International Journal of Pure and Applied Mathematics

Volume 39 No. 1 2007, 125-138

AN SQP ALGORITHM WITH EQUALITY CONSTRAINED SUBPROBLEMS FOR GENERAL CONSTRAINED OPTIMIZATION

Zhijun Luo[1 §], Zhibin Zhu[2], Lirong Wang[3], Ajuan Ren[4]

[1,2,4]School of Computing Science and Mathematics
Guilin University of Electronic Technology
Guilin, 541004, P.R. CHINA
[1]e-mail: ldlzj123@163.com

[3]School of Mathematical Sciences
Guangxi Normal University
Guilin, 541004, P.R. CHINA

Abstract: In this paper, an SQP method is presented to solve general constrained optimization. Firstly the original problem is expanded to parametric programming problems with only inequality constraints, and the expansive problem is equivalent to the original problem if the parameter is suitable. Then an algorithm is proposed which is only necessary to solve one QP subproblem with equality constraints. Thus, the computational cost is reduced. Under some suitable assumptions, the algorithm is proven to be globally and superlinearly convergent.

AMS Subject Classification: 90C30, 65K05

Key Words: general constrained optimization, SQP algorithm, equality constrained quadratic programming, global convergence, superlinear convergence rate

1. Introduction

Sequential quadratic programming (SQP) algorithms are widely acknowledged to be among the most successful algorithms for solving two or three nonlinear

Received: April 18, 2007

§Correspondence author

optimization problems. Because of its superlinear convergence rate, it is a topic of many active researches [10], [4], [8], [6], [3]. However, the traditional algorithms make it necessary to solve relatively complex and highly computational cost QP problems per single iteration, or let the Hessian matrix of the quadratic programming subproblem be positive definite or uniformly positive definite [2], [9], [7]. In order to simplify the structure of the algorithm, weaken hypothesis conditions, reduce the computational cost, and quicken the convergence rate, a lot of authors present many different types algorithms. For example [11] proposed a new SQP method for solving inequality constrained optimization, which is not necessary that the Hessian matrix is positive definite. In this paper we proposed an SQP method for general constrained optimization. Firstly, we make use of the technique which handle the general constrained optimization as an inequality parametric programming in [5]. Then, we make a new quadratic programming with only equality constraints, and propose an SQP type algorithm to solve the general constrained optimization. In the end, under some mild assumptions, we prove that the algorithm is global convergence as well as superlinear convergence.

The plan of the paper is as follows: In Section 2, the algorithm is proposed. In Section 3, we show that the algorithm is globally convergent. While the superlinear convergence rate is analyzed in Section 4.

2. Description of Algorithm

In this paper, we consider the following nonlinear optimization problem.

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g_j(x) \le 0, j \in L_1 = \{1, 2, \cdots, m\}, \\ & g_j(x) = 0, j \in L_2 = \{m+1, \cdots, m+l\}. \end{array} \tag{2.1}$$

Here, $f(x)$, $g_j(x) (j \in L = L_1 \bigcup L_2)$ are continuously differentiable functions. Denote the feasible set $R = \{g_j(x) \le 0,\ j \in L_1;\ g_j(x) = 0, j \in L_2\}$. For some $c > 0$, let $F(x),\ F_c(x) : E^n \longrightarrow E^1$ be defined as follows

$$F(x) = -\sum_{j \in L_2} g_j(x), \qquad F_c(x) = f(x) + cF(x).$$

When $L_2 = \phi$, i.e., $F(x) \equiv 0$, (2.1) is the inequality constrained optimization. Consider the following auxiliary programming

$$\min\{F_c(x) | x \in R_+ = \{x \in E^n | g_j(x) \le 0, j \in L\}\}. \tag{2.2}$$

Obviously, $x \in R$ if and only if $x \in R_+$, and $F(x)=0$. For $x \in R_+$, denote

$$E(x)=\{j \in L_1 \mid g_j(x)=0\} \bigcup L_2.$$

It is well known that standard SQP method for (2.2) generates a decent direction at the point $x \in R_+$ by solving the quadratic programming subproblem

$$\begin{array}{ll} \min & \nabla F_c(x)^T d + \frac{1}{2} d^T H d \\ \text{s.t.} & g_j(x) + \nabla g_j(x)^T d \le 0, j \in L, \end{array} \tag{2.3}$$

where H is the approximate Hessian matrix of Lagrangian function associated with (2.2). However, the direction d is not a feasible direction, and cannot avoid the Maratos effect. So, we define some variants as follows:

$$\begin{array}{c} N(x)=\left(\nabla g_j(x), j \in L\right), \quad D(x)=\operatorname{diag}(D_j(x), j \in L), \\ D_j(x)=\left\{\begin{array}{ll} g_j^2(x), & j \in L_1, \\ 0, & j \in L_2, \end{array}\right. \\ B(x)=\left(N(x)^T N(x)+D(x)\right)^{-1} N(x)^T, \\ \pi(x)=-B(x)\nabla f(x),\ \pi(x;c)=-B(x)\nabla F_c(x). \end{array} \tag{2.4}$$

