GLOBAL CONVERGENCE OF THE POLAK-RIBIÈRE-POLYAK CONJUGATE GRADIENT METHOD WITH AN ARMIJO-TYPE INEXACT LINE SEARCH FOR NONCONVEX UNCONSTRAINED OPTIMIZATION PROBLEMS

ZENG XIN WEI, GUO YIN LI, AND LI QUN QI

ABSTRACT. We propose two algorithms for nonconvex unconstrained optimization problems that employ Polak-Ribière-Polyak conjugate gradient formula and new inexact line search techniques. We show that the new algorithms converge globally if the function to be minimized has Lipschitz continuous gradients. Preliminary numerical results show that the proposed methods for particularly chosen line search conditions are very promising.

1. INTRODUCTION

The conjugate gradient (CG) method has played a special role for solving large-scale nonlinear optimization due to the simplicity of their iteration and their very low memory requirements. In fact, the CG method is not among the fastest or more robust optimization algorithms for nonlinear problems available today, but it remains very popular for engineers and mathematicians who are interested in solving large problems [24, 25]. The nonlinear conjugate gradient method is designed to solve the following unconstrained optimization problem:

$$(1.1) \qquad \min\{f(x) \mid x \in \Re^n\},$$

where $f : \Re^n \to \Re$ is a smooth, nonlinear function whose gradient will be denoted by g. We consider only the case where the method is implemented without regular restarts. The iterative formula of the conjugate gradient methods is given by

$$(1.2) \qquad x_{k+1} = x_k + t_k d_k,$$

where t_k is a step length which is computed by carrying out a line search, and d_k is the search direction defined by

$$(1.3) \qquad d_k = \begin{cases} -g_k & \text{if} \quad k = 1, \\ -g_k + \beta_k d_{k-1} & \text{if} \quad k \geq 2, \end{cases}$$

where β_k is a scalar, and g_k denotes $g(x_k)$. There are at least six formulas for β_k, which are given below:

$$(1.4) \qquad \beta_k^{HS} = \frac{g_k^T(g_k - g_{k-1})}{(g_k - g_{k-1})^T d_{k-1}} \qquad \text{(Hestenses-Stiefel [19], 1952)};$$

$$(1.5) \qquad \beta_k^{FR} = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}} \qquad \text{(Fletcher-Reeves [14], 1964)};$$

$$(1.6) \qquad \beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{g_{k-1}^T g_{k-1}} \qquad \text{(Polak-Ribière-Polyak [26, 27], 1969)};$$

$$(1.7) \qquad \beta_k^{CD} = -\frac{g_k^T g_k}{d_{k-1}^T g_{k-1}} \qquad \text{(Conjugate Descent [13], 1987)};$$

$$(1.8) \qquad \beta_k^{LS} = -\frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T g_{k-1}} \qquad \text{(Liu-Storey [21], 1991)};$$

$$(1.9) \qquad \beta_k^{DY} = \frac{g_k^T g_k}{(g_k - g_{k-1})^T d_{k-1}} \qquad \text{(Dai-Yuan [8], 1999)};$$

$$(1.10) \qquad \beta_k^{DY-HS} = \max\{0, \min\{\beta^{DY}, \beta^{HS}\}\} \qquad \text{(Dai-Yuan [11], 2001)}.$$

Considerable attention has been given to the global convergence for the above methods. Zoutendijk [34] proved that the FR method with exact line search is globally convergent. Al Baali [2] extended this result to the strong Wolfe-Powell line search. Powell [28] proved that the sequence of gradient norms g_k could be bounded away from zero only when

$$(1.11) \qquad \sum_{k \geq 0} \frac{1}{\|d_k\|} < +\infty.$$

So one can prove that the FR method is globally convergent for general functions by using (1.11). However, the global convergence has not been established for the PRP method with the standard Wolfe-Powell line search condition. In fact, Powell proved that even if the step length was chosen to be the least positive minimizer of the one variable function ($\phi_k(t) = f(x_k + td_k), t \in \Re$), the PRP method could cycle infinitely without approaching a solution.

Under the sufficient descent condition

$$(1.12) \qquad g_k^T d_k \leq -c\|g_k\|^2,$$

Powell [28] gave another way to discuss the global convergence of the PRP method with the weak Wolfe-Powell line search. In [28], the parameter β_k in (1.6) is not allowed to be negative, i.e.,

$$(1.13) \qquad \beta_k = \max\{\beta_k^{PRP}, 0\}.$$

By using a complicated line search, Gilbert and Nocedal [15] were able to establish the global convergence result of the PRP and HS methods by restricting the scalar β_k to be nonnegative.

Grippo and Lucidi [17] proposed a new line search condition which was designed to match the requirements of the convergence theory to ensure that the Polak-Ribière-Polyak method is globally convergent for nonconvex problems. However, the method given by Grippo and Lucidi in [17] did not perform better than the PRP method which employed (1.13) and the standard (strong) Wolfe-Powell line

search in the numerical computations. The line search in [17] is as follows: For any given $\gamma > 0$ and $\rho \in (0,1)$, find

$$(1.14) \qquad t_k = \max\{\rho^j \cdot \frac{\gamma|g_k^T d_k|}{\|d_k\|^2}; \ \ j = 0, 1, ...\}$$

such that t_k satisfies

$$(1.15) \qquad f(x_{k+1}) \leq f(x_k) - \delta t_k^2 \|d_k\|^2$$

and

$$(1.16) \qquad -c_1 \|g(x_k + t_k d_k)\|^2 \leq g(x_k + t_k d_k)^T d_{k+1} \leq -c_2 \|g(x_k + t_k d_k)\|^2,$$

where $0 < c_2 < 1 < c_1$ are constants. Dai and Yuan [9, 12] and the authors in [5] gave a further study for the convergence conditions of the PRP method if β_k is defined by (1.13) with the Wolfe-Powell line search conditions. They concluded that for the PRP method, the boundedness of the level set and the restriction (1.13) are indispensable. The global convergence results of the PRP method and some other conjugate gradient methods without line search have been given; see [3, 6, 30].

Under the Armijo inexact line search, Dai [4, 12] discussed the global convergence of the FR and the PRP methods in a systematic way. He concluded that the PRP method is globally convergent if the objective function f is strictly convex and the sufficient descent conditions (1.12) hold. Two new formulas (1.8) and (1.9) were given in [8, 21]. The new nonlinear conjugate gradient formula (1.9) has been proved to have some very nice global convergence properties [8, 11, 12].

Considerable research efforts have been made to obtain efficient and robust methods for finding a step length at a given iterate x_k and a given descent direction d_k; see [1, 16, 22] and [33]. There are at least four types of line search procedures: the Armijo step length procedure, the Armijo-Goldstein step length procedure, the standard Wolfe-Powell step length procedure and the strong Wolfe-Powell step length procedure. The Armijo rule is finding $t_k = \rho^{j_k}$ such that j_k is the smallest nonnegative integer j satisfying

$$(1.17) \qquad f(x_k + \rho^j d_k) - f(x_k) \leq \alpha \rho^j g_k^T d_k,$$

where $\rho \in (0,1)$ and $\alpha \in [0, \frac{1}{2})$. The Armijo-Goldstein rule is that $t_k > 0$ satisfies

$$(1.18) \qquad f(x_k + t_k d_k) - f(x_k) \leq \alpha t_k g_k^T d_k$$

and

$$(1.19) \qquad f(x_k + t_k d_k) - f(x_k) \geq (1 - \alpha) t_k g_k^T d_k$$

where $\alpha \in (0, 1/2)$. The standard Wolfe-Powell criterion is that t_k satisfies (1.18) and

$$(1.20) \qquad g(x_k + t_k d_k)^T d_k \geq \lambda g_k^T d_k,$$

where $\alpha \in (0,1)$ and $\lambda \in (\alpha, 1)$. The strong Wolfe-Powell criterion is that t_k satisfies (1.18) and

$$(1.21) \qquad |g(x_k + t_k d_k)^T d_k| \leq \lambda |g_k^T d_k|.$$

These procedures played a crucial role in the study of minimization problems in the past forty years. Recently, the authors in [31, 32] proposed three new types of line searches by using the function

$$f_k = f + \frac{1}{2}(x - x_k)^T B_k (x - x_k)$$

in (1.17), (1.18), (1.19) and (1.20), where B_k is a simple symmetric and positive definite matrix. More precisely, they gave the following three types of line search conditions.

