IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 2, No 2, March 2013
ISSN (Print): 1694-0814 | ISSN (Online): 1694-0784
www.IJCSI.org

A Superlinearly feasible SQP algorithm for Constrained Optimization

Zhijun Luo[1], Zhibin Zhu[2] and Guohua Chen[1]

[1] Department of Mathematics & Applied Mathematics, Hunan University of Humanities, Science and Technology, Loudi, 417000, P. R. China

[2] School of Computing Science &Mathematics, Guilin University of Electronic Technology, Guilin,541004, P.R.China

Abstract

This paper is concerned with a Superlinearly feasible SQP algorithm algorithm for general constrained optimization. As compared with the existing SQP methods, it is necessary to solve equality constrained quadratic programming sub-problems at each iteration, which shows that the computational effort of the proposed algorithm is reduced further. Furthermore, under some mild assumptions, the algorithm is globally convergent and its rate of convergence is one-step superlinearly.

Keywords: *Constrained Optimization, SQP Algorithm, Global convergence, Superlinear convergence rate.*

1. Introduction

Optimization deals with selecting the best of many possible decisions in real-life environment, constructing computational methods to find optimal solutions, exploring the theoretical properties, and studying the computational performance of numerical algorithms implemented based on computational methods. It is widely and increasingly used in science, management, engineering, economics and other areas. Many Optimization algorithmic and theoretical techniques have been developed and applied(such as [1]-[5],etc). As an iterative method, sequential quadratic programming (SQP) method is more robust and effective for solving constrained optimization problems (see (see[6]-[12]).

We consider the following nonlinear programs:

$$\begin{aligned} &\min\ f(x) \\ &s.t.\quad g_i(x)\le 0,\ i\in I=(1,2,\ldots m) \\ &\qquad\ \ g_i(x)=0,\ i\in E=(m+1,m+2,\ldots m+j). \end{aligned} \tag{1}$$

Where $f(x),\ g_i(x): R^n \to R (i\in I\cup E)$ are continuously differentiable functions.

It generates iteratively the main search direction d_{Qp} by solving the following quadratic programming (QP) sub-problem:

$$\begin{aligned} &\min\ \ \nabla f(x)^T d+\frac{1}{2}d^T Hd \\ &s.t.\quad g_i(x)+\nabla g_i(x)^T d\le 0,\ i\in I, \\ &\qquad\ \ g_i(x)+\nabla g_i(x)^T d=0,\ i\in E. \end{aligned} \tag{2}$$

Where $H\in R^{n\times n}$ is a symmetric positive definite matrix. However, in traditional SQP algorithm, there are two serious drawbacks: 1) SQP algorithms require that the related QP subproblem (2) must be solvable at each iteration. Obviously, this is difficult. 2) There exists Maratos effect [13], that is to say, the unit step-size cannot be accepted although the iterate points are close enough to the optimum of (1).

In [8], a scheme of feasible sequential quadratic programming (FSQP) method is proposed to deal with those shortcomings. Their scheme considers the following inequality constrained problem

$$\begin{aligned} &\min\quad F_c(x)=f(x)-c\sum_{i=m+1}^{m+j} g_i(x) \\ &s.t.\quad g_i(x)\le 0,\ i\in L\triangleq I\cup E. \end{aligned} \tag{3}$$

The SQP direction d_{Qp} is defined as the unique solution of the QP

$$\begin{aligned} &\min\ \ \nabla F_c(x)^T d+\frac{1}{2}d^T Hd \\ &s.t.\quad g_i(x)+\nabla g_i(x)^T d\le 0,\ \ i\in I\cup E. \end{aligned}$$

Where c is an appropriate parameters. Generally, the computational effort of a inequality constraints QP problem is much larger than that of equality constraints P. Spellucci[10] proposed a new method, the d_0 is obtained by solving QP sub-problem with only equality constraints:

$$\begin{aligned} &\min\ \nabla f(x)^T d+\frac{1}{2}d^T Hd \\ &s.t.\quad g_j(x)+\nabla g_j(x)^T d=0,\ j\in A\subseteq I, \end{aligned}$$

where the so-called working set $A\subseteq I$ is suitably determined. If $d_0=0$ and $\lambda\ge 0$ (λ is said to be the corresponding KKT multiplier vector.), the algorithm stops. The most advantage of these algorithms is merely necessary to solve QP sub-problems with only equality constraints. However, if $d_0 = 0$, but $\lambda < 0$, the algorithm will not implement successfully. Recently, in[11] Consider the following QP sub-problem:

$$\min \ \nabla f(x)^T d + \frac{1}{2} d^T H d$$
$$s.t. \ \ p_j(x) + \nabla g_j(x)^T d = 0, \ j \in L.$$

Where $p_j(x)$ is a suitable vector, L is a suitable approximate active, which guarantees to hold that if $d_0 = 0$, then x is a KKT point of (1), i.e., if $d_0 = 0$, then it holds that $\lambda \ge 0$.