For an appropriate small parameter $\sigma > 0$, define

$$I(x)=\left\{j \in L_1 \mid -\sigma|\pi_j(x;c)| \le g_j(x) \le 0\right\}, J(x)=I(x)\bigcup L_2. \tag{2.5}$$

Throughout this paper, the following assumption holds.

H 2.1. The feasible sets are nonempty, i.e., $R \ne \phi$ and $R_+ \ne \phi$; the functions f and $g_j(x) (j \in L)$ are two-times continuously differentiable; $\forall x \in R_+$, the vectors $\{\nabla g_j(x), j \in E(x)\}$ are linearly independent.

According to Lemma 2.2 in [6], we can obtain the following result.

Lemma 2.1. *Suppose that H2.1 holds, then $\forall x \in R_+$, the matrix $(N(x)^T N(x) + D(x))$ is positive definite, and $\pi(x)$, $\pi(x;c)$ satisfy that:*

$$\pi_j(x)=\pi_j(x;c), j \in L_1, \pi_j(x)=\pi_j(x;c)-c, j \in L_2.$$

Lemma 2.2. *If $c > \max\{|\pi_j(x)| : j \in L_2\}$, then x is a KKT point of the problem (2.1) if and only if x is a KKT point of the problem (2.2).*

Due to Lemma 2.2, we have

$$\hat{u}_j = u_j,\ j \in J(x)\backslash L_2;\ \hat{u}_j = u_j + c,\ j \in L_2,$$

where $\hat{u}$ and u are corresponding KKT multipliers of the problem (2.2) and (2.1), respectively.

For (2.2), we consider the following equality constrained quadratic programming

$$\begin{array}{ll} \min & \nabla F_c(x)^T d + \frac{1}{2} d^T H d \\ \text{s.t.} & g_j(x) + \nabla g_j(x)^T d = -\min\{0, \pi_j(x;c)\}, j \in J(x). \end{array} \tag{2.6}$$

Lemma 2.3. *Suppose that* $(d_0(x), \widetilde{u}(x))$ *is a KKT point pair of (2.6), if* $d_0(x) = 0$*, then, we obtain that* x *is the KKT point of the problem (2.2). On the conditions of Lemma 2.2,* x *is a KKT point of the problem (2.1).*

Proof. Let

$$\hat{u}(x) = (\hat{u}_j(x), j \in L), \hat{u}_j(x) = \begin{cases} \widetilde{u}_j(x), & j \in J(x), \\ 0, & j \in L \setminus J(x). \end{cases}$$

If $d_0(x) = 0$, then, from (2.6), we have

$$\begin{array}{l} \nabla F_c(x) + \nabla g_{J(x)}(x) \widetilde{u}(x) = 0, \\ g_j(x) + \min\{0, \pi_j(x;c)\} = 0, j \in J(x). \end{array} \tag{2.7}$$

Here $\nabla g_{J(x)}(x) = \left(\nabla g_j(x),\ j \in J(x)\right)$. Thus $g_j(x) = 0,\ \pi_j(x;c) \geq 0,\ j \in J(x)$, $D_j(x)\hat{u}_j(x) = 0,\ \ j \in L$, and we have

$$\left(N(x)^T N(x) + D(x)\right) \hat{u}(x) = N(x)^T \nabla g_{J(x)}(x) \widetilde{u}(x) = -N(x)^T \nabla F_c(x),$$

i.e.

$$\hat{u}(x) = -\left(N(x)^T N(x) + D(x)\right)^{-1} N(x)^T \nabla F_c(x) = \pi(x;c).$$

So, from (2.7), we obtain

$$\begin{array}{l} \nabla F_c(x) + \nabla g_{J(x)}(x) \widetilde{u}(x) = 0, \\ g_j(x) = 0, \widetilde{u}_j(x) \geq 0, j \in J(x). \end{array}$$

This shows that x is a KKT point of the problem (2.2). According to $c > \max\{|\pi_j(x)| : j \in L_2\}$ from Lemma 2.2, it is easy to see that x is a KKT point of the problem (2.1). □

