AMO - Advanced Modeling and Optimization, Volume 16, Number 1, 2014

A Modified Feasible SQP Method for Constrained Minimax Optimization Problems[1]

Zhijun Luo*, Lirong Wang** and Guohua Chen*

* The Department of Mathematics and Econometrics,

Hunan University of Humanities, Science and Technology, Loudi, 417000,P.R.China

**The Department of Information science and Engineering,

Hunan University of Humanities, Science and Technology, Loudi, 417000,P.R.China

Abstract. This paper presents a modified feasible SQP algorithm for solving constrained minimax optimization problems. At each iteration of the proposed algorithm in this paper, the descent direction is yielded by solving only one quadratic programming through introducing an auxiliary variable. A height-order correction direction is obtained by solving a corresponding quadratic programming. Furthermore, under some mild conditions, the global convergence and superlinear properties are proved. Finally, some numerical results reported show that the algorithm is successful.

Key words. Constrained minimax problems; Feasible SQP method; Global convergence; Superlinear convergence

1. Introduction

In the last few decades, optimization has been a basic tool in all areas of applied mathematics, economics, medicine, engineering and other sciences(such as [1],etc). The minimax problem is a typical non-differentiable nonlinear programming in optimization problems,which wants to obtain the objection functions minimum under conditions of the maximum of the functions, and it can be widely applied in many fields, for example engineering, finance, economics, management and other fields, can be stated as a minimax optimization problem.

In this paper, we consider the following inequality constrained optimization problems:

$$\begin{array}{ll} \min\limits_{x\in R^n} & F(x) \\ \text{s.t.} & g_j(x) \le 0,\ j \in J = \{1,2,\cdots,m'\}, \end{array} \tag{1.1}$$

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

The problem (1.1) can be transformed into a smooth constrained optimization problem in R^{n+1} as follows:

$$\begin{array}{lll} \min & z & \\ \text{s.t.} & f_i(x) \le z, & i \in I, \\ & g_j(x) \le 0, & j \in J. \end{array} \tag{1.2}$$

Obviously, from the problem (1.2), the K-T condition of (1.1) is defined as follows:

[1]Corresponding the author: E-mail: ldlzj123@163.com

AMO -Advanced Modeling and Optimization. ISSN: 1841-4311

$$\begin{cases} \sum\limits_{i\in I} \lambda_i \nabla f_i(x) + \sum\limits_{j\in J} \mu_j \nabla g_j(x) = 0 \\ \sum\limits_{i\in I} \lambda_i = 1 \\ \lambda_i \geq 0, f_i(x) - F(x) \leq 0, \lambda_i(f_i(x) - F(x)) = 0, i \in I \\ \mu_j \geq 0, g_j(x) \leq 0, \mu_j g_j(x) = 0, j \in J, \end{cases} \tag{1.3}$$

where $\lambda_i,\ \mu_j$ are the corresponding vector. Based on the equivalent relationship between the K-T point of(1.2) and the stationary point of (1.1), a lot of methods focus on finding the K-T point of (1.1), namely solving (1.3). And many algorithms have been proposed to solve minimax problem[2]-[8]. Such as Refs.[3]-[4], the minimax problems are discussed with nonmonotone line search, which can effectively avoids Maratos effect. Combine the trust-region methods with the line-search methods and curve-search methods, Wang and Zhang [5] propose A hybrid algorithm for linearly constrained minimax problems. Many other effective algorithms for solving the minimax problems are presented, such as Refs [6]-[8] etc.

Sequential quadratic programming (SQP)method has fast convergence rate, and it is one of the efficient algorithm for solving constrained optimization problems and is studied deeply and widely(see such as [9]-[13] etc). Recently, many researches have extended the popular SQP scheme to the minimax problems (see [14]-[18] etc). For typical SQP method, the standard search direction d should be obtained by solving the following quadratic programming:

$$\begin{array}{ll} \min & \nabla F(x)^T d + \frac{1}{2} d^T H d \\ \text{s.t.} & g_j(x) + \nabla g_j(x)^T d \leq 0,\ j \in J, \end{array} \tag{1.4}$$

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 (1.2), one considers the following quadratic programming through introducing an auxiliary variable z

$$\begin{array}{ll} \min & z + \frac{1}{2} d^T H d \\ \text{s.t.} & f_i(x) + \nabla f_i(x)^T d \leq z,\ i \in I, \\ & g_j(x) + \nabla g_j(x)^T d \leq 0,\ \ j \in J. \end{array} \tag{1.5}$$

However, it is well known that the solution d of (1.5) may not be a feasible descent direction and can not avoid the Maratos effect.