We will develop an improved feasible SQP method for solving optimization problems based on the one in [11]. The traditional FSQP algorithms, in order to prevent iterates from leaving the feasible set, and avoid Maratos effect, it needs to solve two or three QP sub-problems like (3). In our algorithm, per single iteration, it is only necessary to solve equality constrained quadratic programming subproblems and systems of linear equations. Obviously, it is simpler to solve the equality constrained QP problem than to solve the QP problem with inequality constraints. In order to void the Maratos effect, a height-order correction direction is computed by an equality constrained QP problem. Furthermore, its global and superlinear convergence rate are obtained under some suitable conditions. In the end, some limited numerical experiments are given to show that the algorithm is effective.

This paper is organized as follows: In Section 2, we state the algorithm; The well-defined of our approach is also discussed, the accountability of which allows us to present global convergence guarantees under common conditions in Section 3, while in Section 4 we deal with superlinear convergence. Finally, in Section 5, some numerical experiments are implemented.

2. Description of Algorithm

For the sake of simplicity, we denote

$$X = \{x \in R^n \mid g_i(x) \le 0, \ i \in I; g_i(x) = 0, \ i \in E\},$$
$$I(x) = \{i \in I \mid g_i(x) = 0\}, \ \ L(x) = I(x) \cup E,$$
$$X_+ = \{x \in R^n \mid g_i(x) \le 0, \ i \in L\}.$$

The following general assumptions are true throughout the paper.

A1 Feasible sets of (1) and (3) are nonempty, i.e., $X \ne \Phi, \ X_+ \ne \Phi$, and functions $f(x)$, $g_i(x), \ i \in L$ are twice continuously differentiable.

A2 $\forall \ x \in X$, the vectors $\{\nabla g_i(x), \ i \in L(x)\}$ are linearly independent.

Given a point $x \in X_+$, define the following matrices

$$N(x) = (\nabla g_i(x), i \in L), \ \ D(x) = diag(D_i(x), i \in L),$$
$$D_i(x) = \begin{cases} g_i^2(x), & i \in I, \\ 0, & i \in E. \end{cases} \qquad (4)$$
$$B(x) = (N(x)^T N(x) + D(x))^{-1} N(x)^T, \qquad \pi(x) = -B(x) \nabla f(x).$$

For the meaning of above matrices, we establish the following result.

Lemma 1

For all $x \in X_+$, the matrix $(N(x)^T N(x) + D(x))$ is positive definite, thereby, $(N(x)^T N(x) + D(x))$ is nonsingular.

Proof. For all $0 \ne y \in R^{m+j}$, we have

$$\Gamma \triangleq y^T (N(x)^T N(x) + D(x)) y = \|N(x) y\|^2 + \sum_{i \in L} D_i(x) y_i^2 \ge 0.$$

Suppose by contradiction that $\Gamma = 0$, then

$$N(x) y = \sum_{i \in L} \nabla g_i(x) y_i = 0, \ \ D_i(x) y_i = 0, \ i \in L.$$

From the definition of the matrix $D(x)$, it gets that

$$y_i = 0, \ i \in L \setminus L(x)$$

So, we obtain that

$$\sum_{i \in L(x)} \nabla g_i(x) y_i = N(x) y = 0 \ \ .$$

From the assumption A2, it sees that

$$y_i = 0, \ i \in L(x)$$

Thereby, it holds that $y = 0$, which is a contradiction.

So, $\Gamma > 0$, i.e., the matrix $(N(x)^T N(x) + D(x))$ is positive definite.

Lemma 2

If the parameter $c > \max\{|\pi_i(x)| : i \in E\}$, then x is a $K-T$ point for (1) if and only if $K-T$ point for (3) .

Proof. If x is a $K-T$ point for (1), then $x \in X_+$, and there exist a multiplier vector $u = (u_i) \in R^{m+j}$ such that

$$\begin{cases} \nabla f(x) + \sum_{i \in L} u_i \nabla g_i(x) = 0, \\ u_i \nabla g_i(x) = 0, \ \ u_i \ge 0, \ \ i \in I, \\ g_i(x) = 0, \ \ i \in E. \end{cases}$$

Thereby, it holds that

$$\begin{cases} \nabla F_c(x) + \sum_{i \in I} u_i \nabla g_i(x) + \sum_{i \in E} (u_i + c) \nabla g_i(x) = 0, \\ u_i \nabla g_i(x) = 0, \ \ u_i \ge 0, \ \ i \in I, \\ (u_i + c) g_i(x) = 0, \ \ i \in E. \end{cases}$$

Furthermore, in accordance with the definition of the matrix $D(x)$, we have

$$\begin{cases} \nabla f(x) + N(x) u = 0, \\ D(x) u = 0. \end{cases}$$

Which implies that

$$u = -\left((N(x)^T N(x) + D(x))\right)^{-1} N(x)^T \nabla f(x)$$
$$= -B(x)\nabla f(x) = \pi(x).$$

Thus,

$$c > |\pi_i(x)| = |u_i|, \ \ i.e., \ u_i + c = \pi_i(x) + c > 0 \ \ i \in E.$$

Thereby, it holds that x is a K-T point of (3).

