NEW ITERATIVE CONJUGATE GRADIENT METHOD FOR NONLINEAR UNCONSTRAINED OPTIMIZATION

Sabrina Ben Hanachi*, Badreddine Sellami and Mohammed Belloufi

Abstract. Conjugate gradient methods (CG) are an important class of methods for solving unconstrained optimization problems, especially for large-scale problems. Recently, they have been much studied. In this paper, we propose a new conjugate gradient method for unconstrained optimization. This method is a convex combination of Fletcher and Reeves (abbreviated FR), Polak–Ribiere–Polyak (abbreviated PRP) and Dai and Yuan (abbreviated DY) methods. The new conjugate gradient methods with the Wolfe line search is shown to ensure the descent property of each search direction. Some general convergence results are also established for this method. The numerical experiments are done to test the efficiency of the proposed method, which confirms its promising potentials.

1. Introduction

Optimization problem is the process of obtaining the minimization or maximization of an objective function. Unconstrained optimization is a branch of optimization in which we minimize an objective function that depends on real variables with the total absence of restrictions on their values of those variables. Consider the nonlinear following unconstrained optimization problem

$$\min\{f(x), x \in \mathbb{R}^n\}, \tag{1.1}$$

where f is a smooth function and its gradient is available [16]. Over the years the mathematicians have developed several numerical methods for solving this kind of problem (1.1), which include the Steepest Descent (SD) method; Newton method; CG and Quasi-Newton (QN) methods, In this paper, we will focus on the CG method because of its simplicity, lowest memory requirements [6], and especially its usability when the dimension is large.

In 1952, Hestenes and Stiefel [15] suggested a CG method for solving the unconstrained linear optimization problem as it is applied to quadratic functions. Then, in 1964, Fletcher and Reeves [13] extended the CG method for solving nonlinear unconstrained minimization problems. The basis of all CG methods is to generate

Laboratory Informatics and Mathematics (LiM), Mohamed Cherif Messaadia University, Souk-Ahras, Algeria.
*Corresponding author: s.benhanachi@univ-soukahras.dz

TABLE 1. Some well known conjugate gradient parameters.

No	Formula	Authors
1	$\beta_K^{\mathrm{HS}} = \frac{g_{k+1}^T y_k}{y_k^T s_k}$	Hestenes and Stiefel (HS) [15]
2	$\beta_K^{\mathrm{HZ}} = \frac{1}{d_k^T y_k}\left(y_k - 2d_k \frac{\|y_k\|^2}{d_k^T y_k}\right)^T g_{k+1}$	Hager and Zhan (HZ) [14]
3	$\beta_K^{\mathrm{CD}} = \frac{\|g_{k+1}\|^2}{-g_k^T s_k}$	Conjugate Descent (CD) [12]
4	$\beta_K^{\mathrm{LS}} = \frac{g_{k+1}^T y_k}{-g_k^T s_k}$	Liu and Storey (LS) [17]
5	$\beta_K^{\mathrm{BA}} = \frac{\|y_k\|^2}{d_k^T y_k}$	Al-Bayati and Al-Assady (BA) [2]

a sequence x_k starting from an initial estimate $x_0 \in \mathbb{R}^n$, using the following recurrence

$$x_{k+1} = x_k + s_k, \quad s_k = \alpha_k d_k, \qquad k = 0, 1, \ldots, n \tag{1.2}$$

where $\alpha_k > 0$ is a step size obtained by carrying out a one dimensional search, known as the line searches [21], we usually use the inexact one [22, 26], this is due to the fact that when we perform an exact linear search at each iteration, it is hardly feasible in practice and it is quite expensive in time and memory. Among them, the so-called strong Wolfe line search conditions require that [23, 24]

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \delta \alpha_k g_k^T d_k, \tag{1.3}$$

$$\left|g_{k+1}^T d_k\right| \leq -\sigma g_k^T d_k. \tag{1.4}$$

where scalars δ and σ satisfy $0 < \delta \leq \sigma < 1$.

And d_k is the search direction generated by the rule:

$$d_k = \begin{cases} -g_0, & \text{if } k = 0 \\ -g_k + \beta_{k-1} d_{k-1}, & \text{if } k \geq 0, \end{cases} \tag{1.5}$$

where g_k is the gradient of f at the point x_k and β_k is known as the CG parameter. The different choices for the parameter β_k correspond to different conjugate gradient methods. Moreover, we are going to summarize some most popular formulas of the conjugate gradient methods in the following Table 1.

