Journal of Computational Mathematics, Vol.23, No.2, 2005, 217–224.

# A REVISED CONJUGATE GRADIENT PROJECTION ALGORITHM FOR INEQUALITY CONSTRAINED OPTIMIZATIONS *

Wei Wang
(*Department of Applied Mathematics, Tongji University, Shanghai 200092, China*)
(*Department of Mathematics, Shanghai University, Shanghai 200436, China*)

Lian-sheng Zhang
(*Department of Mathematics, Shanghai University, Shanghai 200436, China*)

Yi-fan Xu
(*School of Management, Fudan University, Shanghai 200433, China*)

**Abstract**

A revised conjugate gradient projection method for nonlinear inequality constrained optimization problems is proposed in the paper, since the search direction is the combination of the conjugate projection gradient and the quasi-Newton direction. It has two merits. The one is that the amount of computation is lower because the gradient matrix only needs to be computed one time at each iteration. The other is that the algorithm is of global convergence and locally superlinear convergence without strict complementary condition under some mild assumptions. In addition the search direction is explicit.

*Mathematics subject classification:* 90C30, 65K05.
*Key words*: Constrained optimization, Conjugate gradient projection, Revised direction, Superlinear convergence.

## 1. Introduction

Consider the optimization problem

$$\min\{f(x) : g_j(x) \le 0,\ j \in I,\ x \in R^n\}, \tag{1}$$

where $f(x),\ g_j(x) : R^n \to R,\ j \in I = \{1, 2, \ldots, m\}$.

We know that the quasi-Newton method is one of the most effective methods for solving nonlinear optimal problems because of the property of superlinear convergence. Some people investigated the variable metric algorithms for constrained optimization problems, such as [4, 5, 7, 8, 9, 13]. At present, the research on this topic is still active due to various improvements both in theory and applications. It is one of important results that the search direction of the method is constructed by combining the conjugate projective gradient with the quasi-Newton direction. However, the assumption of strict complementary condition, which is very strong, is necessary for keeping the superlinear convergence. Bonnans and Launay [1] proposed a globally and superlinearly convergent method without strict complementary condition. But it needs sufficiently curvature condition and needs to solve two quadratic sub-programmings in each iteration. Generally, the search directions of constrained quasi-Newton methods are composed of two different approaches. In fact, the search direction is determined by quasi-Newton direction under some conditions and determined by the gradient projection direction under

* Revised December 10, 2003; final revised September 16, 2004.

other conditions in the same iteration[12, 14]. This leads to the great amount of computation generally.

In order to overcome the defects in the stated methods above—strong assumption and great amount of computation, we present a revised conjugate projective gradient method. By means of schemes of $\epsilon$-active constraint set and the explicit conjugate projection gradient direction with respect to a positive definite matrix, our algorithm is more harmonious and effective. There are two merits in our method. The first is that the algorithm is of global convergence and locally superlinear convergence without strict complementary condition. The second is that we only need to compute the conjugate projection matrix one time to obtain two search directions at each iteration, so the method presented greatly decreases the amount of computation.

This paper is organized as follow. In Section 2, we present the algorithm. The convergence of the algorithm is discussed in Section 3. In Section 4, we analyze the rate of convergence of the algorithm. In the last section, we show the numerical tests with four examples.

## 2. Algorithm

Denote $X = \{x : g_j(x) \le 0\}$, and $J(x) = \{j \in I : g_j(x) \ge -\varepsilon\}$. We need following two assumptions in the paper.

**A1.** *$f(x)$ and $g_j(x)$ are continuously differentiable for any $j \in I$.*

**A2.** *$\{\nabla g_j(x),\ j \in J(x)\}$ is linearly independent for $x \in X$.*

Let us introduce some notes. At current point $x_k$, we define

$$\begin{aligned} A_k &= (\nabla g_j(x_k),\ j \in J(x_k)), \\ g_{J_k} &= (g_j(x_k),\ j \in J(x_k))^T. \end{aligned}$$

The conjugate projection w.r.t. a given symmetric positive definite matrix $H_k$ is

$$P_k = H_k - H_k A_k B_k \tag{2}$$

where $B_k = (A_k^T H_k A_k)^{-1} A_k^T H_k$.