Obviously, the solution $d_0(x)$ of the problem (2.6) may not be a feasible direction, so, in view of reference [7], we make an assistance feasible direction as follows:

$$
\begin{aligned}
P(x) &= I_n - N(x)B(x),\\
\omega(x) &= \sum_{j\in L_1} \max\left\{-\pi_j(x), \pi_j(x)g_j^2(x)\right\} - \sum_{j\in L_2}(\pi_j(x)+c)g_j(x),\\
\rho(x) &= \frac{1}{1+|e^T\pi(x;c)|}\left(||P(x)\nabla f(x)||^2+\omega(x)\right),\\
e &= (1,\dots,1)^T \in E^{m+l},\\
v(x) &= (v_j(x), j\in I),\\
v_j(x) &= \begin{cases} g_j^2(x)-\rho(x), & j\in L_1, \quad \pi_j(x)\ge 0,\\ -1-\rho(x), & j\in L_1, \quad \pi_j(x)<0,\\ -g_j(x)-\rho(x), & j\in L_2,\end{cases}\\
q(x) &= \rho(x)\left(-P(x)\nabla f(x) + B(x)^T v(x)\right).
\end{aligned}
\tag{2.8}
$$

Lemma 2.4. (see [8]) *For (2.8), let* $c > \max\{|\pi_j(x)| : j \in L_2\}$*, if* $\rho(x) = 0$*, then* x *is a KKT point of the problem (2.1), if* $\rho(x) > 0$*, then, we have*

$$
\begin{aligned}
&\nabla F_c(x)^T q(x) \le -\frac{1}{2}\rho^2(x) < 0,\\
&\nabla g_j(x)^T q(x) \le -\frac{1}{2}\rho^2(x) < 0, \ j\in\{j\in L | g_j(x)=0\}.
\end{aligned}
\tag{2.9}
$$

Now, the algorithm for the solution of the problem (2.1) can be stated as follows.

Algorithm. Given a starting point $x^0 \in R_+$, and an initial symmetric matrix $H^0 \in E^{n\times n}$. Choose parameters $\xi, \upsilon \in (0,1), \alpha \in (0, \frac{1}{2}), \tau \in (2,3), \delta > 2, c_0, \epsilon \in (0, +\infty)$. Set $k = 0$;

Step 1. Update c_k Computation: $t_k = \left(\max\{|\pi_j(x^k)| : j \in L_2\}\right) + c_0$,

$$
c_k = \begin{cases} \max\{t_k, c_{k-1}+\epsilon\}, & \text{if } \ c_{k-1} < t_k,\\ c_{k-1}, & \text{if } \ c_{k-1} \ge t_k.\end{cases}
$$

Step 2. From (2.4) and (2.5), compute $N_k = N(x^k),\ D_k = D(x^k),\ B_k = B(x^k),\ \pi_k = \pi(x^k),\ \pi(x^k; c_k),\ J_k = J(x^k)$;

Step 3. Obtain $\left(d_0^k, \widetilde{u}^k\right)$ by solving the problem (2.6) at x^k, if $d_0^k = 0$, STOP; if the problem (2.6) have no solution, then go to Step 6;

Step 4. Compute

$$
d_1^k = -B_k^T\left(||d_0^k||^\tau e + G(x^k+d_0^k)\right), d^k = d_0^k + d_1^k, \tag{2.10}
$$

where $e = (1,\dots,1)^T \in R^{|L|}$, $G(x^k+d_0^k) = \left(G_j(x^k+d_0^k), j\in L\right)$,

$$
G_j(x^k+d_0^k) = \begin{cases} g_j(x^k+d_0^k), & j\in J_k;\\ 0, & j \in L\setminus J_k.\end{cases}
$$

If

$$\nabla F_{c_k}(x^k)^T d_0^k \le \min\left\{-\xi||d_0^k||^\delta, -\xi||d^k||^\delta\right\}, \tag{2.11}$$

$$||H_k d_0^k|| \le \xi||d_0^k||^{\frac{1}{2}}, \min\left\{\widetilde{u}_j^k, j \in J_k\right\} \ge -\xi||d_0^k||, \tag{2.12}$$

$$F_{c_k}(x^k + d^k) \le F_{c_k}(x^k) + \alpha \nabla F_{c_k}(x^k)^T d_0^k, \tag{2.13}$$

$$g_j(x^k + d^k) \le 0, \ \ j \in L. \tag{2.14}$$

Let $\lambda_k = 1$, go to Step 7;

Step 5. From (2.8) compute $\rho_k = \rho(x^k)$ and $q^k = q(x^k)$, if $\rho_k = 0$, STOP; otherwise, compute λ_k, the first number λ in the sequence $\{1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \ldots\}$ satisfying

$$F_{c_k}(x^k + \lambda q^k) \le F_{c_k}(x^k) + \upsilon\lambda \nabla F_{c_k}(x^k)^T q^k, \tag{2.15}$$

$$g_j(x^k + \lambda q^k) \le 0, \qquad j \in L, \tag{2.16}$$

let $d^k = q^k$;

Step 6. Update: Let H_{k+1} be a new symmetric approximation of the Hessian matrix. Set $x^{k+1} = x^k + \lambda_k d^k$ and $k = k+1$; Go back to Step 2.

