Journal of Computational Mathematics, Vol.21, No.2, 2003, 247–256.

A ROBUST SQP METHOD FOR OPTIMIZATION WITH INEQUALITY CONSTRAINTS*1)

Juliang Zhang†
(*School of Economics and Manggement, Tsinghua University, Beijing, 100084, China*)
Xiangsun Zhang
(*School of Information Engineering, Beijing University of Posts and Telecommunications, Beijing, 100876, China*)

Abstract

A new algorithm for inequality constrained optimization is presented, which solves a linear programming subproblem and a quadratic subproblem at each iteration. The algorithm can circumvent the difficulties associated with the possible inconsistency of QP subproblem of the original SQP method. Moreover, the algorithm can converge to a point which satisfies a certain first-order necessary condition even if the original problem is itself infeasible. Under certain condition, some global convergence results are proved and local superlinear convergence results are also obtained. Preliminary numerical results are reported.

Key words: nonlinear optimization, SQP method, global convergence, superlinear convergence.

1. Introduction

We consider the following nonlinear programming problem:

$$\begin{array}{ll} \min\limits_{x\in R^n} & f(x) \\ \text{s.t.} & c_i(x) \le 0, \quad i \in I \end{array} \tag{1}$$

where $f : R^n \to R$, $c_i : R^n \to R$, $i \in I$ are continuously differentiable functions. $I = \{1, 2, \ldots, m\}$. Let $g(x) = \nabla f(x)$, $C(x) = (c_1(x), c_2(x), \ldots, c_m(x))^T$ and $A(x) = (\nabla c_1(x), \nabla c_2(x), \ldots, \nabla c_m(x))$. In view of convenience, we usually use f_k for $f(x_k)$, C_k for $C(x_k)$, g_k for $g(x_k)$ and A_k for $A(x_k)$, etc.

SQP algorithms for constrained optimization are iteration-type methods. They generate a sequence of points approximating to the solution by the procedure

$$x_{k+1} = x_k + \lambda_k d_k \tag{2}$$

where x_k is the current point, d_k is a search direction which minimizes a quadratic model subject to linearized constraints and λ_k is the stepsize along the search direction (see details in [12, 18, 25]). For $k \ge 1$ the original SQP method developed by Wilson, Han and Powell employs the following QP subproblem

$$\begin{array}{ll} \min & g_k^T d + \frac{1}{2} d^T B_k d \\ \text{s.t.} & C_k + A_k^T d \le 0 \end{array} \tag{3}$$

where B_k is a symmetric matrix which approximates to the $Hessian$ of the $Lagrangian$ function

$$L(x, \rho) = f(x) + \rho^T C(x) \tag{4}$$

* Received December 1, 2000.
1) This work is supported in part by the National Natural Science Foundation of China (Grant No. 10171055).
† Email address: jlzhang@mail.amt.ac.cn.

where ρ is an approximation to the Lagrangian multiplier vector.

Because of its nice convergence properties(for example, see Han (1977), Powell (1978), Bogg et al (1982)), the SQP method has been absorbing attention from many researchers.

The requisite consistency of the linearized constraints of the QP subproblem (3) is a serious limitation of the SQP method. Within the framework of the method, Powell suggested to solve a modified subproblem at each iteration (Powell(1977)):

$$\begin{array}{ll} \min & g_k^T d + \frac{1}{2} d^T B_k d + \frac{1}{2}\delta_k (1-\mu)^2 \\ \text{s.t.} & \mu_i c_i(x_k) + \nabla c_i(x_k)^T d \leq 0 \end{array} \tag{5}$$

where $\mu_i = \begin{cases} 1, & c_i(x_k) < 0 \\ \mu, & c_i(x_k) \geq 0 \end{cases}$ and $0 \leq \mu \leq 1$, $\delta_k > 0$ is a penalty parameter. With some other technique, the computational investigation provided by Schittkowski (1981,1983) shows that this modification works very well.

However, a simple example presented by Burke and Han (1989) and Burke (1992) indicates that this approach may not be the best one.

On the other hand, based on the trust region strategy, Fletcher(1981) developed the Sl_1QP method for problem (1). Burke and Han (1989) shows that Fletcher's approach is still incomplete. One of the reasons is that the search direction generated by Sl_1QP method may point to the contrary of the optimal point.

Burke and Han (1989) and Burke (1989) presented approaches to overcome difficulties associated with the inconsistency of the QP subproblem (3). A feature different to the other methods is that even when (1) is itself infeasible their methods can converge to a point which meets a certain first-order necessary optimality condition. However, Burke and Han's method is conceptual.

