - F\left(x^*\right)\right] = 0, \quad i \in \widetilde{I}. \end{gathered} \qquad (44)$$

Furthermore, it follows from (23) and (27) that $\zeta_{k+1} \to 0$, $k \in K'$. So, $\zeta_* = 0$ and $\lambda_{j'}^{*0} \triangleq 1 - \zeta_* \sum_{j\in\widetilde{I}}\lambda_j^{*0}\|g_j(x^*)\| = 1$. If one denotes $I' = \widetilde{I} \cup \{j'\}$, $u_{I'}^* = (u_j^* = \lambda_j^{*0} / \sum_{j\in I'}\lambda_j^{*0}, j \in I')$, and $u^* = (u_{I'}^*, 0_{I\setminus I'})$, then (44) implies that x^* is a stationary point with the multiplier vector u^*. □

Theorem 11. *Suppose that (A1)–(A3) hold. Then Algorithm A either stops at a stationary point of* (1) *in a finite number of iterations or generates an infinite sequence $\{x^k\}$, of which at least one accumulation point is a stationary point of* (1). *In such sense, Algorithm A is said to possess weakly global convergence.*

Proof. The cases $\zeta_* = 0$ and $\zeta_* > 0$ are discussed, respectively.

Case A. If $\zeta_* = 0$, from (14), Lemma 4 (ii), and (A2), we know that there exists a constant $\underline{\rho} > 0$ such that $\rho_k \geq \underline{\rho}$ holds for all k large enough. Therefore, in view of (A2) and (23), there exists an infinite index K such that

$$x^k \longrightarrow x^*, \qquad \zeta_k = \left\|d^{k-1,0}\right\| + \left\|v^{k-1}\right\| \longrightarrow 0, \quad k \in K; \quad (45)$$

that is, $d^{k-1,0} \to 0$, $v^{k-1} \to 0$, $k \in K$. So, from (34) and the boundedness of $\{B_k\}$, we have $\lim_{k\in K} d^{k-1} = \lim_{k\in K}(d^{k-1,0} + B_{k-1}v^{k-1}) = 0$. It further follows that $\|x^k - x^{k-1}\| = t_{k-1}\|d^{k-1}\| \to 0$, $k \in K$. Thus, $\lim_{k\in K} x^{k-1} = x^*$. So, according to Lemma 10 and $\lim_{k\in K} d^{k-1} = 0$, it is easy to conclude that x^* is a stationary point of (1).

Case B. If $\zeta_* > 0$, then we have $\zeta_k \geq (1/2)\zeta_*$ for all $k \in K$ large enough. Suppose that x^* is any given accumulation point of

$\{x^k\}$. Since I is a fixed and finite set, from Lemma 9, there always exists an infinite index set K such that

$$I\left(x^k\right) \equiv I_*, \qquad \widetilde{I}_k \equiv \widetilde{I}, \qquad j_k \equiv j', \quad k \in K,$$
$$\left(x^k, d^{k0}, d^k, \lambda^{k0}_{\widetilde{I}}, \lambda^k_{\widetilde{I}}, \zeta_k\right) \longrightarrow \left(x^*, d^{*0}, d^*, \lambda^{*0}_{\widetilde{I}}, \lambda^*_{\widetilde{I}}, \zeta_*\right), \quad k \in K. \tag{46}$$

Suppose by contradiction that x^* is not a stationary point. We first show by contradiction that the given infinite index set K satisfies $g_{j'}(x^k)^T d^k \to 0$, $k \in K$. Otherwise, since $g_{j'}(x^k)^T d^k < 0$ and $g_{j'}(x^k)^T d^k \to g_{j'}(x^*)^T d^*$, $k \in K$, there exists a constant $c > 0$ such that

$$g_{j'}\left(x^k\right)^T d^k \le -c, \quad k \in K \text{ large enough.} \tag{47}$$

Consequently, we prove that the line search inequality (29) holds for all $k \in K$ large enough and $t > 0$ small enough. We denote

$$\omega_k(t) = F\left(x^k + td^k\right) - F\left(x^k\right) - \alpha t F'\left(x^k, d^k\right). \tag{48}$$

It follows from the boundedness of $\{d^k\}$ and Taylor expansion that

$$\begin{aligned} \omega_k(t) &= \max\left\{f_j\left(x^k + td^k\right) - F\left(x^k\right) - \alpha t F'\left(x^k, d^k\right), j \in I\right\} \\ &= \max\left\{f_j\left(x^k\right) - F\left(x^k\right) + t g_j\left(x^k\right)^T d^k \right. \\ &\qquad \left. - \alpha t F'\left(x^k, d^k\right) + o(t), j \in I\right\}. \end{aligned} \tag{49}$$