3. Global and Strong Convergence of Algorithm

In this section, we firstly prove the global convergence of algorithm. Under mild conditions, we will discuss the strong convergence of the proposed Algorithm.

Firstly, from Lemma 2.4 and the algorithm, it is easy to obtain the following results.

Lemma 3.1. *Suppose that $\rho_k \neq 0$, then, the line search in Step 6 of Algorithm is well defined.*

Lemma 3.2. *If the sequence $\{x^k\}$ is bounded, then there exists a k_0, such that $c_k \equiv c_{k_0} \stackrel{\triangle}{=} c$ for all $k \ge k_0$.*

Due to Lemma 3.2 we always assume that $c_k \equiv c$ for all k in the rest of this paper.

Theorem 3.1. *The algorithm either stops at a KKT point x^k of the problem (2.1) in finite iteration, or generates an infinite sequence $\{x^k\}$ any accumulation point x^* of which is a KKT point of the problem (2.1).*

Proof. From Lemma 2.2, Lemma 2.3 and Lemma 2.4, the first statement is obvious. Thus, assume that the algorithm generates an infinite sequence $\{x^k\}$, x^* is an accumulation point. Because $J_k \subseteq L$ is a finite set, there exists an infinite index set K, such that

$$x^k \to x^*, \quad J_k \equiv J, \quad k \in K.$$

If there exists $K_1 \subseteq K$, such that for all $k \in K_1$, $x^{k+1} = x^k + \lambda_k d^k (k \in K_1)$ are generated by Step 6 and Step 7, then from Theorem 3.3 in [5], we can obtain that x^* is a KKT point of problem (2.1). Now we might as well assume that, for all $k \in K$, $x^{k+1} = x^k + \lambda_k d^k$ is generated by Step 5 and Step 7, then in view of (2.11), (2.13), (2.15) and (2.9), we obtain that $\{F_{c_k}(x^k)\}$ is monotone decreasing. Thus, from $\{x^k\}_{k\in K} \to x^*$, together with H 2.1 and Lemma 3.2, we can get

$$F_{c_k}(x^k) \to F_c(x^*),\ k \to \infty. \tag{3.1}$$

From (2.11) and (2.13), it is easy to see that

$$0 = \lim_{k\in K}(F_c(x^{k+1}) - F_c(x^k)) \le \lim_{k\in K} \alpha \nabla F_c(x^k)^T d_0^k \le \lim_{k\in K}(-\alpha\xi||d_0^k||^\delta) \le 0. \tag{3.2}$$

Then, we have $d_0^k \to 0,\ k \in K$ and $H_k d_0^k \to 0,\ k \in K$. From (2.6), we get

$$\begin{aligned}&\nabla F_c(x^k) + H_k d_0^k + \nabla g_{J(x)}(x^k)\widetilde{u}^k = 0,\\ &g_j(x^k) + \min\{0, \pi_j(x^k; c)\} + \nabla g_j(x^k)^T d_0^k = 0,\ j \in J.\end{aligned} \tag{3.3}$$

So, $g_j(x^*) = 0,\ j \in J$, and $J \subseteq E(x^*)$ hold. Thus, from H2.1, we obtain that the matrix $\left(\nabla g_J(x^*)^T \nabla g_J(x^*)\right)$ is nonsingular at x^*, so, for $k \in K$ large enough, the matrix $\left(\nabla g_J(x^k)^T \nabla g_J(x^k)\right)$ is nonsingular, and we have

$$\left(\nabla g_J(x^k)^T \nabla g_J(x^k)\right)^{-1} \to \left(\nabla g_J(x^*)^T \nabla g_J(x^*)\right)^{-1},\ k \in K,\ k \to \infty.$$