These methods are the same if f is a strongly convex quadratic function and α_k is obtained by exact line search since the parameters β_k of these methods are equal and thus generate the same sequence $\{x_k\}_\infty^{k=0}$, but in the opposite case when applied to a general nonlinear function with inexact line searches, we get different sequences $\{x_k\}_\infty^{k=0}$, implying a range of different methods [16].

One of the most useful CG methods is the hybrid method which combines the classical CG methods [5] in order to have a good practical conjugate algorithm. Moreover, we are going to summarize some well known hybrid conjugate gradient methods in Table 2.

In this work we propose another hybrid CG method based on combination of FR, PRP and DY conjugate gradient algorithms for solving unconstrained optimization problems. The corresponding conjugate gradient parameters are

$$\beta_K^{\mathrm{FR}} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}, \tag{1.6}$$

TABLE 2. Hybrid conjugate gradient methods.

No	Formula	Authors
1	$\beta_K^c = (1-\theta_k)\beta_K^{\mathrm{HS}} + \theta_k\beta_K^{\mathrm{DY}}$	Andrei [3]
2	$\beta_K^{hyb} = (1-\theta_k)\beta_K^{\mathrm{LS}} + \theta_k\beta_K^{\mathrm{DY}}$	Liu and Li [16]
3	$\beta_K^{hyb} = (1-\theta_k)\beta_K^{\mathrm{HS}} + \theta_k\beta_K^{\mathrm{FR}}$	Djordjević [10]
4	$\beta_K^c = (1-\theta_k)\beta_K^{\mathrm{HS}} + \theta_k\beta_K^{\mathrm{CD}}$	Zheng *et al.* [27]

$$\beta_K^{\mathrm{DY}} = \frac{\|g_{k+1}\|^2}{y_k^T s_k}, \tag{1.7}$$

$$\beta_K^{\mathrm{PRP}} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}, \tag{1.8}$$

where $y_k = g_{k+1} - g_k$ and $\|.\|$ stands for the Euclidean norm. The above methods correspond to those of Fletcher and Reeves [13], Dai and Yuan [8] and Polak–Ribiere–Polyak [19, 20] respectively.

The paper is summarized as follows: in Section 2, we present the new selected hybrid CG method, and we got the parameters γ_k, δ_k using some techniques, under mild conditions we prove that the selected method with the Wolfe line search generates directions satisfying the sufficient descent condition. Section 3 presents the algorithm.Convergence properties of the new selected method are analyzed in Section 4. In Section 5, we proved the efficiency of our method by giving some numerical comparisons against FR, PRP and DY methods using 30 different test problems from the CUTE [7]. Finally, a brief conclusion is drawn in Section 6.

2. HYBRID CONJUGATE GRADIENT ALGORITHMS

In this section, we will describe a new CG method for unconstrained optimization where the parameter β_k in the proposed method, denoted by β_K^{hyb}, is computed as a convex combination of β_K^{FR}, β_K^{PRP} and β_K^{DY}, *i.e.*

$$\beta_K^{hyb} = \delta_k\beta_K^{\mathrm{FR}} + \gamma_k\beta_K^{\mathrm{PRP}} + (1-\delta_k-\gamma_k)\beta_K^{\mathrm{DY}}, \tag{2.1}$$

where $\delta_k, \gamma_k \in [0,1]$ are named the hybridization parameters that will be determined in a specific way to be described later.

The direction d_k^{hyb}, is given by

$$d_0^{hyb} = -g_0, \quad d_{k+1}^{hyb} = -g_{k+1} + \beta_K^{hyb}s_k. \tag{2.2}$$

From the discussion according to the values of the hybridization parameters δ_k and γ_k, we obtain the following cases