On the other hand, let x be a K-T point of (3), then there exists a multiplier vector $\tilde{u} = (\tilde{u}_i, \ i \in L)$ such that

$$\begin{cases} \nabla F_c(x) + \sum_{i\in I} \tilde{u}_i \nabla g_i(x) + \sum_{i \in E} \tilde{u}_i \nabla g_i(x) = 0, \\ \tilde{u}_i \nabla g_i(x) = 0, \ \ \tilde{u}_i \ge 0, \ \ g_i(x) \le 0, \ i \in L. \end{cases}$$

Therefore,

$$\nabla f(x) + \sum_{i\in I} \tilde{u}_i \nabla g_i(x) + \sum_{i\in E} (\tilde{u}_i - c)\nabla g_i(x) = 0.$$

Denote

$$v = (v_i, \ i \in L), \ \ v_i = \begin{cases} \tilde{u}_i, & i \in I \\ \tilde{u}_i - c, & i \in E, \end{cases}$$

then, it is obvious that

$$\nabla f(x) + N(x)v = 0, \ \ D(x)v = 0.$$

So,

$$v = -\left((N(x)^T N(x) + D(x))\right)^{-1} N(x)^T \nabla f(x)$$
$$= -B(x)\nabla f(x) = \pi(x).$$

Thus, we obtain,

$$\tilde{u}_i = v_i + c = \pi_i(x) + c > 0, \ \ i \in E.$$

And

$$\begin{cases} \nabla f(x) + \sum_{i\in I} \tilde{u}_i \nabla g_i(x) + \sum_{i\in E} (\tilde{u}_i - c)\nabla g_i(x) = 0, \\ \tilde{u}_i \nabla g_i(x) = 0, \ \ \tilde{u}_i \ge 0, \ g_i(x) \le 0, \ i \in I, \\ g_i(x) = 0, \ \ i \in E. \end{cases}$$

which implies that x is a K-T point of (1).

Based on Lemma 2, in the sequel, we consider to solve the problem (3). Given $x \in X_+$, an appropriate index set $L(x) \subseteq J \subseteq L$,

Sub-algorithm A:

Computation of an approximate active set J_k

Step 1 For the current point $x^k \in X$, set $j = 0, \ \varepsilon_{k,j} = \varepsilon_0 \in (0,1)$.

Step 2 Compute

$\tilde{J}_{k,j} = \{i \in I \mid -\varepsilon_{k,j} \le g_i(x^k) \le 0\}, \ J_{k,j} = \tilde{J}_{k,j} \cup E, \ A_{k,j} = (\nabla g_i(x^k), j \in J_{k,j})$

If $\det(A_{k,j}^T A_{k,j}) \ge \varepsilon_j(x^k)$, , let $\tilde{J}_k = \tilde{J}_{k,j}, \ J_k = J_{k,j}, \ A_k = A_{k,j}, \ j_k = j,$ STOP. Otherwise go to Step 3.

Step 3 Let $j = j+1, \ \varepsilon_{k,j} = \frac{1}{2}\varepsilon_{k,j-1}$, and go to Step2.

Lemma 3

For any iteration k, there is no infinite cycle in Sub-algorithm A. Moreover, if $\{x^k\}_{k\in K} \to x^*$, then there exists a constant $\bar{\varepsilon} > 0$,such that $\varepsilon_{k,i_k} \ge \bar{\varepsilon}$, for $k \in K$; k large enough.

Now, the algorithm for the solution of the problem (1) can be stated as follows.

Algorithm A :

Step 0: Initialization:

Given a starting point $x^0 \in X$, and an initial symmetric positive definite matrix $H_0 \in R^{n\times n}$ .Choose parameters $\alpha \in (0, \frac{1}{2}), \ \tau \in (2,3), v > 2, \ \delta \in (2, v), \ \xi > 0, \ \varepsilon > 0,$

Set $k = 0$;

Step1: Computation of an approximate active set J_k by Sub-algorithm A ;

Step2: Update c_k Computation:

From (4), calculate $B_k = B(x^k), \ \pi^k = \pi(x^k),$ let

$$a_k = \max\{|\pi_i^k|, \ i \in E\},$$
$$c_k = \begin{cases} \max\{a_k + c_0, \ c_{k-1} + \varepsilon\}, & c_{k-1} < a_k + c_0, \\ c_{k-1}, & c_{k-1} \ge a_k + c_0. \end{cases}$$

Step3: Computation of the search direction:

3.1 Solve the following equality constrained QP subproblem:

$$\begin{aligned} \min \ \ & \nabla F_c(x^k) + \frac{1}{2} d^T H_k d \\ s.t. \ \ & p_i(x^k) + \nabla g_i(x^k)^T d = 0, \ i \in J_k. \end{aligned} \qquad (5)$$

Where, $p_i(x^k) = \begin{cases} -\pi_i(x^k), & i \in J \setminus E, \ \ \pi_i(x^k) < 0, \\ g_i(x^k), & i \in J \setminus E, \ \ \pi_i(x^k) \ge 0. \end{cases}$

Let $(d_0^k, \ \mu^k)$ be the corresponding K-T point pair, if $d_0^k = 0$, **STOP**.