Then from (3.3) it can be seen that

$$\begin{aligned}\widetilde{u}^k &= -\left(\nabla g_J(x^k)^T \nabla g_J(x^k)\right)^{-1} \nabla g_J(x^k)^T (\nabla F_c(x^k) + H_k d_0^k)\\ &\to -\left(\nabla g_J(x^*)^T \nabla g_J(x^*)\right)^{-1} \nabla g_J(x^*)^T \nabla F_c(x^*) \triangleq \lambda^*, k \in K.\end{aligned} \tag{3.4}$$

So, from (2.12) and (3.3), it is easy to get

$$\begin{aligned}&\nabla F_c(x^*) + \nabla g_J(x^*)\lambda^* = 0,\\ &g_j(x^*) = 0, \lambda_j^* \ge 0, j \in J.\end{aligned} \tag{3.5}$$

This implies that x^* is a KKT point of (2.2). Then, in view of $c > |\pi_j(x^*)|,\ j \in L_2$, we know that x^* is a KKT point of the problem (2.1) from Lemma 2.2. □

To obtain the strong convergence of Algorithm, the following assumption is necessary.

H 3.1. Suppose that the sequence $\{x^k\}$ of points generated by Algorithm is bounded, and has a limit point x^*. The second-order sufficiency conditions with strict complementary slackness are satisfied at the KKT point x^* and corresponding multipliers u^* of the problem (2.1).

According to H3.1 and Proposition 4.1 in article [8], we can obtain the following conclusion.

Lemma 3.3. *Let H2.1∼ H3.1 holds,* $\lim\limits_{k\to\infty} ||x^{k+1} - x^k|| = 0$. *Thereby, the entire sequence* $\{x^k\}$ *converges to* x^*, *i.e.* $x^k \to x^*, k \to \infty$.

4. Superlinear Convergence of Algorithm

In order to obtain superlinear convergence, we make another assumption.

H 4.1. $H_k \to H_*, k \to \infty$, and H_* is positive definite on the subspace $Y(x^*)$, where

$$Y(x^*) = \left\{d \in E^n \mid \nabla g_j(x^*)^T d = 0, j \in E(x^*)\right\}.$$

Lemma 4.1. *Suppose that H2.1 ∼ H3.1 holds, for* k *large enough, then the solution of (2.6) is unique, and*

$$J_k \equiv E(x^*) \overset{\triangle}{=} E_*, \ \lim_{k\to\infty} d_0^k = 0, \ \lim_{k\to\infty} \pi(x^k; c_k) = \hat{u}^*, \lim_{k\to\infty} \widetilde{u}^k = (\hat{u}_j^*, j \in E_*),$$

where $\hat{u}_j^* = u_j^*,\ j \in E_* \backslash L_2;\ \hat{u}_j^* = u_j^* + c,\ j \in L_2$.

Proof. Firstly, we prove $\pi(x^k; c) \to \hat{u}^*,\ k \to \infty$. Denote, $N_* = N(x^*),\ D_* = D(x^*),\ B_* = B(x^*)$, then, because x^* is a KKT point, and u^* is the corresponding multiplier, we have

$$\nabla f(x) + N_* u^* = 0,$$

$$g_j(x^*) u_j^* = 0,\ j \in L.$$

Let $\hat{u}_j^* = u_j^*, \ \ j \in E_* \setminus L_2 . \hat{u}_j^* = u_j^* + c, \ \ j \in L_2$. We have

$$\begin{cases} \nabla F_c(x^*) + N_* \hat{u}^* = 0, \\ D_* \hat{u}^* = 0. \end{cases}$$

This implies that the following equality holds

$$\hat{u}^* = -\left(N_*^T N_* + D_*\right)^{-1} N_*^T \nabla F_c(x^*) = -B_* \nabla F_c(x^*).$$

In addition, from assumption H2.1 and $x^k \to x^*, k \to \infty$, it is easy to obtain

$$\pi(x^k; c) = -B_k \nabla F_c(x^k) \to -B_* \nabla F_c(x^*) = \hat{u}^*.$$

From Lemma 2.2 we obtain $\hat{u}_j^* = u_j^*,\ j \in E_* \backslash L_2;\ \hat{u}_j^* = u_j^* + c,\ j \in L_2$.