1. A modification of the Armijo step length procedure (MA):
    Find $t_k = \rho^{j_k}$ such that j_k is the smallest nonnegative integer j satisfying

$$(1.22) \qquad f(x_k + \rho^j d_k) - f(x_k) \leq \alpha \rho^j g_k^T d_k - \frac{1}{2}(\rho^j)^2 d_k^T B_k d_k,$$

where $\rho \in (0, 1)$ and $\alpha \in (0, \frac{1}{2})$.

2. A modification of the Armijo-Goldstein step length procedure (MAG):
    Find $t_k > 0$ such that

$$f(x_{k+1}) - f(x_k) \leq \alpha t_k g_k^T d_k - \frac{1}{2} t_k^2 d_k^T B_k d_k$$

and

$$f(x_{k+1}) - f(x_k) \geq (1 - \alpha) t_k g_k^T d_k - \frac{1}{2} t_k^2 d_k^T B_k d_k,$$

where $\alpha \in (0, \frac{1}{2})$.

3. A modification of the standard Wolfe-Powell step length procedure (MWP):
    Find $t_k > 0$ such that

$$f(x_{k+1}) - f(x_k) \leq \alpha t_k g_k^T d_k - \frac{1}{2} t_k^2 d_k^T B_k d_k$$

and

$$g(x_{k+1})^T d_k \geq -t_k d_k^T B_k d_k + \lambda g_k^T d_k,$$

where $\alpha \in (0, \frac{1}{2})$ and $\lambda \in (\alpha, 1)$.

For the strong Wolfe-Powell line search conditions, we may give a modification of the strong Wolfe-Powell step length procedure (MSWP) as follows:

4. A modification of the strong Wolfe-Powell step length procedure (MSWP):
    Find $t_k > 0$ such that

$$(1.23) \qquad f(x_{k+1}) - f(x_k) \leq \alpha t_k g_k^T d_k - \frac{1}{2} t_k^2 d_k^T B_k d_k$$

and

$$(1.24) \qquad -t_k d_k^T B_k d_k + \lambda g_k^T d_k \leq g(x_{k+1})^T d_k \leq -t_k d_k^T B_k d_k - \lambda g_k^T d_k,$$

where $\alpha \in (0, \frac{1}{2})$ and $\lambda \in (\alpha, 1)$. (For a proof of the existence of the step length in MA, MAG, MWP and MSWP see [31].)

The purpose of this paper is to study the global convergence behavior of the PRP method by using the structure of B_k. Our motivation is also based on the following facts: 1) the PRP formula (1.6) is more effective when it is applied to nonlinear optimization problems, see [24]; 2) unlike the other line search conditions, the Armijo-type line search condition can use the back-tracking technique. This tends to make finding the step length vary in a predictable manner; 3) the sufficient descent condition (1.12) is a very nice and important property for conjugate gradient methods, so one hopes to guarantee this property for the conjugate gradient methods. For the above three objects, we discuss the global convergence of the PRP method with the following Armijo-type line search (ATLS), which is based on (1.22), the ideas given by Grippo and Lucidi [17] and the structure of the formula (1.6). (For convenience, we let $B_k = \mu I$ for all k with a fixed positive scalar μ.)

The Armijo-type line search (ATLS). Let $\alpha \in [0, 1/2)$, $c \in (0, 1)$, $\mu > 0$ and $\phi_k > 0$ be given. Denote $t_k^j = \rho^j \cdot \phi_k$. The ATLS is to find $t_k = t_k^{j_k}$ where j_k is the smallest number of j such that

$$(1.25) \qquad f(x_k + t_k^j d_k) - f(x_k) \le \alpha\, t_k^j\, g_k^T d_k - \frac{\mu}{2}(t_k^j)^2 \|d_k\|^2$$

and

$$(1.26) \qquad g(x_k + t_k^j d_k)^T Q_k(j) \le -c\|g(x_k + t_k^j d_k)\|^2,$$

where $Q_k(j)$ is defined as

$$(1.27) \qquad Q_k(j) = -g(x_k + t_k^j d_k) + \frac{g(x_k + t_k^j d_k)^T (g(x_k + t_k^j d_k) - g_k)}{\|g_k\|^2} d_k.$$

Here ϕ_k is a parameter which plays an important role for improving the initial step length. In fact, Nocedal [25] pointed out that the efficiency of nonlinear conjugate gradient methods would be greatly improved if one could design a variant of the PRP method that produced well-scaled search directions without increasing the storage requirements of the iteration. In what follows, we propose a way to improve the initial step length by a reasonable choice of ϕ_k in Section 2.

This paper is organized as follows. In Section 2, we first prove an important property for ATLS and then present a PRP algorithm with ATLS (Algorithm 1), the sufficient descent property (1.12) of the Algorithm 1 is also proved in this section. In Section 3, we establish the global convergence of Algorithm 1, and prove the global convergence of the PRP formula with the MSWP. The preliminary numerical results are contained in Section 4. Some final remarks are given in the last section.

2. Algorithm and its properties

Before giving our algorithm, we first prove the following lemma, which shows that the Armijo-type line search given by (1.25), (1.26) and (1.27) is reasonable.

Lemma 2.1. *For any $k \in \mathbb{N}$, let $\alpha \in [0, 1/2)$, $\mu > 0$ and $\phi_k > 0$ be given. If $g_k^T d_k < 0$ holds for all $k \in \mathbb{N}$, then there exists a nonnegative integer j_k such that $t_k = \phi_k \rho^{j_k}$ satisfies ATLS.*

Proof. We first prove that there is a j_0, such that for all $j \ge j_0$, (1.25) holds. Suppose that the conclusion does not hold. Then by passing to the subsequence if necessary, we may assume that for any j,

$$f(x_k + \phi_k \rho^j d_k) - f(x_k) > \alpha\phi_k \rho^j g_k^T d_k - \frac{\mu}{2}(\phi_k \rho^j)^2 \|d_k\|^2.$$

From Taylor's expansion, we have

$$\phi_k \rho^j g_k^T d_k + o(\rho^j) > \alpha\phi_k \rho^j g_k^T d_k - \frac{\mu}{2}(\phi_k \rho^j)^2 \|d_k\|^2.$$

Dividing both sides by ρ^j and letting $j \to \infty$, we obtain

$$\phi_k g_k^T d_k \ge \alpha\phi_k g_k^T d_k.$$

Thus we have $\alpha > 1$ since $g_k^T d_k < 0$ and $\phi_k > 0$. This contradicts to $\alpha \in [0, \frac{1}{2})$. Second, we prove that there exists $j_1 \in (j_0, +\infty)$, such that (1.26) holds. Suppose that the conclusion does not hold. Then for any $j > j_0$, we have

$$g(x_k + \phi_k \rho^j d_k)^T Q_k(j) > -c\|g(x_k + \phi_k \rho^j d_k)\|^2.$$

From the definition of $Q_k(j)$ in (1.27), we can deduce that

$$g(x_k + \phi_k\rho^j d_k)^T(-g(x_k + \phi_k\rho^j d_k) + \frac{g(x_k + \phi_k\rho^j d_k)^T(g(x_k + \phi_k\rho^j d_k) - g_k)}{\|g_k\|^2}d_k)$$
$$> -c\|g(x_k + \phi_k\rho^j d_k)\|^2.$$

Therefore

$$-\|g(x_k + \phi_k\rho^j d_k)\|^2 + \frac{g(x_k + \phi_k\rho^j d_k)^T(g(x_k + \phi_k\rho^j d_k) - g_k)}{\|g_k\|^2}g(x_k + \phi_k\rho^j d_k)^T d_k$$
$$> -c\|g(x_k + \phi_k\rho^j d_k)\|^2.$$

Letting $j \in (j_0, +\infty)$ and $j \to \infty$, by the continuity of g, we obtain that

$$-\|g_k\|^2 \geq -c\|g_k\|^2$$

which implies that $c > 1$ from $g_k \neq 0$. This contradicts to $c \in (0,1)$. Therefore from the above discussions, there exists a j_k such that (1.25) and (1.26) hold. $\quad\square$

