BULL. AUSTRAL. MATH. SOC. 90C30, 65K05
VOL. 73 (2006) [117-127]

GENERALISED MONOTONE LINE SEARCH ALGORITHM FOR DEGENERATE NONLINEAR MINIMAX PROBLEMS

JIN-BAO JIAN, RAN QUAN AND XUE-LU ZHANG

In this paper, nonlinear minimax problems are discussed. Using Sequential Quadratic Programming and the generalised monotone line search technique, we propose a new algorithm for solving degenerate minimax problems. At each iteration of the proposed algorithm, a search direction is obtained by solving a new Quadratic Programming problem which always has a solution. Global convergence can be obtained without the regularity condition of linear independence. Finally, some numerical experiments are reported.

1. INTRODUCTION

In this paper we discuss the minimax problem as follows

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

where $F(x) = \max\{f_j(x), j \in I\}$ with $I = \{1, 2, \ldots, m\}$.

The objective function $F(x)$ is continuous but nondifferentiable even when the $f_j(x)(j = 1, \ldots, m)$ are all differentiable, thus the methods for smooth optimisation problems can not be used directly to solve this kind of nonlinear minimax problem.

Since the Sequential Quadratic Programming method has fast convergence, it has been studied by many authors (see [4, 2]). Several authors have used the idea of the Sequential Quadratic Programming method to solve the minimax problems (see [9, 7]). In [9], to overcome the Maratos effect [3], a correction direction is obtained by solving a Quadratic Programming problem and the nonmonotone line search technique is used, that is, it does not require decreasing of the merit function at every iteration but merely every M iterations. In 1992, combining with trust-region method, Varid [6] uses some slack variables to solve the equivalent problem to the minimax problem (P) as follows

$$\text{(1.1)} \qquad \begin{array}{ll} \min & s \\ \text{subject to} & f_j(x) - s \leqslant 0, \quad j \in I, \end{array}$$

Received 10th October, 2005
Project supported by the National Natural Science Foundation (No.10261001) and Guangxi Science Foundation (No.0236001) of China.

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

and the algorithm has global convergence. Xue and Yang [8] propose another globally convergent method for solving the minimax problem (P), in which the search direction is obtained by explicit formulas with generalised projection technique. But the condition of linear independence is used in all the papers [9, 7, 6, 8].

Obviously, the Karush-Kuhn-Tucker conditions of (1.1) can be stated as follows

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

and these relationships are equivalent to

$$\sum_{j\in I}\lambda_j\nabla f_j(x)=0,\quad \sum_{j\in I}\lambda_j=1;\quad \lambda_j\big(f_j(x)-F(x)\big)=0,\quad \lambda_j\geqslant 0,\quad j\in I, \tag{1.2}$$

where $w\perp y$ indicates orthogonality of any vectors w and y. So, a point $x\in R^n$ is said to be a stationary point of (P) ([1]) if there exists a vector $\lambda=(\lambda_j, j\in I)$ such that (1.2) holds, where λ is said to be multiplier vector.

Based on the equivalent relationship between the Karush–Kuhn–Tucker point of (1.1) and the stationary point of (P), many algorithms focus on finding the stationary point of (P), namely solving (1.2). Thus, deriving from the idea of nonmonotone line search in [9], we propose a new generalised monotone line search algorithm for (P) without linear independence. To get a search direction and reduce the computational cost, basing on an $\varepsilon-$active constraint subset, we construct a new Quadratic Programming subproblem with parameters $\gamma_0, \gamma_j(j\in I)$, which always has a feasible solution and possesses small size, and the parameters are used to speed up the convergence. By solving the Quadratic Programming subproblem, we get a direction of descent. Then we present our "generalised monotone" line search algorithm, that is, the merit function is forced to decrease at every $r+1$ iterations, where $r(\geqslant 0)$ is a integer. If $r=0$, then our algorithm is a usual monotone algorithm, else, it is an r monotone algorithm. Under mild conditions without linear independence, the global convergence can be obtained. At last, some numerical experiments are reported.

2. Algorithm

For convenience of presentation, we use the following notations throughout the remainder of this paper

$$I(x)=\big\{j\in I: f_j(x)=F(x)\big\},\quad g_j(x)=\nabla f_j(x), j\in I,$$

and assume that the following assumption holds in this paper.

