Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 159754, 9 pages
http://dx.doi.org/10.1155/2014/159754

Research Article

A Simple SQP Algorithm for Constrained Finite Minimax Problems

Lirong Wang[1] **and Zhijun Luo**[2]

[1] *The Department of Information Science and Engineering, Hunan University of Humanities, Science and Technology, Loudi 417000, China*
[2] *The Department of Mathematics and Econometrics, Hunan University of Humanities, Science and Technology, Loudi 417000, China*

Correspondence should be addressed to Lirong Wang; ldwlr1234@163.com

Received 30 August 2013; Accepted 7 November 2013; Published 10 February 2014

Academic Editors: Z.-C. Deng, K. Skouri, and K.-C. Ying

A simple sequential quadratic programming method is proposed to solve the constrained minimax problem. At each iteration, through introducing an auxiliary variable, the descent direction is given by solving only one quadratic programming. By solving a corresponding quadratic programming, a high-order revised direction is obtained, which can avoid the Maratos effect. Furthermore, under some mild conditions, the global and superlinear convergence of the algorithm is achieved. Finally, some numerical results reported show that the algorithm in this paper is successful.

1. Introduction

Consider the following constrained minimax optimization problems:

$$\begin{aligned} \min_{x\in R^n} \quad & F(x), \\ \text{s.t.} \quad & g_j(x) \le 0, \quad j \in J = \{1,2,\dots,m_1\}, \\ & h_l(x) \le 0, \quad l \in L = \{1,2,\dots,m_2\}, \end{aligned} \tag{1}$$

where $F(x) = \max\{f_i(x) \mid i \in I = \{1,2,\dots,m\}\}$ and $f_i(x), g_j(x), h_l(x) : R^n \to R$ are continuously differentiable.

Minimax problem is one of the most important non-differentiable optimization problems, and it can be widely applied in many fields (such as [1–4]). In real life, a lot of problems can be stated as a minimax problem, such as financial decision making, engineering design, and other fields which wants to obtain the objection functions minimum under conditions of the maximum of the functions. Since the objective function $F(x)$ is non-differentiable, we cannot use the classical methods for smooth optimization problems directly to solve such constrained optimization problems.

Generally speaking, a lot of the schemes have been proposed for solving minimax problems, by converting the problem (1) to a smooth constrained optimization problem as follows

$$\begin{aligned} \min \quad & z, \\ \text{s.t.} \quad & f_i(x) \le z, \quad i \in I, \\ & g_j(x) \le 0, \quad j \in J, \\ & h_l(x) = 0, \quad l \in L. \end{aligned} \tag{2}$$

Obviously, from the problem (2), the KKT conditions of (1) can be stated as follows:

$$\begin{gathered} \sum_{i\in I}\lambda_i \nabla f_i(x) + \sum_{j\in J}\mu_j \nabla g_j(x) + \sum_{j\in L}\nu_j \nabla h_j(x) = 0, \\ \lambda_i \ge 0, \quad f_i(x) - F(x) \le 0, \quad \lambda_i (f_i(x) - F(x)) = 0, \quad i \in I, \\ \mu_j \ge 0, \quad g_j(x) \le 0, \quad \mu_j g_j(x) = 0, \quad j \in J, \\ \sum_{i\in I}\lambda_i = 1, \qquad h_l(x) = 0, \quad l \in L, \end{gathered} \tag{3}$$

where λ_i, μ_j, and ν_j are the corresponding vector. Based on the equivalent relationship between the K-T point of (2) and the stationary point of (1), a lot of methods focus on finding the K-T point of (1), namely, solving (3). And many algorithms have been proposed to solve minimax problem [5–15]. Such as [5–8], the minimax problems are discussed with nonmonotone line search, which can effectively avoid the Maratos effect. Combining the trust-region methods with the line-search methods and curve-search methods, Wang and Zhang [9] propose a hybrid algorithm for linearly constrained minimax problems. Many other effective algorithms for solving the minimax problems are presented, such as [11–15].

Sequential quadratic programming (SQP) method is one of the efficient algorithms for solving smooth constrained optimization problems because of its fast convergence rate. Thus, it is studied deeply and widely (see, e.g., [16–20], etc.). For typical SQP method, the standard search direction d should be obtained by solving the following quadratic programming:

$$\begin{aligned} \min \quad & \nabla F(x)^T d + \frac{1}{2} d^T H d, \\ \text{s.t.} \quad & g_j(x) + \nabla g_j(x)^T d \le 0, \quad j \in J, \\ & h_l(x^k) + \nabla h_l(x^k)^T d = 0, \quad l \in L, \end{aligned} \tag{4}$$

where H is a symmetric positive definite matrix. Since the objective function $F(x)$ contains the max operator, it is continuous but non-differentiable even if every constrained function $f_i(x)$ $(i \in I)$ is differentiable. Therefore this method may fail to reach an optimum for the minimax problem. In view of this and combining with (2), one considers the following quadratic programming through introducing an auxiliary variable z:

$$\begin{aligned} \min \quad & z + \frac{1}{2} d^T H d, \\ \text{s.t.} \quad & f_i(x) + \nabla f_i(x)^T d \le z, \quad i \in I, \\ & g_j(x) + \nabla g_j(x)^T d \le 0, \quad j \in J, \\ & h_l(x^k) + \nabla h_l(x^k)^T d = 0, \quad l \in L. \end{aligned} \tag{5}$$

However, it is well known that the solution d of (5) may not be a feasible descent direction and can not avoid the Maratos effect. Recently, many researches have extended the popular SQP scheme to the minimax problems (see [21–26], etc.). Jian et al. [22] and Q.-J. Hu and J.-Z. Hu [23] process pivoting operation to generate an ε-active constraint subset associated with the current iteration point. At each iteration of their proposed algorithm, a main search direction is obtained by solving a reduced quadratic program which always has a solution.