In this paper, an effective sequential quadratic programming method for solving the constrained minimax optimization problems(1.1) is proposed. Suppose x^k is the current iteration point, the descent direction d^k is obtained by solving quadratic programming as following:

$$\begin{array}{lll} \min & z + \frac{1}{2} d^T H_k d & \\ \text{s.t.} & f_i(x^k) + \nabla f_i(x^k)^T d - F(x^k) \leq z, & i \in I, \\ & g_j(x^k) + \nabla g_j(x^k)^T d \leq \eta_k z, & j \in J. \end{array} \tag{1.6}$$

where H_k is a symmetric positive definite matrix, η_k is nonnegative auxiliary variable. In order to avoid Maratos effect, a height-order correction direction is computed by corresponding quadratic programming. Under suitable conditions, the theoretical analysis shows that the convergence of our Algorithm can be obtained.

2. Algorithm

For convenience, we denote sets

$$I(x) = \{i \in I | f_i(x) = F(x)\}, \quad J(x) = \{j \in J | g_j(x) = 0\}.$$

Now we state our algorithm as follows.

Algorithm :

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

Step 1 : Compute (d^k, z_k) by the quadratic problem(1.6) at x^k. Let $(\lambda^k,\ \mu^k)$ be the corresponding vector. If $d^k = 0$ then STOP;

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

$$\begin{array}{lll} \min & z + \frac{1}{2}(d^k + d)^T H_k (d^k + d) & \\ \text{s.t.} & f_i(x^k + d^k) + \nabla f_i(x^k + d^k)^T d - F(x^k + d^k) \le z, & i \in I, \\ & g_j(x^k + d^k) + \nabla g_j(x^k + d^k)^T d \le -\eta_k, & j \in J. \end{array} \tag{2.7}$$

$(\tilde{\lambda^k},\ \tilde{\mu^k})$ is the corresponding vector. If $\|\tilde{d^k} > \|d^k\|$, set $\tilde{d^k} = 0$.;

Step 3 : The line search ; Compute t_k, the first number t in the sequence $\{1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \cdots\}$ satisfying:

$$F(x^k + td^k + t^2\tilde{d}^k) \le F(x^k) - \alpha t (d^k)^T H_k d^k \tag{2.8}$$

$$g_j(x^k + td^k + t^2\tilde{d}^k) \le 0, j \in J \tag{2.9}$$

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. Convergence of Algorithm

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

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

H 3.2. *$\forall x \in R^n$,the set of vectors*

$$\left\{\begin{pmatrix} -1 \\ \nabla f_i(x) \end{pmatrix}, i \in I(x); \begin{pmatrix} 0 \\ \nabla g_j(x) \end{pmatrix}, j \in J(x)\right\}$$

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 3.1. *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 (1.6). Then*

(1) $z_k + \frac{1}{2}(d^k)^T H_k d^k \le 0, z_k \le 0$.

(2) If $d^k = 0$, then x^k is a K-T point of problem (1.1).

(3) If $d^k \ne 0$,then $z_k < 0$, moreover, d^k is a feasible direction of descent for (1.1) at point x^k.

proof : (1)For $(0,0) \in R^{n+1}$is a feasible solution of (1.6) and H_k is positive definite, one has

$$z_k + \frac{1}{2}(d^k)^T H_k d^k \le 0, \quad z_k \le -\frac{1}{2}(d^k)^T H_k d^k \le 0. \tag{3.10}$$

(2)Firstly, we prove $d^k = 0 \Leftrightarrow z_k = 0$. If $z_k = 0$, then $\frac{1}{2}(d^k)^T H_k d^k = \frac{1}{2}(d^k)^T H_k d^k + z_k \le 0$. 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