We define

$$d_k^0 = -P_k \nabla f(x_k) - B_k^T g_{J_k}, \tag{3}$$

and

$$\lambda_k = -B_k \nabla f(x_k) + (A_k^T H_k A_k)^{-1} g_{J_k} = \lambda_k^1 + \lambda_k^2, \tag{4}$$

where, $\lambda_k^1 = -B_k \nabla f(x_k)$, $\lambda_k^2 = (A_k^T H_k A_k)^{-1} g_{J_k}$.

If the set $J(x_k) = \emptyset$, the algorithm is an ordinary quasi-Newton method. In the following, we always assume that $J(x_k) \neq \emptyset$.

To simplify, we denote $x_k, P_k, A_k, \cdots$ as $x, P, A, \cdots$, and $J$ stands for $J(x)$.

**Lemma 1.** *$P$ is a positive semi-definite matrix, and $PA = 0$, $BA = E$, where $E$ is a $|J| \times |J|$ identity matrix.*

**Theorem 1.** *If $x \in X, d^0 = 0, \lambda \ge 0$, then $x$ is a KKT point of problem* (1).

*Proof.* From $d^0 = 0$, we have

$$\begin{aligned} 0 &= -H\nabla f(x) + HA(A^T HA)^{-1} A^T H \nabla f(x) - HA(A^T HA)^{-1} g_J \\ &= -H\nabla f(x) - HA\lambda \end{aligned}$$

and

$$0 = A^T d^0 = -A^T P \nabla f(x) - (BA)^T g_J = -g_J.$$

Hence, there exists $\lambda \ge 0$ such that $\nabla f(x) + A\lambda = 0$ and $\lambda_j g_j = 0,\ j \in J$.

Now state the algorithm as follows.

**Algorithm:**
**Step 0.** Given $\alpha \in (0, \frac{1}{2})$, $\beta, \eta \in (0,1)$, $\theta \in (\frac{1}{2}, 1)$; $\epsilon, \delta > 0$. Select $x_0 \in X$ and a symmetric positive definite matrix $H_0$, set $k = 0$.
**Step 1.1.** Set $i = 0, \epsilon^{ki} = \epsilon^k$.
**Step 1.2.** If

$$det(A_{ki}^T A_{ki}) > \epsilon^{ki},$$

where $A_{ki} = (\nabla g_j(x_k) : \ j \in J_{ki})$ and $J_{ki} = \{j \in I : g_j(x_k) > -\epsilon^{ki}\}$, then set $A_k = A_{ki}, J_k = J_{ki}, \epsilon^k = \epsilon^{ki}$ and go to step 2.
**Step 1.3.** Let $i = i+1, \epsilon^{ki} = \frac{1}{2}\epsilon^{k(i-1)}$ and go to step 1.2.
**Step 2.** Compute $d_k^0, \lambda_k$ by (3) and (4). if $d_k^0 = 0$ and $\lambda_k \geq 0$, then stop.
**Step 3.** If $\|d_k^0\| \leq \delta$, $\lambda_{kj} \geq -\eta\|d_k^0\|$ and

$$\begin{cases} f(x_k + d^0) \leq f(x_k) + \alpha(\nabla f(x_k)^T d_k^0 - \lambda_k^T g_{J_k}), \\ g_j(x_k + d^0) \leq 0, \quad j \in I, \end{cases} \tag{5}$$

then set $x_{k+1} = x_k + d_k^0$ and $\delta = \frac{1}{2}\delta$. Go to step 7.
**Step 4.** Let $U_k = (u_{kj}, j \in J_k)^T$ where

$$u_{kj} = \begin{cases} \lambda_{kj}^1, & \lambda_{kj}^1 < 0, \\ -|g_j(x_k)|, & \lambda_{kj}^1 \geq 0. \end{cases}$$