The feasible direction method (MFD) (see [27, 28], etc.) is another effective way for solving smooth constrained optimization problems. An advantage of MFD over the classical SQP method is that a feasible direction of descent can be obtained by solving only one quadratic programming. In this paper, to obtain a feasible direction of descent and reduce the computational cost, we construct a new quadratic programming subproblem. Suppose x^k is the current iteration point; at each iteration, the descent direction d^k is obtained by solving the following quadratic programming subproblem:

$$\begin{aligned} \min \quad & z + \frac{1}{2} d^T H_k d, \\ \text{s.t.} \quad & f_i(x^k) + \nabla f_i(x^k)^T d - F(x^k) \le z, \quad i \in I, \\ & g_j(x^k) + \nabla g_j(x^k)^T d \le \eta_k z, \quad j \in J, \\ & h_l(x^k) + \nabla h_l(x^k)^T d = 0, \quad l \in L, \end{aligned} \tag{6}$$

where H_k is a symmetric positive definite matrix and η_k is nonnegative auxiliary variable. In order to avoid the Maratos effect, a height-order correction direction is computed by the corresponding quadratic programming:

$$\begin{aligned} \min \quad & z + \frac{1}{2} (d^k + d)^T H_k (d^k + d), \\ \text{s.t.} \quad & f_i(x^k + d^k) + \nabla f_i(x^k + d^k)^T d \\ & \quad - F(x^k + d^k) \le z, \quad i \in I, \\ & g_j(x^k + d^k) + \nabla g_j(x^k + d^k)^T d \le \eta_k z, \quad j \in J, \\ & h_l(x^k + d^k) + \nabla h_l(x^k + d^k)^T d = 0, \quad l \in L. \end{aligned} \tag{7}$$

Under suitable conditions, the theoretical analysis shows that the convergence of our algorithm can be obtained.

The plan of the paper is as follow. The algorithm is proposed in Section 2. In Section 3, we show that the algorithm is globally convergent, while the superlinear convergence rate is analyzed in Section 4. Finally, some preliminary numerical tests are reported in Section 5.

2. Description of the Algorithm

Now we state our algorithm as follows.

Algorithm 1.

Step 0. Given initial point $x^0 \in R^n$, define a symmetric positive definite matrix $H_0 \in R^{n\times n}$. Choose parameters $\alpha \in (0, 1/2)$, $\eta_0 > 0$, and $\gamma \in (0, 1)$. Set $k = 0$.

Step 1. Compute (d^k, z_k) by the quadratic problem (6) at x^k. Let $(\lambda^k, \mu^k, \nu^k)$ be the corresponding KKT multipliers vector. If $d^k = 0$, then STOP.

Step 2. Compute $(\tilde{d}^k, \tilde{z}_k)$ by the quadratic problem (7).

Set $(\tilde{\lambda}^k, \tilde{\mu}^k, \tilde{\nu}^k)$ as the corresponding KKT multipliers vector. If $\|\tilde{d}^k\| > \|d^k\|$, set $\tilde{d}^k = 0$.

Step 3 (the line search). A merit function is defined as follows:

$$w(x) = F(x) + r\varphi(x) + r\sum_{l\in L} |h_l(x)|, \tag{8}$$

where $\varphi(x) = \max\{g_j(x), j \in J; 0\}$ and r is a suitable large positive scalar.

Compute t_k, the first number t in the sequence $\{1, 1/2, 1/4, 1/8, \ldots\}$ satisfying

$$w\left(x^k + td^k + t^2\tilde{d}^k\right) \le w\left(x^k\right) - \alpha t\left(d^k\right)^T H_k d^k. \quad (9)$$

Step 4 (update). Obtain H_{k+1} by updating the positive definite matrix H_k using some quasi-Newton formulas. Set $x^{k+1} = x^k + td^k + t^2\tilde{d}^k$, $\eta_{k+1} = \min\{\eta_0, \|d^k\|^\gamma\}$. Set $k := k+1$. Go back to Step 1.

3. Global Convergence of the Algorithm

For convenience, we denote

$$\begin{gathered} I(x) = \{i \in I \mid f_i(x) = F(x)\}, \\ J(x) = \{j \in J \mid g_j(x) = \varphi(x)\}. \end{gathered} \quad (10)$$

In this section, we analyze the convergence of the algorithm. The following general assumptions are true throughout this paper.

(H 3.1) The functions $f_i(x)$, $i \in I$, $g_j(x)$, $j \in J$, and $h_l(x)$, $l \in L$, are continuously differentiable.

(H 3.2) $\forall x \in R^n$; the set of vectors

$$\begin{gathered} \left\{\begin{pmatrix} -1 \\ \nabla f_i(x) \end{pmatrix}, i \in I(x); \begin{pmatrix} 0 \\ \nabla g_j(x) \end{pmatrix}, \ j \in J(x); \right. \\ \left. \begin{pmatrix} 0 \\ \nabla h_l(x) \end{pmatrix}, l \in L \right\} \end{gathered} \quad (11)$$

is linearly independent.

(H 3.3) There exist $a, b > 0$, such that $a\|d\|^2 \le d^T H_k d \le b\|d\|^2$, for all $k \in R$ and $d \in R^n$.

Lemma 2. *Suppose that (H 3.1)–(H 3.3) hold, matrix H_k is symmetric positive definite, and (d^k, z_k) is an optimal solution of* (6). *Then*

(1) $z_k + (1/2)(d^k)^T H_k d^k \le 0$, $z_k \le 0$,

(2) if $d^k = 0$, then x^k is a K-T point of problem (1).

Proof. (1) For $(0, 0) \in R^{n+1}$ is a feasible solution of (6) and H_k is positive definite, one has

$$z_k + \frac{1}{2}\left(d^k\right)^T H_k d^k \le 0, \qquad z_k \le -\frac{1}{2}\left(d^k\right)^T H_k d^k \le 0. \quad (12)$$

Further, if $d^k \neq 0$, then $z_k < 0$.