Next, we propose our modified Polak-Ribière-Polyak method. To this end, we first introduce a reasonable choice for selecting ϕ_k as follows (which is motivated by [20]):

A reasonable choice for ϕ_k. Consider the following quadratic model:

$$q_k(t) = f(x_k) + tg_k^T d_k + \frac{1}{2}t^2 d_k^T \nabla^2 f(x_k)d_k.$$

Note that if $\epsilon > 0$ is sufficiently small, then the following approximate relation holds:

$$\nabla^2 f(x_k)d_k \approx \frac{g(x_k + \epsilon d_k) - g_k}{\epsilon}.$$

Define

$$(2.1) \qquad\qquad z_k = \frac{g(x_k + \epsilon d_k) - g_k}{\epsilon},$$

it follows that

$$q_k(t) \approx f(x_k) + tg_k^T d_k + \frac{1}{2}t^2 d_k^T z_k$$

and

$$q_k'(t) \approx g_k^T d_k + td_k^T z_k.$$

If $d_k^T z_k \neq 0$, by letting $t_0 = \frac{-g_k^T d_k}{d_k^T z_k}$ this implies that $q_k'(t_0) = 0$. In this regard, it motivates us to choose ϕ_k by the following rule: let $\eta > 0$ be a real number (close to 0), then

$$(2.2) \qquad\qquad \phi_k = \begin{cases} \frac{-g_k^T d_k}{d_k^T z_k}, & \text{if} \quad \frac{-g_k^T d_k}{d_k^T z_k} \geq \eta, \\ 1, & \text{else}, \end{cases}$$

where z_k is defined as in (2.1) (as we will see in Section 4, this choice works quite well by selecting $\epsilon = 1e - 8$ and $\eta = 1e - 10$).

From Lemma 2.1 and our choice of ϕ_k, we are now ready to state our modified Polak-Ribière-Polyak conjugate gradient method as follows:

Algortithm 1 (Modified PRP methods: MPRP).

Step 0: Let $x_1 \in \Re^n$, $\alpha \in [0, 1/2)$, $c \in (0, 1)$, $\mu, \epsilon, \eta > 0$. Set $d_1 = -g_1$, $k = 1$. If $g_1 = 0$, then stop.

Step 1: Compute ϕ_k by (2.2) and find a $t_k > 0$ satisfying ATLS.

Step 2: Let $x_{k+1} = x_k + t_k d_k$ and $g_{k+1} = g(x_{k+1})$. If $g_{k+1} = 0$, then stop.

Step 3: Compute β_k by the PRP formula (1.6) and generate d_{k+1} by (1.3).

Step 4: Set $k := k + 1$, go to Step 2.

The following lemma shows that Algorithm 1 has a very nice property due to the definition of Q_k given in (1.27).

Lemma 2.2. *For any given k, if $g_k^T d_k < 0$ and $(t_k, x_{k+1}, g_{k+1}, \beta_k, d_{k+1})$ is generated by Algorithm 1, then*

$$(2.3) \qquad g_{k+1}^T d_{k+1} \leq -c\|g_{k+1}\|^2.$$

Proof. From Lemma 2.1, we can generate a stepsize t_k by ATLS and then generate $x_{k+1}, g_{k+1}, \beta_k$ and d_{k+1} by Algorithm 1. Therefore

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 + \beta_{k+1}^{PRP} g_{k+1}^T d_k$$

$$\leq -\|g_{k+1}\|^2 + \frac{g_{k+1}^T (g_{k+1} - g_k)}{\|g_k\|^2} g_{k+1}^T d_k$$

$$= -\|g(x_k + t_k d_k)\|^2 + \frac{g(x_k + t_k d_k)^T (g(x_k + t_k d_k) - g_k)}{\|g_k\|^2} g(x_k + t_k d_k)^T d_k.$$

Since t_k satisfies (1.26), we have

$$- \|g(x_k + t_k d_k)\|^2 + \frac{g(x_k + t_k d_k)^T (g(x_k + t_k d_k) - g_k)}{\|g_k\|^2} g(x_k + t_k d_k)^T d_k$$

$$\leq -c\|g(x_k + t_k d_k)\|^2.$$

So, we obtain our conclusion (2.3). $\qquad\square$

From Lemma 2.1 and 2.2, we have the following theorem which indicates that Algorithm 1 is well defined and has a nice property—the sufficient descent condition (1.12).

Theorem 2.3. *It holds that either there is a $k_0 \in \mathbb{N}$ such that $g_{k_0} = 0$ or the algorithm generates a sequence $\{x_k\}$ such that the property (1.12) holds for all k.*

Proof. If $g_1 = 0$, then we have finished our proof. Suppose $g_1 \neq 0$. Then we have $d_1 = -g_1$, so $g_1^T d_1 = -\|g_1\|^2 \neq 0$ and

$$g_1^T d_1 \leq -c\|g_1\|^2.$$

Using Lemma 2.1 and Lemma 2.2, we can generate (t_1, x_2, g_2). If $g_2 \neq 0$, we can have β_1 and d_2. Using Lemma 2.2 again, we have

$$g_2^T d_2 \leq -c\|g_2\|^2.$$

Repeating the above discussions and using Lemmas 2.1 and 2.2, we can deduce our conclusion by induction. $\qquad\square$

Remark 2.1. For the FR formula (1.5), we use the following Armijo-type line search: find $t_k = \rho^{j_k}$ such that j_k is the smallest nonnegative integer j satisfying

$$(2.4) \qquad f(x_k + \phi_k \rho^j d_k) - f(x_k) \leq \alpha \phi_k \rho^j g_k^T d_k - \frac{\mu}{2} (\phi_k \rho^j)^2 d_k^T d_k$$

and

(2.5) $$g(x_k + \phi_k\rho^j d_k)^T Q_k^{FR}(j) \le -c\|g(x_k + \phi_k\rho^j d_k)\|^2,$$

where Q_k^{FR} is defined as

(2.6) $$Q_k^{FR}(j) = -g(x_k + \phi_k\rho^j d_k) + \frac{g(x_k + \phi_k\rho^j d_k)^T g(x_k + \phi_k\rho^j d_k)}{\|g_k\|^2} d_k.$$

Then we can have the same algorithm as Algorithm 1. Furthermore, the results of Lemma 2.1 and Lemma 2.2 for the PR method still hold. It is easy to see that (2.5) holds if and only if

(2.7) $$g(x_k + \phi_k\rho^j d_k)^T d_k \le (1-c)\|g_k\|^2.$$

Does the FR formula (1.5) with the line search conditions (2.4) and (2.7) converge globally?

Remark 2.2. For the conjugate descent formula (1.7), we can use the following line search: find $t_k = \rho^{j_k}$ such that j_k is the smallest nonnegative integer j satisfying

(2.8) $$f(x_k + \phi_k\rho^j d_k) - f(x_k) \le \alpha\phi_k\rho^j g_k^T d_k - \frac{\mu}{2}(\phi_k\rho^j)^2 d_k^T d_k$$

and

(2.9) $$g(x_k + \phi_k\rho^j d_k)^T Q_k^{CD}(j) \le -c\|g(x_k + \phi_k\rho^j d_k)\|^2,$$

where Q_k^{CD} is defined as

(2.10) $$Q_k^{CD}(j) = -g(x_k + \phi_k\rho^j d_k) - \frac{g(x_k + \phi_k\rho^j d_k)^T g(x_k + \phi_k\rho^j d_k)}{g_k^T d_k} d_k.$$

Then we can have the same algorithm as Algorithm 1. Furthermore, the results of Lemma 2.1 and Lemma 2.2 for the CD method still hold. It is easy to see from $g_1^T d_1 = -\|g_1\|^2 < 0$ that

(2.11) $$g(x_k + \phi_k\rho^j d_k)^T d_k \le -(1-c)g_k^T d_k$$

holds if and only if (2.9) holds. So for the CD method, (2.9) can be replaced by (2.11). It is known that if $m = 0$ and $1 - c \neq 0$, then the CD formula (1.7) with the line search (2.8) and (2.11) need not converge; see [10] and [12] for details. Does the conjugate descent formula (1.7) with the line search conditions (2.8) and (2.11) converge globally?