3.2 Compute the feasible direction:

3.2.1 Solve the following equality constrained QP subproblem at x^k

$$\begin{aligned} \min \ \ & \nabla F_c(x^k) + \frac{1}{2} d^T H_k d \\ s.t. \ \ & g_i(x^k) + \nabla g_i(x^k)^T d = -\| d_0^k \|^v, \ i \in J_k. \end{aligned} \qquad (6)$$

Let $(d_1^k, \ \lambda^k)$ be the corresponding K-T point pair, if $\nabla F_c(x^k)^T d_1^k \le -\xi \| d_1^k \|^v$, set $d^k = d_1^k$, go to step 3.3;

3.2.2 Solving the following linear problem

$$\begin{aligned} &\min \ z \\ &s.t. \ \nabla F_c(x^k)^T d \le z, \\ &\quad g_i(x^k) + \nabla g_i(x^k)^T d \le z, \ i \in J_k. \end{aligned}$$

Let (d_2^k, z_k) be the solution, set $d^k = -z_k d_2^k$;

3.3 Computation of the high-order revised direction: Solve the following equality constrained QP

$$\begin{aligned} &\min \quad \nabla F_c(x^k) + \frac{1}{2}(d^k + d)^T H_k (d^k + d) \\ &s.t. \quad g_i(x^k + d^k) + \nabla g_i(x^k)^T d = -\| d^k \|^\tau, \ i \in J_k. \end{aligned}$$

Let $(\tilde{d}^k, \tilde{\lambda}^k)$ be the corresponding K-T point pair, if $\| \tilde{d}^k \| > \| d^k \|$, set $\tilde{d}^k = 0$;

Step4: The line search:

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

$$F_{c_k}(x^k + td^k + t^2 \tilde{d}^k) \le F_{c_k}(x^k) + \alpha t \nabla F_{c_k}(x^k)^T d^k, \quad (7)$$

$$g_i(x^k + td^k + t^2 \tilde{d}^k) \le 0, \qquad i \in L. \quad (8)$$

Step5: Update:

Obtain H_{k+1} by updating the positive definite matrix H_k using some quasi-Newton formulas. Set $x^{k+1} = x^k + t_k d^k + t^2 \tilde{d}^k$, and $k = k+1$. Go back to step 1.

3. Global convergence of algorithm

In this section, firstly, it is shown that Algorithm A given in Section 2 is well-defined, then we prove the global convergence of Algorithm A.

The following assumptions are needed in the proof of the global convergence:

A3 $\{x^k\}$ is bounded, which is the sequence generated by the algorithm A, and there exist constants $b \ge a > 0$, such that $a \| y \|^2 \le y^T H_k y \le b \| y \|^2$, for all k and all $y \in R^n$.

According to Algorithm A, it is similar to the proof of lemma 3.1 in [11].we can get the following conclusions.

Lemma 4

Let (d_0^k, μ^k) be the solution of (5). If $d_0^k = 0$, then x^k is a K-T point of (1). If $d_0^k \ne 0$, the direction d^k computed in step 3.2 is a feasible descent direction of (1) at x^k.

Lemma 5

The line search in step4 yields a stepsize $t_k = (\frac{1}{2})^i$ for some finite $i = i(k)$.

Proof.

It is a well-known result according to Lemma 4 and $\alpha \in (0, \frac{1}{2})$. For (7),

$$\begin{aligned} s &\triangleq F_{c_k}(x^k + td^k + t^2 \tilde{d}^k) - F_{c_k}(x^k) - \alpha t \nabla F_{c_k}(x^k)^T d^k \\ &= \nabla F_{c_k}(x^k)^T (td^k + t^2 \tilde{d}^k) + o(t) - \alpha t \nabla F_{c_k}(x^k)^T d^k \\ &= (1-\alpha) t \nabla F_{c_k}(x^k)^T d^k + o(t). \end{aligned}$$

For (8), if

$$i \notin I(x^k), \ g_j(x^k) < 0;$$
$$i \in I(x^k), \ g_j(x^k) = 0, \ \nabla g_j(x^k)^T d^k < 0,$$

so we have

$$\begin{aligned} &g_i(x^k + td^k + t^2 \tilde{d}^k) = \nabla g_i(x^k)^T (td^k + t^2 \tilde{d}^k) + o(t) \\ &= \alpha t \nabla g_i(x^k)^T d^k + O(t). \end{aligned}$$

It is shows that the results hold for t small enough.