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

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

For a given iteration point $x^k \in R^n$ and a symmetric positive definite matrix $H_k = H(x^k)$, we introduce a new Quadratic Programming subproblem as follows:

$$(2.1) \qquad QP(x^k, H_k) \qquad \begin{array}{ll} \min & \gamma_0 z + \frac{1}{2} d^T H_k d \\ \text{subject to} & f_j(x^k) + g_j(x^k)^T d - F(x^k) \leqslant \gamma_j z, \quad j \in I^k_{\varepsilon_k}, \end{array}$$

where $I^k_{\varepsilon_k} = \{j \in I : F(x^k) - f_j(x^k) \leqslant \varepsilon_k\}$ and $\varepsilon_k, \gamma_0, \gamma_j (j \in I^k_{\varepsilon_k})$ are all positive constants. To describe the main characters of (2.1), we give two lemmas as follows.

Lemma 2.1. *Suppose that the matrix H_k is symmetric positive definite. Then*

(i) (2.1) *has a unique optimal solution;*

(ii) (z_k, d^k) *is an optimal solution of* (2.1) *if and only if it is a Karush-Kuhn-Tucker point of* (2.1).

Proof: (i) obviously, $(\overline{z}, \overline{d})=(0,0) \in R^{n+1}$ is a feasible solution of (2.1) since $F(x^k) = \max\{f_j(x^k), j \in I\}$, so the feasible set of (2.1) is not empty. On the other hand. for each feasible solution (z, d) of (2.1), we know that the objective function value of (2.1) satisfies the following inequality:

$$\gamma_0 z + \frac{1}{2} d^T H_k d \geqslant \frac{\gamma_0}{\gamma_j}\left(f_j(x^k) + g_j(x^k)^T d - F(x^k)\right) + \frac{1}{2} d^T H_k d, \quad j \in I^k_{\varepsilon_k} \neq \phi.$$

Therefore, the objective function value of (2.1) is bounded from below because of the positive definite property of H_k. Furthermore, it is not difficult to show that (2.1) always has an optimal solution. In addition, problem (2.1) is equal to the following unconstrained optimisation

$$\min\left\{\max\left\{\frac{\gamma_0}{\gamma_j}\left(f_j(x^k) + g_j(x^k)^T d - F(x^k)\right), j \in I^k_{\varepsilon_k}\right\} + \frac{1}{2} d^T H_k d\right\}.$$

Obviously, the first term of the objective function is convex and the second term is strictly convex. Thus the objective function of the problem above is strictly convex and its optimal solution is unique. Therefore the optimal solution of (2.1) is unique.

(ii) If (z_k, d^k) is a Karush-Kuhn-Tucker point of (2.1), then it is an optimal solution of (2.1) since (2.1) is a convex programming. Conversely, if (z_k, d^k) is an optimal solution of (2.1), note that the Abadie Constraint Qualification holds since the constraints of (2.1) are all linear, then (z_k, d^k) is a Karush-Kuhn-Tucker point of (2.1). The proof is completed. $\square$

Lemma 2.2. *Suppose that H1 holds, the matrix H_k is positive definite and (z_k, d^k) is an optimal solution of* (2.1). *Then*

(i) $\gamma_0 z_k + \left((d^k)^T/2\right) H_k d^k \leqslant 0, \ z_k \leqslant 0; \ d^k = 0 \Leftrightarrow z_k = 0;$

(ii) *if $d^k = 0$, then 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.*

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

PROOF: (i) From the fact that (0,0) is a feasible solution of (2.1) and H_k is positive definite, one has

$$\gamma_0 z_k + \frac{1}{2}(d^k)^T H_k d^k \leqslant 0, \quad z_k \leqslant -\frac{1}{2\gamma_0}(d^k)^T H_k d^k \leqslant 0.$$

If $d^k = 0$, then from the constraints of (2.1) we have

$$F(x^k) - f_j(x^k) + \gamma_j z_k \geqslant 0, \quad j \in I^k_{\varepsilon_k}.$$

Therefore, in view of $\phi \neq I(x^k) \subseteq I^k_{\varepsilon_k}$, one has $z_k \geqslant 0$. Since $z_k \leqslant 0$, we know $z_k = 0$. Conversely, if $z_k = 0$, then

$$\big((d^k)^T/2\big) H_k d^k = 0.$$

Taking into account the positive definite property of H_k, one has $d^k = 0$.