Set $d_k^1 = -P_k \nabla f(x_k) + B_k^T U_k$ and $d_k^2 = -P_k \nabla f(x_k) - B_k^T \|d_k^1\| e$, where $e = (1, 1, ..., 1)^T$.
**Step 5.** Set

$$d_k = (1 - \rho_k) d_k^1 + \rho_k d_k^2, \tag{6}$$

where $\rho_k = \max\{\rho \in (0,1] : \nabla f(x_k)^T((1-\rho)d_k^1 + \rho d_k^2) \leq \theta \nabla f(x_k)^T d_k^1\}$.
**Step 6.** Let $t_k$ be the first one of $t$ in the sequence $\{1, \beta, \beta^2, \ldots\}$ satisfying

$$\begin{cases} f(x_k + t d_k) \leq f(x_k) + \alpha t \nabla f(x_k)^T d_k^1, \\ g_j(x_k + t d_k) \leq 0, \quad\quad j \in I \end{cases} \tag{7}$$

and set $x_{k+1} = x_k + t_k d_k$.
**Step 7.** Update $H_k$ to $H_{k+1}$ by quasi-Newton method. Set $k = k + 1$ and back to step 1.

Based on assumption A2, we can immediately get the following lemma.

**Lemma 2.** *Sequence $\{\epsilon^k\}$ has a low bounder $\bar{\epsilon} > 0$ such that $\epsilon^k \geq \bar{\epsilon}$ for all $k$.*

## 3. Convergence

In this section, we discuss the global convergence of the algorithm. We need additional assumptions and let they hold in the rest of the paper.

**A3.** *For any $k$ and $y \in R^n$, $a\|y\|^2 \leq y^T H_k y \leq b\|y\|^2$, where $b \geq a > 0$ are constants.*
**A4.** *Sequence $\{x_k\}$ generated by the algorithm is bounded.*

**Lemma 3.** *Sequences $\{P_k\}, \{d_k^0\}, \{d_k\}$ and $\{\lambda_k\}$ are all bounded.*

*Proof.* According to A3, $\|H_k\|$ is bounded. So we only need to prove $det((A_k^T H_k A_k)^{-1})$ is bounded from (2), (3), (4) and (6). If it is not bounded, then from A4 there exists a subset $K$ such that

$$x_k \to x^* \in X, \qquad det((A_k^T H A_k)^{-1}) \to \infty, \qquad k \xrightarrow{K} \infty.$$

This implies $det(A_k^T A_k) \to 0$, $(k \xrightarrow{K} \infty)$. It is a contradiction with assumption A2. In view of A1 and the boundness of $\{P_k\}$, we can know the rest of conclusion is true.

**Lemma 4.** $d_k^0 = 0, \quad \lambda_k \geq 0 \quad \Longleftrightarrow \quad d_k^1 = 0.$

*Proof.* Let $d_k^0 = 0,\ \lambda_k \geq 0$. Based on the proof of Theorem1, (3), as well as the definitions of $U_k$ and $d_k^1$, it is easy to learn $d_k^1 = 0$.

Contrarily, from $d_k^1 = 0$, we can get

$$\begin{aligned}\nabla f(x_k)^T d_k^1 &= -\nabla f(x_k)^T P_k \nabla f(x_k) + \nabla f(x_k)^T B_k^T U_k \\ &= -\|P_k^{\frac{1}{2}} \nabla f(x_k)\|^2 - \sum_{\lambda_{kj}^1 < 0} (\lambda_{kj}^1)^2 - \sum_{\lambda_{kj}^1 \geq 0} \lambda_{kj}^1 |g_j| = 0.\end{aligned}$$

Thereby, $P_k \nabla f(x_k) = 0, \lambda_k^1 \geq 0$. Recalling Step 4, we obtain $B_k^T U_k = 0$ from $d_k^1 = 0$ and $P_k \nabla f(x_k) = 0$, therefore $U_k = 0$ which implies $g_{J_k} = 0$. So $d_k^0 = 0$ and $\lambda_k \geq 0$ hold.

