Hindawi Publishing Corporation
Abstract and Applied Analysis
Volume 2014, Article ID 921364, 9 pages
http://dx.doi.org/10.1155/2014/921364

Research Article

Extension of Modified Polak-Ribière-Polyak Conjugate Gradient Method to Linear Equality Constraints Minimization Problems

Zhifeng Dai[1,2]

[1] *College of Business, Central South University, Hunan 410083, China*
[2] *College of Mathematics and Computational Science, Changsha University of Science and Technology, Hunan 410114, China*

Correspondence should be addressed to Zhifeng Dai; zhifengdai823@163.com

Received 19 March 2014; Revised 1 July 2014; Accepted 21 July 2014; Published 6 August 2014

Academic Editor: Daniele Bertaccini

Combining the Rosen gradient projection method with the two-term Polak-Ribière-Polyak (PRP) conjugate gradient method, we propose a two-term Polak-Ribière-Polyak (PRP) conjugate gradient projection method for solving linear equality constraints optimization problems. The proposed method possesses some attractive properties: (1) search direction generated by the proposed method is a feasible descent direction; consequently the generated iterates are feasible points; (2) the sequences of function are decreasing. Under some mild conditions, we show that it is globally convergent with Armijio-type line search. Preliminary numerical results show that the proposed method is promising.

1. Introduction

In this paper, we consider solving the following linear equality constraints optimization problem:

$$\begin{aligned} \min \quad & f(x) \\ \text{s.t.} \quad & Ax = b, \end{aligned} \tag{1}$$

where $f(x) : R^n \to R$ is a smooth function and A is a $m \times n$ matrix of rank $m (m \le n)$. In this paper, the feasible region D and the feasible direction set S are defined, respectively, as follows:

$$D = \{Ax = b\}, \qquad S = \{Ad = 0\}. \tag{2}$$

Taking the negative gradient for a search direction ($d = -\nabla f(x_k)$) is a natural way of solving unconstrained optimization problems. However, this approach does not work for constrained problems, since the gradient may not be a feasible direction. A basic technique to overcome this difficulty was initiated by Rosen [1] in 1960. To obtain a feasible search direction, Rosen projected the gradient into the feasible region; that is,

$$d = -P\nabla f(x_k), \qquad P = I - A^T\left(AA^T\right)^{-1}A. \tag{3}$$

The convergence of Rosen's gradient projection method can be proved by Du; see [2–4].

In fact, Rosen's gradient projection method is an extension of the steepest-descent method. It is well known that the drawback of the steepest-descent method is easy to suffer from zig-zagging specially when the graph of $f(x)$ has an "elongated" form. To overcome the zig-zagging, we want to use the conjugate gradient method to modify the projection direction.

It is well known that nonlinear conjugate gradient methods such as the Polak-Ribière-Polyak (PRP) method [5, 6] are very efficient for large-scale unconstrained optimization problems due to their simplicity and low storage. However, it does not necessarily satisfy the descent conditions $g_k^T d_k \le -c\|g_k\|^2$, $c > 0$.

Recently, Cheng [7] proposed a two-term modified PRP method (called TMPRP), in which the direction d_k is given by

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k^{\mathrm{PRP}}\left(I - \dfrac{g_k g_k^T}{\|g_k\|^2}\right)d_{k-1}, & \text{if } k \ge 1. \end{cases} \tag{4}$$

An attractive property of the TMPRP method is that the direction generated by the method satisfies

$$g_k^T d_k = -\|g_k\|^2, \tag{5}$$

which is independent of any line search. The presented numerical results show some potential advantage of the TMPRP method in Cheng [7]. In fact, we can easily rewrite the above direction d_k (4) as a three-term form:

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k^{\text{PRP}} d_{k-1} - \beta_k^{\text{PRP}} \dfrac{g_k^T d_{k-1}}{\|g_k\|^2} g_k, & \text{if } k \geq 1. \end{cases} \tag{6}$$

In the past few years, researchers have paid increasing attention to the conjugate gradient methods and their applications. Among others, we mention here the following works, for example, [8–29].

In the past few years, some researchers also paid attention to equality constrained problems. Martínez et al. [30] proposed a spectral gradient method for linearly constrained optimization by the following way to obtain the search direction: $d_k \in R^n$ is the unique solution of

$$\begin{aligned} \min \quad & \frac{1}{2} d^T B_k d + \nabla f(x)^T d \\ \text{s.t.} \quad & Ad = 0. \end{aligned} \tag{7}$$

In this algorithm, B_k can be computed by quasi-Newton method in which the approximate Hessians satisfy a weak secant equation. The spectral choice of steplength is embedded into the Hessian approximation and the whole process is combined with a nonmonotone line search strategy.

C. Li and D. H. Li [31] proposed a feasible Fletcher-Reeves conjugate gradient method for solving linear equality constrained optimization problem with exact line search. Their idea is to use original Fletcher-Reeves conjugate gradient method to modify the Zoutendijk direction. The Zoutendijk direction is the feasible steepest descent direction. It is a solution of the following problem:

$$\begin{aligned} \min \quad & \nabla f(x)^T d \\ \text{s.t.} \quad & Ad = 0, \ \|d\| \leq 1. \end{aligned} \tag{8}$$

Li et al. [32] also extended the modified Fletcher-Reeves conjugate gradient method in Zhang et al. [33] to solve linear equality constraints optimization problems which combined with the Zoutendijk feasible direction method. Under some mild conditions, Li et al. [32] showed that the proposed method with Armijo-type line search is globally convergent.