(ii) In view of Lemma 2.1 (ii), we know that the optimal solution (z_k, d^k) of (2.1) is a Karush-Kuhn-Tucker point of (2.1), then there exists a corresponding Karush-Kuhn-Tucker multiplier vector

$$\lambda^k = (\lambda^k_j, j \in I^k_{\varepsilon_k}, 0_{I \setminus I^k_{\varepsilon_k}})$$

such that

$$(2.2) \qquad \begin{cases} \begin{pmatrix} \gamma_0 \\ H_k d^k \end{pmatrix} + \sum\limits_{j=1}^{m} \lambda^k_j \begin{pmatrix} -\gamma_j \\ g_j(x^k) \end{pmatrix} = 0, \\ f_j(x^k) + g_j(x^k)^T d^k - F(x^k) - \gamma_j z_k \leqslant 0, \quad j \in I^k_{\varepsilon_k}, \\ \big(f_j(x^k) + g_j(x^k)^T d^k - F(x^k) - \gamma_j z_k\big) \lambda^k_j = 0, \quad j \in I^k_{\varepsilon_k}, \\ \lambda^k_j \geqslant 0, j \in I^k_{\varepsilon_k}; \quad \lambda^k_j = 0, \quad j \in I \setminus I^k_{\varepsilon_k}. \end{cases}$$

If $d^k = 0$, then we get $z_k = 0$ from Lemma 2.2 (i), and from the first equality of (2.2) we have $\sigma_k \triangleq \sum\limits_{j=1}^{m} \lambda^k_j > 0$ and

$$(2.3) \qquad \begin{cases} \sum\limits_{j=1}^{m} \dfrac{\lambda^k_j}{\sigma_k} g_j(x^k) = 0, \quad \sum\limits_{j=1}^{m} \dfrac{\lambda^k_j}{\sigma_k} = 1, \\ 0 \leqslant \dfrac{\lambda^k_j}{\sigma_k} \perp \big(f_j(x^k) - F(x^k)\big) \leqslant 0, \quad j \in I. \end{cases}$$

Hence x^k is a stationary point of (P) from (1.2).

(iii) Using $\gamma_0 z_k + \big((d^k)^T/2\big) H_k d^k \leqslant 0, d^k \neq 0$ and the positive definite property of the matrix H_k, we know that $z_k < 0$ holds. Furthermore, in view of the constraints of (2.1), one gets

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

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

$$(2.4) \qquad F'(x;d) = \lim_{\lambda\to 0+} \frac{F(x+\lambda d) - F(x)}{\lambda} = \max\{g_j(x)^T d, \ \ j \in I(x)\}.$$

Thus there exists an index $j_{k_0} \in I(x^k)$ such that

$$(2.5) \qquad F'(x^k; d^k) = g_{j_{k_0}}(x^k)^T d^k \leqslant \gamma_{j_{k_0}} z_k < 0,$$

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

Now we give our algorithm as follows.

ALGORITHM A. *Parameters*: integer $r \geqslant 0$, $\varepsilon_0 > 0$, $\tau \in (2,3)$, $\alpha \in (0,0.5)$, $\beta \in (0,1)$, $\xi_1 > 0$, $\xi_2 \geqslant 0$, $\gamma_0 > 0$, $\gamma_j > 0, j \in I$, where $\gamma_0\xi_1 + \alpha\xi_2 \leqslant \gamma = \min\{\gamma_j : j \in I\}$.

Data: $x^{-r} = x^{-r+1} = \cdots = x^0 \in R^n$, a symmetric positive definite matrix $H_0 \in R^{n\times n}$.

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

STEP 1. *Solve* (Quadratic Programming.) Solve (2.1) to get a (unique) solution (z_k, d^k). If $d^k = 0$, stop; otherwise, enter Step 2.

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

$$(2.6) \qquad F(x^k + td^k) \leqslant F_k - \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k),$$

where $F_k = \max\{F(x^{k-l}) : l = 0, 1, 2, \ldots, r\}$.

STEP 3. *Update*: Compute ε_{k+1} and a new symmetric positive definite matrix H_{k+1}, set $x^{k+1} = x^k + t_k d^k$ and $k := k+1$, go back to Step 1.

The following lemma shows that the algorithm is well defined.