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

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 (1.1) when $d^k = 0$. From the problem (1.6), the K-T condition at x^k is defined as follows:

$$\begin{cases} H_k d^k + \sum\limits_{i \in I} \lambda_i^k \nabla f_i(x^k) + \sum\limits_{j \in J} \mu_j^k \nabla g_j(x^k) = 0; \\ \sum\limits_{i \in I} \lambda_i^k + \eta_k \sum\limits_{j \in J} \mu_j^k = 1; \\ \lambda_i \ge 0, \ 0 \le \lambda_i^k \perp (f_i(x^k) + \nabla f_i(x^k)^T d^k - F(x^k) - z_k) \le 0, \ i \in I; \\ \mu_j \ge 0, \ 0 \le \mu_j^k \perp (g_j(x^k) + \nabla g_j(x^k)^T d^k - \eta_k z_k) \le 0, \ j \in J. \end{cases} \tag{3.11}$$

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

$$\begin{cases} \sum\limits_{i \in I} \lambda_i \nabla f_i(x) + \sum\limits_{j \in J} \mu_j \nabla g_j(x) = 0 \\ \sum\limits_{i \in I} \lambda_i = 1 \\ \lambda_i \ge 0, f_i(x) - F(x) \le 0, \lambda_i(f_i(x) - F(x)) = 0, i \in I \\ \mu_j \ge 0, g_j(x) \le 0, \mu_j g_j(x) = 0, j \in J. \end{cases}$$

That is to see the results holds.
(3) If $d^k \ne 0$, together with the positive definite property of the matrix H_k, it is easy to show that $z_k < 0$. On the other hand, the directional derivative $F^{'}(x;d)$ of $F(x)$ at point x along direction d can be expressed as

$$F'(x;d) = \lim_{\lambda \to 0^+} \frac{F(x + \lambda d) - F(x)}{\lambda} = max\{\nabla f_i(x)^T d : i \in I(x).\}$$

Therefore, $F'(x^k; d^k) \le z_k < 0$, i.e., d^k is a descent direction of $F(x)$ at point x^k.
For any $j \in J(x^k)$, it follows that

$$g_j(x^k + td^k) = g_j(x^k) + tg_j(x^k)^T d^k + o(t) \le t\eta_k z_k + o(t) \le 0$$

for $t > 0$ small enough, i.e., d^k is a feasible direction.

From Lemma 3.1, it is obvious, if $d^k \ne 0$, that the line search in step 3 yields is always completed.

Lemma 3.2. *If $d^k \ne 0$, the line search in Step 3 of Algorithm is well defined.*

proof: Similar to the proof of Lemma 3.1(3), we get that $g_j(x^k + td^k + t^2\tilde{d}^k) \leq 0, j \in J$ for $t > 0$ small enough. For $i \in I$, together with(3.10) we have

$$\begin{aligned}
\alpha_k &\triangleq f_i(x^k + td^k + t^2\tilde{d}^k) - F(x^k) + \alpha t(d^k)^T H_k d^k \\
&\leq f_i(x^k) + t\nabla f_j(x^k)^T(d^k + t\tilde{d}^k) + o(t) - F(x^k) + \alpha t(d^k)^T H_k d^k \\
&\leq t(f_i(x^k) + \nabla f_i(x^k)^T d^k - F(x^k)) + (1-t)(f_i(x^k) - F(x^k)) + \alpha t(d^k)^T H_k d^k + o(t) \\
&\leq tz_k + \alpha t(d^k)^T H_k d^k + o(t) \\
&\leq (\alpha - \frac{1}{2})t(d^k)^T H_k d^k + o(t).
\end{aligned}$$

This implies that there exists some $t > 0$ such that $\alpha_k \leq 0$. It is clear that the line search condition (2.8)is satisfied.

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

$$x^k \longrightarrow x^*, \quad H_k \longrightarrow H_*, \quad \eta_k \longrightarrow \eta_*, \quad d^k \longrightarrow d^*, \quad z_k \longrightarrow z_*, \quad \lambda^k \longrightarrow \lambda^*, \quad \mu^k \longrightarrow \mu^*, \quad k \in K. \tag{3.12}$$

Theorem 3.1. *The algorithm either stops at the K-T point x^k of the problem (1.1)in finite number of steps, or generates an infinite sequence $\{x^k\}$ any accumulation point x^* of which is a K -T point of the problem (1.1).*