Recently, Liu and Yuan (2000) presented a method which is a modification to SQP method. Similar to Burke and Han's methods, even when (1) is itself infeasible their method can converge to a point which meets a certain first-order necessary optimality condition. Unlike the other methods, their method solves two subproblem—one is an unconstrained piecewise quadratic subproblem, the other is a quadratic subproblem. Their method has excellent theoretical properties and is implementable.

In this paper, we describe another implementable method which is a modification to SQP method. The algorithm can circumvent difficulties associated with the infeasibility of the QP subproblem. Our method is similar to Liu and Yuan's method. At each iteration it solves two subproblems—one is a linear programming, which is different from Liu and Yuan's, the other is a quadratic subproblem. Since solving a linear programming is much easier than solving a piecewise quadratic programming, the computation at each iteration is less than Liu and Yuan's. Under certain conditions we can prove that the method is globally convergent and locally superlinearly convergent.

Note that in our method we only deal with inequality constraints. In fact, for equality constraints, we can convert it into two inequalities and our method can also deal with it. Therefore, our method can solve optimization problem with general constraints.

Our algorithm can be easily combined with the trust region strategy. Thus the algorithm in this paper can be extended to a trust region algorithm for constrained optimization problem.

The paper is organized as follows. The algorithm model is presented in Section 2. In Section 3, the global convergence results of the algorithm are proved. We discuss the local properties of the algorithm in Section 4. In Section 5, some numerical results are reported. Some discussions are given in Section 6 to conclude the paper.

2. The Algorithm Model

First, we consider the following linear programming subproblem:

$$LP(x_k) \quad \begin{array}{ll} \min\limits_{(d^T,z)^T\in R^{n+1}} & z \\ \text{s.t.} & C_k + A_k^T d \le ze \\ & z \ge 0, \end{array} \tag{6}$$

where $e = (1, 1, \cdots, 1)^T \in R^n$. Let its solution be $(\hat{d}_k^T, z_k)^T$. Then we consider the following quadratic programming subproblem:

$$QP(x_k, B_k) \quad \begin{array}{ll} \min\limits_{d\in R^n} & g_k^T d + \frac{1}{2} d^T B_k d \\ \text{s.t.} & C_k + A_k^T d \le z_k e. \end{array} \tag{7}$$

Obviously, the feasible region of this subproblem is nonempty. In fact, $\hat{d}_k$ is a feasible solution of (7). If we assume that B_k is positive definite, the solution of (7) is unique. Let d_k be the solution of (7). Then d_k is used as the search direction at the current point x_k.

In order to carry out a line search, we use the nondifferentiable exact penalty function as the merit function.

Let

$$\Psi(x) = \max_{i\in I}\{c_i(x), 0\}.$$

Given $d \in R^n$, let

$$\Psi^*(x; d) = \max_{i\in I}\{c_i(x) + \nabla c_i(x)^T d, 0\} - \Psi(x).$$

Let

$$\Phi_\sigma(x) = f(x) + \sigma\Psi(x),$$

and

$$\theta_\sigma(x; d) = g(x)^T d + \sigma\Psi^*(x; d)$$

where $\sigma > 0$ is a penalty parameter. The updating formula of the penalty parameter is given in Step 3 in the algorithm model.

Now we state our algorithm as follows.

Algorithm 2.1 (A Robust Algorithm for Optimization)

Step 0. Given $x_0 \in R^n$, $B_0 \in R^{n\times n}$ is a symmetric positive definite matrix, $\beta \in (0, \frac{1}{2})$, $\gamma \in (0, 1)$, $\sigma_0 > 0$, $k := 0$;

Step 1. Solve (6) to obtain $\hat{d}_k$, z_k. If $\hat{d}_k = 0$ and $z_k \ne 0$, stop;

Step 2. Solve (7) to obtain d_k. If $d_k = 0$, stop;

Step 3. If $\theta_{\sigma_k}(x_k; d_k) > -d_k^T B_k d_k$, then

$$\sigma_k := \max\left\{\frac{g_k^T d_k + d_k^T B_k d_k}{-\Psi^*(x_k; d_k)}, 2\sigma_k\right\}; \tag{8}$$

Step 4. Choose λ_k which is the largest one in the sequence $\{1, \gamma, \gamma^2, \ldots\}$ satisfying

$$\Phi_{\sigma_k}(x_k + \lambda d_k) - \Phi_{\sigma_k}(x_k) \le \lambda\beta\theta_{\sigma_k}(x_k; d_k).$$

Set

$$x_{k+1} = x_k + \lambda_k d_k;$$

Step 5. Modify B_k to obtain B_{k+1}, $k := k + 1$, go to Step 1.