LEMMA 2.3. *The line search at Step 2 can be carried out if $d^k \neq 0$, that is, inequality* (2.6) *holds for $t > 0$ sufficient small.*

PROOF: By contradiction, we assume that (2.6) does not hold for $\lambda = \beta^j, j = 1, 2, \ldots$. Then from (2.5), (2.4), $\alpha \in (0,0.5)$, $\beta \in (0,1)$ and Lemma 2.2 (i), we have

$$\begin{aligned} z_k &\geqslant \frac{1}{\gamma_{j_{k_0}}} F'(x^k; d^k) = \frac{1}{\gamma_{j_{k_0}}} \lim_{j\to\infty} \frac{F(x^k + \beta^j d^k) - F(x^k)}{\beta^j} \\ &\geqslant -\frac{1}{\gamma_{j_{k_0}}} \lim_{j\to\infty} \alpha(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k) \\ &\geqslant -\frac{1}{\gamma}\alpha\xi_1 (d^k)^T H_k d^k + \frac{1}{\gamma}\alpha\xi_2 z_k > \frac{\gamma_0\xi_1}{\gamma} z_k + \frac{\alpha\xi_2}{\gamma} z_k = \frac{\gamma_0\xi_1 + \alpha\xi_2}{\gamma} z_k \geqslant z_k, \end{aligned}$$

which is a contradiction. The proof is finished. $\square$

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

REMARK 2.1. The inequality (2.6) is equivalent to

$$f_j(x^k + td^k) \leqslant F_k - \alpha t\big(\xi_1(d^k)^T H_k d^k - \xi_2 z_k\big), \quad \forall j \in I.$$

3. Global convergence analysis

In this section, we shall discuss the global convergence of the proposed algorithm. If the solution d^k generated at Step 1 equals zero, then Algorithm A stops at x^k, and from Lemma 2.2 (ii) we know that x^k is a stationary point of the problem (P). Thus we 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). First, the following three assumptions are necessary in the rest of this paper.

H2 The sequence $\{H_k\}$ of matrices is uniformly positive definite, that is, there exist two positive constants a and b such that

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

H3 For any $x^0 \in R^n$, the level set $\Omega = \big\{x \in R^n : F(x) \leqslant F(x^0)\big\}$ is compact.

H4 $\inf_k\{\varepsilon_k\} \triangleq \overline{\varepsilon} > 0$.

REMARK 3.1. From (2.6), we know that the sequence $\{x^k\} \subseteq \Omega$, so H3 implies that the sequence $\{x^k\}$ of points generated by Algorithm A is bounded.

REMARK 3.2. If one chooses ε_k by one of the two following cases, then H4 holds automatically.

CASES A. $\varepsilon_k \equiv \varepsilon$ for all k, where ε is a positive sufficiently small constant.

CASES B. $\varepsilon_k = \max\big\{F(x^k) - f_j(x^k) : j \in I\big\} + \varepsilon$. Note that in this case, $I^k_{\varepsilon_k} \equiv I$.

LEMMA 3.1. *Suppose that H1–H3 hold. Then the entire sequence $\big\{F(x^k)\big\}$ is convergent, and the entire sequence $\{t_k d^k\}$ converges to zero.*

PROOF: First, we define index $l(k)$ as follows

$$F(x^{l(k)}) = \max\big\{F(x^l) : l = k - r, k - r + 1, \dots, k\big\}.$$

This together with (2.6) shows that

$$F(x^{l(k+1)}) \leqslant \max\big\{F(x^{k+1-l}) : l = 0, 1, 2, \dots, r, r+1\big\} = \max\big\{F(x^{l(k)}), F(x^{k+1})\big\} = F(x^{l(k)}),$$

which shows that $\{F(x^{l(k)})\}$ is a monotonely non-increasing sequence. Thus $\{F(x^{l(k)})\}$ is convergent, denote

$$\lim_{k\to\infty} F(x^{l(k)}) \triangleq F_*. \tag{3.1}$$

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

On the other hand, from (2.6) and Lemma 2.2 (i), we have

$$F(x^{l(k)}) \leqslant F(x^{l(l(k)-1)}) - \alpha t_{l(k)-1}\Big(\xi_1 + \frac{\xi_2}{2\gamma_0}\Big)(d^{l(k)-1})^T H_{l(k)-1} d^{l(k)-1}. \tag{3.2}$$

This relationship along with (3.1) and H2 gives

$$t_{l(k)-1} d^{l(k)-1} \to 0, \quad \|x^{l(k)} - x^{l(k)-1}\| \to 0, \quad k \to \infty. \tag{3.3}$$

Now, we set $\widehat{l}(k) = l(k+r+2)$ and show, by induction, that for $\forall j \geqslant 1$

$$\lim_{k\to\infty} t_{\widehat{l}(k)-j} d^{\widehat{l}(k)-j} = 0, \quad \lim_{k\to\infty} F(x^{\widehat{l}(k)-j}) = \lim_{k\to\infty} F(x^{l(k)}) = F_*. \tag{3.4}$$