Remark 2.3. For the LS formula (1.8), we can use the following line search: find $t_k = \rho^{j_k}$ such that j_k is the smallest nonnegative integer j satisfying

(2.12) $$f(x_k + \phi_k\rho^j d_k) - f(x_k) \le \alpha\phi_k\rho^j g_k^T d_k - \frac{\mu}{2}(\phi_k\rho^j)^2 d_k^T d_k$$

and

(2.13) $$g(x_k + \phi_k\rho^j d_k)^T Q_k^{LS}(j) \le -c\|g(x_k + \phi_k\rho^j d_k)\|^2,$$

where Q_k^{LS} is defined as

(2.14) $$Q_k^{LS}(j) = -g(x_k + \phi_k\rho^j d_k) - \frac{g(x_k + \phi_k\rho^j d_k)^T (g(x_k + \phi_k\rho^j d_k) - g_k)}{g_k^T d_k} d_k.$$

Then we can have the same algorithm as Algorithm 1. Furthermore, the results of Lemma 2.1 and Lemma 2.2 and for the LS method still hold. We conjecture that the LS formula (1.8) with the line search conditions (2.12) and (2.13) is convergent globally.

3. Global convergence

In this section, we prove global convergence of Algorithm 1 under the following assumptions.

Assumption A. The level set $\Omega = \{x \in R^n : f(x) \leq f(x_1)\}$ is bounded where x_1 is the initial point.

Assumption B. There exists a constant L such that for any $x, y \in \Omega$,

$$(3.1) \qquad \|g(x) - g(y)\| \leq L\|x - y\|.$$

Since $\{f(x_k)\}$ is a decreasing sequence, it is clear that the sequence $\{x_k\}$ generated by Algorithm 1 is contained in Ω, and there exists a constant f^*, such that

$$(3.2) \qquad \lim_{k \to \infty} f(x_k) = f^*.$$

Lemma 3.1. *Suppose that Assumption A holds. Then we have*

$$(3.3) \qquad \lim_{k \to \infty} t_k\|d_k\| = 0$$

and

$$(3.4) \qquad \lim_{k \to \infty} -t_k g_k^T d_k = 0.$$

Proof. From (3.2), we have

$$
\begin{aligned}
\sum_{k=1}^{\infty}(f(x_k) - f(x_{k+1})) &= \lim_{N \to \infty} \sum_{k=1}^{N}(f(x_k) - f(x_{k+1})) \\
&= \lim_{N \to \infty} (f(x_1) - f(x_{k+1})) \\
&= f(x_1) - f^*.
\end{aligned}
$$

Thus

$$\sum_{k=1}^{\infty}(f(x_k) - f(x_{k+1})) < +\infty,$$

which when combined with

$$f(x_k + t_k d_k) - f(x_k) \leq \alpha t_k g_k^T d_k - \frac{\mu}{2}t_k^2\|d_k\|^2$$

yields

$$(3.5) \qquad \sum_{k=1}^{\infty} t_k^2\|d_k\|^2 < +\infty$$

and

$$(3.6) \qquad \sum_{k=1}^{\infty} -t_k g_k^T d_k < +\infty.$$

Therefore, (3.3) and (3.4) hold. $\qquad\square$

The property (3.3) is very important for proving the global convergence of Algorithm 1, and it is not yet known by us whether (3.3) holds for other conjugate gradient methods with another line search condition (for example, the standard Wolfe-Powell conditions).

Lemma 3.2. *Suppose that Assumptions A and B hold. If there exists a constant $\epsilon > 0$ such that for all k,*

$$(3.7) \qquad \|g_k\| \geq \epsilon,$$

then there exists a constant $M_2 > 0$ such that for all k,

$$(3.8) \qquad \|d_k\| \leq M_2.$$

Proof. From the definition of d_k, we have

$$\begin{aligned} \|d_k\| &\leq \|g_k\| + |\beta_k^{PRP}| \|d_{k-1}\| \\ &\leq \|g_k\| + (\|g_k\| \|g_k - g_{k-1}\| / \|g_{k-1}\|^2) \|d_{k-1}\|. \end{aligned}$$

Using Assumption B, we obtain

$$(3.9) \qquad \|d_k\| \leq \|g_k\| + \|g_k\| \frac{L t_{k-1} \|d_{k-1}\|}{\epsilon^2} \|d_{k-1}\|.$$

On the other hand, from the fact that $\{x_k\}$ is bounded, by using Assumption B, we can deduce that there exists $M_3 > 0$ such that for all k,

$$(3.10) \qquad \|g_k\| \leq M_3.$$

Thus (3.9) and (3.10) yield the following inequality:

$$(3.11) \qquad \|d_k\| \leq M_3 + \frac{M_3 L}{\epsilon^2} t_{k-1} \|d_{k-1}\|^2 = M_3 + \left(\frac{L M_3}{\epsilon^2} t_{k-1} \|d_{k-1}\|\right) \|d_{k-1}\|.$$

From (3.3), this implies that there exist a constant $q \in (0, 1)$ and an integer k_0, such that for all $k \geq k_0$,

$$\frac{L M_3}{\epsilon^2} t_{k-1} \|d_{k-1}\| \leq q.$$

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

$$\begin{aligned} \|d_k\| &\leq M_3 + q \|d_{k-1}\| \\ &\leq M_3 (1 + q + q^2 + \cdots + q^{k-k_0-1}) + q^{k-k_0} \|d_{k_0}\| \\ &\leq \frac{M_3}{1-q} + q^{k-k_0} \|d_{k_0}\| \\ &\leq \frac{M_3}{1-q} + \|d_{k_0}\|. \end{aligned}$$

Setting $M_2 = \max\{\|d_1\|, \|d_2\|, \cdots, \|d_{k_0}\|, \frac{M_3}{1-q} + \|d_{k_0}\|\}$, we deduce that for all k, (3.8) holds. $\square$

Lemma 3.3. *Suppose that Assumptions A and B hold. Let $\{x_k\}$ be generated by Algorithm 1. Then there exists a constant $M_1 > 0$ such that for all k,*

$$(3.12) \qquad t_k \geq M_1 \|g_k\|^2 / \|d_k\|^2.$$

Proof. We will divide our proof into two cases.

Case 1. $t_k = 1$. In this case, we have

$$\|g_k\|^2 \leq \frac{1}{c} |g_k^T d_k| \leq (1/c) \|g_k\| \|d_k\|$$

by using (1.12). Hence

$$\|g_k\| \leq \frac{1}{c} \|d_k\|.$$

Using $t_k = 1$, we have

$$\|g_k\|^2 \leq \frac{1}{c^2}\|d_k\|^2 = \frac{t_k}{c^2}\|d_k\|^2.$$

Hence

$$(3.13) \qquad\qquad t_k \geq c^2\|g_k\|^2/\|d_k\|^2.$$

Case 2. $t_k < 1$. In this case, we have $j_k - 1$ is a nonnegative integer. From the definition of t_k, (1.25) and (1.26) cannot be simultaneously satisfied for $t_k/\rho = \phi_k\rho^{j_k-1}$.

Case 2.1. t_k/ρ does not satisfy (1.25). From the definition of t_k, we have

$$f(x_k + (t_k/\rho)d_k) - f(x_k) > \alpha(t_k/\rho)g_k^T d_k - \frac{\mu}{2}(t_k/\rho)^2\|d_k\|^2.$$

Using the Mean Value Theorem in the above inequality, we obtain $\theta_k \in (0,1)$, such that

$$(t_k/\rho) \cdot g(x_k + \theta_k(t_k/\rho)d_k)^T d_k > \alpha(t_k/\rho)g_k^T d_k - \frac{\mu}{2}(t_k/\rho)^2\|d_k\|^2.$$

Dividing both sides of the above inequality by t_k/ρ, we have

$$g(x_k + \theta_k(t_k/\rho)d_k)^T d_k > \alpha g_k^T d_k - \frac{\mu}{2}(t_k/\rho)\|d_k\|^2.$$

Subtracting $g_k^T d_k$ on both sides of the above inequality, we obtain

$$(g(x_k + \theta_k(t_k/\rho)d_k) - g_k)^T d_k > -(1-\alpha)g_k^T d_k - \frac{\mu}{2}(t_k/\rho)\|d_k\|^2,$$

which when combined with Assumption B yields

$$L\theta_k(t_k/\rho)\|d_k\|^2 > -(1-\alpha)g_k^T d_k - \frac{\mu}{2}(t_k/\rho)\|d_k\|^2.$$

Therefore

$$t_k > \frac{2(1-\alpha)\rho}{2L\theta_k + \mu}\frac{(-g_k^T d_k)}{\|d_k\|^2}.$$

Hence, we have

$$(3.14) \qquad t_k > \frac{2c(1-\alpha)\rho}{2L\theta_k + \mu}\frac{\|g_k\|^2}{\|d_k\|^2} \geq \frac{2c(1-\alpha)\rho}{2L + \mu}\frac{\|g_k\|^2}{\|d_k\|^2}$$

by using (1.12) and $\theta_k \in (0,1)$.