(2) Firstly, we prove $d^k = 0 \Leftrightarrow z_k = 0$. If $z_k = 0$, then $(1/2)(d^k)^T H_k d^k = (1/2)(d^k)^T H_k d^k + z_k \le 0$. For the positive definite property of H_k, it has $d^k = 0$. On the other hand, if $d^k = 0$, in view of the constraints

$$f_i\left(x^k\right) + \nabla f_i\left(x^k\right)^T d^k - F\left(x^k\right) \le z_k, \quad i \in I\left(x^k\right), \quad (13)$$

we have $z_k \ge 0$. Combining $z_k \le 0$, we have $z_k = 0$.

Secondly, we show that x^k is a K-T point of problem (1) when $d^k = 0$. From the problem (6), the K-T condition at x^k is defined as follows:

$$\begin{gathered} H_k d^k + \sum_{i \in I} \lambda_i^k \nabla f_i\left(x^k\right) + \sum_{j \in J} \mu_j^k \nabla g_j\left(x^k\right) + \sum_{l \in L} \nu_l^k \nabla h_l\left(x^k\right) = 0, \\ \sum_{i \in I} \lambda_i^k + \eta_k \sum_{j \in J} \mu_j^k = 1, \\ \lambda_i \ge 0, \\ 0 \le \lambda_i^k \perp \left(f_i\left(x^k\right) + \nabla f_i\left(x^k\right)^T d^k - F\left(x^k\right) - z_k\right) \le 0, \\ i \in I, \\ \mu_j \ge 0, \\ 0 \le \mu_j^k \perp \left(g_j\left(x^k\right) + \nabla g_j\left(x^k\right)^T d^k - \eta_k z_k\right) \le 0, \\ j \in J, \\ h_l\left(x^k\right) + \nabla h_l\left(x^k\right)^T d^k = 0, \quad l \in L. \end{gathered} \quad (14)$$

If $d^k = 0$, then $z_k = 0$, and according to the definition of η_k in Step 4, we have $\eta_k = 0$. Furthermore, it holds that

$$\begin{gathered} \sum_{i \in I} \lambda_i \nabla f_i(x) + \sum_{j \in J} \mu_j \nabla g_j(x) = 0, \\ \sum_{i \in I} \lambda_i = 1, \\ \lambda_i \ge 0, \quad f_i(x) - F(x) \le 0, \quad \lambda_i\left(f_i(x) - F(x)\right) = 0, \\ i \in I, \\ \mu_j \ge 0, \quad g_j(x) \le 0, \quad \mu_j g_j(x) = 0, \quad j \in J, \\ h_l\left(x^k\right) = 0, \quad l \in L. \end{gathered} \quad (15)$$

That is to see that the results hold. □

From Lemma 2, it is obvious, if $d^k \neq 0$, that the line search in Step 3 yields is always completed.

Lemma 3. *If $d^k \neq 0$ and if r satisfies $r \ge \|\mu\|_\infty$ and $r \ge \|\nu\|_\infty$, the line search in Step 3 of the algorithm is well defined.*

Proof. Firstly, we consider the functions $f_i(x + td^k + t^2\tilde{d})$, $i \in I$, $g_j(x + td + t^2\tilde{d})$, $j \in J$, and $h_l(x + td + t^2\tilde{d})$, $l \in L$, of the Taylor expansion at x. Then, we obtain

$$w\left(x + td + t^2\tilde{d}\right) = \widetilde{w}(x; td) + o(t), \quad (16)$$

where

$$\begin{aligned}\widetilde{w}(x;td) = t\Bigg\{&\max_{i\in I}\left\{f_i(x)+\nabla f_i(x)^T d\right\}\\&+r\sum_{j\in J}\max\left\{g_j(x)+\nabla g_j(x)^T d,0\right\}\\&+r\sum_{l\in L}\left|h_l(x)+\nabla h_l(x)^T d\right|\Bigg\}.\end{aligned}\tag{17}$$

$\widetilde{w}(x;d)$ is convex as a function of d, and thus we have

$$\widetilde{w}(x;td)-w(x)\le t\{\widetilde{w}(x;d)-w(x)\},\quad \forall t\in[0,1].\tag{18}$$

From the definition of $w(x)$, $\widetilde{w}(x;d)$ and (1), it is easy to obtain

$$\begin{aligned}\widetilde{w}(x;d)-w(x)={}&\max_{i\in I}\left\{f_i(x)+\nabla f_i(x)^T d\right\}\\&-\max_{i\in I}f_i(x)-r\sum_{j\in J}\max\left\{g_j(x),0\right\}\\&-r\sum_{l\in L}\left|h_l(x)\right|.\end{aligned}\tag{19}$$

On the other hand, from the first equation of (14) we can get

$$\begin{aligned}&d^T Hd+\sum_{i\in I}\lambda_i\nabla f_i(x)^T d+\sum_{j\in J}\mu_j\nabla g_j(x)^T d\\&\quad+\sum_{l\in L}\nu_l\nabla h_l(x)^T d=0.\end{aligned}\tag{20}$$

Let $I_f(x;d)=\{i;f_i(x)+\nabla f_i(x)^T d=\max_{l\in I}\{f_l(x)+\nabla f_l(x)^T d\},\ i\in\{1,2,\ldots,m\}\}$. Since the third formula of (14) implies

$$\lambda_i=0,\quad \forall i\in\{1,2,\ldots,m\}\setminus I_f(x;d),\tag{21}$$

then

$$\begin{aligned}&d^T Hd+\sum_{i\in I_f(x;d)}\lambda_i\nabla f_i(x)^T d\\&\quad+\sum_{j\in J}\mu_j\nabla g_j(x)^T d+\sum_{l\in L}\nu_l\nabla h_l(x)^T d=0.\end{aligned}\tag{22}$$

For $i\in I_f(x;d)$, we get

$$\begin{aligned}&\nabla f_i(x)^T d=\max_{l\in I}\left\{f_l(x)+\nabla f_l(x)^T d\right\}-f_i(x),\\&\sum_{i\in I}\lambda_i=1\Longrightarrow\sum_{i\in I_f(x;d)}\lambda_i=1.\end{aligned}\tag{23}$$