First, from H3, (3.1), (3.3) and $\{\widehat{l}(k)\} \subseteq \{l(k)\}$, one has,

$$\big|F(x^{\widehat{l}(k)-1}) - F(x^{l(k)})\big| \leqslant \big|F(x^{\widehat{l}(k)-1}) - F(x^{\widehat{l}(k)})\big| + \big|F(x^{\widehat{l}(k)}) - F(x^{l(k)})\big| \to 0, \quad k \to \infty,$$

this along with (3.3) shows that (3.4) holds for $j = 1$. Suppose that (3.4) holds for $j = \widehat{j}$. Then similar to the proof of (3.3), one has

$$t_{\widehat{l}(k)-\widehat{j}-1} d^{\widehat{l}(k)-\widehat{j}-1} \to 0, \quad \big\|x^{\widehat{l}(k)-\widehat{j}} - x^{\widehat{l}(k)-\widehat{j}-1}\big\| \to 0, \quad k \to \infty.$$

Thus from H3, we obtain

$$\lim_{k\to\infty} F(x^{\widehat{l}(k)-\widehat{j}-1}) = \lim_{k\to\infty} F(x^{\widehat{l}(k)-\widehat{j}}) = \lim_{k\to\infty} F(x^{l(k)}) = F_*.$$

So, (3.4) holds for $j = \widehat{j} + 1$.

Now we turn to complete the rest of the proof. First, for $\forall k$, one has

$$1 \leqslant \widehat{l}(k) - k - 1 = l(k+r+2) - k - 1 \leqslant r + 1, \quad x^{\widehat{l}(k)} = x^{k+1} + \sum_{j=1}^{\widehat{l}(k)-k-1} t_{\widehat{l}(k)-j} d^{\widehat{l}(k)-j}.$$

From the two relationships above, (3.4) and H3, we get

$$\|x^{k+1} - x^{\widehat{l}(k)}\| \to 0, \quad \big|F(x^{k+1}) - F(x^{\widehat{l}(k)})\big| \to 0, \quad F(x^{k+1}) \to F_*, \quad k \to \infty.$$

Secondly, from Step 2 of the proposed algorithm, one has

$$F(x^{k+1}) \leqslant F(x^{l(k)}) - \alpha\xi_1 t_k (d^k)^T H_k d^k + \alpha t_k \xi_2 z_k.$$

Similar to the proof of (3.3), we obtain $t_k d^k \to 0, \ k \to \infty$. □

Lemma 3.2. *Suppose that H1–H4 hold. Then*

(i) *the entire sequences $\{z_k\}$ and $\{d^k\}$ are both bounded;*

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

(ii) *the entire multiplier sequence* $\{\lambda^k_{I^k_{\varepsilon_k}}\}$*is bounded;*

(iii) $\lim\limits_{k\to\infty} d^k = 0$ *and* $\lim\limits_{k\to\infty} z_k = 0$.

PROOF: (i) In view of Lemma 2.2 (i), the constraints of Quadratic Programming (x^k, H_k), H1, H2 and Remark 3.1, there exist two constants $\bar{c}, \tilde{c} > 0$ such that

$$0 \geqslant \gamma_0 z_k + \frac{1}{2}(d^k)^T H_k d^k \geqslant \frac{\gamma_0}{\gamma_j}\big(f_j(x^k) + g_j(x^k)^T d^k - F(x^k)\big) + \frac{1}{2}(d^k)^T H_k d^k$$
$$\geqslant -\bar{c}\cdot\|d^k\| - \tilde{c} + \frac{1}{2}a\|d^k\|^2, \quad \forall j \in I^k_{\varepsilon_k} \neq \phi, \quad \forall k.$$

These inequalities imply that $\{z_k\}$ and $\{d^k\}$ are both bounded.

(ii) From (2.2), one gets $0 \leqslant \lambda^k_{I^k_{\varepsilon_k}}$, $\sum\limits_{j\in I^k_{\varepsilon_k}} \gamma_j \lambda^k_j = \gamma_0$, so $\{\lambda^k_{I^k_{\varepsilon_k}}\}$ is bounded.