**Lemma 5.** *If $x_k$ is not a KKT point of* (1), *then*

*(i).* $\nabla f(x_k)^T d_k^0 - \lambda_k^T g_{J_k} \leq 0,\ \nabla f(x_k)^T d_k^1 < 0$ *and*

*(ii).* $\nabla g_j(x_k)^T d_k^1 \leq 0,\ \nabla g_j(x_k)^T d_k^2 < 0,\ j \in J_k$.

*Proof.* (i). Since $(d_k^0,\ \lambda_k)$ is a KKT pair of the following quadratic programming

$$(QP) \qquad \begin{array}{ll} min & \frac{1}{2} d^T H_k^{-1} d + \nabla f(x_k)^T d \\ s.t. & g_j(x_k) + \nabla g_j(x_k)^T d = 0, \qquad j \in J_k, \end{array}$$

we have

$$\nabla f(x_k)^T d_k^0 - \lambda_k^T g_{J_k} = \nabla f(x_k)^T d_k^0 + \lambda_k^T A_{J_k} d_k^0 = -(d_k^0)^T H_k^{-1} d_k^0 \leq 0.$$

Now we show $\nabla f(x_k)^T d_k^1 < 0$. For $x_k$ is not a KKT point, we have either $d_k^0 \neq 0$ or there is a $j \in J_k$ such that $\lambda_{kj}^1 < 0$. Hence

$$\nabla f(x_k)^T d_k^1 = -(d_k^0)^T H_k^{-1} d_k^0 - \sum_{\lambda_{kj}^1 < 0} (\lambda_{kj}^1)^2 - \sum_{\lambda_{kj}^1 \geq 0} \lambda_{kj}^1 |g_j| < 0$$

(ii). Since $x_k$ is not a KKT point, then $d_k^1 \neq 0$. Thus,

$$A_{J_k}^T d_k^1 = U_k \leq 0, \quad A_{J_k}^T d_k^2 = -\|d_k^1\| e < 0.$$

From (6) and Lemma 5, the following lemma holds evidently.

**Lemma 6.** *If $x_k$ is not the KKT point of* (1), *then*

*(i).* $\nabla f(x_k)^T d_k < 0$.

*(ii).* $\nabla g_j(x_k)^T d_k < 0, \quad j \in J(x_k)$.

**Lemma 7.** *If $x_k$ is not a KKT point of* (1), *then there exists $\bar{t} > 0$ such that* (7) *holds for any $t \in [0, \bar{t}]$.*

*Proof.* By the differential mean value theorem,

$$f(x_k + t d_k) = f(x_k) + t \nabla f(x_k + \xi d_k)^T d_k, \qquad \xi \in [0, t].$$

Since $f(x)$ is continuously differentiable, $\nabla f(x_k)^T d_k \leq \theta \nabla f(x_k) d_k^1 < 0$ and $\alpha/\theta < 1$, there exists $t_0 > 0$ such that

$$f(x_k + t d_k) \leq f(x_k) + \alpha t \nabla f(x_k)^T d_k^1. \qquad \forall\, t \in [0, t_0].$$

In the same way, we can write for any $j$

$$g_j(x_k + t d_k) = g_j(x_k) + t \nabla g_j(x_k + \xi_j d_k) d_k, \qquad \xi_j \in [0, t].$$

If $j \notin J(x_k)$, i.e. $g_j(x_k) < 0$, it is easy to know that there exists sufficiently small $t_j > 0$ such that $g_j(x_k + t d_k) \leq 0,\ \forall t \in [0, t_j]$.

If $j \in J(x_k)$, then $\nabla g_j(x_k)^T d_k < 0$ according to Lemma 6, which deduces that there exists sufficiently small $t_j$ such that $g_j(x_k + td_k) \leq 0, \ \forall t \in [0, t_j]$.

Set $\bar{t} = \min\{t_0, t_1, \ldots, t_m\}$. Then (7) holds for any $t \in [0, \bar{t}\,]$.