In this paper, we will extend the two-term Polak-Ribière-Polyak (PRP) conjugate gradient method in Cheng [7] to solve linear equality constraints optimization problems (1), which combines with the Rosen gradient projection method in Rosen [1]. Under some mild conditions, we show that it is globally convergent with Armijo-type line search.

The rest of this paper is organized as follows. In Section 2, we firstly propose the algorithm and prove the the feasible descent direction. In Section 3, we prove the global convergence of the proposed method. In Section 4, we give some improvement for the algorithm. In Section 5, we report some numerical results to test the proposed method.

2. Algorithm and the Feasible Descent Direction

In this section, we propose a two-term Polak-Ribière-Polyak conjugate gradient projection method for solving the linear equality constraints optimization problem (1). The proposed method is a combination of the well-known Rose gradient projection method and the two-term Polak-Ribière-Polyak (PRP) conjugate gradient method in Cheng [7].

The iterative process of the proposed method is given by

$$x_{k+1} = x_k + \alpha_k d_k, \tag{9}$$

and the search direction d_k is defined by

$$d_k = \begin{cases} -Pg_k, & \text{if } k = 0, \\ -Pg_k + \beta_k^{\text{PRP}} d_{k-1} - \beta_k^{\text{PRP}} \dfrac{g_k^T d_{k-1}}{\|Pg_k\|^2} Pg_k, & \text{if } k \geq 1, \end{cases} \tag{10}$$

where

$$\begin{aligned} P &= I - A^T\left(AA^T\right)^{-1} A, \qquad g_k = \nabla f\left(x_k\right), \\ y_{k-1} &= g_k - g_{k-1}, \qquad \beta_k^{\text{PRP}} = \frac{\left(Pg_k\right)^T y_{k-1}}{\|Pg_{k-1}\|^2}, \end{aligned} \tag{11}$$

and $\alpha_k > 0$ is a steplength obtained by a line search.

For convenience, we call the method (9) and (10) as EMPRP method. Now we prove that the direction d_k defined by (10) and (11) is a feasible descent direction of f at x_k.

Theorem 1. *Suppose that $x_k \in D$, $P = I - A^T(AA^T)^{-1}A$. d_k is defined by* (10). *If $d_k \neq 0$, then d_k is a feasible descent direction of f at x_k.*

Proof. From (9), (10), and the definition of P, we have

$$\begin{aligned} g_k^T d_k &= g_k^T \left(-Pg_k + \beta_k^{\text{PRP}} d_{k-1} - \beta_k^{\text{PRP}} \frac{g_k^T d_{k-1}}{\|Pg_k\|^2} Pg_k \right) \\ &= -\|Pg_k\|^2 + \beta_k^{\text{PRP}} g_k^T d_{k-1} - \beta_k^{\text{PRP}} \frac{g_k^T d_{k-1}}{\|Pg_k\|^2} \|Pg_k\|^2 \\ &= -\|Pg_k\|^2. \end{aligned} \tag{12}$$

This implies that d_k provides a descent direction of f at x_k.

In what follows, we show that d_k is a feasible descent direction of f at x_k. From (9), we have that

$$A\left(Pg_k\right) = A\left(I - A^T\left(AA^T\right)^{-1} A\right) g_k = 0. \tag{13}$$

It follows from (10) and (13) that

$$Ad_k = A\left(-Pg_k + \beta_k^{\mathrm{PRP}} d_{k-1} - \beta_k^{\mathrm{PRP}} \frac{g_k^T d_{k-1}}{\|Pg_k\|^2} Pg_k\right) = \beta_k^{\mathrm{PRP}} Ad_{k-1}. \tag{14}$$

When $k = 0$, we have

$$Ad_0 = APg_0 = 0. \tag{15}$$

It is easy to get from (14) and (15) that, for all k, $Ad_k = 0$ is satisfied. That is, d_k is feasible direction. □

In the remainder of this paper, we always assume that $f(x)$ satisfies the following assumptions.

Assumption A. (i) The level set

$$\Omega = \{x \in R^n \mid f(x) \le f(x_0)\} \tag{16}$$

is bounded.

(ii) Function $f : R^n \to R$ is continuously differentiable and bounded from below. Its gradient is Lipschitz continuous, on an open ball $\mathcal{N}$ containing Ω; that is, there is a constant $L > 0$ such that

$$\|g(x) - g(y)\| \le L\|x - y\|, \quad \forall x, y \in \mathcal{N}. \tag{17}$$

Since $f(x_k)$ is decreasing, it is clear that the sequence $\{x_k\}$ generated by Algorithm 2 is contained in Ω. In addition, we get from Assumption A that there is a constant $\gamma > 0$, such that

$$\|g(x)\| \le \gamma, \quad \forall x \in \Omega. \tag{18}$$

Since the matrix P is a projection matrix, it is reasonable to assume that there is a constant $C > 0$, such that

$$\|Pg(x)\| \le C, \quad \forall x \in \Omega. \tag{19}$$

We state the steps of the algorithm as follows.

Algorithm 2 (EMPRP method with Armijio-type line search).
Step 0. Choose an initial point $x_0 \in D$, $\varepsilon > 0$. Let $k := 0$.

Step 1. Compute d_k by (10), where β_k is computed by (11).

Step 2. If $\|Pg_k\| \le \varepsilon$ stop, else go to Step 3.

Step 3. Given $\delta \in (0, 1/2)$, $\rho \in (0, 1)$. Determine a stepsize $\alpha_k = \max(\rho^j \mid j = 0, 1, 2, \ldots)$ satisfying

$$f(x_k + \alpha_k d_k) \le f(x_k) - \delta\alpha_k^2\|d_k\|^2. \tag{20}$$

Step 4. Let $x_{k+1} = x_k + \alpha_k d_k$, and $k := k + 1$. Go to Step 1.