(iii) We prove $\lim\limits_{k\to\infty} d^k = 0$ first. By contradiction, we assume that $\lim\limits_{k\to\infty} d^k \neq 0$, then there exist an infinite index subset K_1 and a constant $\sigma > 0$ such that $\|d^k\| \geqslant \sigma, k \in K_1$. In view of the approximately active set $I^k_{\varepsilon_k}$ being the subset of the fixed and finite set I, we can also assume without loss of generality that K_1 satisfies

$$x^k \to x^*, \quad I^k_{\varepsilon_k} \equiv \bar{I}, \quad H_k \to H_*, \quad k \in K_1. \tag{3.5}$$

Denote

$$w_k(t) = F(x^k + td^k) - F_k + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k),$$

then we have

$$w_k(t) = \max\{f_j(x^k + td^k) - F_k + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k) : j \in I\}$$
$$= \max\{f_j(x^k) + tg_j(x^k)^T d^k + o(t) - F_k + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k) : j \in I\}.$$

We also denote

$$a_{kj}(t) = f_j(x^k) + tg_j(x^k)^T d^k + o(t) - F_k + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k), j \in I.$$

So from the definition of F_k, one has

$$a_{kj}(t) \leqslant f_j(x^k) + tg_j(x^k)^T d^k + o(t) - F(x^k) + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k), j \in I.$$

Then from Quadratic Programming (x^k, H_k), H2, $\alpha \in (0, 0.5)$ and $\gamma_0\xi_1 + \alpha\xi_2 \leqslant \gamma$, we have for $j \in I^k_{\varepsilon_k} \equiv \bar{I}$

$$a_{kj}(t) \leqslant t\big(f_j(x^k) + g_j(x^k)^T d^k - F(x^k)\big) + \alpha t(\xi_1 (d^k)^T H_k d^k - \xi_2 z_k) + o(t)$$
$$\leqslant t\gamma_j z_k + \alpha t\xi_1 (d^k)^T H_k d^k - \alpha t\xi_2 z_k + o(t)$$
$$\leqslant -\frac{\gamma_j - \alpha\xi_2}{2\gamma_0} t(d^k)^T H_k d^k + \alpha t\xi_1 (d^k)^T H_k d^k + o(t)$$
$$\leqslant t(\alpha\xi_1 - \frac{\gamma_j - \alpha\xi_2}{2\gamma_0})a\sigma^2 + o(t) \leqslant 0.$$

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press

For $j \in I \setminus I^k_{\varepsilon_k}$, in view of H4, we get easily that $a_{kj}(t) \leqslant 0$ for $t > 0$ small enough. Thus (2.6) holds for $t > 0$ small enough and all $k \in K_1$. Thus there exists a constant $\bar{t} > 0$ such that the stepsize $t_k \geqslant \bar{t}$, $k \in K_1$ and $\|t_k d^k\| \geqslant \bar{t}\|d^k\| \geqslant \bar{t}\sigma$, which contradicts Lemma 3.1. So $\lim\limits_{k\to\infty} d^k = 0$.

Finally, we prove $\lim\limits_{k\to\infty} z_k = 0$. From Lemma 2.2 (i) and the constraints of Quadratic Programming (x^k, H_k), one has

$$0 \geqslant \gamma_j z_k \geqslant f_j(x^k) + g_j(x^k)^T d^k - F(x^k) = g_j(x^k)^T d^k, \quad j \in I(x^k) \subseteq I^k_{\varepsilon_k},$$

these along with $\lim\limits_{k\to\infty} d^k = 0$ show that $\lim\limits_{k\to\infty} z_k = 0$. The whole proof is completed. $\square$

Theorem 3.1. *Suppose that* 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).

Proof: If Algorithm A stops at the k-th iteration, then, from Step 1 of Algorithm A and Lemma 2.2 (ii), we know that x^k is a stationary point of (P). Now we suppose that Algorithm A generates an infinite sequence $\{x^k\}$ with corresponding multiplier sequence $\{\lambda^k\}$ and x^* is an accumulation of $\{x^k\}$. In view of Lemma 3.2, we can assume without loss of generality that the infinite index set K_1 in (3.5) satisfies $d^k \to 0, z_k \to 0, \lambda^k \to \lambda^*, k \in K_1$. Now, passing to the limit $k \in K_1$ and $k \to \infty$ in (2.2), we have

$$\begin{cases} \begin{pmatrix} \gamma_0 \\ 0 \end{pmatrix} + \sum\limits_{j=1}^{m} \lambda_j^* \begin{pmatrix} -\gamma_j \\ g_j(x^*) \end{pmatrix} = 0, \\ (f_j(x^*) - F(x^*))\lambda_j^* = 0, \quad \lambda_j^* \geqslant 0, \quad j \in I, \end{cases}$$

therefore $\sigma_* = \sum\limits_{j=1}^{m} \lambda_j^* > 0$ and

$$\sum_{j=1}^{m} \frac{\lambda_j^*}{\sigma_*} g_j(x^*) = 0, \quad \sum_{j=1}^{m} \frac{\lambda_j^*}{\sigma_*} = 1; \quad (f_j(x^*) - F(x^*))\frac{\lambda_j^*}{\sigma_*} = 0, \quad \frac{\lambda_j^*}{\sigma_*} \geqslant 0, \quad j \in I.$$