Thus, (22) implies

$$\begin{aligned}&\max_{i\in I}\left\{f_i(x)+\nabla f_i(x)^T d\right\}\\&\quad=\sum_{i\in I_f(x;d)}\lambda_i f_i(x)-d^T Hd\\&\qquad-\sum_{j\in J}\mu_j\nabla g_j(x)^T d-\sum_{l\in L}\nu_l\nabla h_l(x)^T d.\end{aligned}\tag{24}$$

Substituting the above equality in (19), we can obtain

$$\begin{aligned}&\widetilde{w}(x;d)-w(x)\\&\quad\le-\Bigg\{d^T Hd+\sum_{j\in J}\mu_j\nabla g_j(x)^T d\\&\qquad+r\sum_{j\in J}\max\left\{g_j(x);0\right\}\\&\qquad+\sum_{l\in L}\nu_l\nabla h_l(x)^T d+r\sum_{l\in L}\left|h_l(x)\right|\Bigg\}.\end{aligned}\tag{25}$$

It follows from (14) that

$$\begin{aligned}&\widetilde{w}(x;td)-w(x)\\&\quad\le-t\Bigg\{d^T Hd-\sum_{j\in J}\mu_j g_j(x)\\&\qquad+r\sum_{j\in J}\max\left\{g_j(x);0\right\}\\&\qquad-\sum_{l\in L}\nu_l h_l(x)+r\sum_{l\in L}\left|h_l(x)\right|\Bigg\}.\end{aligned}\tag{26}$$

Considering r satisfies $r\ge\|\mu\|_\infty$ and $r\ge\|\nu\|_\infty$, then we have

$$\widetilde{w}(x;td)-w(x)\le-td^T Hd<0,\quad\forall t\in[0,1].\tag{27}$$

Then, at x^k, we have

$$w\left(x^k+td^k+t^2\tilde{d}^k\right)-w\left(x^k\right)=\widetilde{w}\left(x^k;td^k\right)-w\left(x^k\right)+o(t).\tag{28}$$

Since $\alpha\in(0,1/2)$, for t small enough, it holds that

$$\begin{aligned}&w\left(x^k+td^k+t^2\tilde{d}^k\right)-w\left(x^k\right)\\&\quad\le\alpha\left(\widetilde{w}\left(x^k;td^k\right)-w\left(x^k\right)\right)\le-\alpha td^{kT}H_k d^k.\end{aligned}\tag{29}$$

That is, the line search condition (9) is satisfied. □

In the following of this section, we will show the global convergence of the algorithm. Since $\{d^k,z_k,\lambda^k,\mu^k\}$ is bounded under all the above-mentioned assumptions, we can assume without loss of generality that there exist an infinite index set K and a constant η^* such that

$$\begin{aligned}&x^k\longrightarrow x^*,\quad H_k\longrightarrow H_*,\quad \eta_k\longrightarrow\eta_*,\quad d^k\longrightarrow d^*,\\&z_k\longrightarrow z_*,\quad \lambda^k\longrightarrow\lambda^*,\quad \mu^k\longrightarrow\mu^*,\quad k\in K.\end{aligned}\tag{30}$$

Theorem 4. *The algorithm either stops at the KKT point x^k of the problem* (1) *in finite number of steps or generates an infinite sequence $\{x^k\}$ any accumulation point x^* of which is a KKT point of the problem* (1).

Proof. The first statement is obvious, the only stopping point being in Step 1. Thus, assume that the algorithm generates an infinite sequence $\{x^k\}$ and (30) holds. The cases $\eta_* = 0$ and $\eta_* > 0$ are considered separately.

Case A ($\eta_* = 0$). By Step 4, there exists an infinite index set $K_1 \subseteq K$, such that $d^{k-1} \to 0, k \in K_1$, while, by Step 3, it holds that

$$\begin{aligned}\lim_{k\in K_1} \left\|x^k - x^{k-1}\right\| &= \lim_{k\in K_1} \left\|t_{k-1}d^{k-1} + t_{k-1}^2\tilde{d}^{k-1}\right\| \\ &\le \lim_{k\in K_1} \left(\left\|d^{k-1}\right\| + \left\|\tilde{d}^{k-1}\right\|\right) = 0.\end{aligned} \tag{31}$$

So, the fact that $x^{k-1} \xrightarrow{k\in K_1} x^*$ implies that $d^{k-1} \xrightarrow{k\in K_1} 0$. So, from Lemma 2, it is clear that x^* is a K-T point of (1).

Case B ($\eta_* > 0$). Obviously, it only needs to prove that $d^k \to 0, k \in K$. Suppose by contradiction that $d^* \neq 0$. Since

$$\begin{aligned} &f_i\left(x^k\right) + \nabla f_i\left(x^k\right)^T d^k - F\left(x^k\right) \le z_k, \quad i \in I, \\ &g_j\left(x^k\right) + \nabla g_j\left(x^k\right)^T d^k \le \eta_k z_k, \quad j \in J, \\ &h_l\left(x^k\right) + \nabla h_l\left(x^k\right)^T d = 0, \quad l \in L, \end{aligned} \tag{32}$$

in view of $k \in K, k \to \infty$, we have

$$\begin{aligned} &f_i\left(x^*\right) + \nabla f_i\left(x^*\right)^T d^* - F\left(x^*\right) \le z_*, \quad i \in I, \\ &g_j\left(x^*\right) + \nabla g_j\left(x^*\right)^T d^* \le \eta_* z_*, \quad j \in J, \\ &h_l\left(x^*\right) + \nabla h_l\left(x^*\right)^T d = 0, \quad l \in L. \end{aligned} \tag{33}$$

So, the following corresponding QP subproblem (6) at x^*

$$\begin{aligned} \min_{(d,z)\in R^{n+1}} \quad & z + \frac{1}{2}d^T H_* d, \\ \text{s.t.} \quad & f_i\left(x^*\right) + \nabla f_i\left(x^*\right)^T d - F\left(x^*\right) \le z, \\ & \qquad i \in I, \\ & g_j\left(x^*\right) + \nabla g_j\left(x^*\right)^T d \le \eta_* z, \quad j \in J, \\ & h_l\left(x^*\right) + \nabla h_l\left(x^*\right)^T d = 0, \quad l \in L, \end{aligned} \tag{34}$$

has a nonempty feasible set. Moreover, it is not difficult to show that (z_*, d^*) is the unique solution of (34). So, it holds that

$$\begin{aligned} &z_* < 0, \quad \nabla f_i\left(x^*\right)^T d^* \le z_* < 0, \quad i \in I\left(x^*\right), \\ &\nabla g_j\left(x^*\right)^T d^* \le \eta_* z_* < 0, \quad j \in J\left(x^*\right). \end{aligned} \tag{35}$$

