

A SEQUENTIAL QUADRATIC PROGRAMMING ALGORITHM FOR NONLINEAR MINIMAX PROBLEMS

QING-JIE HU AND JU-ZHOU HU

In this paper, an active set sequential quadratic programming algorithm with non-monotone line search for nonlinear minmax problems is presented. At each iteration of the proposed algorithm, a main search direction is obtained by solving a reduced quadratic program which always has a solution. In order to avoid the Maratos effect, a correction direction is yielded by solving the reduced system of linear equations. Under mild conditions *without the strict complementarity*, the global and superlinear convergence can be achieved. Finally, some preliminary numerical experiments are reported.

1. INTRODUCTION

It is well known that many engineering design problems can be expressed in a minmax form as follows

$$(1.1) \quad (P) \quad \min\{F(x) \mid x \in R^n\},$$

where $F(x) = \max\{f_j(x), j \in I\}$ with $I = \{1, 2, \dots, m\}$, which enables the designer to minimise the maximum violation of design specifications, (see [11, 16, 18, 20]). Since the objective function $F(x)$ contains the max operator, it is continuous but nondifferentiable even when the $f_i(x), i = 1, \dots, m$, are all differentiable. In particular, at points where the max is attained in two or more functions $f_i(x)$, $F(x)$ is nondifferentiable. So, the classical methods for smooth optimisation problems may fail to reach an optimum if they are applied directly to the nonlinear minmax problem.

Many of the schemes that have been proposed for solving minmax problems are based on an equivalent translation of the original problem (1.1) as follows

$$(1.2) \quad (P') \quad \min\{y \mid (x, y) \in R^{n+1} : f_j(x) \leq y, j \in I\},$$

but there are drawbacks to use the form (P') directly to obtain a solution of (1.1) since some properties of the primal problem may be thrown away (see [15]).

Received 7th March, 2007

The authors thank the anonymous referees for their valuable comments and suggestions on the original version of this paper.

Copyright Clearance Centre, Inc. Serial-fee code: 0004-9727/07 \$A2.00+0.00.

Obviously, the Karush–Kuhn–Tucker conditions of (1.2) can be stated as follows

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix} + \sum_{j \in I} \lambda_j \begin{pmatrix} \nabla f_j(x) \\ -1 \end{pmatrix} = 0, \\ \lambda_j \geq 0, \quad f_j(x) - y \leq 0, \quad \lambda_j(f_j(x) - y) = 0, \quad j \in I,$$

and these relationships are equivalent to

$$(1.3) \quad \sum_{j \in I} \lambda_j \nabla f_j(x) = 0, \quad \sum_{j \in I} \lambda_j = 1; \quad \lambda_j(f_j(x) - F(x)) = 0, \quad \lambda_j \geq 0, j \in I.$$

So, a point $x \in R^n$ is called as stationary point of (P) ([5]) if there exists a vector $\lambda = (\lambda_j, j \in I)$ such that (1.3) holds, where λ is said to be multiplier vector.

Since the sequential quadratic programming method has satisfactory convergence, it is one of the most effective algorithms for solving nonlinearly constrained optimisation problems and it is studied widely and deeply, (see [10, 12, 8, 7, 17, 4]). So several authors have extended the popular sequential quadratic programming scheme to the minimax problems (see [1, 9, 14, 13, 21, 22]). Global convergence is usually ensured by means of a suitable line search. Based on the equivalent relationship between the Karush–Kuhn–Tucker point of (P') and the stationary point of (P), many algorithms focus on finding the stationary point of (P), namely solving (1.3). Zhou and Tits proposed an algorithm ([23]): the search direction is generated by solving two quadratic programs, and avoiding the Maratos effect by means of nonmonotone line search. Unfortunately, it obtains only two-step superlinear convergence and needs the strict complementarity (namely the multiplier $\lambda_j > 0$ for j satisfying $f_j(x) = F(x)$) which is rather difficult for testing. Recently, some sequential quadratic programming algorithms have been proposed to overcome the shortcoming of the two-step superlinear convergence and obtain one-step superlinear convergence, such as [21], but their assumptions are rather strong: (1) the iteration sequence $\{x^k\}$ generated by the algorithm is assumed to satisfy $\lim_{k \rightarrow \infty} x^k = x^*$; that is, the algorithm is assumed to be strongly convergent; (2) the step size is supposed to always equal one after finite iterations; (3) the sufficient condition for the superlinear convergence is rather strong.

In this paper, we present an improved sequential quadratic programming algorithm for the minimax problem (1.1). In this algorithm, a main search direction is obtained by solving a quadratic program which always has a solution. In order to avoid the Maratos effect, unlike ([23]), a correction direction is yielded by solving a system of linear equation. Under mild conditions without the strict complementarity, global and superlinear convergence can be obtained. Finally, some preliminary numerical experiments are reported.

The balance of this paper is organised as follows. The algorithm and its properties are presented in Section 2. Global and superlinear convergence are analysed in Section 3

and Section 4, respectively. Numerical results are reported in Section 5. Section 6 is devoted to final remarks.

2. ALGORITHM

For convenience of presentation, for a given $x \in R^n$, we use the following notation throughout the remainder of this paper

$$(2.1) \quad \begin{aligned} f(x) &= (f_j(x), j \in I)^T, \quad I(x) = \{j \in I : f_j(x) = F(x)\}, \\ g_j(x) &= \nabla f_j(x), j \in I, \quad g(x) = (g_j(x), j \in I). \end{aligned}$$

We assume that the following assumptions hold in this paper.

(H1) Functions f_j ($j \in I$) are all at least first order continuously differentiable.

(H2) Vectors $\left\{ \begin{pmatrix} g_j(x) \\ -1 \end{pmatrix}, j \in I(x) \right\}$ in R^{n+1} are linearly independent.

