Hindawi Publishing Corporation
Journal of Applied Mathematics
Volume 2013, Article ID 986317, 6 pages
http://dx.doi.org/10.1155/2013/986317

Research Article

A Conjugate Gradient Type Method for the Nonnegative Constraints Optimization Problems

Can Li

College of Mathematics, Honghe University, Mengzi 661199, China

Correspondence should be addressed to Can Li; canlymathe@163.com

Received 16 December 2012; Accepted 20 March 2013

Academic Editor: Theodore E. Simos

We are concerned with the nonnegative constraints optimization problems. It is well known that the conjugate gradient methods are efficient methods for solving large-scale unconstrained optimization problems due to their simplicity and low storage. Combining the modified Polak-Ribière-Polyak method proposed by Zhang, Zhou, and Li with the Zoutendijk feasible direction method, we proposed a conjugate gradient type method for solving the nonnegative constraints optimization problems. If the current iteration is a feasible point, the direction generated by the proposed method is always a feasible descent direction at the current iteration. Under appropriate conditions, we show that the proposed method is globally convergent. We also present some numerical results to show the efficiency of the proposed method.

1. Introduction

Due to their simplicity and their low memory requirement, the conjugate gradient methods play a very important role for solving unconstrained optimization problems, especially for the large-scale optimization problems. Over the years, many variants of the conjugate gradient method have been proposed, and some are widely used in practice. The key features of the conjugate gradient methods are that they require no matrix storage and are faster than the steepest descent method.

The linear conjugate gradient method was proposed by Hestenes and Stiefel [1] in the 1950s as an iterative method for solving linear systems

$$Ax = b, \quad x \in R^n, \tag{1}$$

where A is an $n \times n$ symmetric positive definite matrix. Problem (1) can be stated equivalently as the following minimization problem

$$\min \frac{1}{2} x^T A x - b^T x, \quad x \in R^n. \tag{2}$$

This equivalence allows us to interpret the linear conjugate gradient method either as an algorithm for solving linear systems or as a technique for minimizing convex quadratic functions. For any $x \in R^n$, the sequence $\{x_k\}$ generated by the linear conjugate gradient method converges to the solution x^* of the linear systems (1) in at most n steps.

The first nonlinear conjugate gradient method was introduced by Fletcher and Reeves [2] in the 1960s. It is one of the earliest known techniques for solving large-scale nonlinear optimization problems

$$\min f(x), \quad x \in R^n, \tag{3}$$

where $f : R^n \to R$ is continuously differentiable. The nonlinear conjugate gradient methods for solving (3) have the following form:

$$x_{k+1} = x_k + \alpha_k d_k,$$
$$d_k = \begin{cases} -\nabla f(x_k), & k = 0, \\ -\nabla f(x_k) + \beta_k d_{k-1}, & k \ge 1, \end{cases} \tag{4}$$

where α_k is a steplength obtained by a line search and β_k is a scalar which deternimes the different conjugate gradient methods. If we choose f to be a strongly convex quadratic and α_k to be the exact minimizer, the nonliear conjugate gradient method reduces to the linear conjugate gradient method.

Several famous formulae for β_k are the Hestenes-Stiefel (HS) [1], Fletcher-Reeves (FR) [2], Polak-Ribière-Polyak (PRP) [3, 4], Conjugate-Descent (CD) [5], Liu-Storey (LS) [6], and Dai-Yuan (DY) [7] formulae, which are given by

$$\beta_k^{\mathrm{HS}} = \frac{\nabla f(x_k)^{\top} y_{k-1}}{d_{k-1}^{\top} y_{k-1}}, \qquad \beta_k^{\mathrm{FR}} = \frac{\|\nabla f(x_k)\|^2}{\|\nabla f(x_{k-1})\|^2}, \tag{5}$$

$$\beta_k^{\mathrm{PRP}} = \frac{\nabla f(x_k)^{\top} y_{k-1}}{\|\nabla f(x_{k-1})\|^2}, \qquad \beta_k^{\mathrm{CD}} = -\frac{\|\nabla f(x_k)\|^2}{d_{k-1}^{\top} \nabla f(x_{k-1})}, \tag{6}$$