Let the sequence $\{x^k\}$ be generated by the algorithm A, without loss of generality, Since there are only finitely many choices for sets $J_k \subseteq L$, and the definition of π^k, we might as well assume that there exists a subsequence K, such that

$$\begin{aligned} &x^k \to x^*, \ \pi^k \to \pi^*, \ p^k \to p^*, \\ &H_k \to H_*, \ J_k \equiv J, \ k \in K, \end{aligned} \quad (9)$$

where J is a constant set. Moreover, according to the assumptions A3, the sequence $\{d_0^k, d_1^k, d^k, \mu^k, \lambda^k\}$ is bounded. Thereby, we might as well assume, too, that

$$\begin{aligned} &d_0^k \to d_0^*, \ d_1^k \to d_1^*, \ d^k \to d^*, \\ &\mu^k \to \mu^*, \ \lambda^k \to \lambda^*, \ k \in K. \end{aligned} \quad (10)$$

Lemma 6

If the sequence $\{x^k\}$ is bounded, then there exists a positive integer k_0, such that

$$c_k = c_{k_0} \triangleq c, \quad for\ all\ k \ge k_0.$$

Proof.

Suppose by contradiction that the result is not true. According to step 2, there exists a sequence $K_1 (K_1 \subseteq K), \ |K_1| = \infty$, such that,

$$c_k = \max\{a_k + c_0, \ c_{k-1} + \varepsilon\}, \ c_{k-1} < a_k + c_0, \ k \in K_1.$$

While $\varepsilon > 0$, hence the definition of a_k in step 2 shows that $c_k \to \infty, \ k \in K_1$, and $\{c_k\}$ is monotone increasing, thereby, it holds that

$$c_k \to \infty, \ k \to \infty.$$

In addition, because the functions $f(x), \ g_i(x) (i \in L)$ are continuously differentiable and the sequence $\{x^k\}$ is bounded, the fact $\pi^k \to \pi^*, k \in K$ shows that

$$\sup\{a_k, \ k \in K\} < \infty.$$

Hence, we have $\lim_{k\in K_1} c_{k-1} < \infty$, which is contradict. Thus, the result is true. Furthermore, in the sequel, we always assume that $c_k \equiv c$. Obviously, it holds that

$$c \geq |\pi_i| + c_0 > |\pi_i^*|, \ \ for\ all\ i \in E.$$

Theorem 1

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

Proof.

The first statement is easy to show, since the only stopping point is in step 3. Thus, assume that the algorithm generates an infinite sequence $\{x^k\}$, (9) and (10) holds. According to Lemma 4, it is only necessary to prove that $d_0^* = 0$. Suppose by contradiction that $d_0^* \neq 0$. From lemma 3, we have $L(x^*) \subseteq J$ and $\det(A_*^T A_*) \geq \bar{\varepsilon}$, where $A_* = (\nabla g_i(x^*),\ i \in J)$. it is not difficult to prove that $d_1^* \neq 0$ is the sole solution of the following QP

$$\min \quad \nabla F_c(x^*) + \frac{1}{2} d^T H_* d$$

$$s.t. \quad g_i(x^*) + \nabla g_i(x^*)^T d = -\| d_0^* \|^\nu,\ i \in J .$$

Thereby, it holds that

$$\nabla F_c(x^*)^T d^* < 0,\ \ \nabla g_i(x^*)^T d^* < 0,\ i \in L(x^*) \subseteq J.$$

Thus, it is easy to see that the step-size t_k obtained in step 4 are bounded away from zero on K, *i.e.*

$$t_k \geq t_* = \inf\{t_k,\ k \in K\} > 0,\ k \in K.$$

In addition, from (7) and lemma 4, it is true that $\{F_c(x^k)\}$ is monotonically decreasing, which implies that

$$F_c(x^k) \to F_c(x^*),\ k \to \infty, k \in K ,$$

So, it holds that

$$0 = \lim_{k\in K}(F_c(x^{k+1}) - F_c(x^k)) \leq \lim_{k\in K} \alpha t_k \nabla F_c(x^k)^T d^k$$

$$\leq \frac{1}{2}\alpha t_* \nabla F_c(x^*)^T d^* < 0.$$

It is a contradiction, which shows $d_0^* = 0$. Thus, x^* is a K-T point of (1).

4. The rate of convergence

Now we prove the sequence $\{x^k\}$ generated by the algorithm is one-step superlinearly convergent. For this purpose, we state some stronger regularity assumptions.

A5 The second-order sufficiency conditions with strict complementary slackness are satisfied at the K-T point x^* and the corresponding multiplier vector u^*.