3. Global Convergence

In what follows, we establish the global convergence theorem of the EMPRP method for general nonlinear objective functions. We firstly give some important lemmas of the EMPRP method.

Lemma 3. *Suppose that $x_k \in D$, $P = I - A^T(AA^T)^{-1}A$. d_k is defined by* (10) *and* (11). *Then we have $(Pg_k)^T d_k = g_k^T d_k$, $(Pg_{k+1})^T d_k = g_{k+1}^T d_k$, $(Pg_k)^T d_k = g_k^T d_k = -\|Pg_k\|^2$.*

Proof. By the definition of d_k, we have

$$\begin{aligned}(Pg_k)^T d_k &= g_k^T\left(I - A^T\left(AA^T\right)^{-1}A\right)d_k \\ &= g_k^T d_k - g_k^T\left(A^T\left(AA^T\right)^{-1}\right)Ad_k = g_k^T d_k.\end{aligned} \tag{21}$$

It follows from (12) and (21) that

$$(Pg_k)^T d_k = g_k^T d_k = -\|Pg_k\|^2. \tag{22}$$

On the other hand, we also have

$$\begin{aligned}(Pg_{k+1})^T d_k &= g_{k+1}^T\left(I - A^T\left(AA^T\right)^{-1}A\right)d_k \\ &= g_{k+1}^T d_k - g_{k+1}^T\left(A^T\left(AA^T\right)^{-1}\right)Ad_k = g_{k+1}^T d_k.\end{aligned} \tag{23}$$

□

Lemma 4. *Suppose that Assumption A holds. $\{x_k\}$ is generated by Algorithm 2. If there exists a constant $\varepsilon > 0$ such that*

$$\|Pg_k\| \ge \varepsilon, \quad \forall k; \tag{24}$$

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

$$\|d_k\| \le M, \quad \forall k. \tag{25}$$

Proof. It follows from (20) that the function value sequence $\{f(x_k)\}$ is decreasing. We also have from (20) that

$$\sum_{i=1}^{\infty} -\delta\alpha_k^2\|d_k\|^2 < +\infty, \tag{26}$$

as f is bounded from below. In particular, we have

$$\lim_{k\to\infty} \alpha_k\|d_k\| = 0. \tag{27}$$

By the definition of d_k, we get from (17), (19), and (24) that

$$\begin{aligned}\|d_k\| &\le \|Pg_k\| + \frac{\|Pg_k\|\,\|y_{k-1}\|\,\|d_{k-1}\|}{\|Pg_{k-1}\|^2} \\ &\quad + \frac{\|Pg_k\|\,\|y_{k-1}\|}{\|Pg_{k-1}\|^2}\cdot\frac{\|g_k\|\,\|d_{k-1}\|\,\|Pg_k\|}{\|Pg_k\|^2} \\ &\le C + \frac{CL\alpha_{k-1}\|d_{k-1}\|}{\varepsilon^2}\|d_{k-1}\| + \frac{\gamma L\alpha_{k-1}\|d_{k-1}\|}{\varepsilon^2}\|d_{k-1}\| \\ &\le C + \left(\frac{CL}{\varepsilon^2} + \frac{\gamma L}{\varepsilon^2}\right)\alpha_{k-1}\|d_{k-1}\|^2.\end{aligned} \tag{28}$$

Since $\alpha_{k-1}d_{k-1} \to 0$, as $k \to \infty$, there exist a constant $r \in (0, 1)$ and an integer k_0, such that the following inequality holds for all $k \ge k_0$:

$$\left(\frac{CL}{\varepsilon^2} + \frac{\gamma L}{\varepsilon^2}\right)\alpha_{k-1}\|d_{k-1}\| \le r. \tag{29}$$

Hence, we have, for any $k > k_0$,

$$\begin{aligned}\|d_k\| &\le C + r\|d_{k-1}\| \le C\left(1 + r + r^2 + \cdots + r^{k-k_0-1}\right) \\ &\quad + r^{k-k_0}\|d_{k_0}\| \le \frac{C}{1-r} + \|d_{k_0}\|.\end{aligned} \tag{30}$$

Letting $M = \max\{\|d_1\|, \|d_1\|, \ldots, \|d_{k_0}\|, C/(1-r) + \|d_{k_0}\|\}$, we can get (25). □

We now establish the global convergence theorem of the EMPRP method for general nonlinear objective functions.

Theorem 5. *Suppose that Assumption A holds. $\{x_k\}$ is generated by Algorithm 2. Then we have*

$$\liminf_{k\to\infty} \|Pg_k\| = 0. \tag{31}$$

Proof. Suppose that $\liminf_{k\to\infty}\|Pg_k\| \neq 0$ for all k. Then there exists a constant $\varepsilon > 0$ such that

$$\|Pg_k\| > \varepsilon, \quad \forall k \ge 0. \tag{32}$$

We now prove (31) by considering the following two cases.

Case (i). $\liminf_{k\to\infty}\alpha_k > 0$. We get from (22) and (27) that $\liminf_{k\to\infty}\|Pg_k\| = 0$. This contradicts assumption (32).