This shows that x^* is a stationary point of (P). $\square$

4. Numerical experiments

In this section, we select the problems 1–5 in [6] to show the efficiency of Algorithm A. The numerical experiments are implemented on MATLAB 6.5 and we use its optimisation toolbox to solve the problem (2.1). The numerical results show that the proposed algorithm is efficient.

In the implementation the approximation Hessian matrix H_k is updated according to the Powell's modification of Broyden–Fletcher–Goldfarb–Shanno formula ([5]) as follows

$$H_{k+1} = H_k - \frac{H_k s^k (s^k)^T H_k}{(s^k)^T H_k s^k} + \frac{\overline{y}^k (\overline{y}^k)^T}{(s^k)^T \overline{y}^k}, \qquad (k \geqslant 0), \tag{4.1}$$

where

$$s^k = x^{k+1} - x^k,\ \ \overline{y}^k = \eta_k y^k + (1 - \eta_k) H_k s^k,\ \ y^k = \nabla_x L(x^{k+1}, \overline{\lambda}^k) - \nabla_x L(x^k, \overline{\lambda}^k,),$$

$$\overline{\lambda}^k = \lambda^k / \sigma_k,\quad \sigma_k = \sum_{j=1}^{m} \lambda_j^k,\quad \nabla_x L(x, \lambda) = \sum_{j=1}^{m} \lambda_j g_j(x),$$

$$\eta_k = \begin{cases} 1, & \text{if } \ (s^k)^T y^k \geqslant 0.2 (s^k)^T H_k s^k; \\ \dfrac{0.8 (s^k)^T H_k s^k}{(s^k)^T H_k s^k - (y^k)^T s^k}, & \text{otherwise.} \end{cases}$$

The numerical results of the proposed algorithm are given in the following tables 1 and 2.

Table 1. *Numerical results of Problems 1–5 for* $\xi_1 = 0.4, \xi_2 = 0.6, \gamma_0 = \gamma_j = 0.7, \forall j \in I$

P	IP	n	m	r	NI	approximate solution x^*	$F(x^*)$	$\|\|d^k\|\|$
1	$(1,-0.01)^T$	2	3	0	11	$(1.139037,0.899559)^T$	1.952224	$3.462285e-7$
				1	52	$(1.139033, 0.899565)^T$	1.952224	$6.927319e-6$
2	$(0.01,0.01)^T$	2	3	0	7	$(1.000000,1.000000)^T$	2.000000	$6.177491e-10$
				1	33	$(1.000000,1.000000)^T$	2.000000	$4.020731e-9$
3	$(0.2,-1,2.3,-0.01)^T$	4	4	0	38	$(0.0,1.0,2.0,-1.0)^T$	-44.000000	$3.087092e-6$
				1	188	$(0.0,1.0,2.0,-1.0)^T$	-44.000000	$3.325487e-7$
4	$(3,1)^T$	2	3	0	10	$(-0.453299, 0.906598)^T$	0.616440	$6.419295e-6$
				1	64	$(-0.453296, 0.906592)^T$	0.616432	$1.529814e-7$
				4	7	$(0.000039, 0.000000)^T$	1.000000	$2.689503e-7$
5	$(1.01,0.9,1)^T$	3	6	0	20	$(0.328259,0.0,0.131319)^T$	3.599719	$7.227559e-7$
				1	74	$(0.328258,0.0,0.131319)^T$	3.599719	$5.854201e-6$

Table 2. *Numerical results of Problems 1–5 for* $\xi_1 = 0.4, \xi_2 = 0.6, \gamma_0 = 0.7, \gamma_j = 0.9, \forall j \in I$