$$\beta_k^{\mathrm{LS}} = -\frac{\nabla f(x_k)^{\top} y_{k-1}}{d_{k-1}^{\top} \nabla f(x_{k-1})}, \qquad \beta_k^{\mathrm{DY}} = \frac{\|\nabla f(x_k)\|^2}{d_{k-1}^{\top} y_{k-1}}, \tag{7}$$

where $y_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$ and $\|\cdot\|$ stands for the Euclidean norm of vectors. In this paper, we focus our attention on the Polak-Ribière-Polyak (PRP) method. The study of the PRP method has received much attention and has made good progress. The global convergence of the PRP method with exact line search has been proved in [3] under strong convexity assumption on f. However, for general nonlinear function, an example given by Powell [8] shows that the PRP method may fail to be globally convergent even if the exact line search is used. Inspired by Powell's work, Gilbert and Nocedal [9] conducted an elegant analysis and showed that the PRP method is globally convergent if β_k^{PRP} is restricted to be nonnegative and α_k is determined by a line search satisfying the sufficient descent condition $g_k^{\top} d_k \leq -c \| g_k\|^2$ in addition to the standard Wolfe conditions. Other conjugate gradient methods and their global convergence can be found in [10–15] and so forth.

Recently, Li and Wang [16] extended the modified Fletcher-Reeves (MFR) method proposed by Zhang et al. [17] for solving unconstrained optimization to the nonlinear equations

$$F(x) = 0, \tag{8}$$

where $F : R^n \to R^n$ is continuously differentiable, and proposed a descent derivative-free method for solving symmetric nonlinear equations. The direction generated by the method is descent for the residual function. Under appropriate conditions, the method is globally convergent by the use of some backtracking line search technique.

In this paper, we further study the conjugate gradient method. We focus our attention on the modified Polak-Ribière-Polyak (MPRP) method proposed by Zhang et al. [18]. The direction generated by MPRP method is given by

$$d_k = \begin{cases} -g(x_k), & k = 0, \\ -g(x_k) + \beta_k^{\mathrm{PRP}} d_{k-1} - \theta_k y_{k-1}, & k > 0, \end{cases} \tag{9}$$

where $g(x_k) = \nabla f(x_k)$, $\beta_k^{\mathrm{PRP}} = g(x_k)^T y_{k-1}/\|g(x_{k-1})\|^2$, $\theta_k = g(x_k)^T d_{k-1}/\|g(x_{k-1})\|^2$, and $y_{k-1} = g(x_k) - g(x_{k-1})$. The MPRP method not only reserves good properties of the PRP method but also possesses another nice property; that it is, always generates descent directions for the objective function. This property is independent of the line search used. Under suitable conditions, the MPRP method with the Armoji-type line search is also globally convergent. The purpose of this paper is to develop an MPRP type method for the nonnegative constraints optimization problems. Combining the Zoutendijk feasible direction method with MPRP method, we propose a conjugate gradient type method for solving the nonnegative constraints optimization problems. If the initial point is feasible, the method generates a feasible point sequence. We also do numerical experiments to test the proposed method and compare the performance of the method with the Zoutendijk feasible direction method. The numerical results show that the method that we propose outperforms the Zoutendijk feasible direction method.

2. Algorithm

Consider the following nonnegative constraints optimization problems:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \geq 0, \end{aligned} \tag{10}$$

where $f : R^n \to R$ is continuously differentiable. Let $x_k \geq 0$ be the current iteration. Define the index set

$$I_k = I(x_k) = \{i \mid x_k(i) = 0\}, \qquad J_k = \{1, 2, \ldots, n\} \setminus I_k, \tag{11}$$

where $x_k(i)$ is the ith component of x_k. In fact the index set I_k is the active set of problem (10) at x_k.