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 (3.12)holds.The cases $\eta_* = 0$, and $\eta_* > 0$are considered separately.
Case A $\eta_* = 0$.
By the Step 4, there exists an infinite index set $K_1 \subseteq K$, such that $d^{k-1} \longrightarrow 0, k \in K_1$. While, by step 3, it holds that

$$\lim_{k\in K_1} \|x^k - x^{k-1}\| = \lim_{k\in K_1} \|t_{k-1}d^{k-1} + t_{k-1}^2\tilde{d}^{k-1}\| \leq \lim_{k\in K_1} (\|d^{k-1}\| + \|\tilde{d}^{k-1}\|) = 0.$$

So, the fact that $x^{k-1} \stackrel{k\in K_1}{\longrightarrow} x^*$, implies that$d^{k-1} \stackrel{k\in K_1}{\longrightarrow} 0$. So, from Lemma 3.1, it is clear that $x^*$is a K-T point of (1.1).
Case B $\eta_* > 0$.
Obviously, it only needs to prove that $d^k \longrightarrow 0, k \in K$. Suppose by contradiction that $d^* \neq 0$. Since

$$\begin{cases} f_i(x^k) + \nabla f_i(x^k)^T d^k - F(x^k) \leq z_k, \ i \in I \\ g_j(x^k) + \nabla g_j(x^k)^T d^k \leq \eta_k z_k, \ j \in J, \end{cases} \tag{3.13}$$

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

$$\begin{cases} f_i(x^*) + \nabla f_i(x^*)^T d^* - F(x^*) \leq z_*, \ i \in I \\ g_j(x^*) + \nabla g_j(x^*)^T d^* \leq \eta_* z_*, \quad j \in J. \end{cases} \tag{3.14}$$

So, the following corresponding QP subproblem (1.6) at x^*

$$\begin{array}{ll} \min\limits_{(d,z)\in R^{n+1}} & z + \frac{1}{2}d^T H_* d \\ s.t & f_i(x^*) + \nabla f_i(x^*)^T d - F(x^*) \leq z, \ i \in I, \\ & g_j(x^*) + \nabla g_j(x^*)^T d \leq \eta_* z, \quad j \in J, \end{array} \tag{3.15}$$

has a nonempty feasible set. Moreover, it is not difficult to show that (z_*, d^*) is the unique solution of (3.15). So, it holds that

$$\begin{cases} z_* < 0, \nabla f_i(x^*)^T d^* \le z_* < 0, \ i \in I(x^*), \\ \nabla g_j(x^*)^T d^* \le \eta_* z_* < 0, \qquad j \in J(x^*). \end{cases} \tag{3.16}$$

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

$$\begin{cases} \nabla f_i(x^k)^T d^k \le \frac{1}{2} \nabla f_i(x^*)^T d^* < 0, \ i \in I(x^*), \\ \nabla g_j(x^k)^T d^k \le \frac{1}{2} \nabla g_j(x^*)^T d^* < 0, \quad j \in J(x^*). \end{cases} \tag{3.17}$$

From (3.17), by imitating the proof of Proposition 3.2 in [10], we know that the stepsize t_k obtained by the line search is bounded away from zero on K, i.e.,

$$t_k \ge t_* = inf\{t_k, \quad k \in K\} > 0. \tag{3.18}$$

In addition, from (2.8) and Lemma 3.1, 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

$$f_i(x^k) \to f_i(x^*), \quad k \in K, \quad i \in I(x^*). \tag{3.19}$$

Hence, from (2.8) and (3.17)-(3.19), we get

$$0 = \lim_{k \in K} (f_i(x^{k+1}) - f_i(x^k)) \le \lim_{k \in K} \alpha t_k \nabla f_i(x^k)^T d^k \le \tfrac{1}{2} \alpha t_* \nabla f_i(x^*)^T d^* < 0. \tag{3.20}$$

It is a contradiction. So, $d^* = 0$. Thereby , according to Lemma 3.1, x^* is a K-T point of problem (1.1).