REMARK 1. It is easy to prove that (H2) is equivalent to the following condition: For each $t \in I(x)$, vectors $\{g_j(x) - g_t(x), j \in I(x) \setminus \{t\}\}$ are linearly independent. This can also be illustrated from the following example: Let $I(x^0) = \{1, 2, 3\}$ and

$$g_1(x^0) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad g_2(x^0) = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \quad g_3(x^0) = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

for some $x^0 \in R^n$. Obviously,

$$\left\{ \begin{pmatrix} g_j(x^0) \\ -1 \end{pmatrix}, j \in I(x^0) \right\}$$

are linearly independent and $\{g_j(x^0) - g_t(x^0), j \in I(x^0) \setminus \{t\}\}$ are linearly independent for each $t \in I(x^0)$.

Let $x^k \in R^n$ be a given iteration point. Based on (H2), we use the following pivoting operation to generate an ϵ -active constraint subset $I_k \supseteq I(x^k)$ such that the matrix $A_k \triangleq \left(\begin{pmatrix} g_j(x^k) \\ -1 \end{pmatrix}, j \in I_k \right)$ is of full column rank.

PIVOTING OPERATION POP.

STEP (i) Select an initial parameter $\epsilon = \epsilon_{k-1} > 0$.

STEP(ii) Generate the ϵ -active constraint subset $I(x^k, \epsilon)$ and matrix M_k by

(2.2)

$$I(x^k, \epsilon) = \{j \in I : -\epsilon \leq f_j(x^k) - F(x^k) \leq 0\}, \quad M_k = \left(\begin{pmatrix} g_j(x^k) \\ -1 \end{pmatrix}, j \in I(x^k, \epsilon) \right).$$

STEP(iii) If $\det(M_k^T M_k) \geq \epsilon$, set $I_k = I(x^k, \epsilon)$, $A_k := M_k$ and $\epsilon_k = \epsilon$, stop; otherwise set $\epsilon := (1/2)\epsilon$ and repeat Step (ii).

In order to show some beneficial properties of the pivoting operation above, which is helpful for discussing the convergence of our algorithm, we present the following lemma, and its proof is similar to [3, Lemmata 1.1 and 2.8], so it is omitted here.

LEMMA 2.1. *Suppose that (H1) and (H2) hold, and let $x^k \in R^n$. Then*

- (i) *the pivoting operation can be finished in a finite number of computations, that is, there is no infinite times of loop between Step (ii) and Step (iii);*
- (ii) *if a sequence $\{x^k\}$ of points has an accumulation point, then there exists an $\bar{\epsilon} > 0$ such that the associated sequence $\{\epsilon_k\}$ of parameters generated by pivoting operation satisfies $\epsilon_k \geq \bar{\epsilon}$ for all k .*

For a given iteration point $x^k \in R^n$ and a symmetric positive matrix $H_k = H(x^k)$ (How to choose H_k will be discussed much later), we introduce a new quadratic program as follows:

$$(2.3) \quad \begin{aligned} &\text{minimize } z + \frac{1}{2}d^T H_k d \\ &\text{such that } f_j(x^k) + g_j(x^k)^T d - F(x^k) \leq z, \quad j \in I_k. \end{aligned}$$

To describe the main characters of (2.3), we give two lemmas as follows.

LEMMA 2.2. *Suppose that the matrix H_k is symmetric positive definite. Then*

- (i) *(2.3) has a unique optimal solution;*
- (ii) *(z_k, d^k) is an optimal solution of (2.3) if and only if it is a Karush-Kuhn-Tucker point of (2.3).*

It is not difficult to finish this proof, and is omitted.

LEMMA 2.3. *Suppose that the assumptions (H1) and (H2) hold, and (z_k, d^k) is an optimal solution of (2.3). Then*

- (i) $z_k + (1/2)(d^k)^T H_k d^k \leq 0, \quad z_k \leq 0; \quad d^k = 0 \Leftrightarrow z_k = 0;$
- (ii) $d^k = 0 \Leftrightarrow x^k$ is a stationary point of (P);
- (iii) if $d^k \neq 0$, then $z_k < 0$, moreover, d^k is a descent direction of $F(x)$ at point x^k .

PROOF: (i) From the fact that (0,0) is a feasible solution of (2.3) and H_k is positive definite, one has

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

If $d^k = 0$, then from the constraints of (2.3) we have

$$F(x^k) - f_j(x^k) + z_k \geq 0, \quad j \in I_k.$$

In view of $\phi \neq I(x^k) \subseteq I_k$, one has $z_k \geq 0$. Since that $z_k \leq 0$, we have $z_k = 0$. Conversely, if $z_k = 0$, then $(1/2)(d^k)^T H_k d^k = (1/2)(d^k)^T H_k d^k + z_k \leq 0$, taking into account the positive definite property of H_k , one has $d^k = 0$.

(ii) In view of Lemma 2.2 (ii), we know that the optimal solution (z_k, d^k) of (2.3) is a Karush–Kuhn–Tucker point of (2.3), then there exists a corresponding Karush–Kuhn–Tucker multiplier vector $\lambda^k = (\lambda_j^k, j \in I_k, 0_{I \setminus I_k})$ such that

$$(2.4) \quad \begin{cases} \begin{pmatrix} 1 \\ H_k d^k \end{pmatrix} + \sum_{j=1}^m \lambda_j^k \begin{pmatrix} -1 \\ g_j(x^k) \end{pmatrix} = 0, \\ f_j(x^k) + g_j(x^k)^T d^k - F(x^k) - z_k \leq 0, \quad j \in I_k, \\ (f_j(x^k) + g_j(x^k)^T d^k - F(x^k) - z_k) \lambda_j^k = 0, \quad j \in I_k, \\ \lambda_j^k \geq 0, j \in I_k; \quad \lambda_j^k = 0, \quad j \in I \setminus I_k. \end{cases}$$

If $d^k = 0$, then we get $z_k = 0$ from Lemma 2.3 (i), and we further have from (2.4)

$$(2.5) \quad \begin{cases} \sum_{j=1}^m \lambda_j^k g_j(x^k) = 0, \quad \sum_{j=1}^m \lambda_j^k = 1, \\ f_j(x^k) - F(x^k) \leq 0, \quad j \in I, \\ (f_j(x^k) - F(x^k)) \lambda_j^k = 0, \quad \lambda_j^k \geq 0, \quad j \in I. \end{cases}$$

Hence x^k is a stationary point of (P) from (1.3).