The purpose of this paper is to develop a conjugate gradient type method for problem (10). Since the iterative sequence is a feasible point sequence, the search directions should be feasible descent directions. Let $x_k \geq 0$ be the current iteration. By the definition of feasible direction, we have that [19] $d \in R^n$ is a feasible direction of (10) at x_k if and only if $d_{I_k} \geq 0$. Similar to the Zoutendijk feasible direction method, we consider the following problem:

$$\begin{aligned} \min \quad & \nabla f(x_k)^T d \\ \text{s.t.} \quad & d_{I_k} \geq 0, \quad \|d\| \leq 1. \end{aligned} \tag{12}$$

Next, we show that, if x_k is not a KKT point of (10), the solution of problem (12) is a feasible descent direction of f at x_k.

Lemma 1. *Let $x_k \geq 0$ and let $\overline{d}$ be a solution of problem* (12)*; then $\nabla f(x_k)^T \overline{d} \leq 0$. Moreover $\nabla f(x_k)^T \overline{d} = 0$ if and only if x_k is a KKT point of problem* (10).

Proof. Since $d = 0$ is a feasible point of problem (12), there must be $\nabla f(x_k)^T \overline{d} \leq 0$. Consequently, if $\nabla f(x_k)^T \overline{d} \neq 0$, there must be $\nabla f(x_k)^T \overline{d} < 0$. This implies that the direction $\overline{d}$ is a feasible descent direction of f at x_k.

We suppose that $\nabla f(x_k)^T \overline{d} = 0$. Problem (12) is equivalent to the following problem:

$$\begin{aligned} \min \quad & \nabla f(x_k)^T d \\ \text{s.t.} \quad & d_{I_k} \geq 0, \quad \|d\|^2 \leq 1. \end{aligned} \tag{13}$$

Then there exist λ_{I_k} and μ such that the following KKT condition holds:

$$\nabla f(x_k) - \begin{pmatrix} \lambda_{I_k} \\ 0 \end{pmatrix} + 2\mu\overline{d} = 0,$$
$$\lambda_{I_k} \geq 0, \qquad \overline{d}_{I_k} \geq 0, \qquad \lambda_{I_k}^T \overline{d}_{I_k} = 0, \tag{14}$$
$$\mu \geq 0, \qquad \left\|\overline{d}\right\| \leq 1, \qquad \mu\left(\left\|\overline{d}\right\|^2 - 1\right) = 0.$$

Multiplying the first of these expressions by $\overline{d}$, we obtain

$$\nabla f(x_k)^T \overline{d} - \lambda^T \overline{d} + 2\mu\left\|\overline{d}\right\|^2 = 0, \tag{15}$$

where $\lambda = \begin{pmatrix} \lambda_{I_k} \\ 0 \end{pmatrix}$. By combining the assumption $\nabla f(x_k)^T \overline{d} = 0$ with the second and the third expressions of (14), we find that $\mu = 0$. Substituting it into the first expressions of (14), we obtain that

$$\nabla f_{I_k}(x_k) - \lambda_{I_k} = 0, \qquad \nabla f_{J_k}(x_k) = 0. \tag{16}$$

Let $\lambda_i = 0$, $i \in J_k$; then $\lambda_i \geq 0$, $i \in I_k \cup J_k$. Moreover, we have

$$\nabla f(x_k) - \begin{pmatrix} \lambda_{I_k} \\ \lambda_{J_k} \end{pmatrix} = 0,$$
$$\lambda_i \geq 0, \quad x_k(i) \geq 0, \quad \lambda_i x_k(i) = 0, \quad i \in I_k \cup J_k. \tag{17}$$

This implies that x_k is a KKT point of problem (10).

On the other hand, we suppose that x_k is a KKT point of problem (10). Then there exist λ_i, $i \in I_k \cup J_k$, such that the following KKT condition holds:

$$\nabla f(x_k) - \begin{pmatrix} \lambda_{I_k} \\ \lambda_{J_k} \end{pmatrix} = 0,$$
$$\lambda_i \geq 0, \quad x_k(i) \geq 0, \quad \lambda_i x_k(i) = 0, \quad i \in I_k \cup J_k. \tag{18}$$

From the second of these expressions, we get $\lambda_{J_k} = 0$. Substituting it into the first of these expressions, we have $\nabla f_{I_k}(x_k) = \lambda_{I_k} \geq 0$ and $\nabla f_{J_k}(x_k) = 0$, so that $\nabla f(x_k)^T \overline{d} = \nabla f_{I_k}(x_k)^T \overline{d}_{I_k} = \lambda_{I_k}^T \overline{d}_{I_k} \geq 0$. However, we had shown that $\nabla f(x_k)^T \overline{d} \leq 0$, so $\nabla f(x_k)^T \overline{d} = 0$.