For $x^k \to x^*, d^k \to d^*, k \in K$, it is clear, for $k \in K$, k large enough, that

$$\begin{aligned} &\nabla f_i\left(x^k\right)^T d^k \le \frac{1}{2}\nabla f_i\left(x^*\right)^T d^* < 0, \quad i \in I\left(x^*\right), \\ &\nabla g_j\left(x^k\right)^T d^k \le \frac{1}{2}\nabla g_j\left(x^*\right)^T d^* < 0, \quad j \in J\left(x^*\right). \end{aligned} \tag{36}$$

From (36), by imitating the proof of [17, Proposition 3.2], we know that the stepsize t_k obtained by the line search is bounded away from zero on K; that is,

$$t_k \ge t_* = \inf\left\{t_k, k \in K\right\} > 0. \tag{37}$$

In addition, from (9) and Lemma 2, it follows that $\{f(x^k)\}$ is monotonous decreasing. So, considering $\{x^k\}_K \to x^*$ and the hypothesis (H 3.1), one holds that

$$f_i\left(x^k\right) \longrightarrow f_i\left(x^*\right), \quad k \in K, \ i \in I\left(x^*\right). \tag{38}$$

Hence, from (9) and (36)–(38), we get

$$\begin{aligned} 0 &= \lim_{k\in K}\left(f_i\left(x^{k+1}\right) - f_i\left(x^k\right)\right) \\ &\le \lim_{k\in K} \alpha t_k \nabla f_i\left(x^k\right)^T d^k \le \frac{1}{2}\alpha t_* \nabla f_i\left(x^*\right)^T d^* < 0. \end{aligned} \tag{39}$$

It is a contradiction. So, $d^* = 0$. Thereby, according to Lemma 2, x^* is a KKT point of problem (1). □

4. Rate of Convergence

In this section, we show the convergence rate of the algorithm. For this purpose, we add the following some stronger regularity assumptions.

(H 4.1) The functions $f_i(x)$ $(i \in I)$, $g_j(x)$ $(j \in J)$, and $h_l(x)$ $(l \in L)$ are twice continuously differentiable.

(H 4.2) The sequence x^k generated by the algorithm possesses an accumulation point x^*, and $H_k \to H_*, k \to \infty$.

(H 4.3) The second-order sufficiency conditions with strict complementary slackness are satisfied at the KKT point x^*; that is, it holds that $\lambda_i > 0, i \in I(x^*)$, $\mu_j > 0$, $j \in J(x^*)$, and

$$d^T \nabla_{xx}^2 L\left(x^*, \lambda^*, \mu^*, \nu^*\right) d > 0, \quad 0 \neq d \in S^*, \tag{40}$$

where

$$\begin{aligned} &\nabla_{xx}^2 L\left(x^*, \lambda^*, \mu^*, \nu^*\right) \\ &\quad = \sum_{i\in I} \lambda_i^* \nabla^2 f_i\left(x^*\right) + \sum_{j\in J} \mu_j^* \nabla^2 g_j\left(x^*\right) \\ &\qquad + \sum_{l\in L} \nu_l^* \nabla^2 h_l\left(x^*\right) \\ &\quad = \sum_{i\in I(x^*)} \lambda_i^* \nabla^2 f_i\left(x^*\right) + \sum_{j\in J(x^*)} \mu_j^* \nabla^2 g_j\left(x^*\right) \\ &\qquad + \sum_{l\in L} \nu_l^* \nabla^2 h_l\left(x^*\right), \\ &S^* = \Big\{d \in R^n \mid \nabla f_i\left(x^*\right)^T d = \nabla f_{i_k}\left(x^*\right)^T d, \\ &\qquad \forall i \in I\left(x^*\right), i_k \in I\left(x^*\right), \\ &\qquad \nabla g_j\left(x^*\right)^T d = 0, \\ &\qquad \forall j \in J\left(x^*\right), \nabla h_l\left(x^*\right)^T d = 0, \forall i \in I\left(x^*\right)\Big\}. \end{aligned} \tag{41}$$

According to the all stated assumptions (H 4.1)–(H 4.3) and [21, Theorem 2], we have the following results.

Lemma 5. *The KKT point x^* of problem* (1) *is isolated.*

Lemma 6. *The entire sequence $\{x^k\}$ converges to x^*; that is, $x^k \to x^*, k \to \infty$.*

Proof. The result of this lemma is similar to the proof of [19, Lemma 4.1]. □

Lemma 7. *For k large enough, it holds that*

(1) $d^k \to 0$ and $z_k \to 0$,

(2) $\lambda^k \to \lambda^*$, $\mu^k \to \mu^*$, and $\nu^k \to \nu^*$.

Lemma 8. *For k large enough, $\widetilde{d}^k$ obtained by Step 2 satisfies*

(1)

$$\left\|\widetilde{d}^k\right\| = O\left(\left\|d^k\right\|^2\right), \tag{42}$$

(2)

$$\begin{aligned} h_l\left(x^k + d^k + \widetilde{d}^k\right) &= O\left(\left\|d^k\right\|^3\right), \quad \forall l \in L, \\ g_j\left(x^k + d^k + \widetilde{d}^k\right) &= O\left(\left\|d^k\right\|^3\right), \quad \forall j \in J(x). \end{aligned} \tag{43}$$

Proof. (1) The result can be proven similarly to the proof of [5, Proposition 3.1] or [19, Lemma 4.3].