Conversely, if x^k is a stationary point of (P), then $z_k := 0$ together $d^k := 0$ satisfies (2.4) and $(0,0)$ is the unique optimal solution of (2.3) from Lemma 2.2. Therefore $d^k = 0$.

(iii) Using $z_k + (1/2)(d^k)^T H_k d^k \leq 0, d^k \neq 0$ and the positive definite property of the matrix H_k , we know that $z_k < 0$ holds. Furthermore, in view of the constraints of (2.3), one gets

$$g_j(x^k)^T d^k \leq z_k + F(x^k) - f_j(x^k) = z_k < 0, \quad j \in I(x^k).$$

On the other hand, it is easy to know that the directional derivative $F'(x; d)$ of $F(x)$ at point x along direction d can be expressed as

$$(2.6) \quad F'(x; d) = \lim_{\lambda \rightarrow 0^+} \frac{F(x + \lambda d) - F(x)}{\lambda} = \max\{g_j(x)^T d, \quad j \in I(x)\}.$$

Thus

$$(2.7) \quad F'(x^k; d^k) \leq z_k < 0,$$

and d^k is a descent direction of $F(x)$ at point x^k . The whole proof is completed. □

Now we give the details of our algorithm as follows.

ALGORITHM A.

Parameters: $\epsilon_{-1} > 0, \tau \in (2, 3), \alpha \in (0, 0.5), \beta \in (0, 1), \gamma < 0$.

Data: $x^0 \in R^n$, a symmetric positive definite matrix $H_0 \in R^{n \times n}$ (Usually, H_0 is chosen as a unity matrix).

STEP 0. *Initialisation:* Let $k := 0$.

STEP 1. *Pivoting operation:* Set parameter $\varepsilon = \varepsilon_{k-1}$, generate an active constraint set I_k by the pivoting operation and let ε_k be the corresponding termination parameter.

STEP 2. *Quadratic programming:* Solve (2.3) to get a solution (z_k, d^k) with the corresponding Karush–Kuhn–Tucker multiplier vector $\lambda_{I_k}^k = (\lambda_j^k, j \in I_k)$. If $d^k = 0$, then x^k is a stationary point of problem (P) and stop; otherwise, enter Step 3.

STEP 3. *Generate a correction direction \tilde{d}^k :* Compute direction \tilde{d}^k by solving the following system of linear equations

$$(2.8) \quad \begin{bmatrix} \tilde{H}_k & A_k \\ A_k^T & 0 \end{bmatrix} \begin{bmatrix} \tilde{t} \\ \tilde{\gamma} \end{bmatrix} = \begin{bmatrix} 0 \\ -\|d^k\|^r e - \tilde{f}_k \end{bmatrix},$$

where

$$\tilde{H}_k = \begin{bmatrix} H_k & 0 \\ 0 & 1 \end{bmatrix}, \quad \tilde{t} = \begin{bmatrix} \tilde{d} \\ \tilde{z} \end{bmatrix}, \quad e = (1, 1, \dots, 1)^T \in R^{|I_k|}$$

and $\tilde{f}_k = (\tilde{f}_j^k, j \in I_k)$, $\tilde{f}_j^k = f_j(x^k + d^k) - f_j(x^k) - g_j(x^k)^T d^k$, $j \in I_k$. If $\|\tilde{d}^k\| \geq \|d^k\|$, set $\tilde{d}^k = 0$.

STEP 4. *Perform line search:* Compute the step size t_k , the first number t of the sequence $\{1, \beta, \beta^2, \dots\}$ satisfying

$$(2.9) \quad F(x^k + td^k + t^2\tilde{d}^k) \leq \max_{l=0,1,2} F(x^{k-l}) - \alpha t(d^k)^T H_k d^k.$$

STEP 5 *Update:* Generate a new symmetric positive definite matrix H_{k+1} using the formula proposed by [2], set $x^{k+1} = x^k + t_k d^k + t_k^2 \tilde{d}^k$ and $k := k + 1$. Go back to Step 1.

To show that the algorithm is well defined, we introduce the following lemma.

LEMMA 2.4. *The line search in Step 4 can be carried out if $d^k \neq 0$, that is, there exists $\bar{t}_k > 0$ such that (2.9) holds.*

PROOF: By contradiction, we assume that (2.9) does not hold for $\lambda = \beta^j$, $j = 1, 2, \dots$, then from (2.7), (2.6), $\alpha \in (0, 0.5)$, $\beta \in (0, 1)$ and Lemma 2.3 (i), we have

$$\begin{aligned} z_k &\geq F'(x^k; d^k) = \lim_{j \rightarrow \infty} \frac{F(x^k + \beta^j d^k) - F(x^k)}{\beta^j} = \lim_{j \rightarrow \infty} \frac{F(x^k + \beta^j d^k + \beta^{2j} \tilde{d}^k) - F(x^k)}{\beta^j} \\ &\geq \lim_{j \rightarrow \infty} \frac{F(x^k + \beta^j d^k + \beta^{2j} \tilde{d}^k) - \max_{l=0,1,2} F(x^{k-l})}{\beta^j} \\ &\geq - \lim_{j \rightarrow \infty} \alpha (d^k)^T H_k d^k > -\frac{1}{2} (d^k)^T H_k d^k \geq z_k, \end{aligned}$$

which is a contradiction. The proof is completed. □

3. GLOBAL CONVERGENCE

In this section, we shall establish the global convergence of the proposed algorithm. If the solution d^k generated at Step 2 equals zero, then Algorithm A stops at x^k , and from Lemma 2.3 (ii) we know that x^k is a stationary point of the problem (P). And if $d^k \neq 0$, one knows from Lemma 2.3 (iii) that d^k is a descent direction of $F(x)$ at point x^k .