By the proof of Lemma 1 we find that $\nabla f_{I_k}(x_k) \geq 0$ and $\nabla f_{J_k}(x_k) = 0$ are necessary conditions of the fact that x_k is a KKT point of problem (10). We summarize these observation results as the following result. □

Lemma 2. *Let $x_k \geq 0$; then x_k is a KKT point of problem* (10) *if and only if $\nabla f_{I_k}(x_k) \geq 0$ and $\nabla f_{J_k}(x_k) = 0$.*

Proof. Firstly, we suppose that x_k is a KKT point of problem (10). Similar to the proof of Lemma 1, it is easy to get that $\nabla f_{I_k}(x_k) \geq 0$ and $\nabla f_{J_k}(x_k) = 0$.

Secondly, we suppose that $\nabla f_{I_k}(x_k) \geq 0$ and $\nabla f_{J_k}(x_k) = 0$. Let $\lambda_{I_k} = \nabla f_{I_k}(x_k) \geq 0$, $\lambda_{J_k} = 0$; then the KKT condition (18) holds, so that x_k is a KKT point of problem (10).

Based on the above discussion, we propose a conjugate gradient type method for solving problem (10) as follows. Let feasible point x_k be current iteration. For the boundary of the feasible region $x_{k_{I_k}} = 0$, we take

$$d_{k_i} = \begin{cases} 0, & g_i(x_k) > 0, \\ -g_i(x_k), & g_i(x_k) \leq 0, \end{cases} \quad \forall i \in I_k, \tag{19}$$

where $g_i(x_k) = \nabla f_i(x_k)$. For the interior of the feasible region $x_{k_{J_k}} > 0$, similar to the direction d_k in the MPRP method, we define $d_{k_{J_k}}$ by the following formula:

$$d_{k_{J_k}}^{\mathrm{MPRP}} = \begin{cases} -g_{J_k}(x_k), & k = 0, \\ -g_{J_k}(x_k) + \beta_k^{\mathrm{PRP}} d_{k-1_{J_k}} - \theta_k^{\mathrm{MPRP}} y_{k-1}, & k > 0, \end{cases} \tag{20}$$

where $g_{J_k}(x_k) = \nabla f_{J_k}(x_k)$, $\beta_k^{\mathrm{PRP}} = g_{J_k}(x_k)^T y_{k-1} / \|g(x_{k-1})\|^2$, $\theta_k^{\mathrm{MPRP}} = g_{J_k}(x_k)^T d_{k-1_{J_k}} / \|g(x_{k-1})\|^2$, and $y_{k-1} = g_{J_k}(x_k) - g_{J_k}(x_{k-1})$.

It is easy to see from (19) and (20) that

$$-\left\|g_{I_k}(x_k)\right\|^2 \leq g_{I_k}(x_k)^T d_{k_{I_k}} \leq 0,$$
$$g_{J_k}(x_k)^T d_{k_{J_k}} = -\left\|g_{J_k}(x_k)\right\|^2. \tag{21}$$

The above relations indicate that

$$g(x_k)^T d_k = g_{I_k}(x_k)^T d_{k_{I_k}} + g_{J_k}(x_k)^T d_{k_{J_k}}$$
$$\leq -\left\|g_{J_k}(x_k)\right\|^2, \tag{22}$$

$$g(x_k)^T d_k \geq -\left\|g_{I_k}(x_k)\right\|^2 - \left\|g_{J_k}(x_k)\right\|^2$$
$$= \left\|g(x_k)\right\|^2, \tag{23}$$

where $g(x_k) = \nabla f(x_k)$. □

Theorem 3. *Let $x_k \geq 0$, d_k be defined by* (19) *and* (20) *then*

$$g(x_k)^T d_k \leq 0. \tag{24}$$

Moreover, x_k is a KKT point of problem (10) *if and only if $g(x_k)^T d_k = 0$.*

Proof. Clearly, inequality (22) implies that

$$g(x_k)^T d_k \leq 0. \tag{25}$$

If x_k is a KKT point of problem (10), similar to the proof of Lemma 1, we also get that $g(x_k)^T d_k = 0$.