4. Rate of convergence

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

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

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

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

$$d^T \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) d > 0, \ 0 \ne d \in S^*,$$

where,

$$\begin{aligned} \nabla^2_{xx} L(x^*, \lambda^*, \mu^*) &= \sum_{i \in I} \lambda_i^* \nabla^2 f_i(x^*) + \sum_{j \in J} \mu_j^* \nabla^2 g_j(x^*) \\ &= \sum_{i \in I(x^*)} \lambda_i^* \nabla^2 f_i(x^*) + \sum_{j \in J(x^*)} \mu_j^* \nabla^2 g_j(x^*), \end{aligned}$$

$$S^* \quad = \{d \in R^n \mid \nabla f_i(x^*)^T d = \nabla f_{i_k}(x^*)^T d, \forall i \in I(x^*), i_k \in I(x^*), \nabla g_j(x^*)^T d = 0, \ \forall j \in J(x^*)\}.$$

According to the all stated assumptions H4.1-H4.3 and Lemma 4.1 in [11], we have the following results

Lemma 4.1. *The KKT point x^* of problem (1.1) is isolated.*

Lemma 4.2. *The entire sequence $\{x^k\}$ converges to x^*, i.e., $x^k \to x^*, k \to \infty$.*

Lemma 4.3. *For k large enough, it holds that*
1) $d^k \to 0, z_k \to 0, \lambda^k \to \lambda^*, \mu^k \to \mu^*$.
2) $\widetilde{d}^k$ obtained by step 2 satisfies,

$$\left\|\widetilde{d}^k\right\| = O\left(\|d^k\|^2\right),\ g_j(x^k + d^k + \widetilde{d}^k) = O\left(\|d^k\|^3\right) \forall j \in J(x).$$

.

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

H 4.4. *The matrix sequence H_k satisfies that*

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

where,

$$P_k = I_n - A_k(A_k^T A_k)^{-1}A_k^T,$$
$$A_k = A_k(x^k) = ((\nabla f_i(x^k) - \nabla f_{i_k}(x^k)),\ \nabla g_j(x^k)), (i \in I(x^k)\backslash\{i_k\}, j \in J(x^k).$$

According to Lemma 4.2 and Lemma 4.3, it is easy to know

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

Lemma 4.4. *For k large enough, under the above-mentioned assumptions, $t_k \equiv 1$.*

Proof. It is only necessary to show that the inequalities (2.8) and (2.9) are satisfied with $t = 1$ for k large enough. Firstly, we prove that

$$F(x^k + d^k + \widetilde{d}^k) \le F(x^k) - \alpha d^{kT} H_k d^k. \tag{4.1}$$

From (1.6), (3.11) and Taylor expansion, we have

$$\begin{aligned} f_i(x^k + d^k) &= f_i(x^k) + \nabla f_i(x^k)^T d^k + O(\|d^k\|^2) \\ &= f(x^k) + z_k + O(\|d^k\|^2), \qquad i \in I(X^*), \end{aligned}$$

$$\begin{aligned} f_j(x^k + d^k) &= f_j(x^k) + \nabla f_j(x^k)^T d^k + O(\|d^k\|^2) \\ &= f(x^k) + z_k + O(\|d^k\|^2), \qquad j \in I(X^*). \end{aligned}$$

Hence,

$$f_i(x^k + d^k) = f_j(x^k + d^k) + O(\|d^k\|^2), \qquad \forall i, j \in I(x^*). \tag{4.2}$$

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

$$f_i(x^k + d^k + \widetilde{d}^k) = f_j(x^k + d^k + \widetilde{d}^k) + O(\|d^k\|^3), \qquad \forall i, j \in I(x^*). \tag{4.3}$$

On the other hand, the facts that $d^k \to 0,\ \widetilde{d}^k \to 0$ 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

$$\begin{aligned} F(x^k + d^k + \widetilde{d}^k) &= \max_{l \in I}\{f_l(x^k + d^k + \widetilde{d}^k)\} \\ &= f_{j_k}(x^k + d^k + \widetilde{d}^k) = f_j(x^k + d^k + \widetilde{d}^k) + O(\|d^k\|^3), \quad \forall j \in I(x^*) \end{aligned} \tag{4.4}$$

Hence, combining $\sum\limits_{l\in I(x^*)} \lambda_i^k = 1$ with (4.4), we have

$$\begin{array}{l} F(x^k + d^k + \widetilde{d}^k) = \sum\limits_{i\in I(x^*)} \lambda_i^k f_i(x^k + d^k + \widetilde{d}^k) + O(\|d^k\|^3) \\ = \sum\limits_{i\in I(x^*)} \lambda_i^k (f_i(x^k) + \nabla f_i(x^k)^T (d^k + \widetilde{d}^k) + \frac{1}{2}(d^k)^T \nabla^2 f_i(x^k) d^k) + o(\|d^k\|^2) \end{array} \tag{4.5}$$