P	IP	n	m	r	NI	approximate solution x^*	$F(x^*)$	$\|\|d^k\|\|$
1	$(1,-0.01)^T$	2	3	0	19	$(1.139040,0.899558)^T$	1.952224	$2.457516e-6$
				1	50	$(1.139046, 0.899553)^T$	1.952224	$8.382028e-6$
2	$(0.01,0.01)^T$	2	3	0	17	$(1.000000,1.000000)^T$	2.000000	$4.775926e-9$
				1	41	$(1.000000,1.000000)^T$	2.000000	$2.857378e-9$
3	$(0.2,-1,2.3,-0.01)^T$	4	4	0	74	$(0.0,1.0,2.0,-1.0)^T$	-44.000000	$7.732998e-6$
				1	150	$(0.0,1.0,2.0,-1.0)^T$	-44.000000	$9.266799e-6$
4	$(3,1)^T$	2	3	0	21	$(-0.453293, 0.906592)^T$	0.616432	$2.641059e-6$
				1	51	$(-0.453295, 0.906592)^T$	0.616432	$2.903402e-6$
				4	28	$(-0.453305, 0.906592)^T$	0.616432	$7.167424e-6$
5	$(1.01,0.9,1)^T$	3	6	0	34	$(0.328257,0.0,0.131319)^T$	3.599719	$3.382266e-6$
				1	44	$(0.328258,0.0,0.131319)^T$	3.599719	$2.844776e-6$

During the numerical experiments, we set parameters $\varepsilon_0 = 0.02$, $\overline{\varepsilon} = 0.001$, $\alpha = 0.1$, $\beta = 0.5$. In formula (4.1), we select $H_0 = E$, where $E \in \Re^{n \times n}$ is an identity matrix. Execution is terminated if $\|d^k\| < 10^{-5}$ or $\|x^{k+1} - x^k\| < 10^{-6}$. Parameter ε_k is adjusted by the following formula

$$\varepsilon_{k+1} = \begin{cases} \varepsilon_k, & \text{if} \quad \varepsilon_k \leqslant \bar{\varepsilon}; \\ \varepsilon_k/2, & \text{otherwise.} \end{cases}$$

The columns of the two tables have the following meanings:

n: the dimension of x; m: the number of $f_j(x)$; IP: the initial point; NI: the number of iterations.

From the iteration results on problems 1 to 5 above, the following observations can be made. First, it can be seen that in Table 1 all problems except problem 4 and all problems in Table 2 are solved faster for $r = 0$ than the cases for $r \geqslant 1$. Secondly, suitable values of parameters $\gamma_0, \gamma_j (j \in I)$ may speed up the convergence. In both Tables, we only set the parameters $\gamma_j, j \in I$ with different values. And we can see that for $\gamma_j = 0.9, j \in I$ in Table 2, problems 3–5 with $r = 1$ have faster convergence than the same cases for $\gamma_j = 0.7, j \in I$. And for $\gamma_j = 0.7, j \in I$ in Table 1, problem 2, problems 1,3,5 with $r = 0$ and problem 4 with $r = 0, 4$ have faster convergence than the same cases for $\gamma_j = 0.9, j \in I$.

References

[1] S.P. Han, 'Variable metric methods for minimizing a class of nondifferentiable functions', *Math. Programming* **20** (1981), 1–13.

[2] 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.

[3] N. Maratos, *Exact penalty function algorithm for finite dimensional and control optimization problems*, (Ph.D. thesis) (University of London, London UK, 1978).

[4] E.R. Panier and A.L. Tits, 'On combining feasibility, descent and superlinear convergence in inequality constrained optimization', *Math. Programming* **59** (1993), 261–276.

[5] M.J. Powell, *The convergence of variable metric methods for nonlinearly constrained optimization calculations*, (R.R. Meyer and S.M. Robinson, Editors), Nonlinear Promgramming **3** (Academic Press, New York, 1978).

[6] A. Vardi, 'New minimax algorithm', *J. Optim. Theory Appl.* **75** (1992), 613–634.

[7] Y. Xue, 'The sequential quadratic programming method for solving minimax problem', *J. Systems Science Systems Engg. (PRC)* **22** (2002), 355–364.

[8] Y. Xue and H.Z. Yang, 'Descent algorithm for solving minimax optimization problem', *J. Beijing Plotechnic University* **27** (2001), 255–261.

[9] J.L. Zhou and A.L. Tits, 'Nonmonotone line search for minimax problems', *J. Optim. Theory Appl.* **76** (1993), 455–476.

College of Mathematics and Informatics Science
Guangxi University
530004, Nanning
Peoples Republic of China
e-mail: jianjb@gxu.edu.cn

https://doi.org/10.1017/S0004972700038673 Published online by Cambridge University Press