If $g(x_k)^T d_k = 0$, by (22), we can get that

$$g_{I_k}(x_k)^T d_{k_{I_k}} = 0,$$
$$g_{J_k}(x_k)^T d_{k_{J_k}} = -\left\|g_{J_k}(x_k)\right\|^2 = 0. \tag{26}$$

The equality $g_{I_k}(x_k)^T d_{k_{I_k}} = 0$ and the definition of $d_{k_{I_k}}$ (19) imply that

$$g_{I_k}(x_k) \geq 0. \tag{27}$$

Let $\lambda_{I_k} = g_{I_k}(x_k) \geq 0$; $\lambda_{J_k} = 0$, then the KKT condition (18) also holds, so that x_k is a KKT point of problem (10).

By combining (22) with Theorem 3, we conclude that d_k defined by (19) and (20) provides a feasible descent direction of f at x_k, if x_k is not a KKT point of problem (10). □

Based on the above process, we propose an MPRP type method for solving (10) as follows.

Algorithm 4 (MPRP type method).

Step 0. Given constants $\rho \in (0,1)$, $\delta > 0$, $\epsilon > 0$. Choose the initial point $x_0 \geq 0$; Let $k := 0$.

Step 1. Compute $d_k = (d_{k_{I_k}}, d_{k_{J_k}})$ by (19) and (20). If $|g(x_k)^T d_k| \leq \epsilon$, then stop. Otherwise, go to the next step.

Step 2. Determine $\alpha_k = \max\{\rho^j, j = 0, 1, 2, \ldots\}$ satisfying $x_k + \alpha_k d_k \geq 0$ and

$$f(x_k + \alpha_k d_k) \leq f(x_k) - \delta \alpha_k^2 \|d_k\|^2. \tag{28}$$

Step 3. Let the next iteration be $x_{k+1} = x_k + \alpha_k d_k$.

Step 4. Let $k := k + 1$ and go to Step 1.

It is easy to see that the sequence $\{x_k\}$ generated by Algorithm 4 is a feasible point sequence. Moreover, it follows from (28) that the function value sequence $\{f(x_k)\}$ is decreasing. In addition if $f(x)$ is bounded from below, we have from (28) that

$$\sum_{k=0}^{\infty} \alpha_k^2 \|d_k\|^2 < \infty. \tag{29}$$

In particular we have

$$\lim_{k\to\infty} \alpha_k \|d_k\| = 0. \tag{30}$$

Next, we prove the global convergence of Algorithm 4 under the following assumptions.

Assumption A. (1) The level set $\omega = \{x \in R^n \mid f(x) \leq f(x_0)\}$ is bound.

(2) In some neighborhood N of ω, f is continuously differentiable, and its gradient is the Lipschitz continuous; namely, there exists a constant $L > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L \|x - y\|, \quad \forall x, y \in N. \tag{31}$$

Clearly, Assumption A implies that there exists a constant γ_1 such that

$$\|\nabla f(x)\| \leq \gamma_1, \quad \forall x \in N. \tag{32}$$

Lemma 5. *Suppose that the conditions in Assumption A hold; $\{x_k\}$ and $\{d_k\}$ are the iterative sequence and the direction sequence generated by Algorithm 4. If there exists a constant $\epsilon > 0$ such that*

$$\|g(x_k)\| \geq \epsilon, \quad \forall k, \tag{33}$$

then there exists a constant $M > 0$ such that

$$\|d_k\| \leq M, \quad \forall k. \tag{34}$$

Proof. By combining (19), (20), and (33) with Assumption A, we deduce that

$$\begin{aligned}
\|d_k\| &\leq \left\|d_{k_{I_k}}\right\| + \left\|d_{k_{J_k}}^{\mathrm{MPRP}}\right\| \\
&\leq \gamma_1 + \left\|g_{J_k}(x_k)\right\| + 2\frac{\left\|g_{J_k}(x_k)\right\| \|y_{k-1}\| \left\|d_{k-1_{J_k}}^{\mathrm{MPRP}}\right\|}{\|g(x_{k-1})\|^2} \\
&\leq 2\gamma_1 + \frac{2\gamma_1 L \alpha_{k-1} \left\|d_{k-1_{J_k}}^{\mathrm{MPRP}}\right\|}{\epsilon^2} \left\|d_{k-1_{J_k}}^{\mathrm{MPRP}}\right\|.
\end{aligned} \tag{35}$$