Case (ii). $\liminf_{k\to\infty}\alpha_k = 0$. That is, there is an infinite index set K such that

$$\lim_{k\in K, k\to\infty} \alpha_k = 0. \tag{33}$$

When $k \in K$ is sufficiently large, by the line search condition, $\rho^{-1}\alpha_k$ does not satisfy inequality (20). This means

$$f(x_k + \alpha_k d_k) - f(x_k) > -\delta\rho^{-2}\alpha_k^2\|d_k\|^2. \tag{34}$$

By the mean-value theorem and inequality (17), there is a $t_k \in (0,1)$ such that $x_k + t_k\rho^{-1}\alpha_k d_k \in N$ and

$$\begin{aligned}&f\left(x_k + \rho^{-1}\alpha_k d_k\right) - f(x_k) \\ &\quad = \rho^{-1}\alpha_k g\left(x_k + t_k\rho^{-1}\alpha_k d_k\right)^T d_k \\ &\quad = \rho^{-1}\alpha_k g_k^T d_k + \rho^{-1}\alpha_k\left(g\left(x_k + t_k\rho^{-1}\alpha_k d_k\right) - g_k\right)^T d_k \\ &\quad \le \rho^{-1}\alpha_k g_k^T d_k + \rho^{-1}\alpha_k\left\|g\left(x_k + t_k\rho^{-1}\alpha_k d_k\right) - g_k\right\|\|d_k\| \\ &\quad \le \rho^{-1}\alpha_k g_k^T d_k + L\rho^{-2}\alpha_k^2\|d_k\|^2.\end{aligned} \tag{35}$$

Substituting the last inequality into (34), for all $k \in K$ sufficiently large, we have from (12) that

$$\|Pg_k\|^2 = -g_k^T d_k \le \rho^{-1}(L+\delta)\alpha_k\|d_k\|^2. \tag{36}$$

Since $\{d_k\}$ is bounded and $\lim_{k\to\infty}\alpha_k = 0$, the last inequality implies

$$\lim_{k\in K, k\to\infty} \|Pg_k\| = 0. \tag{37}$$

This also yields a contradiction. The proof is then complete. □

4. Improvement for Algorithm 2

In this section, we propose techniques for improving the efficiency of Algorithm 2 in practical computation which is about computing projections and the stepsize of the Armijo-type line search.

4.1. Computing Projection. In this paper, as in Gould et al. [34], instead of computing a basis for null space of matrix A, we choose to work directly with the matrix of constraint gradients, computing projections by normal equations. As the computation of the projection is a key step in the proposed method, following Gould et al. [34], this projection can be computed in an alternative way.

Let

$$g^+ = -Pg, \qquad P = I - A^T\left(AA^T\right)^{-1}A. \tag{38}$$

We can express this as

$$g^+ = -g + A^T v, \tag{39}$$

where v is the solution of

$$AA^T v = Ag. \tag{40}$$

Noting that (40) is the normal equation. Since A is a $m \times n$ matrix of rank $m (m \le n)$, AA^T is symmetric positive definite matrix. We use the Doolittle (called LU) factorization of AA^T to solve (40).

For the matrix A is constant, the factorization of AA^T needs to be carried out only once at the beginning of the iterative process. Using a Doolittle factorization of AA^T, (40) can be computed in the following form:

$$L\left(U^T v\right) = Ag \Longleftrightarrow \begin{cases} Ly = Ag \\ U^T v = y, \end{cases} \tag{41}$$

where L, U is the Doolittle factor of AA^T.

4.2. Computing Stepsize. The drawback of the Armijo line search is how to choose the initial stepsize α_k. If α_k is too large then the procedure needs to call much more function evaluations. If α_k is too small then the efficiency of related algorithm will be decreased. Therefore, we should choose an adequate initial stepsize α_k at each iteration. In what follows, we propose a way to generate the initial stepsize.

We first estimate the stepsize determined by the exact line search. Support at the moment that f is twice continuously differentiable. We denote by $G(x)$ the Hessian of f at x and abbreviate $G(x_k)$ as G_k. Notice that the exact line search stepsize $\overline{\alpha}_k$ satisfies

$$-g_k^T d_k = \left(g\left(x_k + \overline{\alpha}_k d_k\right) - g_k\right)^T d_k \approx \overline{\alpha}_k d_k^T G_k d_k. \tag{42}$$

This shows that scalar $\delta_k \triangleq -g_k^T d_k / d_k^T G_k d_k$ is an estimation to $\overline{\alpha}_k$. To avoid the computation of the second derivative, we further estimate $\overline{\alpha}_k$ by letting

$$\gamma_k = \frac{-\epsilon_k g_k^T d_k}{d_k^T\left(g\left(x_k + \epsilon_k d_k\right) - g_k\right)}, \tag{43}$$

where positive sequence satisfies $\{\epsilon_k\} \to 0$ as $k \to \infty$. Let the initial stepsize of the Armijo line search be an approximation to

$$\delta_k \triangleq \frac{-g_k^T d_k}{d_k^T G_k d_k} \approx \frac{-\epsilon_k g_k^T d_k}{d_k^T \left(g\left(x_k + \epsilon_k d_k\right) - g_k\right)} \triangleq \gamma_k. \tag{44}$$

It is not difficult to see that if ϵ_k and $\|d_k\|$ are sufficiently small, then δ_k and γ_k are good estimation to $\overline{\alpha}_k$.

So to improve the efficiency of EMPRP method in practical computation, we utilize the following line search process.

Line Search Process. If inequality,

$$f\left(x_k + |\gamma_k| d_k\right) \le f\left(x_k\right) + \delta |\gamma_k|^2 \|d_k\|^2, \tag{45}$$

holds, then we let $\alpha_k = |\gamma_k|$. Otherwise we let α_k be the largest scalar in the set $\{|\gamma_k|\rho^i, i = 0, 1, \ldots\}$ such that inequality (45) is satisfied.