Algorithm 2.1 is similar to the methods proposed by Burke and Han (1989), Burke(1989), Zhou (1997) and Liu and Yuan (2000). But it is not identical to one of these. Comparing to Burke and Han (1989), Burke(1989) and Zhou (1997) we do not employ bound constraints. The algorithm can be implemented in the way as SQP algorithm. The difference between Algorithm 2.1 and Liu and Yuan's is that Liu and Yuan's method solves a piecewise quadratic programming and a quadratic programming but Algorithm 2.1 solves a linear programming and a quadratic programming. So the computation of Algorithm 2.1 at each iteration is less than Liu and Yuan's.

3. Global Convergence

Throughout this work, we assume that the following conditions hold.

Assumption 3.1:

(1). $f(x)$, $c_i(x), i \in I$ are twice continuously differentiable;

(2). B_k, $k = 1, 2 \ldots$ are positive definite and there exist two positive constants M_1 and M_2 such that

$$M_1 \|d\|^2 \leq d^T B_k d \leq M_2 \|d\|^2, \tag{9}$$

for all $d \in R^n$ and $k \geq 1$;

(3). $\{x_k\}$, $\{d_k\}$ are uniformly bounded.

It is a common assumption for convergence analysis of SQP methods that $\{x_k\}$ is bounded. Since the objective function of (7) is coercive, the condition that $\{d_k\}$ is bounded is reasonable. Because no restriction are imposed on the constraint functions, the cluster point of the sequence generated by our algorithm can be one of three different types of points. Similar to Yuan (1995), Liu and Yuan (2000), we give their definitions as follows.

Definition 3.1. *$x \in R^n$ is called*

(1). a strong stationary point of problem (1) if x is feasible and there exists a vector $\rho = (\rho_1, \rho_2, \ldots, \rho_m)^T \in R^m$ such that

$$g(x) + A(x)\rho = 0, \tag{10}$$

$$\rho_i \geq 0, \rho_i c_i(x) = 0, i \in I; \tag{11}$$

(2). an infeasible stationary point of problem (1) if x is infeasible and

$$\min_{d \in R^n} \max_{i \in I} \{c_i(x) + \nabla c_i(x)^T d, 0\} = \Psi(x); \tag{12}$$

(3). a singular stationary point of problem (1) if x is feasible and there exists an infeasible sequence $\{v_k\}$ converging to x such that

$$\lim_{k \to \infty} \frac{\min_{d \in R^n} \max_{i \in I} \{c_i(v_k) + \nabla c_i(v_k)^T d, 0\}}{\Psi(v_k)} = 1. \tag{13}$$

Definition 3.1 is related to our algorithm closely. It should be noted that there are some difference between our definition and that of Yuan (1995), Liu and Yuan (2000). A strong stationary point defined above is precisely a K–T point of problem (1).

It is similar to Lemma 3.4 in Liu and Yuan (2000) that we can prove the following lemma, which describes the properties of infeasible stationary point and singular stationary point.

Lemma 3.1. *If $x \in R^n$ is an infeasible stationary point or a singular stationary point, there exist $\rho_0 \geq 0$ and $\rho \in R^m$ such that the following first-order necessary condition*

$$\rho_0 g(x) + \sum_{i=1}^{m} \rho_i \nabla c_i(x) = 0 \tag{14}$$

$$\rho_i \geq 0, i = 0, 1, \ldots, m \tag{15}$$

holds.

The following lemma shows that if our algorithm stops after finite many iterations, the last iterate point must be a strong stationary point or an infeasible stationary point of problem (1).

Lemma 3.2. *If Algorithm 3.1 terminates at x_k, then x_k is either an infeasible stationary point or a strong stationary point.*

Proof. (i). If the algorithm terminates at Step 2, then $z_k = \Psi(x_k)$ and $z_k \neq 0$. Since $z_k \neq 0$, x_k is an infeasible point. Now we prove that

$$\min_{d \in R^n} \max_{i \in I} \{c_i(x_k) + \nabla c_i(x_k)^T d, 0\} = \Psi(x_k). \tag{16}$$

In fact, if (16) does not hold, then there exists d_k^0 such that

$$\hat{z} = \min_{d\in R^n} \max_{i\in I}\{c_i(x_k) + \nabla c_i(x_k)^T d_k^0, 0\} < \Psi(x_k).$$

Obviously, $({d_k^0}^T, \hat{z})^T$ is a feasible point of subproblem (6). It follows from the definition of $\hat{z}_k$ that

$$z_k \le \hat{z} < \Psi(x_k),$$

which contradicts the fact $z_k = \Psi(x_k)$.