By (30), there exists a constant $\gamma \in (0,1)$ and an integer k_0 such that the following inequality holds for all $k \geq k_0$:

$$\frac{2L\gamma_1}{\epsilon^2} \alpha_{k-1} \left\|d_{k-1_{J_k}}^{\mathrm{MPRP}}\right\| \leq \gamma. \tag{36}$$

Hence, we have for any $k \geq k_0$

$$\begin{aligned}
\|d_k\| &\leq 2\gamma_1 + \gamma \|d_{k-1}\| \\
&\leq 2\gamma_1 \left(1 + \gamma + \gamma^2 + \cdots + \gamma^{k-k_0-1}\right) \\
&\quad + \gamma^{k-k_0} \left\|d_{k_0}\right\| \\
&\leq \frac{2\gamma_1}{1-\gamma} + \left\|d_{k_0}\right\|.
\end{aligned} \tag{37}$$

Let

$$M = \max\left\{\|d_1\|, \|d_2\|, \ldots, \left\|d_{k_0}\right\|, \frac{2\gamma_1}{1-\gamma} + \left\|d_{k_0}\right\|\right\}. \tag{38}$$

Then

$$\|d_k\| \leq M, \quad \forall k. \tag{39}$$

□

Theorem 6. *Suppose that the conditions in Assumption A hold. Let $\{x_k\}$ and $\{d_k\}$ be the iterative sequence and the direction sequence generated by Algorithm 4. Then*

$$\liminf_{k\to\infty} \left|g(x_k)^T d_k\right| = 0. \tag{40}$$

Proof. We prove the result of this theorem by contradiction. Assume that the theorem is not true; then there exists a constant $\varepsilon > 0$ such that

$$\left|g(x_k)^T d_k\right| \geq \epsilon, \quad \forall k. \tag{41}$$

So by combining (41) with (23), it is easy to see that (33) holds.

(1) If $\liminf_{k\to\infty} \alpha_k > 0$, we get from (30) that $d_k \to 0$, so that $\lim_{k\to\infty} |g(x_k)^T d_k| = 0$. This contradicts assumption (41).

(2) If $\liminf_{k\to\infty}\alpha_k = 0$, there is an infinite index set K such that

$$\lim_{k\in K,\, k\to\infty} \alpha_k = 0. \tag{42}$$

It follows from Step 2 of Algorithm 4, that when $k \in K$ is sufficiently large, $\rho^{-1}\alpha_k$ does not satify $f(x_k+\alpha_k d_k) \le f(x_k) - \delta\alpha_k^2 \| d_k\|^2$; that is

$$f\left(x_k + \rho^{-1}\alpha_k d_k\right) - f\left(x_k\right) > -\delta\rho^{-2}\alpha_k^2\|d_k\|^2. \tag{43}$$

By the mean-value theorem, Lemma 1, and Assumption A, there is $h_k \in (0,1)$ such that

$$\begin{aligned} f&\left(x_k + \rho^{-1}\alpha_k d_k\right) - f\left(x_k\right) \\ &= \rho^{-1}\alpha_k g\left(x_k + h_k\rho^{-1}\alpha_k d_k\right)^T d_k \\ &= \rho^{-1}\alpha_k g(x_k)^T d_k \\ &\quad + \rho^{-1}\alpha_k \left(g\left(x_k + h_k\rho^{-1}\alpha_k d_k\right) - g\left(x_k\right)\right)^T d_k \\ &\le \rho^{-1}\alpha_k g(x_k)^T d_k + L\rho^{-2}\alpha_k^2\|d_k\|^2. \end{aligned} \tag{44}$$