**Lemma 8.** *Let $x^*$ be an cluster point of $\{x_k\}$ generated by Algorithm and let $\{x_k\}_K$ be any subsequence converging to $x^*$. Then, $\{d_k^0\}_K \to 0$, $\{\lambda_k\}_K \to \lambda^* \geq 0$. $\lambda^*$ is the Lagrangian multiplier associated with $x^*$, i.e., $x^*$ is a KKT point of* (1).

*Proof.* From Lemma 5 and assumptions, it is known that $\{f(x_k)\}$ is descending and

$$f(x_k) \to f(x^*), \quad k \xrightarrow{K} \infty.$$

Therefore, because $\nabla f(x_k)^T d_k^0 - \lambda_k^T g_{J_k} = -(d_k^0)^T H_k^{-1} d_k^0$, there would be

$$0 \leftarrow f(x_{k+1}) - f(x_k) \leq \begin{cases} -\alpha (d_k^0)^T H_k^{-1} d_k^0, & \text{if } x_{k+1} = x_k + d_k^0 \\ \alpha t_k \nabla f(x_k)^T d_k^1, & \text{otherwise.} \end{cases} \tag{8}$$

If $x_{k+1} (k \in K)$ is decided by $x_k + d_k^0$, from A3 and Step 3 we obtain

$$\{d_k^0\} \xrightarrow{K} (d^0)^* = 0, \qquad \{\lambda_k\} \xrightarrow{K} \lambda^* \geq 0.$$

That is to say $x^*$ is a KKT point.

If $x_{k+1} (k \in K)$ is decided by $x_k + t_k d_k$, we only need to prove $d_k^1 \xrightarrow{K} (d^1)^* = 0$ by the Lemma 4. Suppose it is not true, that is $d_k^1 \xrightarrow{K} (d^1)^* \neq 0$ , then there exists $d' > 0$ such that $\|d_k^1\| \geq d'$ for all large enough $k \in K$. Next, we will show that there exists $t' > 0$ such that $t_k \geq t'$ for all large enough $k \in K$. Therefore, $t_k d_k^1 \not\to 0$.

Since $d_k^1 \to d^* \neq 0$, there is $\sigma_1 > 0, \rho' > 0$ such that for large enough $k$

$$\nabla f(x_k)^T d_k \leq \theta \nabla f(x_k)^T d_k^1 \leq -\sigma_1 < 0, \quad \rho_k > \rho' > 0.$$

If $j \notin J_0(x^*)$ (where $J_0(x) = \{j \in I : g_j(x) = 0\}$), then $g_j(x_k) \to g_j(x^*) < 0$. Hence, there exists $\sigma_2 > 0$ such that for large enough $k$, $g_j(x_k) \leq -\sigma_2 < 0$.

If $j \in J_0(x^*)$, then from (), $g_j(x_k) \to g_j(x^*) = 0$ implies that $\nabla g_j(x_k)^T d_k^2 \to -\|(d^1)^*\| \leq -d' < 0, \quad k \xrightarrow{K} \infty$.

Hence, we get that for large enough $k$

$$\nabla g_j(x_k)^T d_k \leq -\frac{1}{2} \rho' d' < 0.$$

and it is easy to know that there exists $t' > 0$ such that $t_k \geq t'$ and $t_k d_k^1 \not\to 0$.

On the other hand, according to (8) and the definition of $d_k^1$,

$$0 \leftarrow \nabla f(x_k)^T d_k^1 = -(d_k^0)^T H_k^{-1} d_k^0 - \sum_{\lambda_{kj}^1 < 0} (\lambda_{kj}^1)^2 - \sum_{\lambda_{kj}^1 \geq 0} \lambda_{kj}^1 |g_j| \leq 0$$

Therefore we get$(d^0)^* = 0, \ \lambda^* \geq 0 \quad \Rightarrow \quad (d^1)^* = 0$. This is a contradiction.

From Lemma 8, we immediately have the convergence of the algorithm.