We further assume that an infinite sequence $\{x^k\}$ of points is generated by Algorithm A, and the consequent task is to show that every accumulation point x^* of $\{x^k\}$ is a stationary point of problem (P). Firstly, the following assumption is necessary in the rest of this paper.

(H3) The sequence $\{H_k\}$ of matrices is uniformly positive definite, that is, there exist two positive constants a and b such that

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

(H4) For any $x^0 \in R^n$, the set $\Omega = \{x \in R^n : f(x) \leq f(x^0)\}$ is compact.

In the rest of this paper, we suppose that x^* is a given accumulation point of $\{x^k\}$, therefore, in view of I_k being the subset of the fixed and finite set I and Lemma 2.1, we may assume without loss of generality that there exist an infinite index set K and a constant $\bar{\epsilon} > 0$ such that

$$(3.1) \quad x^k \rightarrow x^*, \quad I(x^k) \equiv I_* \neq \phi, \quad I_k \equiv I', \forall k \in K; \quad \epsilon_k \geq \bar{\epsilon}, \forall k.$$

LEMMA 3.1. ([23]) *The sequence $\{x^k\}$ is bounded and the sequence $\{t_k d^k\}$ and $\{x^{k+1} - x^k\}$ both converge to zero.*

LEMMA 3.2. *Suppose that the assumptions (H1)–(H3) hold. Then*

- (i) *the sequences $\{z_k, k \in K\}$, $\{d^k, k \in K\}$ and $\{\tilde{d}^k, k \in K\}$ are all bounded;*
- (ii) $\lim_{k \in K} d^k = \lim_{k \in K} \tilde{d}^k = 0, \quad \lim_{k \in K} z_k = 0.$

PROOF: (i) Due to the fact that $(0,0)$ is a feasible solution of (2.3), combining assumption (H3) and the constraints of (2.3), we have

$$\begin{aligned} 0 &\geq z_k + \frac{1}{2}(d^k)^T H_k d^k \geq f_j(x^k) + g_j(x^k)^T d^k - F(x^k) + \frac{1}{2}(d^k)^T H_k d^k \\ &= g_j(x^k)^T d^k + \frac{1}{2}(d^k)^T H_k d^k \geq -\|g_j(x^k)\| \cdot \|d^k\| + \frac{1}{2}a\|d^k\|^2, \quad \forall j \in I(x^k), \forall k. \end{aligned}$$

These inequalities show that $\{z_k, k \in K\}$ and $\{d^k, k \in K\}$ are all bounded. Taking into account the definition of \tilde{d}^k at Step 3 of Algorithm A, we can conclude that $\{\tilde{d}^k, k \in K\}$ is bounded.

(ii) Similar to the proof of [23, Theorem 3.1], we can prove $\lim_{k \in K} d^k = 0$. This shows that conclusion (ii) is true. □

LEMMA 3.3. *The whole multiplier sequence $\{\lambda^k = (\lambda_{I_k}^k, 0_{I \setminus I_k})\}$ is bounded.*

PROOF: From (2.4) we get $\sum_{j=1}^m \lambda_j^k = 1$ and $\lambda_j^k \geq 0, j \in I$. Thus sequence $\{\lambda^k\}$ is bounded. □

Now, it is sufficient to establish the following globally convergent theorem for the proposed algorithm.

THEOREM 3.1. *Suppose that the assumptions (H1)–(H4) hold, then the proposed Algorithm A either stops at a stationary point of problem (P) in a finite number of iterations, or generates an infinite sequence $\{x^k\}$ of points such that each accumulation x^* of $\{x^k\}$ is a stationary point of (P).*

The proof is similar to the one of [23, Theorem 3.1].

4. RATE OF CONVERGENCE

Firstly, we give a proposition as follows, which would be useful in the rest discussions.

PROPOSITION 4.1. *Suppose that the assumptions (H1) and (H2) hold. Then the multiplier vector corresponding to a stationary point \tilde{x} of (P) is unique.*

PROOF: Suppose that $\tilde{\lambda}, \tilde{\mu}$ are two multiplier vectors corresponding to a same stationary point \tilde{x} . Then we have from (1.3)

$$\sum_{j \in I(\tilde{x})} \tilde{\lambda}_j \begin{pmatrix} g_j(\tilde{x}) \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \quad \sum_{j \in I(\tilde{x})} \tilde{\mu}_j \begin{pmatrix} g_j(\tilde{x}) \\ -1 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix}, \quad \tilde{\lambda}_{I \setminus I(\tilde{x})} = \tilde{\mu}_{I \setminus I(\tilde{x})} = 0.$$

These two equations give $\sum_{j \in I(\tilde{x})} (\tilde{\mu}_j - \tilde{\lambda}_j) \begin{pmatrix} g_j(\tilde{x}) \\ -1 \end{pmatrix} = 0$. Then $\tilde{\lambda} = \tilde{\mu}$ follows from (H2). □

THEOREM 4.1. *Suppose that the assumptions (H1)–(H4) hold. Then*

$$\lim_{k \rightarrow \infty} d^k = \lim_{k \rightarrow \infty} \tilde{d}^k = 0, \quad \lim_{k \rightarrow \infty} z_k = 0.$$

The proof is similar to the one of [23, Theorem 3.1].