Substituting the last inequality into (43), we get for all $k \in K$ sufficiently large

$$0 \le -g(x_k)^T d_k \le \rho^{-1}\left(L+\delta\right)\alpha_k\|d_k\|^2. \tag{45}$$

Taking the limit on both sides of the equation, then by combining $\| d_k \| \le M$ and recalling $\lim_{k\in K,\, k\to\infty}\alpha_k = 0$, we obtain that $\lim_{k\in K,\, k\to\infty}|g(x_k)^T d_k| = 0$. This also yields a contradiction. □

3. Numerical Experiments

In this section, we report some numerical experiments. We test the performance of Algorithm 4 and compare it with the Zoutendijk method.

The code was written in Matlab, and the program was run on a PC with 2.20 GHz CPU and 1.00 GB memory. The parameters in the method are specified as follows. We set $\rho = 1/2$, $\delta = 1/10$. We stop the iteration if $|\nabla f(x_k)^T d_k| \le 0.0001$ or the iteration number exceeds 10000.

We first test Algorithm 4 on small and medium size problems and compared them with the Zoutendijk method in the total number of iterations and the CPU time used. The test problems are from the CUTE library [20]. The numerical results of Algorithm 4 and the Zoutendijk method are listed in Table 1. The columns have the following meanings.

$P(i)$ is the number of the test problem, Dim is the dimension of the test problem, Iter is the number of iterations, and Time is CPU time in seconds.

We can see from Table 1 that Algorithm 4 has successfully solved 12 test problems, and the Zoutendijk method has successfully solved 8 test problems. From the number of iterations, Algorithm 4 has 12 test results better than Zoutendijk method. From the computation time, Algorithm 4 performs much better than the Zoutendijk method did. We then test Algorithm 4 and the Zoutendijk method on two problems with a larger dimension. The problem of VARDIM comes from [20], and the following problem comes from [16]. The results are listed in Tables 2 and 3.

Table 1: The numerical results.

$P(i)$	Dim	Algorithm 4		Zoutendijk method	
		Iter	Time	Iter	Time
3	2	1973	1.5710	—	—
4	2	201	0.2290	—	—
6	2	30	0.0160	—	—
	3	35	0.0160	—	—
	4	39	0.0470	—	—
	10	124	0.1210	—	—
	50	220	0.5370	—	—
8	3	44	0.0150	40	0.2188
11	3	3	0.0000	4	0.1094
15	4	10	0.0160	20	0.1563
18	6	322	0.0690	1936	12.0938
19	11	438	0.5440	8338	72.4219
23	50	12	0.0300	4	0.5000
24	100	142	0.3750	—	—
25	100	38	0.0810	6	0.3438
26	100	8	0.0470	6	0.1250
	1000	4	47.9060	4	190.1406

Table 2: Test results for VARDIM with various dimensions.

Problem	Dim	Algorithm 4		Zoutendijk method	
		Iter	Time	Iter	Time
VARDIM	1000	46	13.4485	—	—
	2000	55	49.0090	—	—
	3000	65	97.1020	—	—
	4000	78	164.6213	—	—
	5000	90	271.0340	—	—

Table 3: Test results for Problem 1 with various dimensions.

Problem	Dim	Algorithm 4		Zoutendijk method	
		Iter	Time	Iter	Time
Problem 1	1000	17	0.1400	8	110.2578
	2000	26	16.8604	8	263.2660
	3000	39	39.6561	11	554.0310
	4000	51	68.1729	30	910.1090
	5000	55	110.5660	—	—

Problem 1. The nonnegative constraints optimization problem

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & x \ge 0, \end{aligned} \tag{46}$$

with Engval function $f : R^n \to R$ is defined by

$$f(x) = \sum_{i=2}^{n} \left\{ \left(x_{i-1}^2 + x_i^2 \right)^2 - 4x_{i-1} + 3 \right\}. \tag{47}$$

We can see from Table 2 that Algorithm 4 has successfully solved the problem of VARDIM whose scale varies from 1000 dimensions to 5000 dimensions. However, the Zoutendijk method fails to solve the problem of VARDIM with larger dimension. From Table 3, although the number of iterations of Algorithm 4 is more than the Zoutendijk method, the computation time of Algorithm 4 is less than the Zoutendijk method, and this feature becomes more evident as increase of the dimension of the test problem.