A 6 $H_k \to H_*,\ k \to \infty,$ *and* H_* is positive definite on the subspace $Y(x^*)$, where

$$Y(x^*) = \{d \in R^n \mid \nabla g_i(x^*)^T d = 0,\ i \in L(x^*)\}.$$

A 7 Let the sequence of matrixes $\{H_k\}$ satisfy that

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

$$\Leftrightarrow \| P_k(H_k - \nabla_{xx}^2 L(x^k, u^k))d^k \| = o(\| d^k \|),$$

where

$$P_k = I_n - A_k(A_k^T A_k)^{-1} A_k^T,$$

$$\nabla_{xx}^2 L(x^k, \lambda^k) = \nabla^2 F_c(x^k) + \sum_{i\in L(x^*)} \lambda_i^k \nabla^2 g_i(x^k),$$

$$\nabla_{xx}^2 L(x^*, u^*) = \nabla^2 f(x^*) + \sum_{i\in L} u_i^* \nabla^2 g_i(x^*).$$

The first task is to show that, the entire sequence $\{x^k\}$ converges to x^*.

Lemma 7

Under above conditions, the entire sequence $\{x^k\}$ converges to x^*, i.e. $x^k \to x^*,\ k \to \infty.$

Proof. When $\nabla F_c(x^k)^T d_1^k \leq -\xi \| d_1^k \|^\nu$, holds, we have $F_c(x^{k+1}) - F_c(x^k) \leq \alpha t_k\ \nabla F_c(x^k)^T d^k \leq -\alpha t_k \xi \| d^k \|^\nu$.

When $d^k = -z_k d_2^k$, from the assumptions A3, there exists a constant $c > 0$,such that $-z_k \geq c\|d^k\|$. Thereby, from Lemma 4, it holds that

$$F_c(x^{k+1}) - F_c(x^k) \leq \alpha t_k\ \nabla F_c(x^k)^T d^k$$

$$\leq -\alpha t_k z_k^2 \leq -\alpha t_k c^2 \| d^k \|^2 .$$

Through the above analysis, we have

$$F_c(x^{k+1}) - F_c(x^k) \leq -t_k \alpha \| d^k \|^2 \min\{c^2, -\xi \| d^k \|^{\nu-2}\},\ \forall k.$$

From $F_c(x^k) \to F_c(x^*),\ k \to \infty$, it is easy to see that

$$t_k \|d^k\| \to 0,\ k \to \infty.$$

So, we have

$$\|x^{k+1} - x^k\| \leq t_k \|d^k\| + t_k^2 \|\tilde{d}^k\| \leq 2t_k \|d^k\| \to 0,\ k \to \infty.$$

Thereby, according to the assumptions A5 and Proposition 4.1 in [13], we have $x^k \to x^*,\ k \to \infty.$

Lemma 8

For k large enough, it holds that

1) $J_k \equiv L(x^*) = I(x^*) \cup E,\ d_0^k \to 0$,

2) π^k is obtained by (4) satisfy that $\pi^k \to u^*$,

3) $\{d_0^k,\ d_1^k,\ d^k,\ \tilde{d}^k,\ \lambda^k\}$ obtained in step 3 satisfy that

$$d^k \equiv d_1^k,\ \| d^k \| \sim \| d_0^k \|,\ \ \| \tilde{d}^k \| \sim O(\| d^k \|^2),$$
$$\mu_i^k \to u_i^*(i \in I(x^*)),\ \ \mu_i^k \to u_i^* + c > 0 (i \in E),\ k \to \infty,$$
$$\lambda_i^k \to u_i^*(i \in I(x^*)),\ \lambda_i^k \to u_i^* + c (i \in E),\ \ k \to \infty.$$

Proof.

1) Firstly, from Theorem 1 and Lemma 7, it holds that $d_0^k \to 0,\ k \to \infty$. Then, to prove that $J_k \equiv L(x^*) = I(x^*) \cup E$. According to definitions of J_k and $\tilde{J}_k$ Sub-algorithm A, we only prove that $\tilde{J}_k = I(x^*)$, On one hand, from Lemma 3, we obtain, for k large enough, that $I(x^*) \subseteq \tilde{J}_k$. On the other hand, we suppose by contradiction that $\tilde{J}_k \subseteq I(x^*)$ is not holds, then there exist constants i_0 and $\phi > 0$, such that

$$g_{i_0}(x^*) < -\phi < 0,\ \ i_0 \in \widetilde{J_k}.$$

According to $d_0^k \to 0,\ x^k \to x^*, k \to \infty$ and the continuity of $g_{i_0}(x)$, for k large enough, it holds that

$$p_{i_0}^k + \nabla g_{i_0}(x^k)^T d_0^k$$
$$= \begin{cases} -\pi_{i_0}(x^k) + \nabla g_{i_0}(x^k)^T d_0^k \geq -\frac{1}{2}\pi_{i_0}(x^k) > 0, & \pi_{i_0}(x^k) < 0 \\ g_{i_0}(x^k) + \nabla g_{i_0}(x^k)^T d_0^k \leq -\frac{1}{2}\phi < 0, & \pi_{i_0}(x^k) \geq 0, \end{cases}$$

which is contradictory with the fact $i_0 \in \widetilde{J_k}$,then, $\tilde{J}_k \subseteq I(x^*)$, i.e. $\tilde{J}_k = I(x^*)$, so, $J_k \equiv L(x^*)$.