Now we prove that $J_k \equiv E_*$ holds. When $j \in E_*$, we get $u_j^* > 0$, from assumption H3.1, we have $\hat{u}_j^* > 0$. Then, since $x^k \to x^*,\ \pi(x^k; c) \to \widetilde{u}^*, k \to \infty$ and the functions $g_j(x)(j \in L)$ are smooth functions (for k large enough) we obtain

$$\pi_j(x^k; c) > 0,\ 0 \geq g_j(x^k) \to g_j(x^*) = 0,\ j \in E_*.$$

Then, from the definition of J_k, we obtain $j \in J_k$, i.e. $E_* \subseteq J_k$ (for k large enough). Now, we prove that $J_k \subseteq E_*$. Suppose that the desired conclusion is false. Then, there exists a constant j_0 and infinite set K, such that

$$j_0 \in J_k \setminus E_*,\ g_{j_0}(x^*) < 0, g_{j_0}(x^k) \geq -\sigma |\pi_{j_0}(x^k; c)|,\ \forall k \in K.$$

Let $k \to \infty,\ k \in K$, we have

$$0 > g_{j_0}(x^*) \geq -\sigma \hat{u}_{j_0}^*,\ \hat{u}_{j_0}^* > 0,$$

which is contradictory with the strict complementary slackness condition. So, $J_k \subseteq E_*$, i. E.$J_k = E_*$.

Furthermore, from the above analysis, when k is large enough, the quadratic programming (2.6) becomes the form as following:

$$\begin{array}{ll} \min & \nabla F_c(x^k)^T d + \frac{1}{2} d^T H_k d \\ \text{s.t.} & g_j(x^k) + \nabla g_j(x^k)^T d = 0, j \in E_*. \end{array} \tag{4.1}$$

Obviously, the feasible region of the quadratic programming (4.1) is nonempty because the facts assumption H2.1 holds, $x^k \to x^*, k \to \infty$ and k large enough. Because H_* is a positive definite matrix, then, (4.1) has only a unique solution.

Now, we prove that $d_0^k \to 0,\ \widetilde{u}^k \to (\hat{u}_j^*,\ j \in E_*),\ k \to \infty$. By (4.1), it can be seen that

$$\begin{array}{l} \nabla F_c(x^k) + H_k d_0^k + \nabla g_{E_*}(x^k) \widetilde{u}^k = 0, \\ g_{E_*}(x^k) + \nabla g_{E_*}(x^k)^T d_0^k = 0, \end{array}$$

i.e.

$$\begin{pmatrix} H_k & \nabla g_{E_*}(x^k) \\ \nabla g_{E_*}(x^k)^T & 0 \end{pmatrix} \begin{pmatrix} d_0^k \\ \widetilde{u}^k \end{pmatrix} = -\begin{pmatrix} \nabla F_c(x^k) \\ g_{E_*}(x^k) \end{pmatrix}.$$

Denote

$$G_k = \begin{pmatrix} H_k & \nabla g_{E_*}(x^k) \\ \nabla g_{E_*}(x^k)^T & 0 \end{pmatrix}, G_* = \begin{pmatrix} H_* & \nabla g_{E_*}(x^*) \\ \nabla g_{E_*}(x^*)^T & 0 \end{pmatrix}.$$

Because $x^k \to x^*, H_k \to H_*$, the matrix $\nabla g_{E_*}(x^*)$ is nonsingular at x^*, and H_* is a positive definite matrix, it is obvious that G_k and G_* are meaning, for k large enough, and $G_k^{-1} \to G_*^{-1}$. From $g_{E_*}(x^*) = 0$, we have

$$\begin{pmatrix} d_0^k \\ \widetilde{u}^k \end{pmatrix} = -G_k^{-1} \begin{pmatrix} \nabla F_c(x^k) \\ g_{E_*}(x^k) \end{pmatrix} \to -G_*^{-1} \begin{pmatrix} \nabla F_c(x^*) \\ 0 \end{pmatrix}.$$

In addition, from x^* is a KKT point of the problem (2.2), we have

$$\nabla F_c(x^*) = -\nabla g_{E_*}(x^*)\hat{u}^*_{E_*},$$

where $\hat{u}^*_{E_*} = (\hat{u}^*_j, j \in E_*)$. So, it is easy to obtain

$$\begin{aligned} \begin{pmatrix} d_0^k \\ \widetilde{u}^k \end{pmatrix} &\to -G_*^{-1} \begin{pmatrix} \nabla F_c(x^*) \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} H_* & \nabla g_{E_*}(x^*) \\ \nabla g_{E_*}(x^*)^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} H_* & \nabla g_{E_*}(x^*) \\ \nabla g_{E_*}(x^*)^T & 0 \end{pmatrix} \begin{pmatrix} 0 \\ \hat{u}^*_{E_*} \end{pmatrix} \\ &= \begin{pmatrix} 0 \\ \hat{u}^*_{E_*} \end{pmatrix}, \end{aligned}$$

i.e. $d_0^k \to 0, \widetilde{u}^k \to (\hat{u}^*_j, j \in E_*), k \to \infty$. □