**Theorem 2.** *The algorithm either stops at a KKT point of problem* (1) *in finite steps or generates a infinite sequence whose any cluster point is the KKT point of* (1).

## 4. The Rate of Convergence

In this section, we discuss the convergent rate of the algorithm. We replace assumption A1 by A1$'$ and add another two assumptions. They hold in this section.

**A1′.** *$f(x), g_j(x)$ are twice continuously differentiable for any $j \in I$.*
**A5.** *Strong second-order sufficient condition holds, i.e.,*

$$d^T \nabla^2_{xx} L(x^*, \lambda^*) d > 0, \qquad \forall\ d \in \ker \nabla g_{\hat{J}(x^*)}(x^*) \backslash \{0\},$$

*where $L(x, \lambda) = f(x) + \lambda^T g(x)$, $g(x) = (g_1(x), \cdots g_m(x))^T$, $\hat{J}(x^*) = \{j \in J(x^*) : (\lambda^*)_j > 0\}$ and $(x^*, \lambda^*)$ is the KKT pair of problem (1).*
**A6.** $\|(H_k - \nabla^2_{xx} L(x^*, \lambda^*)) d_k^0\| = \circ(\|d_k^0\|)$.
**Lemma 9.** *Let $(x^*, \lambda^*)$ be a KKT pair of (1). Then, there exists a convex neighborhood $\Omega$ of $(x^*, \lambda^*)$ and a positive $\mu$ such that for all $(x_k, \lambda_k) \in \Omega$, the matrix*

$$\hat{M}_k = \begin{pmatrix} \nabla^2 L(x_k, \lambda_k) & A_k \\ A_k^T & 0 \end{pmatrix}$$

*is nonsingular and $\|\hat{M}_k^{-1}\| \leq \mu$.*

*Proof.* See [2, Proposition 3.1].

**Lemma 10.** *For large enough $k$, the iterative implementation will come into Step 3 and will not go away from Step 3 under the assumptions A1', A5 and A6.*

*Proof.* If the conclusion is not true, then the iterative process will come into Step 6 infinite times. Since $(d_k^0, \lambda_k)$ is the KKT pair of the sub-problem (QP), the following equation holds

$$\left[ \begin{pmatrix} H_k^{-1} - \nabla^2 L(x_k, \lambda_k) & 0 \\ 0 & 0 \end{pmatrix} + \hat{M}_k \right] \begin{pmatrix} d_k^0 \\ \lambda_{J_k} \end{pmatrix} = \begin{pmatrix} -\nabla f(x_k) \\ -g_{J_k}(x_k) \end{pmatrix}.$$

By assumption A6, the formulation above implies

$$\hat{M}_k \begin{pmatrix} d_k^0 \\ \lambda_{J_k} \end{pmatrix} = \begin{pmatrix} -\nabla f(x_k) \\ -g_{J_k}(x_k) \end{pmatrix} + \circ(\|d_k^0\|),$$

and furthermore

$$\hat{M}_k \begin{pmatrix} x_k + d_k^0 - x^* \\ \lambda_{J_k} - \lambda^*_{J_k} \end{pmatrix} = \begin{pmatrix} -\nabla f(x_k) + \nabla^2 L(x_k, \lambda_k)(x_k - x^*) - A_k \lambda^*_{J_k} \\ -g_{J_k}(x_k) + A_k^T (x_k - x^*) \end{pmatrix} + \circ(\|d_k^0\|).$$

Let $(x_k, \lambda_k) \in \Omega$ . In view of assumption A1′, $\nabla L(x^*, \lambda^*) = 0$ and mean-value theorem, we have