In order to obtain the superlinearly convergent rate of the proposed algorithm, from Proposition 4.1 above, we further suppose that the following assumption holds.

(H5)

- (i) The functions $f_j(x)$ ($j \in I$) are all twice continuously differentiable for any $x \in R^n$;
- (ii) The sequence $\{x^k\}$ generated by Algorithm A possesses an accumulation point x^* with the corresponding unique multipliers μ^* (by Theorem 3.1, x^* is a stationary point of problem (P)), such that the stationary point pair

(x^*, μ^*) of problem (P) satisfies the following strong second-order sufficiency conditions for some index $t_0 \in I_+^*$

$$d^T \nabla_{xx}^2 L(x^*, \mu^*) d > 0, \quad \forall d \in R^n : d \neq 0, \quad (g_j(x^*) - g_{t_0}(x^*))^T d = 0, \quad j \in I_+^*,$$

where

$$\nabla_{xx}^2 L(x^*, \mu^*) = \sum_{j \in I} \mu_j^* \nabla^2 f_j(x^*) = \sum_{j \in I_+^*} \mu_j^* \nabla^2 f_j(x^*), \quad I_+^* \triangleq \{j \in I : \mu_j^* > 0\}.$$

REMARK 2. Since the assumption (H5) above does not include the so-called strict complementary condition, namely $\mu_j^* > 0$ for all $j \in I(x^*)$, which is assumed to hold in assumption in [23, A5] and [22, Assumption 3.4], it is a weaker hypothesis.

Now we prove that x^* is an isolated stationary point of (P) under certain conditions.

LEMMA 4.1. *Suppose that assumptions (H2) and (H5) hold. Then x^* is an isolated stationary point of (P).*

The proof is similar to the one of [6, Theorem 1.2.5].

THEOREM 4.2. *Suppose that the assumptions (H2)–(H5) hold. Then $\lim_{k \rightarrow \infty} x^k = x^*$.*

PROOF: From Lemma 4.1, we know that x^* is an isolated stationary point of (P). Furthermore, one can conclude x^* is an isolated limit point of $\{x^k\}$ and this together with Theorem 4.1 (ii) implies $\lim_{k \rightarrow \infty} x^k = x^*$ (see [6, Theorem 1.1.5]). □

LEMMA 4.2. *Under all the above-mentioned assumptions, when k is sufficiently large, the matrix*

$$M_k \stackrel{\text{def}}{=} \begin{bmatrix} \tilde{H}_k & A_k \\ A_k^T & 0 \end{bmatrix}.$$

is nonsingular, furthermore, there exists a constant $C > 0$ such that $\|M_k^{-1}\| \leq C$.

The proof of Lemma 4.2 is similar to that of [6, Lemma 2.2.2], and is omitted.

LEMMA 4.3. *Suppose that the assumptions (H2) and (H3) hold. Then $\|\tilde{d}^k\| = O(\|d^k\|^2)$.*

PROOF: Taking into account Taylor expansion, the definition of \tilde{f}_j^k and Theorem 4.1 (i), we obtain

$$\tilde{f}_j^k = f_j(x^k + d^k) - f_j(x^k) - g_j(x^k)^T d^k = O(\|d^k\|^2).$$

So, by using the Lemma 4.2, $\tau \in (2, 3)$ and (2.8), we have $\|\tilde{d}^k\| = O(\|d^k\|^2)$. □

LEMMA 4.4. *If assumptions (H2)–(H5) are all satisfied, then the Karush–Kuhn–Tucker multiplier $\lambda_{I_k}^k$ of (2.3) corresponding to (z_k, d^k) satisfies $\lim_{k \rightarrow \infty} \lambda^k = \mu^*$ with $\lambda^k = (\lambda_{I_k}^k, 0_{I \setminus I_k})$.*

PROOF: We assume by contradiction that $\lim_{k \rightarrow \infty} \lambda^k \neq \mu^*$, then there exists an infinite subset K and a constant $\bar{\alpha} > 0$ such that

$$\|\lambda^k - \mu^*\| \geq \bar{\alpha}, k \in K.$$

In view of $\lim_{k \rightarrow \infty} x^k = x^*$ and the boundedness of $\{\lambda^k\}$, there exists another infinite set $K' \subseteq K$ such that

$$(4.1) \quad x^k \rightarrow x^*, \quad \|\lambda^k - \mu^*\| \geq \bar{\alpha}, \quad \lambda^k \rightarrow \lambda^*, k \in K' \subseteq K.$$

Taking into account of Theorem 4.1 and passing to the limit $k \in K'$ and $k \rightarrow \infty$ in (2.4), we have

$$\begin{cases} \begin{pmatrix} 1 \\ 0 \end{pmatrix} + \sum_{j=1}^m \lambda_j^* \begin{pmatrix} -1 \\ g_j(x^*) \end{pmatrix} = 0, \\ f_j(x^*) - F(x^*) \leq 0, \quad (f_j(x^*) - F(x^*))\lambda_j^* = 0, \quad \lambda_j^* \geq 0, \quad j \in I. \end{cases}$$

From above we know that (x^*, λ^*) is a stationary point pair of (P), thus $\lambda^* = \mu^*$ (since the multiplier vector is unique), which contradicts (4.1). So the whole proof is finished. \square

In order to obtain the superlinearly convergent rate of the proposed algorithm, first of all, we should guarantee that the step size unit is accepted by the line search for k large enough. For this purpose, the following additional assumptions are necessary.

(H6) Suppose that $\|(\nabla_{xx}^2 L(x^k, \lambda^k) - H_k)d^k\| = o(\|d^k\|)$.