2) From the definition of π^k, we have

$$\pi(x^k) = -(N(x^k)^T N(x^k) + D(x^k))^{-1} N(x^k)^T \nabla f(x^k)$$
$$= -B(x^k)\nabla f(x^k),$$

Then, $k \to \infty$, it holds that

$$\pi(x^k) \to -(N(x^*)^T N(x^*) + D(x^*))^{-1} N(x^*)^T \nabla f(x^*)$$
$$= -B(x^*)\nabla f(x^*).$$

In addition, the fact (x^*, u^*) is a K-T point pair of (1), it is implies that

$$\begin{cases} \nabla f(x^*) + \sum_{i \in L} u_i^* \nabla g_i(x^*) = \nabla f(x^*) + N^* u^* = 0, \\ u_i^* \nabla g_i(x^*) = 0,\ \ g_i(x^*) \leq 0,\ u_i^* \geq 0,\ \ i \in I, \\ g_i(x^*) = 0,\ \ i \in E. \end{cases}$$

Denote

$$D_* = diag(D_*, i \in L),\ D_i(x) = \begin{cases} g_i^2(x^*), & i \in I, \\ 0, & i \in E. \end{cases}$$

Then $D_k \to D_* (k \to \infty)$, and $D_* u^* = 0$, so, it shows that

$$u^* = -(N(x^*)^T N(x^*) + D(x^*))^{-1} N(x^*)^T \nabla f(x^*)$$
$$= -B(x^*)\nabla f(x^*).$$

According to definitions of π^k, it holds that $\pi^k \to u^*$.

3) it is similar to the proof of Lemma4.2 in [11].

Lemma 9

For k large enough, $t \equiv 1$, i.e. $x^{k+1} = x^k + d^k + \tilde{d}^k$.

Proof. It is only necessary to prove that

$$F_{c_k}(x^k + d^k + \tilde{d}^k) \leq F_{c_k}(x^k) + \alpha \nabla F_{c_k}(x^k)^T d^k,$$
$$g_i(x^k + d^k + \tilde{d}^k) \leq 0,\ \ \ i \in L.$$

it is similar to the proof of Lemma4.3 in [10].

In view of Lemma 9 and Theorem 5.2 in [9], we may obtain the following theorem:

Theorem 2

Under all above-mentioned assumptions, the algorithm is superlinearly convergent, i.e., the sequence $\{x^k\}$ generated by the algorithm satisfies that

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

5. Numerical experiments

In this section, we carry out numerical experiments based on the Algorithm A. The code of the proposed algorithm is written by using MATLAB 7.0 and utilized the optimization toolbox. The results show that the algorithm is effective. During the numerical experiments, it is chosen at random some parameters as follows:

$$\varepsilon_0 = 0.5,\ \alpha = 0.25,\ \tau = 2.25,\ v\text{-}3,\ \delta{=}2.5,\ \xi{=}0.5\ H_0 = I,$$

Where I is the unit matrix. H_k is updated by the BFGS formula [7].

$$H_{k+1} = BFGS(H_k, s^k, y^k),$$

Where,

$$s^k = x^{k+1} - x^k,\ y^k = \theta \hat{y}^k + (1-\theta) H_k s^k,$$
$$\hat{y}^k = \nabla F_{c_k}(x^{k+1}) - \nabla F_{c_k}(x^k) + \sum_{j=1}^m u_j^k (\nabla g_j(x^{k+1}) - \nabla g_j(x^k)),$$
$$\theta = \begin{cases} 1, & if\ \hat{y}^{kT} s^k \geq 0.2 (s^k)^T H_k s^k, \\ \frac{0.8 (s^k)^T H_k s^k}{(s^k)^T H_k s^k - \hat{y}^{kT} s^k}, & otherwise. \end{cases}$$

This algorithm has been tested on some problems from Ref.[14], a feasible initial point is either provided or obtained easily for each problem. The results are summarized in Table 1—Tabe 3. The columns of this table has the following meanings:

No.: the number of the test problem in [14];
n: the number of variables;
|C1|, |C2|: give the number of inequality and equality constraints, respectively;
NT : the number of iterations;
CPU: the total time taken by the process (unit: millisecond);
FV : the final value of the objective function.