5. Numerical Experiments

This section reports some numerical experiments. Firstly, we test the EMPRP method and compare it with the Rose gradient projection method in [1] on low dimensional problems. Secondly, we test the EMPRP method and compare it with the spectral gradient method in Martínez et al. [30] and the feasible Fletcher-Reeves conjugate gradient method in Li et al. [32] on large dimensional problems. In the line search process, we set $\epsilon_k = 10^{-6}$, $\rho = 0.3$, $\delta = 0.02$.

The methods in the tables have the following meanings.

(i) "EMPRP" stands for the EMPRP method with the Armijio-type line search (20).

(ii) "ROSE" stands for Rose gradient projection method in [1] with the Armijio-type line search (20). That is, in Algorithm 2, the direction $d_k = -Pg_k$.

(iii) "SPG" stands for the spectral gradient method with the nonmonotone line search in Martínez et al. [30], where $M = 10$, $P = 2$.

(iv) "FFR" stands for the feasible modified Fletcher-Reeves conjugate gradient method with the Armijio-type line search in Li et al. [32].

We stop the iteration if the condition $\|Pg_k\| \le \varepsilon$ is satisfied, where $\varepsilon = 10^{-5}$. If the iteration number exceeds 10^5, we also stop the iteration. Then we call it failure. All of the algorithms are coded in Matlab 7.0 and run on a personal computer with a 2.0 GHZ CPU processor.

5.1. Numerical Comparison of EMPRP and ROSE. We test the performance of EMPRP and ROSE methods on the following test problems with given initial points. The results are listed in Table 1. n stands for the dimension of tested problem and n_c stands for the number of constraints. We will report the following results: the CPU time *Time* (in seconds), the number of iterations *Iter*, the number of gradient evaluations *Geval*, and the number of function evaluations *Feval*.

Problem 1 (HS28 [35]). The function HS28 in [35] is defined as follows:

$$\begin{aligned} f(x) &= \left(x_1 + x_2\right)^2 + \left(x_2 + x_3\right)^2, \\ \text{s.t.} \quad & x_1 + 2x_2 + 3x_3 - 1 = 0, \end{aligned} \tag{46}$$

with the initial point $x_0 = (-4, 1, 1)$. The optimal solution $x^* = (0.5, -0.5, 0.5)$ and optimal function value $f(x^*) = 0$.

Problem 2 (HS48 [35]). The function HS48 in [35] is defined as follows:

$$\begin{aligned} f(x) &= \left(x_1 - 1\right)^2 + \left(x_2 - x_3\right)^2 + \left(x_4 - x_5\right)^2, \\ \text{s.t.} \quad & x_1 + x_2 + x_3 + x_4 + x_5 - 5 = 0, \\ & x_3 + 2\left(x_4 + x_5\right) + 3 = 0, \end{aligned} \tag{47}$$

with the initial point $x_0 = (3, 5, -3, 2, -2)$. The optimal solution $x^* = (1, 1, 1, 1, 1)$ and optimal function value $f(x^*) = 0$.

Problem 3 (HS49 [35]). The function HS49 in [35] is defined as follows:

$$\begin{aligned} f(x) &= \left(x_1 - x_2\right)^2 + \left(x_3 - 1\right)^2 + \left(x_4 - 1\right)^4 + \left(x_5 - 1\right)^6, \\ \text{s.t.} \quad & x_1 + x_2 + x_3 + 4x_4 - 7 = 0, \\ & x_3 + 5x_5 - 6 = 0, \end{aligned} \tag{48}$$

with the initial point $x_0 = (10, 7, 2, -3, 0.8)$. The optimal solution $x^* = (1, 1, 1, 1, 1)$ and optimal function value $f(x^*) = 0$.

Problem 4 (HS50 [35]). The function HS50 in [35] is defined as follows:

$$\begin{aligned} f(x) &= \left(x_1 - x_2\right)^2 + \left(x_2 - x_3\right)^2 + \left(x_3 - x_4\right)^2 + \left(x_4 - x_5\right)^2, \\ \text{s.t.} \quad & x_1 + 2x_2 + 3x_3 - 6 = 0, \\ & x_2 + 2x_3 + 3x_4 - 6 = 0, \\ & x_3 + 2x_4 + 3x_5 - 6 = 0, \end{aligned} \tag{49}$$

with the initial point $x_0 = (35, -31, 11, 5, -5)$. The optimal solution $x^* = (1, 1, 1, 1, 1)$ and optimal function value $f(x^*) = 0$. Moreover, we extend the dimension of function HS51 [35] to 10, 20 with the initial point $x_0 = (35, -31, 11, \ldots)$. The optimal solution $x^* = (1, 1, \ldots, 1)$ and optimal function value $f(x^*) = 0$.

Problem 5 (HS51 [35]). The function HS51 in [35] is defined as follows:

$$\begin{aligned} f(x) &= \left(x_1 - x_2\right)^2 + \left(x_2 + x_3 - 2\right)^2 + \left(x_4 - 1\right)^2 + \left(x_5 - 1\right)^2, \\ \text{s.t.} \quad & x_1 + 3x_2 - 4 = 0, \\ & x_3 + x_4 - 2x_5 = 0, \\ & x_2 - x_5 = 0, \end{aligned} \tag{50}$$

Table 1: Test results for Problems 1–5 with given initial points.