(2) We have

$$\begin{aligned} &h_l\left(x^k + d^k + \widetilde{d}^k\right) \\ &\quad = h_l\left(x^k + d^k\right) + \nabla h_l\left(x^k + d^k\right)^T \widetilde{d}^k + O\left(\left\|\widetilde{d}^k\right\|^2\right) \\ &\quad = h_l\left(x^k + d^k\right) + \left(\nabla h_l\left(x^k\right) + O\left\|d^k\right\|\right)^T \widetilde{d}^k + O\left(\left\|\widetilde{d}^k\right\|^2\right) \\ &\quad = h_l\left(x^k + d^k\right) + \nabla h_l\left(x^k\right)^T \widetilde{d}^k + O\left(\left\|d^k\right\|^3\right) \\ &\quad = O\left(\left\|d^k\right\|^3\right), \quad l \in L. \end{aligned} \tag{44}$$

Analogously, the other result is not difficult to be shown. □

To get the superlinearly convergent rate of the above proposed algorithm, the following additional assumption is necessary.

(H 4.4) The matrix sequence H_k satisfies that

$$\left\|P_k\left(H_k - \nabla^2_{xx} L\left(x^k, \lambda^k, \mu^k, \nu^k\right)\right) d^k\right\| = o\left(\left\|d^k\right\|\right), \tag{45}$$

where

$$\begin{aligned} P_k &= I_n - A_k\left(A_k^T A_k\right)^{-1} A_k^T, \\ A_k &= A_k\left(x^k\right) = \left(\left(\nabla f_i\left(x^k\right) - \nabla f_{i_k}\left(x^k\right)\right),\right. \\ &\qquad \left.\nabla g_j\left(x^k\right), \nabla h_l\left(x^k\right)\right), \\ &\qquad \left(i \in I\left(x^k\right) \setminus \{i_k\}, j \in J\left(x^k\right), l \in L\left(x^k\right)\right). \end{aligned} \tag{46}$$

According to Lemmas 6 and 8, it is easy to know

$$\begin{aligned} &\left\|P_k\left(H_k - \nabla^2_{xx} L\left(x^k, \lambda^k, \mu^k, \nu^k\right)\right) d^k\right\| \\ &\quad = o\left(\left\|d^k\right\|\right) \Longleftrightarrow \left\|P_k\left(H_k - \nabla^2_{xx} L\left(x^*, \lambda^*, \mu^*, \nu^*\right)\right) d^k\right\| \\ &\quad = o\left(\left\|d^k\right\|\right). \end{aligned} \tag{47}$$

Lemma 9. *For k large enough, under the above-mentioned assumptions, $t_k \equiv 1$.*

Proof. It is only necessary to prove that

$$w\left(x^k + d^k + \widetilde{d}^k\right) \le w\left(x^k\right) - \alpha d^{kT} H_k d^k. \tag{48}$$

From (6) and (14), we have

$$\begin{aligned} f_i\left(x^k + d^k\right) &= f_i\left(x^k\right) + \nabla f_i\left(x^k\right)^T d^k + O\left(\left\|d^k\right\|^2\right) \\ &= f\left(x^k\right) + z_k + O\left(\left\|d^k\right\|^2\right), \quad i \in I\left(X^*\right), \\ f_j\left(x^k + d^k\right) &= f_j\left(x^k\right) + \nabla f_j\left(x^k\right)^T d^k + O\left(\left\|d^k\right\|^2\right) \\ &= f\left(x^k\right) + z_k + O\left(\left\|d^k\right\|^2\right), \quad j \in I\left(X^*\right). \end{aligned} \tag{49}$$

Hence,

$$f_i\left(x^k + d^k\right) = f_j\left(x^k + d^k\right) + O\left(\left\|d^k\right\|^2\right), \quad \forall i, j \in I\left(x^*\right). \tag{50}$$

Similarly, together with $\|\widetilde{d}^k\| = O(\|d^k\|^2)$, it is easy to get

$$\begin{aligned} &f_i\left(x^k + d^k + \widetilde{d}^k\right) \\ &\quad = f_j\left(x^k + d^k + \widetilde{d}^k\right) + O\left(\left\|d^k\right\|^3\right), \quad \forall i, j \in I\left(x^*\right). \end{aligned} \tag{51}$$

On the other hand, the facts that $d^k \to 0$ and $\widetilde{d}^k \to 0$ imply that $I(x^k + d^k + \widetilde{d}^k) \subseteq I(x^*)$ (k large enough). Thus, for $j_k \in I(x^k + d^k + \widetilde{d}^k) \subseteq I(x^*)$, we have

$$\begin{aligned} F\left(x^k + d^k + \widetilde{d}^k\right) &= \max_{l \in I}\left\{f_l\left(x^k + d^k + \widetilde{d}^k\right)\right\} \\ &= f_{j_k}\left(x^k + d^k + \widetilde{d}^k\right) = f_j\left(x^k + d^k + \widetilde{d}^k\right) \\ &\quad + O\left(\left\|d^k\right\|^3\right), \quad \forall j \in I\left(x^*\right). \end{aligned} \tag{52}$$

By the definition of $w(x)$ and Lemma 8, we have

$$w\left(x^k+d^k+\tilde{d}^k\right)=\max_{l\in I}\left\{f_l\left(x^k+d^k+\tilde{d}^k\right)\right\}+O\left(\left\|d^k\right\|^3\right). \tag{53}$$

Multiplying both sides of (52) by λ_i^k and adding them, combining $\sum_{l\in I(x^*)}\lambda_i^k=1$ with (53), we get

$$w\left(x^k+d^k+\tilde{d}^k\right)=\max_{i\in I(x^*)}\left\{f_i\left(x^k+d^k+\tilde{d}^k\right)\right\}+O\left(\left\|d^k\right\|^3\right). \tag{54}$$