$$\begin{aligned}
& \| - \nabla f(x_k) + \nabla^2 L(x_k, \lambda_k)(x_k - x^*) - A_k \lambda^*_{J_k} \| \\
\leq\ & \|\nabla f(x^*) - \nabla f(x_k) + \nabla^2 f(x_k)(x_k - x^*)\| \\
& + \| \sum_{J_k} (\lambda^*)_j [\nabla g_j(x^*) - \nabla g_j(x_k) + \nabla^2 g_j(x_k)(x_k - x^*)] \| \\
& + \| \sum_{J_k} [\lambda_{kj} - (\lambda^*)_j] \nabla^2 g_j(x_k)(x_k - x^*) \| \\
\leq\ & \| \int_0^1 [\nabla^2 f(x_k + t(x^* - x_k)) - \nabla^2 f(x_k)](x_k - x^*) dt \| \\
& + c_1 \sum_{J_k} \| \int_0^1 [\nabla^2 g_j(x_k + t(x^* - x_k)) - \nabla^2 g_j(x_k)](x_k - x^*) dt \| \\
& + c_2 \sum_{J_k} \|\lambda_{kj} - (\lambda^*)_j\| \cdot \|x_k - x^*\| \\
=\ & \circ(\|x_k - x^*\|)
\end{aligned}$$

where $c_1, c_2$ are constants.

In the same way, from $J_k \subseteq J(x^*)$, $g_{J_k}(x^*) = 0$ and assumption A1$'$, we can get that

$$\begin{aligned} & \|-g_{J_k}(x_k) + A_k^T(x_k - x^*)\| = \|g_{J_k}(x^*) - g_{J_k}(x_k) + A_k^T(x_k - x^*)\| \\ \leq & \sum_{J_k} \|g_j(x^*) - g_j(x_k) + \nabla g_j(x_k)^T(x_k - x^*)\| = \circ(\|x_k - x^*\|). \end{aligned}$$

Thus, no matter wether $x_k$ is decided by Step 3 or Step 6, if set $z_{k+1} = x_k + d_k^0$ , due to $\hat{J}(x^*) \subseteq J_k$, we always have

$$\|z_{k+1} - x^*\| = \circ(\|x_k - x^*\|), \qquad \|\lambda_{J_k} - \lambda_{J_k}^*\| = \circ(\|x_k - x^*\|). \tag{9}$$

Suppose for $k$ large enough, $x_{k+1}$ is derived from Step 3. Then $\|d_k^0\| \leq \delta$ , and from the first expression of (9)

$$\|d_k^0\| = \|x_{k+1} - x_k\| \geq \|x_k - x^*\| - \|x_{k+1} - x^*\| \geq \frac{1}{2}\|x_k - x^*\|.$$

Namely, $\|x_k - x^*\| \leq 2\delta$. Therefore,

$$\begin{aligned} \|d_{k+1}^0\| &= \|z_{k+2} - x_{k+1}\| \leq \|z_{k+2} - x^*\| + \|x_{k+1} - x^*\| \\ &\leq 2\|x_{k+1} - x^*\| \leq \frac{1}{4}\|x_k - x^*\| \leq \frac{1}{2}\delta. \end{aligned}$$

And from the second expression of (9), for $j \in J_{k+1}$

$$\lambda_{k+1,j} \geq -\|\lambda_{k+1} - \lambda^*\| \geq -\frac{\eta}{2}\|x_{k+1} - x^*\| \geq -\eta\|z_{k+2} - x_{k+1}\| = -\eta\|d_{k+1}^0\|.$$

Above two inequalities illuminate that the $(k+2)$th iteration will be in Step 3 again and $x_{k+2} = x_{k+1} + d_{k+1}^0$. Therefore, the iterative process will keep in Step 3 from now on. It contradicts that iterative process will come into Step 6 infinite times.

Based on Lemma 10, we can see when $k$ is large enough the algorithm will implement the Newton steps and will not change. Thus the following theorem holds.

**Theorem 3.** *Under all stated assumptions in the paper, the algorithm is superlinearly convergent, i.e. for large enough $k$,*

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

## 5. Numerical Test

We present the results of numerical test according to our algorithm. Four tested problems are chosen from [10] and computed by MATLAB. In the process, we select $\alpha = 0.3$, $\theta = 0.75$, $\beta = \eta = 0.5$ and $H_k$ updated by the BFGS formulation. In the following table, No. is the number of tested problems in reference [10]; Size shows the number of variables and the number of constraints; $x_0$ is the initial point; NIT stands for the iterative times; $f(x_k)$ is the final objective function value computed by the algorithm.