THEOREM 4.3. *Under all above assumptions (H2)–(H6), the step size in Algorithm A always equals to one, that is, $t_k \equiv 1$, if k is sufficient large.*

PROOF: It is sufficient to show that (2.9) holds for $t = 1$. Firstly, in view of Taylor expansion, Theorems 4.1, 4.2 and Lemma 4.2, we get

$$(4.2) \quad \begin{aligned} f_i(x^k + d^k + \tilde{d}^k) &= f_i(x^k + d^k) + g_i(x^k + d^k)^T \tilde{d}^k + O(\|\tilde{d}^k\|^2) \\ &= f_i(x^k + d^k) + g_i(x^k)^T \tilde{d}^k + O(\|d^k\|^3), i \in J_k. \end{aligned}$$

Where $J_k = \{i \mid f_j(x^k) + g_j(x^k)^T d^k - F(x^k) = z_k\}$.

From (2.8), we also have

$$(4.3) \quad A_k^T \begin{pmatrix} \tilde{d}^k \\ \tilde{z}_k \end{pmatrix} = -\|d^k\|^\tau e - \tilde{f}^k, \quad g_i(x^k)^T \tilde{d}^k = \tilde{z}_k - \|d^k\|^\tau - \tilde{f}_i^k, i \in J_k \subseteq I_k.$$

Hence from (4.2), (4.3) and the definition of \tilde{f}_i^k , we have

$$\begin{aligned} f_i(x^k + d^k + \tilde{d}^k) &= f_i(x^k + d^k) + \tilde{z}_k - \|d^k\|^\tau - \tilde{f}_i^k + O(\|d^k\|^3) \\ &= \tilde{z}_k - \|d^k\|^\tau + f_i(x^k) + g_i(x^k)^T d^k + O(\|d^k\|^3) \\ &= F(x^k) + z_k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3), \quad i \in J_k \subseteq I_k. \end{aligned}$$

So

$$f_j(x^k + d^k + \tilde{d}^k) = F(x^k) + z_k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3), j \in J_k.$$

Thus from the two formulas above, we obtain

$$(4.4) \quad f_i(x^k + d^k + \tilde{d}^k) = f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3), \forall i, j \in J_k.$$

Now we shall show that for k large enough

$$F(x^k + d^k + \tilde{d}^k) = f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3), \forall j \in J_k.$$

In view of $I(x^k + d^k + \tilde{d}^k) \subseteq I(x^*)$, $J_k \subseteq I(x^*) \subseteq I_k$ for k large enough, one has

$$F(x^k + d^k + \tilde{d}^k) = \max\{f_i(x^k + d^k + \tilde{d}^k), i \in J_k, f_i(x^k + d^k + \tilde{d}^k), i \in I(x^*) \setminus J_k\}.$$

CASE A. If there exists an index $j_0 \in J_k$ such that $F(x^k + d^k + \tilde{d}^k) = f_{j_0}(x^k + d^k + \tilde{d}^k)$, then from (4.4) we have

$$F(x^k + d^k + \tilde{d}^k) = f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3), \forall j \in J_k.$$

CASE B. If $f_j(x^k + d^k + \tilde{d}^k) < F(x^k + d^k + \tilde{d}^k), \forall j \in J_k$, then there exists an index $j_1 \in I(x^*) \setminus J_k \subseteq I_k \setminus J_k$ such that

$$F(x^k + d^k + \tilde{d}^k) = f_{j_1}(x^k + d^k + \tilde{d}^k).$$

From Taylor expansion, we have

$$f_{j_1}(x^k + d^k + \tilde{d}^k) = f_{j_1}(x^k + d^k) + g_{j_1}(x^k)^T \tilde{d}^k + O(\|d^k\|^3),$$

In view of (2.8), similar to (4.3), one gets

$$g_{j_1}(x^k)^T \tilde{d}^k = \tilde{z}_k - \|d^k\|^\tau - \tilde{f}_{j_1}^k, j_1 \in I(x^*) \setminus J_k \subseteq I_k \setminus J_k.$$

Thus

$$\begin{aligned} f_{j_1}(x^k + d^k + \tilde{d}^k) &= f_{j_1}(x^k + d^k) + \tilde{z}_k - \|d^k\|^\tau + f_{j_1}(x^k) + g_{j_1}(x^k)^T d^k \\ &\quad - f_{j_1}(x^k + d^k) + O(\|d^k\|^3) \\ &= f_{j_1}(x^k) + g_{j_1}(x^k)^T d^k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3) \\ &< F(x^k) + z_k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3), \end{aligned}$$

that is

$$F(x^k + d^k + \tilde{d}^k) < F(x^k) + z_k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3), i \in J_k,$$

combining

$$f_i(x^k + d^k + \tilde{d}^k) = F(x^k) + z_k + \tilde{z}_k - \|d^k\|^\tau + O(\|d^k\|^3),$$

imply that

$$F(x^k + d^k + \tilde{d}^k) = f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3), \forall j \in J_k.$$

On the other hand, from Theorem 4.1 it is easy to get $J_k \subseteq I(x^*)$ for k large enough. Thus, from (2.4), Lemma 3.2, Taylor expansion and Lemma 4.2, we get

$$\sum_{j \in J_k} \lambda_j^k = 1, \quad \lambda_j^k F(x^k + d^k + \tilde{d}^k) = \lambda_j^k f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3), \quad j \in J_k,$$

$$\begin{aligned} & F(x^k + d^k + \tilde{d}^k) \\ (4.5) \quad &= \sum_{j \in J_k} \lambda_j^k F(x^k + d^k + \tilde{d}^k) = \sum_{j \in J_k} \lambda_j^k f_j(x^k + d^k + \tilde{d}^k) + O(\|d^k\|^3) \\ &= \sum_{j \in J_k} \lambda_j^k (f_j(x^k) + g_j(x^k)^T (d^k + \tilde{d}^k) + \frac{1}{2} (d^k)^T \nabla^2 f_j(x^k) d^k) + o(\|d^k\|^2). \end{aligned}$$