In addition, for k large enough, we have

$$\begin{aligned}
&L\left(x^k+d^k+\tilde{d}^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)\\
&\quad=\sum_{i\in I}\lambda_i^k f_i\left(x^k+d^k+\tilde{d}^k\right)\\
&\qquad+\sum_{j\in J}\mu_j^k g_j\left(x^k+d^k+\tilde{d}^k\right)\\
&\qquad+\sum_{l\in L}\nu_l^k h_l\left(x^k+d^k+\tilde{d}^k\right)\\
&\quad=\sum_{i\in I(x^*)}\lambda_i^k f_i\left(x^k+d^k+\tilde{d}^k\right)\\
&\qquad+\sum_{j\in J_g(x^*)}\mu_j^k g_j\left(x^k+d^k+\tilde{d}^k\right)\\
&\qquad+\sum_{l\in L}\nu_l^k h_l\left(x^k+d^k+\tilde{d}^k\right)\\
&\quad=\sum_{i\in I(x^*)}\lambda_i^k f_i\left(x^k+d^k+\tilde{d}^k\right)+O\left(\left\|d^k\right\|^3\right).
\end{aligned} \tag{55}$$

Combining the above equation with (54) we can obtain

$$\begin{aligned}
&w\left(x^k+d^k+\tilde{d}^k\right)\\
&\quad=L\left(x^k+d^k+\tilde{d}^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)+O\left(\left\|d^k\right\|^3\right)\\
&\quad=L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)+\nabla_x L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)^T\left(d^k+\tilde{d}^k\right)\\
&\qquad+\frac{1}{2}\left(d^k+\tilde{d}^k\right)^T\nabla_{xx}^2L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)\\
&\qquad\times\left(d^k+\tilde{d}^k\right)+o\left(\left\|d^k\right\|^2\right)\\
&\quad=L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)+\nabla_x L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)^T\left(d^k+\tilde{d}^k\right)\\
&\qquad+\frac{1}{2}\left(d^k\right)^T\nabla_{xx}^2L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)\left(d^k\right)+o\left(\left\|d^k\right\|^2\right).
\end{aligned} \tag{56}$$

From the KKT condition (14) implies $\nabla_x L(x^k,\lambda_i^k,\mu_j^k,\nu_l^k)=-H_kd^k$; then we get

$$\begin{aligned}
&\nabla_x L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)^T\left(d^k+\tilde{d}^k\right)\\
&\quad=-d^{kT}H_kd^k+o\left(\left\|d^k\right\|^2\right),\qquad\sum_{i\in I(x^*)}\lambda_i^k=1.
\end{aligned} \tag{57}$$

Thus,

$$\begin{aligned}
&w\left(x^k+d^k+\tilde{d}^k\right)\\
&\quad=L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)-d^{kT}H_kd^k\\
&\qquad+\frac{1}{2}\left(d^k\right)^T\nabla_{xx}^2L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)\\
&\qquad\times\left(d^k\right)+o\left(\left\|d^k\right\|^2\right)\\
&\quad=w\left(x^k\right)-\alpha d^{kT}H_kd^k+\frac{1}{2}\left(d^k\right)^T\\
&\qquad\times\left(\nabla_{xx}^2L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)-H_k\right)\left(d^k\right)\\
&\qquad-\frac{1}{2}d^{kT}H_kd^k+\alpha d^{kT}H_kd^k\\
&\qquad+L\left(x^k,\lambda_i^k,\mu_j^k,\nu_l^k\right)-w\left(x^k\right)+o\left(\left\|d^k\right\|^2\right)\\
&\quad\le w\left(x^k\right)-\alpha d^{kT}H_kd^k\\
&\qquad-\left(\frac{1}{2}-\alpha\right)d^{kT}H_kd^k+o\left(\left\|d^k\right\|^2\right).
\end{aligned} \tag{58}$$

For k large enough, according to $\alpha\in(0,1/2)$, it holds that

$$w\left(x^k+d^k+\tilde{d}^k\right)\le w\left(x^k\right)-\alpha d^{kT}H_kd^k. \tag{59}$$

□

From Lemma 9 and the method of [29, Theorem 5.2], we can get the following.

Theorem 10. *Under all stated assumptions, the algorithm is superlinearly convergent; that is, the sequence* $\{x^k\}$ *generated by the algorithm satisfies* $\|x^{k+1}-x^*\|=o(\|x^k-x^*\|)$.

5. Numerical Experiments

In this section, we select several problems to show the efficiency of the algorithm in Section 2. Some preliminary numerical experiments are tested on an Intel(R) Celeron(R) CPU 2.40 GHz computer. The code of the proposed algorithm is written by using MATLAB 7.0 and utilized the optimization toolbox to solve the quadratic programmings (6) and (7). The results show that the proposed algorithm is efficient.

During the numerical experiments, are chosen at random some parameters as follows: $\alpha=0.25$, $\eta_0=1$, $\gamma=0.5$, and $H_0=I$, the $n\times n$ unit matrix. H_k is updated by the BFGS

Table 1: Numerical results of Algorithm 1.

Number	n, m, m_1, m_2	NT	IP	FV
1 ([10, Problem 1])	2, 3, 0, 0	11	$(1, 5)^T$	1.952224
2 ([10, Problem 4])	2, 3, 0, 0	10	$(3, 1)^T$	0.616234
3 ([11, Problem 1])	2, 3, 2, 0	7	$(0, 0)^T$	1.952224
4 ([11, Problem 2])	2, 6, 2, 0	12	$(1, 3)^T$	0.616432
5 ([11, Problem 4])	2, 3, 2, 0	10	$(4, 2)^T$	2.250000
6 ([11, Problem 5])	4, 4, 3, 0	32	$(0, 1, 1, 0)^T$	−44.000000
7 ([11, Problem 6])	2, 3, 2, 0	4	$(0, 1)^T$	2.000000
8 ([26, Problem 5])	2, 2, 0, 1	12	$(0, 4)^T$	−5.875407
9 ([26, Problem 6])	3, 3, 1, 2	5	$(2, 3, 2)^T$	−3.934502
10 ([26, Problem 7])	10, 8, 0, 3	54	$(0, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0)^T$	$2.3339e + 003$

formula [16]. In the implementation, the stopping criterion of Step 1 is changed to If $\|d_0^k\| \le 10^{-6}$, STOP.