Case 2.2. t_k does not satisfy (1.26). In this case, we have

$$-c\|g(x_k + (t_k/\rho)d_k)\|^2 < g(x_k + (t_k/\rho))d_k)^T Q_k(j_k - 1).$$

Using the definition of Q_k given in (1.27), we have

$$-c\|g(x_k + (t_k/\rho)d_k)\|^2 < -\|g(x_k + (t_k/\rho)d_k)\|^2$$
$$+ \frac{g(x_k + (t_k/\rho)d_k)^T(g(x_k + (t_k/\rho)d_k) - g_k)}{\|g_k\|^2}g(x_k + (t_k/\rho)d_k)^T d_k.$$

Hence

$$-c\|g(x_k + (t_k/\rho)d_k)\|^2 < -\|g(x_k + (t_k/\rho)d_k)\|^2$$
$$+ \frac{\|g(x_k + (t_k/\rho)d_k)\|^2}{\|g_k\|^2}\|g(x_k + (t_k/\rho)d_k) - g_k\|\|d_k\|.$$

Dividing both sides of the above inequality by $\|g(x_k + (t_k/\rho)d_k)\|^2$, we have

$$(3.15) \qquad -c \leq -1 + \frac{\|g(x_k + (t_k/\rho)d_k) - g_k\|\|d_k\|}{\|g_k\|^2}.$$

Using Assumption B, we obtain,

$$-c \leq -1 + \frac{L(t_k/\rho)\|d_k\|^2}{\|g_k\|^2}.$$

Hence

(3.16) $$t_k > ((1-c)\rho/L)(\|g_k\|^2/\|d_k\|^2).$$

This finishes the proof of Case 2.2.

Letting

$$M_1 = \min\{c^2, \; \frac{2(1-\alpha)\rho}{2L+\mu}, \; \frac{(1-c)\rho}{L}\},$$

we obtain (3.12) from the discussions in Case 1 and Case 2. $\qquad\square$

By using the above lemmas, we are now ready to establish the following global convergence theorem for Algorithm 1.

Theorem 3.4. *Suppose that Assumptions A and B hold. Let $\{x_k\}$ be generated by Algorithm 1. Then*

(3.17) $$\liminf_{k\to\infty} \|g_k\| = 0.$$

Proof. Suppose that the conclusion does not hold. Then there exists a constant $\epsilon > 0$ such that for all k,

(3.18) $$\|g_k\| \geq \epsilon.$$

By Lemma 3.2, we can obtain a constant $M_1 > 0$, such that for all k,

$$t_k \geq M_1\|g_k\|^2/\|d_k\|^2,$$

which when combined with Lemma 3.3 yields

$$\|g_k\|^2 \leq \frac{M_2}{M_1}t_k\|d_k\|.$$

This make a contradiction to (3.18) by using (3.3) and letting $k \to \infty$ in the above inequality. Therefore, (3.17) holds. $\qquad\square$

The proof of the global convergence result of Algorithm 1 is very different from those given in the literatures. The general outline of the proofs in the literatures is that, assuming that the conclusion does not hold, i.e., (3.7) holds, one can derive that

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} = +\infty,$$

which contradicts the Zoutendijk condition. In [5], they gave a general and positive result for conjugate methods with the strong Wolfe line search (see Corollary 2.4 of [5]). Corollary 2.4 of [5] is also based on the Zoutendijk condition. Moreover, if a conjugate gradient method fails to converge ((3.7) holds), one can easily see from Corollary 2.4 in [5] that the length of the direction will converge to infinity. Unlike the proofs mentioned above for the global convergence results given for CG methods, we did not use the Zoutendijk condition here; furthermore, we proved that the search direction is bounded if (3.7) holds.

Remark 3.1. The global convergence properties of the PRP method with (1.13) (PRP^+) and SWP has been widely discussed; see [3, 5, 9, 11, 12, 15, 17, 25, 30] for details. It is known from [9] and [12] that the PRP^+ with the SWP is globally convergent if $g_k^T d_k \leq 0$ for all k. In this remark, we will prove a similar result for the PRP with the MSWP, i.e., we prove the global convergence of the following algorithm if for all $k, g_k^T d_k \leq 0$.

Algortithm 2 (PRP formula (1.6) with MSWP)**.**
    Step 0: Let $x_1 \in \Re^n$, $\alpha \in (0, 1)$ and $\lambda \in (\alpha, 1)$. Set $d_1 = -g_1, k = 1$. If $g_1 = 0$, then stop.
    Step 1: Find a $t_k > 0$ satisfying MSWP.
    Step 2: Let $x_{k+1} = x_k + t_k d_k$ and $g_{k+1} = g(x_{k+1})$. If $g_{k+1} = 0$, then stop.
    Step 3: Compute β_k by PRP formula (1.6) and generate d_{k+1} by (1.3).
    Step 4: Set $k = k + 1$, go to Step 2.

If for all $k, g_k^T d_k \leq 0$, then Algorithm 2 is well defined and has the following global convergence result. (Note that for convenience, we let $B_k = \mu I$ for all k with a fixed positive scalar μ.)

Theorem 3.5. *Suppose that Assumptions A and B hold. Let $\{x_k\}$ be generated by Algorithm 2. Suppose that for all $k \in \mathbb{N}$ we have*

$$(3.19) \qquad\qquad g_k^T d_k \leq 0.$$

Then it holds that

$$\liminf_{k \to \infty} \|g_k\| = 0.$$

Proof. Using Assumption B and the MSWP, we obtain

$$-(1 - \lambda)g_k^T d_k - \mu t_k \|d_k\|^2 \leq (g_{k+1} - g_k)^T d_k \leq L t_k \|d_k\|^2.$$

So

$$(3.20) \qquad\qquad t_k \geq \frac{-(1 - \lambda)g_k^T d_k}{(L + \mu)\|d_k\|^2},$$

which when combined with (3.19) yields

$$\sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} \leq \frac{L + \mu}{1 - \lambda} \sum_{k=1}^{\infty} (-t_k g_k^T d_k).$$

Using (3.6), we have the following Zoutendijk condition:

$$(3.21) \qquad\qquad \sum_{k=1}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty.$$

On the other hand, we have that for all $k \geq 2$,

$$d_k + g_k = \beta_k^{PRP} d_{k-1}.$$

Squaring both sides of the above equation, we may get

$$\|d_k\|^2 = -\|g_k\|^2 - 2g_k^T d_k + (\beta_k^{PRP})^2 \|d_{k-1}\|^2,$$

which when combined with (3.19) yields

$$\|d_k\|^2 \geq (\beta_k^{PRP})^2 \|d_{k-1}\|^2 - \|g_k\|^2.$$

From (1.3) we have

$$g_k^T d_k - \beta_k^{PRP} g_k^T d_{k-1} = -\|g_k\|^2.$$

According to the MSWP, we have

$$|g_k^T d_k| + \lambda|\beta_k^{PRP}||g_{k-1}^T d_{k-1}| + \mu t_{k-1}|\beta_k^{PRP}| \, \|d_{k-1}\|^2 \geq \|g_k\|^2.$$

Squaring both sides of the above inequality and using the inequality

$$(a + b + c)^2 \leq 3(a^2 + b^2 + c^2) \quad (\forall a, b, c > 0),$$

we have

$$(g_k^T d_k)^2 + \lambda^2(\beta_k^{PRP})^2(g_{k-1}^T d_{k-1})^2 \geq \frac{1}{3}\|g_k\|^4 - \mu^2(\beta_k^{PRP})^2 t_{k-1}^2\|d_{k-1}\|^4.$$