Also, from (2.4) and Lemma 4.2, one has

$$(4.6) \quad \sum_{j \in J_k} \lambda_j^k g_j(x^k)^T (d^k + \tilde{d}^k) = -(d^k)^T H_k d^k + o(\|d^k\|^2),$$

and

$$(4.7) \quad \sum_{j \in J_k} \lambda_j^k f_j(x^k) \leq \sum_{j \in J_k} \lambda_j^k F(x^k) = F(x^k).$$

So, from (4.5)–(4.7), (H3) and (H5), we have

$$\begin{aligned} & F(x^k + d^k + \tilde{d}^k) \\ & \leq \max_{l=0,1,2} F(x^{k-l}) - (d^k)^T H_k d^k + \frac{1}{2} (d^k)^T \left(\sum_{j \in J_k} \lambda_j^k \nabla^2 f_j(x^k) \right) d^k + o(\|d^k\|^2) \\ & = \max_{l=0,1,2} F(x^{k-l}) - \frac{1}{2} (d^k)^T H_k d^k + \frac{1}{2} (d^k)^T \left(\sum_{j \in J_k} \lambda_j^k \nabla^2 f_j(x^k) - H_k \right) d^k + o(\|d^k\|^2) \\ & = \max_{l=0,1,2} F(x^{k-l}) - \alpha (d^k)^T H_k d^k + \left(\alpha - \frac{1}{2} \right) (d^k)^T H_k d^k + o(\|d^k\|^2) \\ & \leq \max_{l=0,1,2} F(x^{k-l}) - \alpha (d^k)^T H_k d^k + \left(\alpha - \frac{1}{2} \right) a \| (d^k) \|^2 + o(\|d^k\|^2). \end{aligned}$$

Taking into account $\alpha \in (0, 1/2)$, we have for k large enough

$$F(x^k + d^k + \tilde{d}^k) \leq \max_{l=0,1,2} F(x^{k-l}) - \alpha (d^k)^T H_k d^k,$$

that is (2.9) holds for $t = 1$ and k large enough. So the whole proof is finished. □

To analyse the superlinear convergence, we further give a lemma as follows.

LEMMA 4.5. *Suppose that the assumptions (H2)–(H5) hold and let*

$$R_k = R(x^k) = (g_j(x^k) - g_{t_0}(x^k), j \in J_k \setminus \{t_0\}), \quad P_k = E_n - R_k(R_k^T R_k)^{-1} R_k^T.$$

Then for k large enough the following matrix

$$(4.8) \quad G_k = \begin{pmatrix} P_k \nabla_{xx}^2 L(x^*, \mu^*) & R_k \\ R_k^T & 0 \end{pmatrix}$$

is nonsingular and there exists a constant c such that $\|G_k^{-1}\| \leq c$.

The proof of this Lemma is similar to the one of Lemma 4.2, and is omitted.

THEOREM 4.4. *Let the assumptions (H2)–(H6) be satisfied. Then the proposed algorithm A is superlinearly convergent, that is, the sequence $\{x^k\}$ generated by algorithm A satisfies*

$$\|x^{k+1} - x^*\| = o(\|x^k - x^*\|).$$

The proof is similar to the one of [6, Theorem 2.2.3].

5. NUMERICAL EXPERIMENTS

In this section, we test some practical problems based on the proposed algorithm. The numerical experiments are implemented on MATLAB 6.5, under Windows XP and 1000MHZ CPU. The (2.3) and (2.8) are solved by the Optimisation Toolbox.

A slight modification of the Broyden–Fletcher–Goldfarb–Shanno formula, which is proposed in [2], is adopted in the algorithm.

$$(5.1) \quad H_{k+1} = H_k - \frac{H_k s^k (s^k)^T H_k}{(s^k)^T H_k s^k} + \frac{\tilde{y}^k (\tilde{y}^k)^T}{(s^k)^T \tilde{y}^k}, \quad (k \geq 0)$$

where

$$s^k = x^{k+1} - x^k, \quad \tilde{y}^k = y^k + a_k (\gamma_k s^k + \bar{A}_k \bar{A}_k^T s^k), \quad \gamma_k = \min\{\|d_0^k\|^2, \xi\}, \quad \xi \in (0, 1)$$

$$y^k = \nabla_x L(x^{k+1}, \lambda^k) - \nabla_x L(x^k, \lambda^k), \quad \bar{A}_k = (g_j(x^k), j \in I(x^k)),$$

$$\nabla_x L(x, \lambda) = \sum_{j \in I(x)} \lambda_j g_j(x),$$

$$a_k = \begin{cases} 0, & \text{if } (s^k)^T y^k \geq \delta \|s^k\|^2, \delta \in (0, 1); \\ 1, & \text{if } 0 \leq (s^k)^T y^k < \delta \|s^k\|^2; \\ 1 + \frac{\gamma_k \|s^k\|^2 - (s^k)^T y^k}{\gamma_k \|s^k\|^2 + (s^k)^T \bar{A}_k (\bar{A}_k)^T s^k}, & \text{otherwise.} \end{cases}$$

During the numerical experiments, we set

$$H_0 = I, \quad \tau = 2.5, \quad \beta = 0.6, \quad \alpha = 0.45, \quad \varepsilon_{-1} = 2.1,$$

where I is an unity matrix. The test problem in Table 5.1 are selected from [21] and [19]. The initial points for the selected problems are as same as the ones in [21] and [19]. The columns of Table 5.1 have the following meanings: The **prob** column lists the test problem taken from [21] and [19]. The columns labelled **Ni** give the number of iterations required to solve the problem. The columns labelled **objective**, **dnorm** and **eps** denote the final objective value, the norm of d^k and the step criterion threshold ε , respectively.