Name	n	n_c	EMPRP				ROSE			
			Time	Iter	Geval	Feval	Time	Iter	Geval	Feval
HS28	3	1	0.0156	20	22	21	0.0468	71	72	143
HS48	5	2	0.0156	26	28	27	0.0468	65	66	122
HS49	5	2	0.0156	29	30	30	0.0780	193	194	336
HS50	5	3	0.0156	22	23	23	0.0624	76	77	139
HS51	5	3	0.0156	15	16	16	0.0624	80	81	148
HS52	5	3	0.0156	30	31	31	0.0312	70	71	141
HS50E1	10	8	0.0156	16	17	17	0.0312	63	64	127
HS50E2	20	18	0.0156	15	16	16	0.0936	63	64	127

Table 2: Test results for Problem 6 with given initial points.

k	n	n_c	EMPRP		FFR		SPCG	
			Time	Iter	Time	Iter	Time	Iter
50	99	49	0.0568	73	0.0660	80	0.0256	70
100	199	99	0.1048	103	0.1060	120	0.0756	90
200	399	199	0.0868	108	0.2660	123	0.0904	97
300	599	299	0.2252	121	0.4260	126	0.7504	123
400	799	399	0.3142	121	0.5150	126	0.9804	153
500	999	499	0.4045	138	0.7030	146	1.4320	176
1000	1999	999	2.6043	138	2.1560	146	3.3468	198
2000	3999	1999	8.2608	138	6.8280	146	9.1266	216
3000	5999	2999	22.4075	138	15.1720	146	28.2177	276
4000	7999	3999	40.1268	138	35.0460	146	56.0604	324
5000	9999	4999	105.6252	138	85.0460	146	128.1504	476

with the initial point $x_0 = (2.5, 0.5, 2, -1, 0.5)$. The optimal solution $x^* = (1, 1, 1, 1, 1)$ and optimal function value $f(x^*) = 0$.

From Table 1, we can see that the EMPRP method performs better than the Rosen gradient projection method in [1], which implies that the EMPRP method can improve the computational efficiency of the Rosen gradient projection method for solving linear equality constrained optimization problems.

5.2. Numerical Comparison of EMPRP, FFR, and SPCG. In this subsection, we test the EMPRP method and compare it with the spectral gradient method (called SPCG) in Martínez et al. [30] and the feasible Fletcher-Reeves conjugate gradient method (called FFR) in Li et al. [32] on the following large dimensional problems with given initial points. The results are listed in Tables 2, 3, 4, 5, and 6. We will report the following results: the CPU time *Time* (in seconds), the number of iterations *Iter*.

Problem 6. Given a positive integer k, the function is defined as follows:

$$f(x) = \frac{1}{2}\sum_{i=1}^{k-2}(x_{k+i+1} - x_{k+i})^2, \quad \text{s.t.} \quad x_{k+i} - x_{i+1} + x_i = i, \quad i = 1, \ldots, k-1, \tag{51}$$

with the initial point $x_0 = (1, 2, \ldots, k, 2, 3, \ldots, k)$. The optimization function value $f(x^*) = 0$. This problem comes from Martínez et al. [30].

Problem 7. Given a positive integer n, the function is defined as follows:

$$f(x) = \sum_{i=1}^{n} \cos\left(2\pi x_i * \sin\left(\frac{\pi}{20}\right)\right), \quad \text{s.t.} \quad x_i - x_{i+1} = 0.4 \quad i = 1, \ldots, n-1, \tag{52}$$

with the initial point $x_0 = (1, 0.6, 0.2, \ldots, 1 - 0.4^{n-1})$. This problem comes from Asaadi [36] and is called MAD6.

Problem 8. Given a positive integer k, the function is defined as follows:

$$f(x) = \frac{1}{2}\sum_{i=1}^{k-2}(x_{k+i+1} - x_{k+i})^4, \quad \text{s.t.} \quad x_{k+i} - x_{i+1} + x_i = i, \quad i = 1, \ldots, k-1, \tag{53}$$

with the initial point $x_0 = (1, 2, \ldots, k, 2, 3, \ldots, k)$. The optimization function value $f(x^*) = 0$.

Table 3: Test results for Problem 7 with given initial points.

n	n_c	EMPRP		FFR		SPCG	
		Time	Iter	Time	Iter	Time	Iter
200	99	0.1468	83	0.1679	94	0.1286	79
400	199	0.2608	94	0.2860	103	0.2266	90
600	299	0.6252	121	0.8422	132	0.7504	133
800	399	1.4075	123	1.6510	138	1.4177	149
1000	499	3.280	141	2.7190	146	2.8790	168
2000	999	8.4045	189	4.6202	202	5.1280	190
3000	1499	12.0042	219	9.7030	248	10.7864	260
4000	1999	17.2608	254	13.4255	288	19.1266	316
5000	2499	20.6252	321	19.0940	330	29.1504	383

Table 4: Test results for Problem 8 with given initial points.

k	n	n_c	EMPRP		FFR		SPCG	
			Time	Iter	Time	Iter	Time	Iter
50	99	49	0.0668	78	0.0750	83	0.0286	75
100	199	99	0.0988	113	0.1462	130	0.0826	106
200	399	199	0.1978	123	0.2960	133	0.1864	124
300	599	299	0.3436	128	0.5260	136	0.3504	156
400	799	399	0.4142	134	0.7143	136	0.5804	186
500	999	499	0.6039	152	0.9030	158	0.9331	228
1000	1999	999	2.9065	152	2.9570	158	3.1260	268
2000	3999	1999	8.3408	152	7.8280	158	8.2266	298
3000	5999	2999	19.6086	152	17.4350	158	19.8672	330
4000	7999	3999	46.3268	152	38.1460	158	56.1256	368
5000	9999	4999	75.3972	152	58.3867	158	125.7680	398

Table 5: Test results for Problem 9 with given initial points.