In summary, the results from Tables 1–3 show that Algorithm 4 is more efficient than the Zoutendijk method and provides an efficient method for solving nonnegative constraints optimization problems.

Acknowledgment

This research is supported by the NSF (11161020) of China.

References

[1] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems," *Journal of Research of the National Bureau of Standards*, vol. 49, pp. 409–436, 1952.

[2] R. Fletcher and C. M. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, pp. 149–154, 1964.

[3] B. Polak and G. Ribire, "Note sur la convergence de directions conjugees," *Revue Française d'Informatique et de Recherche Opérationnelle*, vol. 16, pp. 35–43, 1969.

[4] B. T. Polyak, "The conjugate gradient method in extremal problems," *USSR Computational Mathematics and Mathematical Physics*, vol. 9, no. 4, pp. 94–112, 1969.

[5] R. Fletcher, *Practical Methods of Optimization*, John Wiley & Sons Ltd., Chichester, UK, 2nd edition, 1987.

[6] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms. I. Theory," *Journal of Optimization Theory and Applications*, vol. 69, no. 1, pp. 129–137, 1991.

[7] Y. H. Dai and Y. Yuan, "A nonlinear conjugate gradient method with a strong global convergence property," *SIAM Journal on Optimization*, vol. 10, no. 1, pp. 177–182, 1999.

[8] M. J. D. Powell, "Convergence properties of algorithms for nonlinear optimization," *SIAM Review*, vol. 28, no. 4, pp. 487–500, 1986.

[9] J. C. Gilbert and J. Nocedal, "Global convergence properties of conjugate gradient methods for optimization," *SIAM Journal on Optimization*, vol. 2, no. 1, pp. 21–42, 1992.

[10] R. Pytlak, "On the convergence of conjugate gradient algorithms," *IMA Journal of Numerical Analysis*, vol. 14, no. 3, pp. 443–460, 1994.

[11] G. Li, C. Tang, and Z. Wei, "New conjugacy condition and related new conjugate gradient methods for unconstrained optimization," *Journal of Computational and Applied Mathematics*, vol. 202, no. 2, pp. 523–539, 2007.

[12] X. Li and X. Zhao, "A hybrid conjugate gradient method for optimization problems," *Natural Science*, vol. 3, no. 1, pp. 85–90, 2011.

[13] Y. H. Dai and Y. Yuan, "An efficient hybrid conjugate gradient method for unconstrained optimization," *Annals of Operations Research*, vol. 103, pp. 33–47, 2001.

[14] W. W. Hager and H. Zhang, "A new conjugate gradient method with guaranteed descent and an efficient line search," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 170–192, 2005.

[15] D.-H. Li, Y.-Y. Nie, J.-P. Zeng, and Q.-N. Li, "Conjugate gradient method for the linear complementarity problem with S-matrix," *Mathematical and Computer Modelling*, vol. 48, no. 5-6, pp. 918–928, 2008.

[16] D.-H. Li and X.-L. Wang, "A modified Fletcher-Reeves-type derivative-free method for symmetric nonlinear equations," *Numerical Algebra, Control and Optimization*, vol. 1, no. 1, pp. 71–82, 2011.

[17] L. Zhang, W. Zhou, and D. Li, "Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search," *Numerische Mathematik*, vol. 104, no. 4, pp. 561–572, 2006.

[18] L. Zhang, W. Zhou, and D.-H. Li, "A descent modified Polak-Ribière-Polyak conjugate gradient method and its global convergence," *IMA Journal of Numerical Analysis*, vol. 26, no. 4, pp. 629–640, 2006.

[19] D. H. Li and X. J. Tong, *Numerical Optimization*, Science Press, Beijing, China, 2005.

[20] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "Testing unconstrained optimization software," *ACM Transactions on Mathematical Software*, vol. 7, no. 1, pp. 17–41, 1981.