Denote $a_{kj}(t) = f_j(x^k) - F(x^k) + t g_j(x^k)^T d^k - \alpha t F'(x^k, d^k) + o(t)$. For $j \in I_* \setminus \{j'\} \subseteq \widetilde{I}$, according to (18) in Lemma 4 (ii), it is easy to know that there exists a constant $\sigma > 0$ such that $\|g_j(x^k)\| \ge \sigma$ for all $k \in K$. From Lemma 8 (ii), one has for $k \in K$ large enough and $t > 0$ small enough

$$\begin{aligned} a_{kj}(t) &= t g_j\left(x^k\right)^T d^k - \alpha t F'\left(x^k, d^k\right) + o(t) \\ &\le (1-\alpha) t g_j\left(x^k\right)^T d^k + o(t) \\ &\le (1-\alpha) t \zeta_k \left\|g_j\left(x^k\right)\right\| g_{j'}\left(x^k\right)^T d^k + o(t) \\ &\le -\frac{1}{2}(1-\alpha)\zeta_* \sigma c t + o(t) \le 0. \end{aligned} \tag{50}$$

For $j = j'$, one has for $k \in K$ large enough and $t > 0$ small enough $a_{kj}(t) \le (1-\alpha) t g_{j'}(x^k)^T d^k + o(t) \le -(1-\alpha)ct + o(t) \le 0$.

For $j \in I \setminus I_*$, from the POP and Lemma 4 (ii), we can obtain $f_j(x^k) - F(x^k) < -\varepsilon_k \le -\bar{\varepsilon}$. So, $a_{kj}(t) \le -\bar{\varepsilon} + O(t) \le 0$ holds for $k \in K$ large enough and $t > 0$ small enough.

Summarizing the analysis above, we know that there exists a constant $t_* > 0$ such that the step size $t_k \ge t_*$, $\forall k \in K$. Furthermore, for $\forall j \in I_*$, $g_j(x^k)^T d^k \le -\min\{(1/2)\zeta_* \sigma, 1\}c$. So, $F'(x^k, d^k) \le -\min\{(1/2)\zeta_* \sigma, 1\}c$. Combining (29), we have

$$\begin{aligned} F\left(x^{k+1}\right) - F\left(x^k\right) &\le \alpha t_k F'\left(x^k, d^k\right) \\ &\le -\alpha t_* \min\left\{\frac{1}{2}\zeta_* \sigma, 1\right\} c, \quad k \in K. \end{aligned} \tag{51}$$

Since the whole sequence $\{F(x^k)\}$ is decreasing and $\lim_{k \in K} F(x^k) = F(x^*)$, we know that $\lim_{k \to \infty} F(x^k) = F(x^*)$. So, passing to the limit for $k \in K$ and $k \to \infty$ in inequality (51), one has $-\alpha t_* \min\{(1/2)\zeta_* \sigma, 1\}c \ge 0$, which contradicts the fact that $\alpha > 0$, $t_* > 0$, $\zeta_* > 0$, and $c > 0$. The contradiction shows that (40) holds. Furthermore, one can conclude from Lemma 10 that x^* is a stationary point of (1), which contradicts the assumption. The whole proof is completed. □

4. Numerical Results

In this section, some preliminary numerical tests on 5 typical problems from [26] are reported, and the computation results show that Algorithm A is efficient. All the numerical experiments were implemented on MATLAB 7.0, under Windows XP and 2.2 GHz CPU. The BFGS formula with Powell's modification [27] is adopted in the algorithm, and H_0 is the identity matrix. The parameters were selected as $\alpha = 0.2$, $\beta = 0.6$, and $\varepsilon_{-1} = 1.2$. In addition, execution is terminated if one of the following termination criteria is satisfied:

(a) $\|d^k\| \le 10^{-5}$,

(b) $\|x^{k+1} - x^k\| \le 10^{-5}$.

The computational results are reported in Table 1, and the columns of Table 1 have the following meanings: IP: the initial point; n: the number of variables; m: the number of functions $f_j(x)$; ALG: the type of algorithm; NI: the number of iterations. "Algo A" represents Algorithm A in this paper, "J2006-1" and "J2006-2" represent the algorithms in [13], and "Hu2009" represents the algorithm in [15].

From Table 1, we can see that our algorithm can find the solutions of the test problems with a small number of iterations, and the computational results illustrate that our algorithm executes well for those problems. The numerical results are comparative with the algorithms in [13, 15]. Furthermore, we only need to solve two systems of linear equations with the same coefficient matrix per iteration. Considering that these linear systems have the same decomposed coefficient matrix, the computational cost per iteration of Algorithm A is typically low. This shows the potential advantage of our algorithm when applied to solving problems with large numbers of constraints. In addition, the parameters in the proposed algorithm are few, and the stability of the algorithm is very well.