Using $\beta \in (0,1)$, we have

$$(g_k^T d_k)^2 + (\beta_k^{PRP})^2(g_{k-1}^T d_{k-1})^2 \geq \frac{1}{3}\|g_k\|^4 - \mu^2(\beta_k^{PRP})^2 t_{k-1}^2\|d_{k-1}\|^4.$$

Therefore, we can deduce that

$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} = \frac{1}{\|d_k\|^2}\left((g_k^T d_k)^2 + \frac{\|d_k\|^2}{\|d_{k-1}\|^2}(g_{k-1}^T d_{k-1})^2\right)$$

$$\geq \frac{1}{\|d_k\|^2}\left((g_k^T d_k)^2 + (\beta_k^{PRP})^2(g_{k-1}^T d_{k-1})^2 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2}\|g_k\|^2\right)$$

$$\geq \frac{1}{\|d_k\|^2}\left(\frac{1}{3}\|g_k\|^4 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2}\|g_k\|^2 - \mu^2(\beta_k^{PRP})^2 t_{k-1}^2\|d_{k-1}\|^4\right).$$

Hence

(3.22)
$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} + \mu^2(\beta_k^{PRP})^2\frac{t_{k-1}^2\|d_{k-1}\|^4}{\|d_k\|^2}$$

$$\geq \frac{1}{\|d_k\|^2}\left(\frac{1}{3}\|g_k\|^4 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2}\|g_k\|^2\right).$$

Noting that (by the Cauchy-Schwartz inequality and the Assumption B)

(3.23)
$$|\beta_k^{PRP}| = |\frac{g_k^T(g_k - g_{k-1})}{g_{k-1}^T g_{k-1}}| \leq \frac{Lt_{k-1}\|g_k\|^2\|d_{k-1}\|}{\|g_{k-1}\|^2}$$

$$\leq \frac{M_3^2 L}{\epsilon_1^2}t_{k-1}\|d_{k-1}\| = \tilde{M}t_{k-1}\|d_{k-1}\|,$$

where $\tilde{M} = \frac{M_3^2 L}{\epsilon_1^2}$, it follows that

(3.24)
$$\frac{(g_k^T d_k)^2}{\|d_k\|^2} + \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} + \mu^2\tilde{M}^2\frac{t_{k-1}^4\|d_{k-1}\|^6}{\|d_k\|^2}$$

$$\geq \frac{1}{\|d_k\|^2}\left(\frac{1}{3}\|g_k\|^4 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2}\|g_k\|^2\right).$$

Next we will complete the proof by contradiction. Suppose that the conclusion does not hold, i.e., there exists a constant $\epsilon_1 > 0$ such that for all k,

(3.25)
$$\|g_k\| \geq \epsilon_1.$$

In particular, we have

(3.26)
$$\|g_k\|^2 \leq \frac{1}{\epsilon_1^2}\|g_k\|^4.$$

From the Zoutendijk condition (3.21) it follows that

$$(3.27) \qquad \lim_{k\to\infty} \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2} = 0.$$

This implies that for all large k,

$$(3.28) \qquad \frac{1}{3}\|g_k\|^4 - \frac{(g_{k-1}^T d_{k-1})^2}{\|d_{k-1}\|^2}\|g_k\|^2 \geq \frac{1}{6}\|g_k\|^4.$$

On the other hand, from Lemma 3.2 there exists $\bar{M}_2 > 0$ such that $\|d_k\| \leq \bar{M}_2$ for all k. This implies that

$$\|d_k\| \geq \|g_k\| - |\beta_k^{PRP}|\|d_{k-1}\| \geq \epsilon_1 - |\beta_k^{PRP}|\bar{M}_2.$$

It follows from (3.3) and (3.23) that there exists $k_0 \in \mathbb{N}$ such that for $k \geq k_0$,

$$\|d_k\| \geq \epsilon_1/2.$$

Consequently, from $\|d_k\| \leq \bar{M}_2$ and (3.5) we obtain

$$(3.29) \qquad \sum_{k=k_0}^{\infty} \frac{t_{k-1}^4\|d_{k-1}\|^6}{\|d_k\|^2} \leq \frac{4\bar{M}_2^2}{\epsilon_1^2} \sum_{k=k_0}^{\infty} t_{k-1}^4\|d_{k-1}\|^4 < \infty.$$

Using (3.24), (3.28), (3.29) and the Zoutendijk condition (3.21) we have

$$(3.30) \qquad \sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < +\infty.$$

However, noting that (by (3.25) and $\|d_k\| \leq \bar{M}_2$)

$$\frac{\|d_k\|^2}{\|g_k\|^4} \leq \frac{\bar{M}_2^2}{\epsilon_1^4},$$

we have

$$\sum_{k=1}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} \geq \sum_{k=1}^{\infty} \frac{\epsilon_1^4}{\bar{M}_2^2} = +\infty.$$

This makes a contradiction and completes the proof. $\qquad\square$

Remark 3.2. In this remark, we give an algorithm (Algorithm 3) using the LS formula and the line search condition MSWP and conjecture that Algorithm 3 is convergent globally if the descent condition (3.19) holds for all k.

Algortithm 3 (LS formula (1.8) with MSWP)**.**
  Step 0. Given $x_1 \in \Re^n$, set $d_1 = -g_1, k = 1$. If $g_1 = 0$, then stop.
  Step 1. Find a $t_k > 0$ satisfying MSWP.
  Step 2. Let $x_{k+1} = x_k + t_k d_k$ and $g_{k+1} = g(x_{k+1})$. If $g_{k+1} = 0$, then stop.
  Step 3. Compute β_k by LS formula (1.8) and generate d_{k+1} by (1.3).
  Step 4. Set $k = k + 1$, go to Step 2.

Remark 3.3. Similarly to the PRP method, the HS method need not converge under the exact line search used. Thus a natural question is: Does the HS formula (1.4) with the MSWP converge globally? This might be a topic for further research.

4. NUMERICAL EXPERIMENTS

In this section, we report the numerical results for Algorithm 1 and Algorithm 2. The problems that we tested are from [23]. For each test problem, the termination condition is $\|g_k\| \leq 10^{-6}$ or the iteration exceeds 5000. In order to rank the iterative numerical methods, one can compute the total of the function and gradient evaluations by the formula

$$(4.1) \qquad\qquad N_{total} = N_f + \theta \cdot N_g$$

where N_f and N_g denote the number of function evaluations and gradient evaluations, respectively, and θ is some integer. From the results on automatic differentiation (cf. [7, 18]), the value of θ can be chosen from 2 to 5 (hence one gradient evaluation is equivalent to at least 2 of the function evaluations if automatic differentiation is used). Notice the following two facts: (1) The variant of PRP methods and DY-HS methods are regarded as the most efficient conjugate methods of the existing methods; (2) We are interested in conjugate gradient methods such that they have not only global convergence property but also good numerical performance; (3) The conjugate gradient methods are especially popular and useful for large-scale problems. Hence we choose our two new methods to compare with several variants of PRP methods and the DY-HS method for some large-scale problems. Tables 4.1.1, 4.1.2, 4.3.1 and 4.3.2 show the computation results, where the columns have the following meanings:

Problem: the name of the test problem in MATLAB;

Dim: the dimension of the problem;

NI: the number of iterations:

N_f: the number of function evaluations:

N_g: the number of gradient evaluations.

Our numerical report contains three parts. In Subsection 4.1, we tested the MPRP method with different parameters c and μ. In Subsection 4.2, we tested the PRPMSWP method with different parameters μ. Finally, using the parameter suggested in Subsection 4.1 and 4.2, we compared our two new methods with other variants of PRP methods and the DY-HS method.

4.1. **Numerical experiments for the MPRP method with different parameters.** From Table 4.1.1, it is obvious that when the parameters c or μ become smaller, the numbers of the iteration, function evaluation and gradient evaluation are also decreased. This phenomenon is not very hard to understand since the condition of ATLS will be easier to satisfy when c or μ become smaller. From the test, it is suggested here that c and μ should be smaller than 0.1.