Lemma 4.2. *For k large enough:*
1) d_1^k and d^k obtained in Step 5 satisfy:

$$||d^k|| \sim ||d_0^k||,\ ||d_1^k|| = O\left(||d_0^k||^2\right). \tag{4.2}$$

2) There exists constants $b, \eta > 0$, such that

$$\sum_{j \in E_*} \widetilde{u}_j^k g_j(x^k) \le -\eta z_k, z_k = \left(\sum_{j \in E_*} g_j^2(x^k)\right)^{\frac{1}{2}}, \tag{4.3}$$

$$-(d_0^k)^T H_k d_0^k \le -b||d_0^k||^2 + o(z_k), \tag{4.4}$$

$$\nabla F_c(x^k)^T d_0^k \le -b||d_0^k||^2. \tag{4.5}$$

Proof. 1) In view of Lemma 4.1 and H3.1, we know that $\pi_j(x^k; c) > 0, \widetilde{u}_j^k > 0, j \in E_*$. Then from (2.6) we have

$$g_j(x^k) + \nabla g_j(x^k)^T d_0^k = 0,\ j \in E_*. \tag{4.6}$$

So

$$g_j(x^k + d_0^k) = g_j(x^k) + \nabla g_j(x^k)^T d_0^k + O\left(||d_0^k||^2\right) = O\left(||d_0^k||^2\right),\ j \in E_*,$$

i.e.

$$||G(x^k + d_0^k)|| = O\left(||d_0^k||^2\right).$$

Considering $B_k \to B_*,\ \tau \in (2,3)$, we obtain $||d_1^k|| = O\left(||d_0^k||^2\right),\ ||d^k|| \sim ||d_0^k||$.

2) According to $\widetilde{u}_j^k > 0,\ j \in E_*$, there exists a constant $\eta > 0$, such that

$$\sum_{j \in E_*} \widetilde{u}_j^k g_j(x^k) = -\sum_{j \in E_*} \widetilde{u}_j^k \left|g_j(x^k)\right| \le -\sum_{j \in E_*} \eta \left|g_j(x^k)\right| \le -\eta z_k.$$

Denote

$$P_* = I_n - A_*(A_*^T A_*)^{-1} A_*^T, P_k = I_n - A_k(A_k^T A_k)^{-1} A_k^T, \tag{4.7}$$

where $A_* = \nabla g_{E_*}(x^*), A_k = \nabla g_{E_*}(x^k)$. Let

$$d_0^k = P_* d_0^k + y_k, y_k = A_*(A_*^T A_*)^{-1} A_*^T d_0^k, \tag{4.8}$$

and $G(x^k) = \left(g_j(x^k), j \in E_*\right)$. Then by (4.6), it can be seen that

$$y_k = A_*(A_*^T A_*)^{-1}(A_* - A_k)^T d_0^k - A_*(A_*^T A_*)^{-1} G(x^k),$$

so,

$$||y_k|| = O\left(||d_0^k||\right) = o\left(||d_0^k||\right) + O(z_k).$$

Hence,

$$\begin{aligned}
-(d_0^k)^T H_k d_0^k &= -\left(P_* d_0^k + y_k\right)^T H_k \left(P_* d_0^k + y_k\right) \\
&= -\left(P_* d_0^k\right)^T H_* \left(P_* d_0^k\right) + \left(P_* d_0^k\right)(H_* - H_k)\left(P_* d_0^k\right) + O\left(||d_0^k|| \cdot ||y_k||\right) \\
&= -\left(P_* d_0^k\right)^T H_* \left(P_* d_0^k\right) + o\left(||d_0^k||^2\right) + o(z_k).
\end{aligned}$$

We know that $d_0^k \to 0, P_* d_0^k \in Y(x^*)$, so, there exists a constant $b_1 > 0$, such that

$$\begin{aligned}
-(d_0^k)^T H_k d_0^k &\le -b_1 ||P_* d_0^k||^2 + o\left(||d_0^k||^2\right) + o(z_k) \\
&= -b_1 ||d_0^k - y_k||^2 + o\left(||d_0^k||^2\right) + o(z_k)
\end{aligned}$$

$$= -b_1||d_0^k||^2 + o\left(||d_0^k||^2\right) + o(z_k) \leq -b||d_0^k||^2 + o(z_k).$$

In addition, from (3.3) and (4.6), we have

$$\nabla F_c(x^k)^T d_0^k = -(d_0^k)^T H_k d_0^k - (A_k^T d_0^k)^T \widetilde{u}^k = -(d_0^k)^T H_k d_0^k + \sum_{j\in E_*} \widetilde{u}_j^k g_j(x^k)$$
$$\leq -b||d_0^k||^2 + o(z_k) - \eta z_k \leq -b||d_0^k||^2. \quad (4.9)$$