This algorithm has been tested on some problems from [10, 11, 26]. The results are summarized in Table 1. The columns of this table have the following meanings:

Number: the number of the test problem in [10, 11] or [26];

n: the dimension of the problem;

m: the number of objective functions;

m_1: the number of inequality constraints;

m_2: the number of equality constraints;

NT: the number of iterations;

IP: the initial point;

FV: the final value of the objective function.

6. Concluding Remarks

In this paper, we propose a simple feasible sequential quadratic programming algorithm for inequality constrained minimax problems. With the help of the technique of method of feasible direction, at each iteration, a main search direction is obtained by solving only one reduced quadratic programming subproblem. Then, a correction direction is yielded by solving another quadratic programming to avoid Maratos effect and guarantee the superlinear convergence under mild conditions. The preliminary numerical results also show that the proposed algorithm is effective.

As further work, we can get the main search direction by other techniques, for example, sequential systems of linear equations technique. And we can also consider removing the strict complementarity.

Conflict of Interests

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

Acknowledgments

The authors would like to thank the anonymous referee for the careful reading and helpful comments and suggestions that led to an improved version of this paper. Project was supported by the Foundation of Hunan Provincial Education Department under Grant nos. 12A077 and 13C453 and Scientific Research Fund of Hunan University of Humanities, Science and Technology of China (no. 2012QN04).

References

[1] X. Cai, K. L. Teo, X. Yang, and X. Y. Zhou, "Portfolio optimization under a minimax rule," *Management Science*, vol. 46, no. 7, pp. 957–972, 2000.

[2] A. R. Pankov, E. N. Platonov, and K. V. Semenikhin, "Minimax quadratic optimization and its application to investment planning," *Automation and Remote Control*, vol. 62, no. 12, pp. 1978–1995, 2001.

[3] A. Baums, "Minimax method in optimizing energy consumption in real-time embedded systems," *Automatic Control and Computer Sciences*, vol. 43, no. 2, pp. 57–62, 2009.

[4] E. Y. Rapoport, "Minimax optimization of stationary states in systems with distributed parameters," *Journal of Computer and Systems Sciences International*, vol. 52, no. 2, pp. 165–179, 2013.

[5] J. L. Zhou and A. L. Tits, "Nonmonotone line search for minimax problems," *Journal of Optimization Theory and Applications*, vol. 76, no. 3, pp. 455–476, 1993.

[6] L. Grippo, F. Lampariello, and S. Lucidi, "Nonmonotone line search technique for newton's method," *SIAM Journal on Numerical Analysis*, vol. 23, no. 4, pp. 707–716, 1986.

[7] 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.

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

[9] F. Wang and K. Zhang, "A hybrid algorithm for nonlinear minimax problems," *Annals of Operations Research*, vol. 164, no. 1, pp. 167–191, 2008.

[10] Y. Xue, "A SQP method for minimax problems," *Journal of System Science and Math Science*, vol. 22, pp. 355–364, 2002 (Chinese).

[11] B. Rustem and Q. Nguyen, "An algorithm for the inequality-constrained discrete min-max problem," *SIAM Journal on Optimization*, vol. 8, no. 1, pp. 265–283, 1998.

[12] 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.

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

[14] Y. Feng, L. Hongwei, Z. Shuisheng, and L. Sanyang, "A smoothing trust-region Newton-CG method for minimax problem," *Applied Mathematics and Computation*, vol. 199, no. 2, pp. 581–589, 2008.

[15] L. H. Ma, Y. Zhang, C. N. Yang et al., "A neural network model for equality and inequality constrained minimax problems," *Information Technology Journal*, vol. 11, no. 11, pp. 1655–1659, 2012.

[16] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," in *Numerical Analysis*, pp. 144–157, Springer, Berlin, Germany, 1978.

[17] E. R. Panier and A. L. Tits, "Superlinearly convergent feasible method for the solution of inequality constrained optimization problems," *SIAM Journal on Control and Optimization*, vol. 25, no. 4, pp. 934–950, 1987.

[18] Z. Wan, "A modified SQP algorithm for mathematical programs with linear complementarity constraints," *Acta Scientiarum Naturalium Universitatis Normalis Hunanensis*, vol. 26, pp. 9–12, 2001.

[19] Z. Zhu and S. Wang, "A superlinearly convergent numerical algorithm for nonlinear programming," *Nonlinear Analysis*, vol. 13, no. 5, pp. 2391–2402, 2012.

[20] Z. Luo, Z. Zhu, G. Chen et al., "A superlinearly convergent SQP algorithm for constrained optimization problems," *Journal of Computational Information Systems*, vol. 9, no. 11, pp. 4443–4450, 2013.

[21] Z. Zhu and C. Zhang, "A superlinearly convergent sequential quadratic programming algorithm for minimax problems," *Chinese Journal of Numerical Mathematics and Applications*, vol. 27, no. 4, pp. 15–32, 2005.

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

[23] Q.-J. Hu and J.-Z. Hu, "A sequential quadratic programming algorithm for nonlinear minimax problems," *Bulletin of the Australian Mathematical Society*, vol. 76, no. 3, pp. 353–368, 2007.

[24] 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.

[25] W. Xue, C. Shen, and D. Pu, "A new non-monotone SQP algorithm for the minimax problem," *International Journal of Computer Mathematics*, vol. 86, no. 7, pp. 1149–1159, 2009.

[26] J. B. Jian, X. L. Zhang, R. Quan, and Q. Ma, "Generalized monotone line search SQP algorithm for constrained minimax problems," *International Journal of Control*, vol. 58, no. 1, pp. 101–131, 2009.

[27] G. Zoutendijk, *Methods of Feasible Directions*, Elsevier, Amsterdam, The Netherlands, 1960.

[28] M. M. Kostreva and X. Chen, "A superlinearly convergent method of feasible directions," *Applied Mathematics and Computation*, vol. 116, no. 3, pp. 231–244, 2000.

[29] F. Facchinei and S. Lucidi, "Quadratically and superlinearly convergent algorithms for the solution of inequality constrained minimization problems," *Journal of Optimization Theory and Applications*, vol. 85, no. 2, pp. 265–289, 1995.