(ii). If the algorithm terminates at Step 3, then $d_k = 0$ is the solution of subproblem (7). In this case, $d_k = 0$ satisfies the following K–T conditions

$$g_k + B_k d + A_k \rho = 0, \tag{17}$$

$$\rho_i \ge 0, i \in I, \tag{18}$$

$$C_k + A_k^T d \le z_k e, \tag{19}$$

$$\rho_i(c_i(x_k) + \nabla c_i(x_k)^T d - z_k) = 0, \quad i \in I. \tag{20}$$

Now we prove that $z_k = 0$. If $z_k \ne 0$, then x_k is infeasible. Since the algorithm does not stop at Step 2, then $z_k < \Psi(x_k)$. This shows that $d = 0$ is not a feasible point of subproblem (7). This is a contradiction. So $z_k = 0$. In this case, (17)-(20) turn to

$$g_k + A_k \rho = 0,$$

$$\rho_i \ge 0, i \in I,$$

$$C_k \le 0,$$

$$\rho_i c_i(x_k) = 0, \quad i \in I.$$

Namely x_k is a strong stationary point.

The following lemma shows that the line search procedure is well defined.

Lemma 3.3. *In Step 4 the line search procedure is well defined.*

Proof. Since the algorithm arrives at step 4, then we have

$$\theta_{\sigma_k}(x_k; d_k) \le -d_k^T B_k d_k \le -M_1 \|d_k\|^2 < 0. \tag{21}$$

If the algorithm is not feasible, then $\forall \lambda \in (0,1)$ we have

$$\Phi_{\sigma_k}(x_k + \lambda d_k) - \Phi_{\sigma_k}(x_k) > \lambda\beta\theta_{\sigma_k}(x_k; d_k),$$

i.e.,

$$\frac{\Phi_{\sigma_k}(x_k + \lambda d_k) - \Phi_{\sigma_k}(x_k)}{\lambda} > \beta\theta_{\sigma_k}(x_k; d_k). \tag{22}$$

Let $\lambda \to 0$, we obtain

$$g_k^T d_k + \sigma_k \Psi'(x_k; d_k) > \beta\theta_{\sigma_k}(x_k; d_k). \tag{23}$$

From Lemma 2.1 in Zhou (1997), we have

$$\Psi'(x_k; d_k) \le \Psi^*(x_k; d_k). \tag{24}$$

(23), (24) imply that

$$(1-\beta)\theta_{\sigma_k}(x_k; d_k) > 0. \tag{25}$$

Note that $\beta \in (0, \frac{1}{2})$, (25) contradicts (21). This contradiction shows that the lemma holds.

If $\sigma_k \to \infty$, by Lemma 4.2 of Yuan (1992), $\lim_{k\to\infty} \Psi(x_k)$ exists.

Lemma 3.4. *If* $\sigma_k \to \infty$ *and* $\lim_{k\to\infty} \Psi(x_k) > 0$*, then there exists a subsequence of* $\{x_k\}$ *converging to an infeasible stationary point.*

Proof. Let S be the set of accumulation points of $\{x_k\}$. If the lemma is not true, for any $x \in S$, $\Psi(x) \ne 0$ and (12) does not hold, then there exists $v > 0$ such that for k large enough,

$$\min_{\|d\|_2 \le \delta} \max_{i\in I}\{c_i(x_k) + \nabla c_i(x_k)^T d), 0\} \le \Psi(x_k) - v \tag{26}$$

where δ is a positive constant.

Let d_k^0 be a vector such that $\|d_k^0\| \le \delta$ and that

$$z_k^0 = \max_{i\in I}\{c_i(x_k) + \nabla c_i(x_k)^T d_k^0, 0\} = \min_{\|d\|_2 \le \delta} \max_{i\in I}\{c_i(x_k) + \nabla c_i(x_k)^T d, 0\}.$$

Hence

$$\Psi^*(x_k; d_k^0) \leq -v.$$

Since $({d_k^0}^T, z_k^0)^T$ is a feasible solution of subproblem (6),

$$z_k \leq z_k^0.$$

So

$$z_k - \Psi(x_k) \leq -v. \tag{27}$$

On the other hand, by Step3 and $\sigma_k \to \infty$,

$$\frac{g_k^T d_k + d_k B_k d_k}{-\Psi^*(x_k; d_k)} \to \infty.$$

Note that d_k and g_k are bounded, then we have

$$\Psi^*(x_k; d_k) \to 0. \tag{28}$$

The definition of Ψ^* and (27) imply that

$$\begin{array}{rcl} \Psi^*(x_k; d_k) & = & \max_{i\in I}\{c_i(x_k) + \nabla c_i(x_k)^T d_k, 0\} - \Psi(x_k) \\ & \leq & z_k - \Psi(x_k) \\ & \leq & -v < 0. \end{array} \tag{29}$$

(29) contradicts to (28). This contradiction shows that the lemma is true.