To ensure the step size unit can be accepted, the following assumption about the symmetric matrix satisfied:

H 4.2. Let

$$\left\|P_k\left(H_k - \nabla_{xx}^2 \widetilde{L}(x^k, \widetilde{u}^k)\right) d_0^k\right\| = o\left(||d_0^k||\right),$$

which, obviously is equivalent to

$$\left\|P_k\left(H_k - \nabla_{xx}^2 L(x^*, u^*)\right) d_0^k\right\| = o\left(||d_0^k||\right),$$

where

$$\nabla_{xx}^2 \widetilde{L}(x^k, \widetilde{u}^k) = \nabla^2 F_c(x^k) + \sum_{j\in E_*} \widetilde{u}_j^k \nabla^2 g_j(x^k), \nabla_{xx}^2 L(x^*, u^*)$$
$$= \nabla^2 F_c(x^*) + \sum_{j\in L} u_j^* \nabla^2 g_j(x^*).$$

Lemma 4.3. *For k large enough, the inequalities of Step 5 are satisfied. i.e.* $x^{k+1} = x^k + d^k, \lambda_k \equiv 1$.

Proof. Firstly, in view of Lemma 4.1, Lemma 4.2 and H3.1, it follows that (2.12) holds. Considering $\delta > 2$, (4.2) and (4.5), so (2.11) holds. Thus, in order to finish the proof of this lemma, we only need to prove that (2.13) and (2.14) are true. According to Lemma 4.1 and H3.1, for k large enough, the perturbation term of the right side of quadratic programming (2.6) is disappeared. Then, (2.6) becomes the following quadratic programming.

$$QP_k \qquad \begin{array}{ll} \min & \nabla F_c(x^k)^T d + \frac{1}{2} d^T H_k d \\ \text{s.t.} & g_j(x^k) + \nabla g_j(x^k)^T d = 0, \quad j \in E_*. \end{array}$$

So, from Lemma 4.2, imitating Theorem 4.2 in [6], it is easy to prove the conclusion holds. □

Moreover, in view of Lemma 4.3 and the way of Theorem 5.2 in [1], we obtain the following theorem.

Theorem 4.1. *Under all above-mentioned assumptions, the algorithm is superlinearly convergent, i.e., the sequence* $\{x^k\}$ *generated by the algorithm satisfies* $||x^{k+1} - x^*|| = o(||x^k - x^*||)$.

Acknowledgements

This work was supported in part by the NNSF (No.10501009, 60471039) of P.R. China.

References

[1] F. Facchinei, S. Lucidi, Quadraticly and Superlinearly Convergent for the Solution of Inequality Constrained Optimization Problem, *JOTA*, **85** (1995), 265-289.

[2] M. Fukushima, A successive quadratic programming algorithm with global and superlinear convergence properties, *Mathematical Programming*, **35** (1986), 253-264.

[3] Z.Y. Gao, F. Wu, A superlinearly feasibble method for nonlinear constraints, *Acta Mathematica Sinica, Ser. A*, **6** (1997), 895-900.

[4] S.P. Han, A globally convergent method for nonlinear programming, *Journal of Optimization Theory and Applications* (1977), 297-309.

[5] J.B. Jan, Algorithm of sequential systems of linear equations with superlinear and quadratical convergence for general constrained optimization, *Journal of Engineering Mathematics*, **8** (2003), 24-30.

[6] J.B. Jian, C.M. Tang, Q.J. Hu, H.Y. Zheng, A feasible descent SQP algorithm for general constrained optimization without strict complementarity, *Journal of Computational and Applied Mathematics*, **180** (2005), 391-412.

[7] J.B. Jian, K.C. Zhang, Subfeasible direction method with strong convergence for inequality constrained optimization, *Journal of Xi'An Jiao Tong University*, **33**, No. 8 (1999), 88-91.

[8] E.R. Panier, A.L. Tits, A superlinearly convergent feasible method for the solution of inequality constrained optimization problems, *SIAM J. Control Optimization*, **4** (1987), 934-950.

[9] P. Spellucci, An SQP method for general nonlinear programs using only equality constrained subproblems, *Mathematical Programming*, **82** (1998), 413-448.

[10] R.B. Wilson, *A Simplical Algorithm for Concave Programming*, Ph.D. Thesis, Graduate School of Business Administration, Harvard University (1963).

[11] Z.B. Zhu, A new SQP algorithm for inequality constained optimization, *Mathmematica Numerica*, **11** (2004), 413-426.