k	n	n_c	EMPRP		FFR		SPCG	
			Time	Iter	Time	Iter	Time	Iter
50	99	49	0.0568	65	0.0640	70	0.0266	62
100	199	99	0.0868	109	0.1362	125	0.0726	103
200	399	199	0.2110	120	0.2862	128	0.2064	124
300	599	299	0.3250	127	0.5062	132	0.4524	160
400	799	399	0.4543	138	0.7456	139	0.6804	192
500	999	499	0.6246	156	0.9268	155	1.0876	258
1000	1999	999	3.9245	156	3.4268	162	3.4560	279
2000	3999	1999	9.3404	156	8.6240	162	8.6280	320
3000	5999	2999	20.6125	156	19.6548	162	20.8656	358
4000	7999	3999	48.2890	156	44.4330	162	57.7680	386
5000	9999	4999	95.4680	156	83.8650	162	128.8760	420

Problem 9. Given a positive integer k, the function is defined as follows:

$$f(x) = \sum_{i=1}^{k-2} 100\left(x_{k+i+1} - x_{k+i}\right)^2 + \left(1 - x_{k+i}\right)^2, \quad \text{s.t.} \quad x_{k+i} - x_{i+1} + x_i = i, \quad i = 1, \ldots, k-1, \tag{54}$$

with the initial point $x_0 = (1, 2, \ldots, k, 2, 3, \ldots, k)$. The optimization function value $f(x^*) = 0$.

Problem 10. Given a positive integer n, the function is defined as follows:

$$f(x) = \sum_{i=1}^{n} \cos^2\left(2\pi x_i * \sin\left(\frac{\pi}{20}\right)\right), \quad \text{s.t.} \quad x_i - x_{i+1} = 0.4 \quad i = 1, \ldots, n-1, \tag{55}$$

with the initial point $x_0 = (1, 0.6, 0.2, \ldots, 1 - 0.4 * (n-1))$.

Table 6: Test results for Problem 10 with given initial points.

n	n_c	EMPRP		FFR		SPCG	
		Time	Iter	Time	Iter	Time	Iter
200	99	0.1842	88	0.1984	98	0.1488	92
400	199	0.2908	104	0.3260	113	0.2348	100
600	299	0.8256	132	0.9422	142	0.9804	182
800	399	1.6078	136	1.7512	148	2.6177	259
1000	499	2.9801	148	2.8192	156	3.1790	268
2000	999	7.8045	199	5.1202	212	8.1288	320
3000	1499	12.8042	248	9.9035	256	14.7862	360
4000	1999	19.2432	304	14.4258	298	22.1268	386
5000	2499	29.6846	382	20.1932	378	32.1422	393

From Tables 2–6, we can see the EMPRP method and the FFR method in [32] perform better than the SPCG method in Martínez et al. [30] for solving large-scale linear equality constrained optimization problems, as the EMPRP method and the FFR method all are first order methods. But the SPCG method in Martínez et al. [30] needs to compute B_k with quasi-Newton method in which the approximate Hessians satisfy a weak secant equation. However, as the EMPRP method also needs to compute projection, the FFR method in [32] performs better than the EMPRP method when the test problem becomes large.

6. Conclusions

In this paper, we propose a new conjugate gradient projection method for solving linear equality constrained optimization problem (1), which combines the two-term modified Polak-Ribière-Polyak (PRP) conjugate gradient method in Cheng [7] with the Rosen projection method. The proposed method also can be regarded as an extension of the recently developed two-term modified Polak-Ribière-Polyak (PRP) conjugate gradient method in Cheng [7]. Under some mild conditions, we show that it is globally convergent with the Armijio-type line search.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The author thanks the anonymous referees for their valuable comments and suggestions that improved the presentation. This work is supported by the NSF of China Grants 11301041, 11371154, 71371065, and 71371195 and the Ministry of Education Humanities and Social Sciences Project Grants 12YJC790027, Project funded by China Postdoctoral Science Foundation, Natural Science Foundation of Hunan Province.

References

[1] J. B. Rosen, "The gradient projection method for nonlinear programming. I. Linear constraints," *SIAM Journal on Applied Mathematics*, vol. 8, no. 1, pp. 181–217, 1960.

[2] D. Z. Du and X. S. Zhang, "A convergence theorem of Rosen's gradient projection method," *Mathematical Programming*, vol. 36, no. 2, pp. 135–144, 1986.

[3] D. Du, "Remarks on the convergence of Rosen's gradient projection method," *Acta Mathematicae Applicatae Sinica*, vol. 3, no. 2, pp. 270–279, 1987.

[4] D. Z. Du and X. S. Zhang, "Global convergence of Rosen's gradient projection method," *Mathematical Programming*, vol. 44, no. 1, pp. 357–366, 1989.

[5] E. Polak and G. Ribire, "Note surla convergence de directions conjuguees," *Rev Francaise informat Recherche Operatinelle 3e Annee*, vol. 16, pp. 35–43, 1969.

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

[7] W. Cheng, "A two-term PRP-based descent method," *Numerical Functional Analysis and Optimization*, vol. 28, no. 11, pp. 1217–1230, 2007.

[8] 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.

[9] 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.

[10] N. Andrei, "Accelerated scaled memoryless BFGS preconditioned conjugate gradient algorithm for unconstrained optimization," *European Journal of Operational Research*, vol. 204, no. 3, pp. 410–420, 2010.

[11] G. Yu, L. Guan, and W. Chen, "Spectral conjugate gradient methods with sufficient descent property for large-scale unconstrained optimization," *Optimization Methods and Software*, vol. 23, no. 2, pp. 275–293, 2008.