Similarly, we have the following results:

Lemma 3.5. *If $\sigma_k \to \infty$ and $\lim_{k\to\infty} \Psi(x_k) = 0$, then there exists a subsequence of $\{x_k\}$ which converges to a singular stationary point of problem (1).*

Proof. Let x be any cluster point of $\{x_k\}$. Then x is a feasible point of problem (1). The condition $\sigma_k \to \infty$ implies that there exists an infinite subsequence $\{x_k : k \in \mathcal{K}\}$ such that $\Psi(x_k) \neq 0$, $\forall k \in \mathcal{K}$.

If the results is not true, then for any convergent subsequence $\{x_k : k \in \hat{K}\}$ ($\hat{K} \subset \mathcal{K}$), (13) does not hold. Hence there exists a positive number v such that (26) holds. Similar to Lemma 3.4, the proof can be completed.

The above two lemmas imply that σ_k is bounded if no subsequence of $\{x_k\}$ converges to an infeasible stationary point or a singular stationary point of problem (1).

In the following, we assume that σ_k is bounded above. Without loss of generality, we can assume that $\sigma_k = \sigma$, $k = 1, 2, \ldots$.

Lemma 3.6. *Suppose that $x_k \to \bar{x}$, $B_k \to \bar{B}$, then $z_k \to \bar{z}$, $d_k \to \bar{d}$, where z_k, $\bar{z}$ are the solutions of $LP(x_k)$, $LP(\bar{x})$ respectively and d_k, $\bar{d}$ are the solutions of $QP(x_k, B_k)$ and $QP(\bar{x}, \bar{B})$ respectively.*

Proof. The first part can be implied from the linear programming sensitivity analysis theory.

Now we prove the second part. Assume that $\{d_k\}$ does not converge to $\bar{d}$, then there exists a subsequence $\{d_s\} \subset \{d_k\}$ converging to $d' \neq \bar{d}$. By the first part, we know that

$$z_k \to \bar{z}. \tag{30}$$

For any feasible solution d of $QP(\bar{x}, \bar{B})$, there exists a feasible solution d_m of $QP(x_s, B_s)$ such that

$$d_m \to d. \tag{31}$$

Since d_s is the solution of $QP(x_s, B_s)$,

$$g_s^T d_s + \frac{1}{2} d_s^T B_s d_s \leq g_s^T d_m + \frac{1}{2} d_m^T B_s d_m.$$

Let $s \to \infty$, $m \to \infty$, we obtain

$$g(\bar{x})^T d' + \frac{1}{2} {d'}^T \bar{B} d' \leq g(\bar{x})^T d + \frac{1}{2} d^T \bar{B} d$$

i.e., d' is a solution of $QP(\bar{x}, \bar{B})$. This contradicts to that $QP(\bar{x}, \bar{B})$ has the unique solution.

Lemma 3.7. *Assume that $\sigma_k = \sigma$, $k = 1, 2, \ldots$ $\{x_k\}$ is an infinite sequence generated by the algorithm and $\{x_k : k \in \hat{K}\}$ is a convergent subsequence. Then $d_k \to 0$ for $k \in \hat{K}$ and $k \to \infty$.*

Proof. We proceed by contradiction. Suppose that there exists an infinite subsequence $\{x_k : k \in K' \subset \hat{K}\}$ and a positive constant η such that $\|d_k\| \geq \eta$, $\forall k \in K'$. From Step 3 and Assumption 3.1(2), we have

$$\theta_{\sigma_k}(x_k; d_k) \leq -d_k^T B_k d_k \leq -M_1 \eta, \quad \forall k \in K'.$$

It is similar to Proposition 3.2 of De O. Pantoja and Mayne (1991) that we can prove that there exists a positive constant $\lambda_0 > 0$ such that

$$\lambda_k \geq \lambda_0, \quad \forall k \in K'.$$

By Step 4,

$$\begin{aligned} \Phi_\sigma(x_{k+1}) - \Phi_\sigma(x_k) &\leq \lambda_k \beta \theta_\sigma(x_k; d_k) \\ &\leq -\lambda_0 \beta M_1 \eta, \quad \forall k \in K'. \end{aligned} \tag{32}$$