5. Concluding Remarks

In this paper, a QP-free algorithm without solving any QP and QCQP subproblems is presented for unconstrained nonlinear

Table 1: Numerical results of Algorithm A.

P	IP	n	m	ALG	NI	Approximate solution x^*	$F(x^*)$	$\|d^k\|$
1	$(1,-0.01)^T$	2	3	Algo A	7	$(1.1390, 0.8995)^T$	1.952	$1.715e-6$
				J2006-1	11	$(1.1390, 0.8995)^T$	1.952	$3.462e-7$
				J2006-2	19	$(1.1390, 0.8995)^T$	1.952	$2.457e-6$
				Hu2009	8	—	1.952	$4.248e-6$
2	$(0.01, 0.01)^T$	2	3	Algo A	7	$(1.0000, 0.9999)^T$	2.000	$3.543e-5$
				J2006-1	7	$(1.0000, 1.0000)^T$	2.000	$6.177e-10$
				J2006-2	17	$(1.00000, 1.00000)^T$	2.000	$4.775e-9$
				Hu2009	7	—	2.000	$1.987e-14$
3	$(0.2,-1,2.3,-0.01)^T$	4	4	Algo A	12	$(0.0, 1.0, 2.0, -1.0)^T$	−44.00	$2.206e-5$
				J2006-1	38	$(0.0, 1.0, 2.0, -1.0)^T$	−44.00	$3.087e-6$
				J2006-2	74	$(0.0, 1.0, 2.0, -1.0)^T$	−44.00	$7.733e-6$
				Hu2009	11	—	−44.00	$8.294e-6$
4	$(3,1)^T$	2	3	Algo A	11	$(-0.45330, 0.90659)^T$	0.616	$9.617e-6$
				J2006-1	10	$(-0.45329, 0.90659)^T$	0.616	$6.419e-6$
				J2006-2	21	$(-0.45329, 0.90659)^T$	0.616	$2.641e-6$
				Hu2009	12	—	0.616	$8.797e-6$
5	$(1,1,1)^T$ $(1.01, 0.9, 1)^T$	3	6	Algo A	13	$(0.3282, 0.0, 0.1313)^T$	3.599	$7.343e-6$
				J2006-1	20	$(0.3282, 0.0, 0.1313)^T$	3.599	$7.227e-7$
				J2006-2	34	$(0.3282, 0.0, 0.1313)^T$	3.599	$3.382e-6$
				Hu2009	14	—	3.599	$1.832e-6$

finite minimax problems. At each iteration, only two systems of linear equations with the same coefficient matrix need to be solved. The proposed algorithm does not need any penalty parameters and barrier parameters which are difficult to deal with. Furthermore, under some mild assumptions, the global convergence is attained. As further work of this method, we think that there are still some problems worthy of discussing. For example, the assumption (A1) is different from the linearly independent assumption in common use, and any of them cannot derive the other. The discussion that the assumption (A1) is a constraint qualification needs further consideration. In addition, one should also take into account improving it to have superlinear convergence and generalizing it to solve minimax problems with inequality constraints.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (11271086), the Guangxi Natural Science Foundation (2013GXNSFBA019017, 2012GXNSFAA053007), and Science Foundation of Guangxi Education Department (2013YB080, ZD2014107).

References

[1] C. Charalambous and A. R. Conn, "An efficient method to solve the minimax problem directly," *SIAM Journal on Numerical Analysis*, vol. 15, no. 1, pp. 162–187, 1978.

[2] S. P. Han, "Variable metric methods for minimizing a class of nondifferentiable functions," *Mathematical Programming*, vol. 20, no. 1, pp. 1–13, 1981.

[3] M. Gaudioso and M. F. Monaco, "A bundle type approach to the unconstrained minimization of convex nonsmooth functions," *Mathematical Programming*, vol. 23, no. 2, pp. 216–226, 1982.

[4] W. Hare and J. Nutini, "A derivative-free approximate gradient sampling algorithm for finite minimax problems," *Computational Optimization and Applications*, vol. 56, no. 1, pp. 1–38, 2013.

[5] R. B. Kearfott, S. Muniswamy, Y. Wang, X. Li, and Q. Wang, "On smooth reformulations and direct non-smooth computations for minimax problems," *Journal of Global Optimization*, vol. 57, no. 4, pp. 1091–1111, 2013.

[6] F. G. Vazquez, H. Günzel, and H. Th. Jongen, "On logarithmic smoothing of the maximum function," *Annals of Operations Research*, vol. 101, pp. 209–220, 2001.