TABLE 4.1.1. Tests results for MPRP methods with different parameters

Problem	Dim	c	ρ	$\mu = 0.01$ $NI/N_f/N_g$	$\mu = 0.1$ $NI/N_f/N_g$	$\mu = 1$ $NI/N_f/N_g$
ROSEX	1000	1e-8	1e-4	34/50/84	34/50/84	114/196/310
		1e-2	1e-4	34/50/84	34/50/84	114/196/310
		0.1	1e-4	34/50/84	34/50/84	109/189/298
		0.5	1e-4	48/86/134	48/86/134	117/208/325
		0.9	1e-4	63/249/312	63/249/312	99/316/415
	2000	1e-8	1e-4	34/50/84	34/50/84	114/196/310
		1e-2	1e-4	34/50/84	86/670/98	114/196/310
		0.1	1e-4	34/50/84	91/690/104	109/189/298
		0.5	1e-4	48/86/134	77/580/96	117/208/325
		0.9	1e-4	63/249/312	63/249/312	99/316/415
	5000	1e-8	1e-4	34/50/84	34/50/84	114/196/310
		1e-2	1e-4	34/50/84	34/50/84	114/196/310
		0.1	1e-4	34/50/84	34/50/84	109/189/298
		0.5	1e-4	48/86/134	48/86/134	117/208/325
		0.9	1e-4	63/249/312	63/249/312	99/316/415

4.2. Numerical experiments for the MSWP method with different parameters.
In this subsection, we will test the following four methods:

PRPMSWP1: Algorithm 2 with $\alpha = 0.01$, $\lambda = 0.1$ and $\mu = 0.01$.

PRPMSWP2: Algorithm 2 with $\alpha = 0.01$, $\lambda = 0.1$ and $\mu = 0.1$.

PRPMSWP3: Algorithm 2 with $\alpha = 0.01$, $\lambda = 0.1$ and $\mu = 1$.

From Table 4.2.1, it is obvious that when the parameter μ becomes smaller, the numbers of the iteration, function evaluation and gradient evaluation are also decreased. Moreover, the test suggests that the parameter μ in MSWP should be smaller than 0.1.

TABLE 4.2.1. Tests results for MSWP/SWP methods

Problem	Dim	$\mu = 0.01$ $NI/N_f/N_g$	$\mu = 0.1$ $NI/N_f/N_g$	$\mu = 1$ $NI/N_f/N_g$
ROSEX	1000	24/111/85	24/309/201	28/320/206
	2000	24/111/85	24/309/201	28/320/206
	5000	24/111/85	24/309/201	28/320/206

4.3. Numerical experiments for the three variants of the PRP method.
In this subsection, we will test the following six CG methods:

MPRP: Algorithm 1 with $\epsilon = 1e-8$, $\eta = 1e-10$, $\rho = 1e-4$, $\alpha = 0.1$, $c = 0.01$ and $\mu = 0.1$.

PRPSWP: the PRP formula with the SWP, where $\alpha = 0.01$ and $\lambda = 0.1$:

PRP+SWP: the PRP+ formula with the SWP, where $\alpha = 0.01$ and $\lambda = 0.1$.

PRPGL: the PRP formula with the GL line search conditions (1.14), (1.15) and (1.16), where $\rho = 1e-4$; $\delta = 0.1$; $c_2 = 0.05$; $c_1 = 150$ and $\gamma = 0.5$.

DY-HS: the DY-HS formula with the SWP, where $\alpha = 0.01$ and $\lambda = 0.1$.

PRPMSWP: the Algorithm 2 with the MSWP, where $\alpha = 0.01$, $\lambda = 0.1$ and $\mu = 0.01$.

TABLE 4.3.1. Test reults for the PRPSWP/PRP+SWP/PRPMSWP/
DY-HS/PRPGL/MPRP methods

		PRPSWP	PRP+SWP	PRPMSWP	DY-HS	PRPGL	MPRP
Problem	Dim	$NI/N_f/N_g$	$NI/N_f/N_g$	$NI/N_f/N_g$	$NI/N_f/N_g$	$NI/N_f/N_g$	$NI/N_f/N_g$
ROSEX	1000	24/111/85	25/116/91	24/111/85	24/127/101	103/568/568	34/50/84
	2000	24/111/85	25/116/91	24/111/85	24/127/101	103/568/568	34/50/84
	5000	24/111/85	25/116/91	24/111/85	24/127/101	103/568/568	34/50/84
SINGX	1000	187/636/546	108/353/305	103/376/226	29/153/124	> 5000	22/23/45
	2000	178/565/463	234/736/536	152/504/433	29/153/124	> 5000	22/23/45
	5000	178/565/463	234/736/536	152/504/433	29/153/124	> 5000	22/23/45
TRIG	1000	56/114/109	63/138/138	57/114/109	58/133/125	366/367/367	68/89/157
	2000	58/125/119	63/138/138	58/125/119	57/123/119	366/367/367	68/89/157
	5000	58/125/119	63/138/138	58/125/119	57/123/119	366/367/367	68/89/157
IE	1000	7/15/8	7/15/8	7/15/8	7/15/8	14/19/19	7/8/15
	2000	7/15/8	7/15/8	7/15/8	7/15/8	14/19/19	7/8/15
	5000	7/15/8	7/15/8	7/15/8	7/15/8	14/19/19	7/8/15
TRID	1000	33/138/83	34/76/72	33/66/62	32/136/82	116/753/753	27/28/55
	2000	34/141/84	37/86/80	34/62/57	34/141/84	116/753/753	27/28/55
	5000	36/148/88	37/86/80	35/70/65	36/142/84	116/753/753	27/28/55

Table 4.3.1 shows the numerical test of these CG methods (here "> 5000" means the iteration exceeds 5000).

In order to rank these methods, we compute the total number of function and gradient evaluations by the following formula $N_{total} = N_f + \theta \cdot N_g$ (here we calculate two extremal cases $\theta = 2$ and $\theta = 5$ respectively). Note that the PRP-SWP method has been considered to be the conjugate gradient method with the best performance in most of the literatures. Therefore, in this part, we compare the PRP+SWP, PRPGL, PRPMSWP, DY-HS, MPRP methods with the PRP-SWP method as follows: for each testing example i, compute the total numbers of function evaluations and gradient evaluations required by the evaluated method j ($EM(j)$) and the PRPSWP method by the formula (4.1) (in case the iteration exceeds 5000, we set the corresponding value of N_f and N_g by $N_f = N_g = 5000$), and denote them by $N_{total,i}(EM(j))$ and $N_{total,i}(PRPSWP)$; then calculate the ratio by the following rule:

$$r_i(EM) = \frac{N_{total,i}(EM)}{N_{total,i}(PRPSWP)}.$$

Finally, the relative efficiency of the method i is calculated by the geometric mean of these ratios over all the test problems, that is,

$$r(EM) = (\prod_{i \in S} r_i(EM))^{1/|S|},$$

where S denotes the set of the test problems and $|S|$ the number of elements in S. One advantage of the above rule is that, the comparison is relative and hence is not dominated by a few problems for which the method requires a great deal of function evaluations and gradient functions.

According to the above rule, it is clear that $r(RPRSWP) = 1$. The values of $r(PRPMSWP)$, $r(PRP + WWP)$, $r(PRPGL)$, $r(DY\text{-}HS)$ and $r(MPRP)$ are listed in Table 4.3.2. From Table 4.3.2, we can see that (1) the average performances of the MRPR method are the best among the six CG methods; (2) the PRPGL method is less efficient than the other five CG methods; (3) the average performances of PRPMSWP are slightly better than the PRPSWP and PRP+SWP method but less efficient than the DY-HS method. Notice that MPRP method not

only possesses the global convergence property but also has good numerical performance (according to our preliminary numerical test). Hence, we prefer to choose the MPRP method.