TABLE I. *THE DETAIL INFORMATION OF NUMERICAL EXPERIMENTS*

NO.	n, |C1|, |C2|	NT	CPU
HS32	3, 4, 1	10	0
HS63	3, 3, 2	46	0
HS81	5, 10, 3	48	10
HS100	7, 4, 0	15	62
HS113	10, 8, 0	95	50

TABLE II. *THE APPROXIMATE OPTIMAL SOLUTION* x^* *FOR TABLE*

NO.	the approximate optimal solution x^*
HS32	$(0.0000000000, 0.0000000000, 1.0000000000)^T$
HS63	$(3.5121213421, 0.2169879415, 3.5521711546)^T$
HS81	$(-1.7171435704, 1.5957096902, 1.8272457529, -0.7636430782, -0.7636430782)^T$
HS100	$(2.3304993729, 1.9513723729, -0.4775413929, 4.3657262336, -0.6244869704, 1.0381310185, 1.59422671167)^T$
HS113	$(2.1719963713, 2.3636829737, 8.7739257385, 5.0959844880, 0.9906547650, 1.4305739789, 1.3216442082, 9.82872580788, 8.2800916701, 8.3759266639)^T$

TABLE III. *THE FINAL VALUE OF THE OBJECTIVE FUNCTION FOR TABLE*

NO.	FV
HS32	1.0000000000E + 00
HS63	9.6171517213E + 02
HS81	5.3949847770E − 02
HS100	6.8063005737E+002
HS113	2.4306209068E + 01

Acknowledgments

This work was supported in part by the National Nature Science Foundation of China (No. 11061011) and the Foundation of Hunan Provincial Education Department Under Grant (NO. 12A077).

References

[1] E.Rafajowicz, K. Stycze, W. Rafajowicz. “A modified filter SQP method as a tool for optimal control of nonlinear systems with spatio-temporal dynamics,” International Journal of Applied Mathematics and Computer Science, 22, 2012, pp. 313-326.

[2] L Cai, D Zhu. “A line search filter inexact SQP method for nonlinear equality constrained optimization,” Journal of Systems Science and Complexity, 25, 2012, pp. 950-963.

[3] X Yan, Q Wu, C Zhang, etc. “An Efficient Function Optimization Algorithm based on Culture Evolution,” International Journal of Computer Science Issues, Vol. 9(5), No 2, 2012, pp .11-18.

[4] J Umamaheswari and Dr.G.Radhamani, “Quadratic Program Optimization using Support Vector Machine for CT Brain Image Classification,” International Journal of Computer Science Issues, Vol. 9(4), No 1, 2012, pp. 305-310.

[5] A. F. Izmailov and M. V. Solodov. “A Truncated SQP Method Based on Inexact Interior-Point Solutions of Sub-problems,” SIAM J. Optim. Vol. 20(5), 2010, pp. 2584-2613.

[6] C.L. Hao, X.W. Liu. “A trust-region filter-SQP method for mathematical programs with linear complementarity constraints,” Journal of Industrial and Management Optimization, vol 7(4), 2011, pp. 1041–1055.

[7] M. J. D. Powell. “A Fast Algorithm for Nonlinearly Constrained Optimization Calculations,” In:Waston, G.A.(ed). *Numerical Analysis. Springer, Berlin*, 1978.

[8] P. T. Boggs, and J. W. Tolle. “A Strategy for Global Convergence in a Sequential Quadratic Programming Algorithm,” SIAM J. Num. Anal., 26, 1989, pp. 600-623.

[9] C.T. Lawrence, A.L. Tits, “A computationally efficient feasible sequential quadratic programming algorithm,” SIAM J. Optimiz. Vol 11, 2001, pp. 1092–1118.

[10] P. Spellucci. “An SQP Method for General Nonlinear Programs Using Only Equality Constrained Subproblems,” Mathematical Programming, vol. 82(3), 1998, pp. 413–448.

[11] Z.B. Zhu “A Feasible Equality Constrained SQP Algorithm for nonlinear programming,” Acta Mathematica Sinica(Chinese Series), Vol 50, No.2, 2007, pp. 281-290.

[12] Z.B.Zhu, S.Wang. “A superlinearly convergent numerical algorithm for nonlinear programming,” Nonlinear Analysis, Vol 13, 2012, pp. 239-2402.

[13] F. Facchinei, S. Lucidi. “Quadratically and superlinearly convergent for the solution of inequality constrained optimization problem,” JOTA , Vol 85, 1995, pp. 265–289.

[14] W. Hock, K. Schittkowski, Test examples for nonlinear programming codes, Lecture Notes in Economics and Mathematical Systems, vol. 187, Springer, Berlin, 1981.

Zhijun Luo received his M.S.degree from Guilin University of Electronic Technology, Guilin, China, in 2008. Now he is a lecturer in Hunan University of Humanities, Science and Technology, Loudi, China. His research interests cover optimization theory, algorithms and their applications.

Zhibin Zhu received his Ph.D. degree in Computational Mathematics from Xi'an Jiaotong University, Xi'an, China, in 2004. Now he is a professor in Guilin University of Electronic Technology, Guilin, China, and PhD supervisor in University of Electronic Science and Technology of China, Chengdu, China. His research interests cover optimization theory, algorithms and their applications in the engineering field.

Guohua Chen received his Ph.D. degree from Hunan University, Changsha, China, in 2009. Now he is a associate professor in Hunan University of Humanities, Science and Technology, Loudi, China.. His research interests cover Operations research, management science, computational intelligence, mathematical modeling. He is a senior member of the CCF (China Computer Federation).