(32) can imply that

$$\sum_{k \in K'} (\Phi_\sigma(x_{k+1}) - \Phi_\sigma(x_k)) \leq -\sum_{k \in K'} \lambda_0 \beta M_1 \eta = -\infty.$$

Hence

$$\begin{aligned} \lim_{n \to \infty} \Phi_\sigma(x_n) - \Phi_\sigma(x_0) &= \lim_{n \to \infty} \sum_{k=0}^{n-1} (\Phi_\sigma(x_{k+1}) - \Phi_\sigma(x_k)) \\ &\leq \sum_{k \in K'} (\Phi_\sigma(x_{k+1}) - \Phi_\sigma(x_k)) \\ &\leq -\sum_{k \in K'} \lambda_0 \beta M_1 \eta = -\infty. \end{aligned}$$

This contradicts to Assumption 3.1.

Theorem 3.1. *Suppose that $\sigma_k = \sigma$ and $\{x_k\}$ is an infinite sequence generated by our algorithm. $\{x_k : k \in \hat{K}\}$ is a subsequence converging to x^*. Then x^* is a strong stationary point.*

Proof. By Assumption 3.1 (2), $\{B_k\}$ is a bounded sequence. Without loss of generality, we can assume that $\{B_k : k \in \hat{K}\}$ converges to B^*. By Lemma 3.6, 3.7 $d^* = 0$ is the solution of $QP(x^*, B^*)$, i.e., there exists a vector $\rho^* \in R^m$ such that

$$g(x^*) + A(x^*)\rho^* = 0,$$
$$C(x^*) \leq z^*,$$
$$\rho_i^*(c_i(x^*) - z^*) = 0, \quad i \in I,$$
$$\rho_i^* \geq 0, i \in I.$$

It is similar to Lemma 3.2 (ii) that we can prove that $z^* = 0$. Then x^* is a strong stationary point of problem (1).

4. Superlinear Convergence

To analyze local superlinear convergence of the algorithm, we make the following assumption, which is similar to that in Liu and Yuan (2000) :

Assumption 4.1:

(1). $x_k \to x^*$, where x^* is a Kuhn–Tucker point (strong stationary point) of problem (1);

(2). Let $I^* = \{i \in I : c_i(x^*) = 0\}$, $\nabla c_i(x^*) (i \in I^*)$ are linearly independent;

(3). $\sigma_k = \sigma$, for $k \geq \hat{k}$, where σ is a constant, $\hat{k}$ is a sufficiently large integer;

By Assumption 4.1 (1) (2), we know that $z_k = 0$ for sufficiently large k. Then (7) is equivalent to subproblem (3). Hence our algorithm is equivalent to standard SQP method for k sufficiently large.

Assumption 4.2: Suppose that ρ^* is a *Lagrangian* multiplier vector associated with x^*:

(1). The strict complementarity condition holds at (x^*, ρ^*);

(2). $\nabla^2 L(x^*, \rho^*)$ is positive definite for all nonzero d in the null space $\{d : \nabla c_i(x^*)^T d = 0, i \in I^*\}$, where $L(x, \rho) = f(x) + C(x)^T \rho$.

(3). $\lim_{k\to\infty} \frac{\|P^*(B_k - \nabla^2 L(x^*,\rho^*))d_k\|_2}{\|d_k\|_2} = 0$, where P^* is a projection matrix on the null space $\{d : \nabla c_i(x^*)^T d = 0, i \in I^*\}$.

Under Assumption 4.2, Boggs et al (1982) proved the following results for SQP method

Lemma 4.1. *Suppose that Assumption 4.1, 4.2 holds, then*

$$\lim_{k\to\infty} \frac{\|x_k + d_k - x^*\|}{\|x_k - x^*\|} = 0.$$

A superlinear convergence step may be truncated due to the nonsmoothness of the merit function, which is known as "the Maratos effect" (see Maratos (1978), Yuan and Sun (1997)). In order to avoid this case, the second-order correction technique is considered by Mayne and Polak (1982), Coleman and Conn (1982), Fletcher (1982), De O. Pantoja and Mayne (1991) and so on. For our algorithm, we solve the linear equation system

$$c_i(x_k + d_k) + \nabla c_i(x_k)^T d = 0, \quad i \in I(x_k) \triangleq \{i \in I | \rho_k^i > 0\} \tag{33}$$

where ρ_k is the *Lagrangian* multiplier of $QP(x_k, B_k)$. Let $\hat{d}_k$ be the least norm solution of (33). If (33) is inconsistent or the norm of its least norm solution is greater than the norm of $\|d_k\|$ we set $\hat{d}_k = 0$. After we obtain the second-order correction step $\hat{d}_k$, the line search in Step 4 is replaced by the following arc search:

Step 4'. Let λ_k be the largest one of the sequence $\{1, \gamma, \gamma^2, \ldots\}$ satisfying

$$\Phi_{\sigma_k}(x_k + \lambda d_k + \lambda^2 \hat{d}_k) - \Phi_{\sigma_k}(x_k) \le \lambda\beta\theta_{\sigma_k}(x_k; d_k). \tag{34}$$

Similar to the analysis in Mayne and Polak (1982), Yuan and Sun (1997), DE O. Pantoja and Mayne (1991), we have the following results:

Theorem 4.1. *Under Assumption 4.1, 4.2, $\{x_k\}$ is an infinite sequence generated by Algorithm 2.1 in which Step 4 is replaced by Step 4'. Then*

$$\lim_{k\to\infty} \frac{\|x_k + d_k + \hat{d}_k - x^*\|}{\|x_k - x^*\|} = 0 \tag{35}$$

and there exists a sufficiently large K_2 such that $\lambda_k = 1$ for $k \ge K_2$. Then $\{x_k\}$ converges Q-superlinearly.

5. Numerical Results

A MATLAB subroutine was programmed to test our algorithm. The standard internal functions LP and QP in Optimization Toolbox are used to solve subproblem (6) and (7) respectively in our program.

For each problem, the standard initial point is used. We choose initial parameters $\beta = 0.25$, $\gamma = 0.5$, $\sigma = 1$ and $\epsilon = 10^{-6}$. The initial Lagrangian Hessian estimate $B_0 = I$ and B_k is updated by the damped BFGS formula ([18]):

$$B_{k+1} = B_k - \frac{B_k s_k s_k^T B_k}{s_k^T B_k s_k} + \frac{y_k y_k^T}{s_k^T y_k},$$

where

$$y_k = \begin{cases} \hat{y}_k, & \hat{y}_k s_k \ge 0.2 s_k^T B_k s_k, \\ \tau_k \hat{y}_k + (1 - \tau_k) B_k s_k, & otherwise, \end{cases}$$

and $\hat{y}_k = g_{k+1} - g_k + (A_{k+1} - A_k)\rho_k$, $s_k = x_{k+1} - x_k$, $\tau_k = 0.8 s_k^T B_k s_k / (s_k^T B_k s_k - s_k^T \hat{y}_k)$, ρ_k is a multiplier associated with (7).

The first test problem that we solved is taken from Sahba (1987) (see Liu and Yuan (2000) also) and the standard starting point is $x_0 = (0, 5)^T$. As in Liu and Yuan (2000), the solution given by our algorithm is $(-0.886227, 0.886226)$, which is an approximate minimum point. IF (the number of functional calculation) is 8 and IG (the number of gradient calculation) is 8. The other test problems are from Hock and Schittkowski (1981). The numerical results are listed in Table 1. However, we do not list the solutions and the residuals given by our method since the solutions given by our method are the same as the solutions given in Hock and Schittkowski (1981). Comparing the results given by our method with the results in Liu and Yuan (2000), our algorithm is comparable to VMCWD and Liu and Yuan's method. Although our algorithm is proposed for optimization problems with inequality constraints, it can handle equality constraint problem, for example, we can solve HS42 which has only equality constraints. Therefore, our method can solve optimization problem with general constraints.

Table 1

Problem	IF	IG
HS22	7	6
HS42	59	26
HS43	55	26
HS44	4	4
HS76	7	7
HS86	7	5
HS113	19	14

6. Conclusion

In this paper, we proposed a new robust SQP method, which can overcome the difficulties associated with the infeasibility of QP subproblem. In this algorithm, we solve two subprolems— one is a linear programming and the other is a quadratic programming. Since solving a linear programming is very easy, so the method can be implemented easily. Theoretical analysis and numerical experiments show that the method has excellent theoretic properties and notable numerical efficiency.

Acknowledgements: The authors are grateful to the referees for their valuable comments in improving this paper.

References

[1] Boggs, P.T., Tolle, J.W., Wang, P., *On the local convergence of quasi-Newton methods for constrained optimization*, SIAM J. on Control and Optimization, **20**(1982), 161–171.

[2] Boggs, P.T., Kearsley, A.J., Tolle, J.W., *A global convergence analysis of an algorithm for large-scale nonlinear optimization problems*, SIAM J. Optimization, **9**(1999), 833–862.

[3] Bonnans, J.F., Launay, G., *Sequential quadratic programming with penalization of displacement*, SIAM J. Optimization, **54**(1995), 792–812.

[4] Burke, J.V., *A sequential quadratic programming method for potentially infeasible mathematical programs*, J. Math. Annal. Appl. **139**(1989), 319–351.