TABLE 4.3.2. Relative efficiency for the PRPSWP/PRP+SWP/PRPMSWP/ DY-HS/PRPGL/MPRP methods

θ	r(PRPSWP)	r(PRP+SWP)	r(PRPMSWP)	r(DY-HS)	r(PRPGL)	r(MPRP)
2	1.0000	0.9842	0.8020	0.5637	7.8999	0.3514
5	1.0000	0.9850	0.8050	0.5607	8.2127	0.4006

5. FINAL REMARKS

In this paper, we have carefully studied methods related to the PRP nonlinear conjugate gradient formula with some Armijo type line searches. The global convergence of the PRP method for nonconvex optimization problems is given by Grippo and Lucidi in [17] with strong line search conditions, and the PRPGL method did not perform better than the PRP with the SWP in the numerical computations. From our preliminary numerical results, one of our methods (MPRP) given in this paper not only possesses the global convergence, but also outperforms the other variants of PRP methods. Finally, from the numerical results given in this paper, we can see that the performance of the two methods (MPRP and PRPMSWP) are dependent strictly on the choices of the parameter μ. It would be more interesting to define B_k as a matrix of a certain simple structure that carries some second order information of the objection function. This might be a topic of further research.

ACKNOWLEDGMENTS

REFERENCES

[1] L. Armijo, *Minimization of functions having Lipschitz conditions partial derivatives*, Pacific Journal of Mathematics, 16 (1966), pp. 1-3. MR0191071 (32:8480)

[2] A. Al-Baali, *Descent property and global convergence of the Fletcher-Reeves method with inexact line search*, IMA J. Numer. Anal, 5 (1985), pp. 121-124 . MR777963 (86d:49043)

[3] X. Chen and J. Sun, *Global convergence of a two-parameter family of conjugate gradient methods without line search*, J. Comput. Appl. Math. 146 (2002), pp. 37-45. MR1923444

[4] Y. Dai, *Convergence of nonlinear conjugate methods*, J. Comput. Math., 19 (2001), pp. 539-549. MR1851852 (2002f:65085)

[5] Y. Dai, J. Han, G. Liu, D. Sun, H. Yin and Y. Yan, *Convergence properties of nonlinear conjugate methods*, SIAM Journal of Optimization, 2 (1999), pp. 345-358.

[6] Y. Dai, *Convergence of Polak-Ribière-Polyak conjugate gradient method with constant stepsizes*, Manuscript, Institute of Computational Mathematics abd Scientific/Engineering Computing, Chinese Academy of Sciences, 2001.

[7] Y. Dai and Q. Ni, *Testing different conjugate gradient methods for large-scale unconstrained optimization*, J. Comput. Math. 21 (2003), no. 3, pp. 311–320. MR1978635

[8] Y. Dai and Y. Yuan, *A nonlinear conjugate gradient with a strong global convergence properties*, SIAM Journal of Optimization, 10 (2000), pp. 177-182. MR1740963 (2000i:90074)

[9] Y. Dai and Y. Yuan, *Further studies on the Polak-Ribière-Polyak method*, Research Report ICM-95-040, Institute of Computational Mathematics and Scientific/ Engineering Computing, Chinese Academy of Sciences, 1995.

[10] Y. Dai and Y. Yuan, *Convergence properties of the conjugate descent method*, Advances in Mathematics, 6 (1996), pp. 552-562. MR1453164

[11] Y. Dai and Y. Yuan, *An efficient hybird conjugate gradient method for unconstrained optimization*, Annals of Operations Research, 103 (2001), 33-47. MR1868442 (2002m:90107)

[12] Y. Dai and Y. Yuan, *Nonlinear Conjugate Gradient Methods*, Science Press of Shanghai, 2000.

[13] R. Fletcher, *Practical Method of Optimization, Vol I: Unconstrained Optimization*, Second edition, Wiley, New York, 1997. MR585160 (83i:65055a)

[14] R. Fletcher and C. Reeves, *Function minimization by conjugate gradients*, Compute. J., 7 (1964), pp. 149-154. MR0187375 (32:4827)

[15] J. C. Gibert and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM Journal of Optimization, 2 (1992), pp. 21-42. MR1147881 (92k:90089)

[16] L. Grippo, F. Lampariello and S. Lucidi, *A nonmonotone line search technique for Newton's methods,* SIAM J. Numer. Anal., 23 (1986), pp. 707-716. MR849278 (87g:90105)

[17] L. Grippo and S. Lucidi, *A globally convergent version of the Polak-Ribière gradient method*, Mathematics Programming, 78 (1997), pp. 375-391. MR1466138 (98k:90080)

[18] A. Griewank, *On automatic differentiation, in: Mathematical Programming: Recent Developments and Applications*, M. Iri and K. Tanabe, eds., Kluwer Academic Publishers, (1989), pp. 84-108. MR1114312 (92k:65002)

[19] M. R. Hestenes and E. Stiefel, *Method of conjugate gradient for solving linear equations*, J. Res. Nat. Bur. Stand ., 49 (1952), pp. 409-436. MR0060307 (15:651a)

[20] D.H. Li, *A descent modified Polak "Ribiére" Polyak conjugate gradient method and its global convergence*, IMA Journal of Numerical Analysis (2006) 26, 629-640. MR2263891 (2007f:90152)

[21] Y. Liu and C. Storey, *Efficient generalized conjugate gradient algorithms, part 1: theory*, Journal of Optimization Theory and Application, 69 (1992), pp. 129-137. MR1104590 (92e:90077)

[22] G. McCormick, *A modification of Armijo's step-size rule for negative curvature*, Mathematical Programming, 13 (1977), pp. 111-115. MR0461907 (57:1889)

[23] J. J. Morè, B. S. Garbow, and K. E. Hillstrome, *Testing unconstrained optimization software*, AVM Trans. Math. Software, 7 (1981), pp. 17-41. MR607350 (83b:90144)

[24] J. Nocedal, *Theory of algorithm for unconstrained optimization*, Acta Numerica, Cambridge University Press, 1992 MR1165726 (93b:90087)

[25] J. Nocedal, *Conjugate gradient methods and nonlinear optimization*, in Linear and Nonlinear Conjugate Gradient Related Methods, L. Adams and J.L. Nazareth, ed., SIAM Philadelphia, 1995, pp. 9-23. MR1446295

[26] E. Polak and G. Ribière, *Note Sur la convergence de directions conjugèes*, Rev. Francaise Informat Recherche Operationelle, 3e Annèe, 16 (1969), pp. 35-43. MR0255025 (40:8232)

[27] B. T. Polyak, *The conjugate gradient method in extreme problems*, USSR Comp. Math. and Math. Phys., 9 (1969), pp. 94-112.

[28] M. J. D. Powell, *Nonconvex minimization calculations and the conjugate gradient method*, In Lecture Notes in Mathematics, Springer-Verlag, Berlin, 1066 (1984) pp. 122-141. MR760460 (85g:49035)

[29] H. Qi, J. Han and G. Liu, *A modification of Hestenes-Stiefel conjugate gradient method*, Chinese Annals of Mathematics, 3 (1996), pp. 177-184.

[30] J. Sun and J. Zhang, *Global convergence of conjugate gradient methods without line search*, Annals of Operations Research, 103 (2001), 161-173. MR1868449 (2003b:90140)

[31] Z. Wei, L. Qi and X. Chen, *An SQP-type method and its application in stochastic programming*, to appear in: Journal of Optimization Theory and Applications, 1 (2003). MR1961035 (2003m:90166)

[32] Z. Wei, L. Qi and S. Ito, *New step-size rules for optimization problems*, Department of Mathematics and Information Science, Guangxi University, Nanning, Guangxin, P. R. China, October, 2000.

[33] Y. Yuan and W. Sun, *Theory and Methods of Optimization*, Science Press of China, 1999.

[34] G. Zoutendijk, *Nonlinear programming computational methods*, Integer and Nonlinear Programming, Jabadie, ed., North-Holland, Amsterdam, 1970, pp. 37-86. MR0437081 (55:10015)

DEPARTMENT OF MATHEMATICS AND INFORMATION SCIENCE, GUANGXI UNIVERSITY, NANNING, GUANGXI, PEOPLE'S REPUBLIC OF CHINA

E-mail address: `zxwei@gxu.edu.cn`

DEPARTMENT OF MATHEMATICS AND INFORMATION SCIENCE, GUANGXI UNIVERSITY, NANNING, GUANGXI, PEOPLE'S REPUBLIC OF CHINA

Current address: Department of Applied Mathematics, The University of New South Wales, Australia

E-mail address: `g.li@unsw.edu.au`

DEPARTMENT OF APPLIED MATHEMATICS, THE HONG KONG POLYTECHNIC UNIVERSITY, HUNG HOM, KOWLOON, HONG KONG

E-mail address: `maqilq@polyu.edu.hk`