[12] G. Yuan, "Modified nonlinear conjugate gradient methods with sufficient descent property for large-scale optimization problems," *Optimization Letters*, vol. 3, no. 1, pp. 11–21, 2009.

[13] Q. Li and D.-H. Li, "A class of derivative-free methods for large-scale nonlinear monotone equations," *IMA Journal of Numerical Analysis*, vol. 31, no. 4, pp. 1625–1635, 2011.

[14] Y. Xiao and H. Zhu, "A conjugate gradient method to solve convex constrained monotone equations with applications in compressive sensing," *Journal of Mathematical Analysis and Applications*, vol. 405, no. 1, pp. 310–319, 2013.

[15] Z. Dai, "Two modified HS type conjugate gradient methods for unconstrained optimization problems," *Nonlinear Analysis: Theory, Methods & Applications*, vol. 74, no. 3, pp. 927–936, 2011.

[16] Y.-Y. Chen and S.-Q. Du, "Nonlinear conjugate gradient methods with Wolfe type line search," *Abstract and Applied Analysis*, vol. 2013, Article ID 742815, 5 pages, 2013.

[17] S. Y. Liu, Y. Y. Huang, and H. W. Jiao, "Sufficient descent conjugate gradient methods for solving convex constrained nonlinear monotone equations," *Abstract and Applied Analysis*, vol. 2014, Article ID 305643, 12 pages, 2014.

[18] Z. F. Dai, D. H. Li, and F. H. Wen, "Robust conditional value-at-risk optimization for asymmetrically distributed asset returns," *Pacific Journal of Optimization*, vol. 8, no. 3, pp. 429–445, 2012.

[19] C. Huang, C. Peng, X. Chen, and F. Wen, "Dynamics analysis of a class of delayed economic model," *Abstract and Applied Analysis*, vol. 2013, Article ID 962738, 12 pages, 2013.

[20] C. Huang, X. Gong, X. Chen, and F. Wen, "Measuring and forecasting volatility in Chinese stock market using HAR-CJ-M model," *Abstract and Applied Analysis*, vol. 2013, Article ID 143194, 13 pages, 2013.

[21] G. Qin, C. Huang, Y. Xie, and F. Wen, "Asymptotic behavior for third-order quasi-linear differential equations," *Advances in Difference Equations*, vol. 30, no. 13, pp. 305–312, 2013.

[22] C. Huang, H. Kuang, X. Chen, and F. Wen, "An LMI approach for dynamics of switched cellular neural networks with mixed delays," *Abstract and Applied Analysis*, vol. 2013, Article ID 870486, 8 pages, 2013.

[23] Q. F. Cui, Z. G. Wang, X. Chen, and F. Wen, "Sufficient conditions for non-Bazilevic functions," *Abstract and Applied Analysis*, vol. 2013, Article ID 154912, 4 pages, 2013.

[24] F. Wen and X. Yang, "Skewness of return distribution and coefficient of risk premium," *Journal of Systems Science & Complexity*, vol. 22, no. 3, pp. 360–371, 2009.

[25] F. Wen and Z. Liu, "A copula-based correlation measure and its application in Chinese stock market," *International Journal of Information Technology & Decision Making*, vol. 8, no. 4, pp. 787–801, 2009.

[26] F. Wen, Z. He, and X. Chen, "Investors' risk preference characteristics and conditional skewness," *Mathematical Problems in Engineering*, vol. 2014, Article ID 814965, 14 pages, 2014.

[27] F. Wen, X. Gong, Y. Chao, and X. Chen, "The effects of prior outcomes on risky choice: evidence from the stock market," *Mathematical Problems in Engineering*, vol. 2014, Article ID 272518, 8 pages, 2014.

[28] F. Wen, Z. He, X. Gong, and A. Liu, "Investors' risk preference characteristics based on different reference point," *Discrete Dynamics in Nature and Society*, vol. 2014, Article ID 158386, 9 pages, 2014.

[29] Z. Dai and F. Wen, "Robust CVaR-based portfolio optimization under a genal a_ne data perturbation uncertainty set," *Journal of Computational Analysis and Applications*, vol. 16, no. 1, pp. 93–103, 2014.

[30] J. M. Martínez, E. A. Pilotta, and M. Raydan, "Spectral gradient methods for linearly constrained optimization," *Journal of Optimization Theory and Applications*, vol. 125, no. 3, pp. 629–651, 2005.

[31] C. Li and D. H. Li, "An extension of the Fletcher-Reeves method to linear equality constrained optimization problem," *Applied Mathematics and Computation*, vol. 219, no. 23, pp. 10909–10914, 2013.

[32] C. Li, L. Fang, and X. Cui, "A feasible fletcher-reeves method to linear equality constrained optimization problem," in *Proceedings of the International Conference on Apperceiving Computing and Intelligence Analysis (ICACIA '10)*, pp. 30–33, December 2010.

[33] L. Zhang, W. J. Zhou, and D. H. Li, "Global convergence of a modified Fletcher-Reeves conjugate gradient method with Armijo-type line search," *Numerische Mathematik*, vol. 104, no. 2, pp. 561–572, 2006.

[34] N. I. M. Gould, M. E. Hribar, and J. Nocedal, "On the solution of equality constrained quadratic programming problems arising in optimization," *SIAM Journal on Scientific Computing*, vol. 23, no. 4, pp. 1376–1395, 2001.

[35] W. Hock and K. Schittkowski, *Test examples for nonlinear programming codes*, Lecture Notes in Economics and Mathematical Systems, Springer, New York, NY, USA, 1981.

[36] J. Asaadi, "A computational comparison of some non-linear programs," *Mathematical Programming*, vol. 4, no. 1, pp. 144–154, 1973.