From the KKT condition (3.11) implies

$$\begin{array}{l} \sum\limits_{i\in I(x^*)} \lambda_i^k \nabla f_i(x^k)^T (d^k + \widetilde{d}^k) = -(d^k)^T H_k d^k) + o(\|d^k\|^2) \\ \sum\limits_{i\in I(x^*)} \lambda_i^k f_i(x^k) \le F(x^k). \end{array}$$

Thus,

$$\begin{array}{l} F(x^k + d^k + \widetilde{d}^k) \\ \le F(x^k) - d^{kT} H_k d^k + \frac{1}{2} d^{kT} (\sum\limits_{i\in I(x^*)} \lambda_i^k \nabla^2 f_i(x^k)) d^k + o(\|d^k\|^2), \\ = F(x^k) - \frac{1}{2} d^{kT} H_k d^k + \frac{1}{2} d^{kT} (\sum\limits_{i\in I(x^*)} \lambda_i^k \nabla^2 f_i(x^k) - H_k) d^k + o(\|d^k\|^2), \\ = F(x^k) - \alpha d^{kT} H_k d^k - (\frac{1}{2} - \alpha) d^{kT} H_k d^k + o(\|d^k\|^2), \end{array}$$

for k large enough, according to $\alpha \in (0, \frac{1}{2})$, it holds that

$$F(x^k + d^k + \widetilde{d}^k) \le F(x^k) - \alpha d^{kT} H_k d^k.$$

Secondly, we show that $g_j(x^k + d^k + \widetilde{d}^k) \le 0$, which implies that the (2.9) holds for $t = 1$. For $j \in J \setminus J(x^*)$, this is always satisfied since $g_j(x^*) < 0,\ d^k \to 0,\ \widetilde{d}^k \to 0$ and the continuity of $g_j(x)$. When $j \in J(x^*)$, expanding $g_j(x^k + d^k + \widetilde{d}^k)$ around $x^k + d^k$, we have

$$\begin{array}{l} g_j(x^k + d^k + \widetilde{d}^k) = g_j(x^k + d^k) + \nabla g_j(x^k + d^k)^T \widetilde{d}^k + o(\|\widetilde{d}^k\|^2), \\ = g_j(x^k + d^k) + \nabla g_j(x^k)^T \widetilde{d}^k + O(\|\widetilde{d}^k\|^3), \end{array}$$

From (2.7), we obtain that

$$\nabla g_j(x^k)^T \widetilde{d}^k \le -\|d^k\|^\gamma - g_j(x^k + d^k).$$

i.e.

$$g_j(x^k + d^k + \widetilde{d}^k) \le -\|d^k\|^\gamma + O(\|\widetilde{d}^k\|^3).$$

According to$\gamma \in (2, 3)$, it holds that $g_j(x^k + d^k + \widetilde{d}^k) \le 0, j \in J(x^*)$.

From Lemma 4.4, we can get the following theorem

Theorem 4.1. *Under all stated assumptions, the algorithm is superlinearly convergent, i.e., 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. The code of the proposed algorithm is written by using MATLAB 7.0 and utilized the optimization toolbox to solve the quadratic programmings (1.6) and (2.7). The results show that the proposed algorithm is efficient.

During the numerical experiments, it is chosen at random some parameters as follows: $\alpha = 0.25, \eta_0 = 1$ and $H_0 = I$, the $n \times n$ unit matrix.H_k is updated by the BFGS formula [9]. In the implementation, the stopping criterion of Step 1 is changed to $If\ \|d_0^k\| \leq 10^{-6} \quad STOP$.

This algorithm has been tested on some problems from Ref.[13] and [14]. The results are summarized in Table 1. The columns of this table has the following meanings:
No.: the number of the test problem inRef.[13] or [14];
n : the dimension of the problem;
m : the number of objective functions;
$m^{'}$: the number of inequality constraints;
NT: the number of iterations;
IP: the initial point;
FV: the final value of the objective function.