[7] S. Xu, "Smoothing method for minimax problems," *Computational Optimization and Applications*, vol. 20, no. 3, pp. 267–279, 2001.

[8] E. Polak, J. O. Royset, and R. S. Womersley, "Algorithms with adaptive smoothing for finite minimax problems," *Journal of Optimization Theory and Applications*, vol. 119, no. 3, pp. 459–484, 2003.

[9] J. Li and J. Huo, "Inexact smoothing method for large scale minimax optimization," *Applied Mathematics and Computation*, vol. 218, no. 6, pp. 2750–2760, 2011.

[10] E. Y. Pee and J. O. Royset, "On solving large-scale finite minimax problems using exponential smoothing," *Journal of Optimization Theory and Applications*, vol. 148, no. 2, pp. 390–421, 2011.

[11] E. Polak, D. Q. Mayne, and J. E. Higgins, "Superlinearly convergent algorithm for min-max problems," *Journal of Optimization Theory and Applications*, vol. 69, no. 3, pp. 407–439, 1991.

[12] Y. H. Yu and L. Gao, "Nonmonotone line search algorithm for constrained minimax problems," *Journal of Optimization Theory and Applications*, vol. 115, no. 2, pp. 419–446, 2002.

[13] J.-B. Jian, R. Quan, and X.-L. Zhang, "Generalised monotone line search algorithm for degenerate nonlinear minimax problems," *Bulletin of the Australian Mathematical Society*, vol. 73, no. 1, pp. 117–127, 2006.

[14] J.-B. Jian, R. Quan, and Q.-J. Hu, "A new superlinearly convergent SQP algorithm for nonlinear minimax problems," *Acta Mathematicae Applicatae Sinica: English Series*, vol. 23, no. 3, pp. 395–410, 2007.

[15] Q.-J. Hu, Y. Chen, N.-P. Chen, and X.-Q. Li, "A modified SQP algorithm for minimax problems," *Journal of Mathematical Analysis and Applications*, vol. 360, no. 1, pp. 211–222, 2009.

[16] Z. Zhu, X. Cai, and J. Jian, "An improved SQP algorithm for solving minimax problems," *Applied Mathematics Letters*, vol. 22, no. 4, pp. 464–469, 2009.

[17] F. Wang and Y. Wang, "Nonmonotone algorithm for minimax optimization problems," *Applied Mathematics and Computation*, vol. 217, no. 13, pp. 6296–6308, 2011.

[18] F.-S. Wang and C.-L. Wang, "An adaptive nonmonotone trust-region method with curvilinear search for minimax problem," *Applied Mathematics and Computation*, vol. 219, no. 15, pp. 8033–8041, 2013.

[19] J.-B. Jian and M.-T. Chao, "A sequential quadratically constrained quadratic programming method for unconstrained minimax problems," *Journal of Mathematical Analysis and Applications*, vol. 362, no. 1, pp. 34–45, 2010.

[20] G.-D. Ma and J.-B. Jian, "An ε-generalized gradient projection method for nonlinear minimax problems," *Nonlinear Dynamics*, vol. 75, no. 4, pp. 693–700, 2014.

[21] B. Rustem, S. Žaković, and P. Parpas, "An interior point algorithm for continuous minimax: implementation and computation," *Optimization Methods & Software*, vol. 23, no. 6, pp. 911–928, 2008.

[22] B. Rustem, S. Žaković, and P. Parpas, "Convergence of an interior point algorithm for continuous minimax," *Journal of Optimization Theory and Applications*, vol. 136, no. 1, pp. 87–103, 2008.

[23] E. Obasanjo, G. Tzallas-Regas, and B. Rustem, "An interior-point algorithm for nonlinear minimax problems," *Journal of Optimization Theory and Applications*, vol. 144, no. 2, pp. 291–318, 2010.

[24] D. Z. Du, "A gradient projection algorithm for nonlinear constraints," *Acta Mathematicae Applicatae Sinica*, vol. 8, no. 1, pp. 7–16, 1985.

[25] J. B. Jian, D. L. Han, and Q. J. Xu, "A new sequential systems of linear equations algorithm of feasible descent for inequality constrained optimization," *Acta Mathematica Sinica (English Series)*, vol. 26, no. 12, pp. 2399–2420, 2010.

[26] A. Vardi, "New minimax algorithm," *Journal of Optimization Theory and Applications*, vol. 75, no. 3, pp. 613–634, 1992.

[27] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," in *Numerical Analysis*, vol. 630 of *Lecture Notes in Mathematics*, pp. 144–157, Springer, Berlin, Germany, 1978.