The detailed information of the solutions to the test problems is listed in the following Table 5.1. It can be seen from Table 5.1 that the proposed algorithm is effective, since it can successfully reach a near-optimal point for all the tested problems. However, we also observe that the numerical results reported here have more number of function evaluations than those reported in [21], in that our algorithm needs the PIVOTING technique which results in a reduction of the size of the quadratic programming subproblem.

Table 5.1 Numerical results

Prob	Ni	objective	dnorm	eps
1	10	1.9522	4.2478e-006	0.1e-04
2	31	2.0000	1.9866e-015	0.1e-04
3	11	-44.0000	8.2939e-006	0.1e-04
4	22	0.6164	8.7974e-006	0.1e-04
5	15	3.5997	1.8327e-006	0.1e-04
8	23	680.6301	1.5379e-006	0.1e-04
9	34	24.3062	2.5866e-006	0.1e-04
Vardi-3	10	-48.0158	4.5843e-008	0.1e-04

6. CONCLUDING REMARKS

In this paper, we propose a nonmonotone descent sequential quadratic programming algorithm for nonlinear minimax problems. At each iteration, by solving a reduced quadratic programming subproblem, a main search direction is obtained. Then we correct the main search direction by solving a reduced system of linear equation. A nonmonotone line search is performed on $F(x)$ to obtain the next iteration point. Under weaker conditions *without the strict complementarity*, the global and one-step superlinear convergent properties are obtained. Preliminary numerical results also show that the proposed algorithm is effective.

REFERENCES

- [1] A.R. Conn and Y. Li, 'An efficient algorithm for nonlinear minimax problems', Report CS-88-41, University of Waterloo, Waterloo, Ontario, Canada.
- [2] J.F.A. De, O. Pantoja and D.Q. Mayne, 'Exact penalty function algorithm with simple updating of the penalty parameter', *J. Optim. Theory Appl.* **69** (1991), 441–467.

- [3] Z.-Y. Gao, G.-P. He and F. Wu, 'A method of sequential systems of linear equations with arbitrary initial point', *Sci. China Ser. A* **27** (1997), 24–33.
- [4] S.P. Han, 'A globally convergent method for nonlinear programming', *J. Optim. Theory Appl.* **22** (1977), 297–309.
- [5] S.P. Han, 'Variable metric methods for minimizing a class of nondifferentiable functions', *Math. Programming* **20** (1981), 1–13.
- [6] J.-B. Jian, *Researches on superlinearly and quadratically convergent algorithms for nonlinearly constrained optimization*, (Ph. D. Thesis) (School of Xi'an Jiaotong University, Xi'an, China, 2000).
- [7] J.-B. Jian and C.-M. Tang, 'An SQP feasible descent algorithm for nonlinear inequality constrained optimization without strict complementarity', *Comput. Math. Appl.* **49** (2005), 223–238.
- [8] J.-B. Jian, K.-C. Zhang and S.-J. Xue, 'A superlinearly and quadratically convergent SQP type feasible method for constrained optimization', *Appl. Math. J. Chinese Univ. Ser. B* **15** (2000), 319–331.
- [9] L. Luskan, 'A compact variable metric algorithm for nonlinear minimax approximation', *Computing* **36** (1986), 355–373.
- [10] D.Q. Mayne and E. Polak, 'A superlinearly convergent algorithm for constrained optimization problems', *Math. Programming Stud.* **16** (1982), 45–61.
- [11] D.Q. Mayne, E. Polak and A. Sangiovanni-Vincenteli, 'Computer-aided design via optimization: A review', *Automatica* **18** (1982), 147–154.
- [12] E.R. Panier and A.L. Tits, 'On combining feasibility, descent and superlinear convergence in inequality constrained optimization', *Math. Programming* **59** (1993), 261–276.
- [13] R.A. Polyak, 'Smooth optimization methods for minimax problems', *SIAM J. Control Optim.* **26** (1988), 1274–1286.
- [14] E. Polak, D.Q. Mayne and J.E. Higgins, 'A Superlinearly convergent algorithm for minimax problems', Proceedings of the 28th IEEE Conference on Decision and Control, Tampa, Florida **1** (1989), 894–898.
- [15] E. Polak, D.Q. Mayne and J.E. Higgins, 'Superlinearly convergent algorithm for min-max problems', *J. Optim. Theory Appl.* **69** (1991), 407–439.
- [16] E. Polak, D.Q. Mayne and D.M. Stimler, 'Control system design via semi-infinite optimization: A review', Proceedings of the IEEE **72** (1984), 1777–1794.
- [17] M.J.D. Powell, 'A fast algorithm for nonlinearly constrained optimization calculation', in *Numerical Analysis (Proc. 7th Biennial Conf., Univ. Dundee, 1977)*, Lecture Notes in Math. **630** (Springer-Verlag, Berlin, 1978), pp. 144–157.
- [18] S.E. Salcudean, *Algorithms for optimal design of feedback compensators*, (Ph.D. Thesis) (Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, 1986).
- [19] A. Vardi, 'New minimax algorithm', *J. Optim. Theory Appl.* **75** (1992), 613–634.
- [20] T.L.S. Wu, *Delightmimo: An interactive system for optimization-based multivariable control system design*, Ph.D. Thesis (Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, CA, 1986).
- [21] X. Yi, 'The sequential quadratic programming method for solving minimax problem', *J. Systems Sci. Math. Sci.* **22** (2002), 355–364.

- [22] Y.H. Yu and L. Gao, 'Nonmonotone line search algorithm for constrained minimax problems', *J. Optim. Theory Appl.* **115** (2002), 419–446.
- [23] J.L. Zhou and A.L. Tits, 'Nonmonotone line search for minimax problems', *J. Optim. Theory Appl.* **76** (1993), 455–476.

Department of Information
Hunan Business College
410205, Changsha
Peoples Republic China
e-mail: hqj0525@126.com.cn

Institute of Applied Mathematics
Hunan University
410082, Changsha
Peoples Republic of China