Table 1

NO.	n,m, $m^{'}$	NT	IP	FV
1 (Problem 1 in [14])	2, 3, 0	11	$(1,5)^T$	1.952224
2 (Problem 4 in [14])	2, 3, 0	10	$(3,1)^T$	0.616234
3 (Problem 1 in [13])	2, 3, 2	7	$(0,0)^T$	1.952224
4 (Problem 2 in [13])	2, 6, 2	12	$(1,3)^T$	0.616432
5 (Problem 4 in [13])	2, 3, 2	10	$(4,2)^T$	2.250000
6 (Problem 5 in [13])	4, 4, 3	32	$(0,1,1,0)^T$	-44.000000
7 (Problem 6 in [13])	2, 3, 2	4	$(0,1)^T$	2.000000

Acknowledgment

This research was supported by the Foundation of Hunan Provincial Education Department Under Grant (NO. 12A077,12C0743 and 13C453) and Scientific Research Fund of Hunan University of Humanities,Science and Technology of China(No.2012QN04).

References

[1] S. A. Rozhenko, G. A. Mikhailov, Minimax Parametric Optimization of Numerical-Statistical 'Method of Similar Trajectories for Solution of Radiation Transfer Theory Problems[J], Russian Journal of Numerical Analysis and Mathematical Modelling, 28(2), 2013, 201-212

[2] J.L. Zhou. A.L. Tits, Nonmonotone Line Search Method for Minimax Problems[J], Journal of Optimization Theory and Applications, 76, 1993, 455-476

[3] Y.H. Yu, L. Gao, Nonmonotone Line Search Algorithm for Constrained Minimax Problems[J], Journal of Optimization Theory and Applications, 115(2), 2002, 419-446

[4] F. Wang, Y. Wang, Nonmonotone Algorithm for Minimax Optimization Problems[J], Applied Mathematics and Computation, 217(13), 2011, 6296-6308

[5] F. S. Wang, K. C. Zhang, A hybrid Algorithm for Nonlinear Minimax Problems[J], Annals of Operations Research, 164(1), 2008, 167-191

[6] E. Obasanjo, G. Tzallas-Regas, and B. Rustem, An Interior-point Algorithm for Nonlinear Minimax Problems[J], Journal of Optimization Theory and Applications, 144(2), 2010, 291-318

[7] B. Rustem, S. Zakovic, P. Parpas, An Interior Point Algorithm for Continuous Minimax: Implementation and Computation[J], Optimisation Methods &Software, 23(6), 2008, 911-928

[8] F. Ye, H. Liu, etc, A Smoothing Trust-region Newton-CG Method for Minimax Problem[J], Applied Mathematics and Computation, 199(2), 2008,581-589

[9] M. J. D. Powell. A Fast Algorithm for Nonlinearly Constrained Optimization Calculations. In:Waston, G.A.(ed). Numerical Analysis. Springer,Berlin 1978, 144-157.

[10] Panier E R, Tits A L. A superlinearly convergent feasible method for the solution of inequality constrained optimization problems[J]. SIAM Journal on control and Optimization, 1987, 25(4): 934-950.

[11] Z.B.Zhu, S. Wang, A Superlinearly Convergent Numerical Algorithm for Nonlinear Programming[J], Nonlinear Analysis, 13(5), 2012, 239-2402

[12] Z.Luo, Z.Zhu, G.Chen, etal, A Superlinearly Convergent SQP Algorithm for Constrained Optimization Problems[J], Journal of Computational Information Systems, 9(11), 2013, 4443-4450

[13] B.Rustem, Q.Nguyen, An algorithm for the inequality-constrained discrete minimax problem, SIAM Journal on Optimization, 8(1), 1991, 265-283

[14] Y. Xue, A SQP Method for Minimax Problems[J], Journal of System Science and Math Science(in Chinese), 22, 2002, 355-364

[15] J.B.Jian, R. Quan, Q.J. Hu, A New Superlinearly Convergent SQP Algorithm for Nonlinear Minimax Problems[J], Acta Mathematicae Applicatae Sinica(English Series), 23(3), 2007, 395-410

[16] Q.J.Hu, Y. Chen,etc, A Modified SQP Algorithm for Minimax Problems[J], Journal of Mathematical Analysis and Applications, 360(1), 2009, 211-222

[17] Z.B.Zhu, X.Cai, J.B.Jian, An Improved SQP Algorithm for Solving Minimax Problems[J], Applied Mathematics Letters, 22(4),2009,464-469

[18] W.J. Xue, C.G. Shen, D.G. Pu, A New Non-Monotone SQP Algorithm for the Minimax Problem[J], International Journal of Computer Mathematics, 86(7), 2009, 1149-1159