**Table**

| No. | Size | $x_0$ | NIT | $x_k$ | $f(x_k)$ |
|---|---|---|---|---|---|
| 264 | 4, 3 | (-2,1,-1,1) | 30 | (-0.00867,0.98530, 2.00290,-0.99880) | -43.944 |
| 268 | 5, 5 | (1,1,1,1,1) | 15 | (0.9927, 1.9919, -0.9974, 2.9815, -3.9656) | $4.33 \times 10^{-5}$ |
| 269 | 5,3 | (0,0,0,0,0) | 6 | ( -0.7674, 0.2558, 0.6279, -0.1162, 0.2558) | 4.0928 |
| 285 | 15,10 | (0,0,...,0) | 40 | * | -8245.9 |

where $* = (1.0387, 0.96163, 0.97695, 0.98887, 0.97449, 1.0477, 0.85701, 0.97957, 1.0146, 1.0524, 1.0127, 0.9992, 0.97353, 1.0134, 1.005)$.

The optimal solution and the optimal value:

No.264: $x^* = (0, 1, 2, -1)$ and $f(x^*) = -44$ ;

No.268: $x^* = (1, 2, -1, 3, -4)$ and $f(x^*) = 0$;

No.269: be not exactly clear;

No.285: $x^* = (1, 1, \cdots, 1)$ and $f(x^*) = -8252$.

**Acknowledgment.** The authors are grateful to the referees for their careful reading of the manuscript and useful advices.

## References

[1] J.F. Bonnans and G. Launay, Sequential quadratic programming with penalization of the displacement, *SIAM J. Optimization*, **54**:4 (1995), 792-812.

[2] F. Facchinei and S. Lucidi, Quadratically and superlinearly convergent for the solution of inequality constrained minimization problems, *JOTA*, **85** (1995), 265-289.

[3] Z.Y. Gao, G.P. He and F. Wu, The algorithm of sequential system of linear equation for arbitray initial point, Science in China (Series A), 27 (1997), 24-33.

[4] S.P. Han, Superlinearly conergence variable metric algorithms for general nonlinear programming problems, *Mathematical Programming*, **11** (1976), 263-282.

[5] H. Kleimmichel and K. Schoneleld, Newton-type methods for nonlinearly constrained programms, Proceedings of the 20th Jahrestagung “Mathematische Optimierung”, Humboldt-Universitat, Zu Burlin, Seminarberichte, (1988), 53-57.

[6] H. Kleinmichel, C. Richetr and K. Schonefeld, On a class of hybrid methods for smooth constrained optimizatin, *JOTA*, **73** (1992), 465-499.

[7] D.Q. Mayne and E. Polak, A superlinearly convergent algorithm for constrained optimization problems, *Math. Programming Study*, **16** (1982), 45-61.

[8] M.J.D. Powell, The convergence of variable metric method for nonlinear constrained optimization calculations, Nonlinear Programming 3, Academic Press, New York, (1978), 27-63.

[9] D.G. Pu, Y. Zhou and H.Y. Zhang, A QP free feasible method, *J. Comput. Math.,* **22** (2004), 651-660.

[10] K.Schittkowski, More test examples for nonlinear programmingcodes, Springer-Verlag, 1987.

[11] Y.F. Xu, Quasi-Newton methods: analysis and algorithm, Ph. D. Dissertation, Institute of Applied Mathematics, Chinese Academy of Sciences, January 1998.

[12] Y.F. Xu, W. Wang, A QP-free and suoerlinearly convergent algorithm for inequality constrained optimizations, *Acta mathematica scientia*, **21B** (2001), 121-130.

[13] Y. Yuan, Linear systems associated with numerical methods for constrained optimization, *J. Comput. Math.,* **21** (2003), 71-84.

[14] J.L. Zhang, C.Y. Wang, A new conjugate projection gradient method, *OR Transaction*, **3**:2 (1999), 61-70.