[5] Burke, J.V., Han, S.P., *A robust SQP method*, Mathematical Programming, **43**(1989), 277–303.

[6] Burke, J.V., Han, S.P., *A Gauss-Newton approach to solving genelized inequalities*, Mathematics of Operation Resaerch, **11**(1986), 632–643.

[7] Coleman, T.F., Conn, A.R., *Nonlinear programming via an exact penalty function: asymptotic analysis*, Mathematical Programming, **24**(1982), 123–136.

[8] DE O. Pantoja, J.F.A., Mayne, D.Q., *Exact penalty function algorithm with simple updating of the penalty parameter*, Journal of Optimization Theory and Application, **69**(1991), 441–467.

[9] Facchinei, F., *Robust recursive quadratic programming algorithm model with global and superlinear convergence properties*, Journal of Optimization Theory and Aplication, **92**(1997), 543–579.

[10] Fletcher, R. *Practical Methods for Optimization, Vol. 2, Constrained Optimization* (John Wiley and Sons. Chichester, 1981).

[11] Han, S.P., *Superlinearly convergent variable metric algorithm for general nonlinear programming problems*, Mathematical Programming, **11**(1976), 263–282.

[12] Han, S.P., *A global convergence method for nonlinear programming*, Journal of Optimization Theory and Application, **22**(1977), 297–309.

[13] Hock, W., Schittkowski, K., *Test examples for nonlinear programming codes*, Lecture notes in Eco. and Math. Systems 187, 1981.

[14] Liu, X.W., Yuan, Y., *A robust algorithm for optimization with general equality and inequalty constraints*, SIAM J. on Scientific and Computing, **22**:2(2000), 517–534.

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

[16] Mayne, D.Q., Polak, E., *A superlinealy convergent algorithm for constrained optimization problems*, Mathematical Programming Study, **16**(1982), 45–61.

[17] Powell, M.J.D., *Algorithm for nonlinear constraints that use Lagrange functions*, Mathematical Programming, **14**(1978), 224–248.

[18] Powell, M.J.D., *A fast algorithms for nonlinearly constrained optimization calculations*, in Proc. 1977 Dundee Biennial Conference on Numerical Analysi, Watson.G.A., (ed.), Springer-Verlag, Berlin, 1978, 144–157.

[19] Powell, M.J.D., *The convergence of variable metric methods for nonlinear constrained optimization calculations*, In Nonlinear Programming 3, O.L. Mangasarian, R.R. Meyer and S.M. Robinson, eds., Academic Press, Newyork, 1978, 27–63.

[20] Powell, M.J.D., *Variable metric methods for constrained optimization*, in Bächen etc. (ed.), Mathematical programming–The state of art; Springer-Verlag, Berlin, 1983, 288–311.

[21] Rockafeller, R.T., *Convex Analysis*, Princeton University Press, Princeton, New York, 1970.

[22] Sahba, M., *Globally convergent algorithm for nonlinearly constrained optimization problems*, Journal of Optimization Theory and Application, **52**(1987), 291–309.

[23] Schittkowski, K., *The nonlinear programming method of Wilson, Han and Powell with an augmented Lagrange type line search function, Part 1: convergence analysis*, Numer. Math. **38**(1981), 83–114.

[24] Schittkowski, K., *On the convergence of a sequential quadratic programming method with an augmented Lagrange line search function*, Math. Operations for Sci. Statist., Ser. Optimization, vol. 14, pp. 197–216.

[25] Wilson, R.B.,*A simplicial algorithm for concave programming*, Ph.D. thesis, Harvard University, Cambridge, MA, 1963.

[26] Yuan, Y.X., *A new trust region algorithm for nonlinear optimization*, Technical Report, Computing Center, Academy of Chinese Science, Beijing, China, 1992.

[27] Yuan, Y.X., *Local convergence and numerical results of a new trust region algorithm*, Technical Report, Computing Center, Academy of Chinese Science, Beijing, China, 1993.

[28] Yuan, Y., Sun, W., *Theories and Methods for Optimization*, (in Chinese) Scientific Publish House, Beijing, 1997.

[29] Zhang, J.L., Wang, C.Y., *A modified SQP method and its superlinear convergence*, OR Transaction(Chinese), **4**(2000), 32–40.

[30] Zhao, R.A, Wu, F., *Nonlinear Optimization Theorem and Methods*, (in Chinese) Zhejiang Scientific Publish House, China, 1992.

[31] Zhou, G.L., *A modified SQP method and its global convergence,* J. Global Optimization, **11**(1997), 193–205.