– If $\delta_k = 1$ and $\gamma_k = 0$, then $\beta_K^{hyb} = \beta_K^{\mathrm{FR}}$.
– If $\delta_k = 0$ and $\gamma_k = 0$, then $\beta_K^{hyb} = \beta_K^{\mathrm{DY}}$.
– If $\delta_k = 0$ and $\gamma_k = 1$, then $\beta_K^{hyb} = \beta_K^{\mathrm{PRP}}$.
– If $\delta_k = 0$ and $0 < \gamma_k < 1$, then $\beta_K^{hyb} = \gamma_k\beta_K^{\mathrm{PRP}} + (1-\gamma_k)\beta_K^{\mathrm{DY}}$ *i.e.* β_K^{hyb} is a convex combination of β_K^{PRP} and β_K^{DY}. See [25].
– If $\gamma_k = 0$ and $0 < \delta_k < 1$, then $\beta_K^{hyb} = \delta_k\beta_K^{\mathrm{FR}} + (1-\delta_k)\beta_K^{\mathrm{DY}}$ *i.e.* β_K^{hyb} is a convex combination between β_K^{FR} and β_K^{DY}. See [1].
– If $1-\delta_k-\gamma_k = 0, 0 < \delta_k, \gamma_k < 1$, then $\gamma_k = 1-\delta_k$. Then $\beta_K^{hyb} = \delta_k\beta_K^{\mathrm{FR}} + (1-\delta_k)\beta_K^{\mathrm{PRP}}$ *i.e.* β_K^{hyb} is a convex combination between β_K^{FR} and β_K^{PRP}. See [9].
– If $0 < \delta_k, \gamma_k < 1$, then we have a new hybrid CG method as a convex combination of three methods "FR, PRP and DY". Which we will focus on studying in the topic of our research.

2.1. New hybrid conjugate gradient algorithm

If $\delta_k \in]0,1[$ and $\gamma_k \in]0,1[$, then β_K^{hyb} is a convex combination among three parameters β_K^{FR}, β_K^{PRP} and β_K^{DY}.

Theorem 2.1. *If the relations* (2.1) *and* (2.2) *hold, then*

$$d_{K+1}^{hyb} = \delta_k d_{K+1}^{\mathrm{FR}} + \gamma_k d_{K+1}^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) d_{K+1}^{\mathrm{DY}}. \tag{2.3}$$

Proof. We have

$$d_{k+1}^{hyb} = -g_{k+1} + \beta_K^{hyb} s_k.$$

According to the relation (2.1), the last form becomes

$$d_{k+1}^{hyb} = -g_{k+1} + \left[\delta_k \beta_K^{\mathrm{FR}} + \gamma_k \beta_K^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) \beta_K^{\mathrm{DY}} \right] s_k. \tag{2.4}$$

Considering again the relation (2.4) and after adding and subtracting the value $(\delta_k + \gamma_k)(g_{k+1})$, we get

$$d_{k+1}^{hyb} = -(1 - \delta_k + \delta_k - \gamma_k + \gamma_k) g_{k+1} + \left[\delta_k \beta_K^{\mathrm{FR}} + \gamma_k \beta_K^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) \beta_K^{\mathrm{DY}} \right] s_k,$$
$$= \delta_k \left(-g_{k+1} + \beta_K^{\mathrm{FR}} s_k \right) + \gamma_k \left(-g_{k+1} + \beta_K^{\mathrm{PRP}} s_k \right) + (1 - \delta_k - \gamma_k) \left(-g_{k+1} + \beta_K^{\mathrm{DY}} s_k \right).$$

Finally, we get (2.3). □

Our motivation to select the parameters δ_k, γ_k in such a manner that the search direction d_{k+1} satisfies the conjugacy condition *i.e.* $(d_{k+1}^T y_k = 0)$.

We have

$$d_{k+1}^{hyb} = -g_{k+1} + \left[\delta_k \beta_K^{\mathrm{FR}} + \gamma_k \beta_K^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) \beta_K^{\mathrm{DY}} \right] s_k. \tag{2.5}$$

Having in view the relations (1.6)–(1.8), the last relation becomes

$$d_{k+1}^{hyb} = -g_{k+1} + \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2} s_k + \gamma_k \frac{g_{k+1}^T y_k}{\|g_k\|^2} s_k + (1 - \delta_k - \gamma_k) \frac{\|g_{k+1}\|^2}{y_k^T s_k} s_k, \tag{2.6}$$

multiplying (2.6) by y_k^T from the left and using the conjugacy condition, we obtain

$$y_k^T d_{k+1}^{hyb} = -y_k^T g_{k+1} + \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \left(y_k^T s_k \right) + \gamma_k \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(y_k^T s_k \right) + (1 - \delta_k - \gamma_k) \|g_{k+1}\|^2,$$

$$y_k^T g_{k+1} = \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \left(y_k^T s_k \right) + \gamma_k \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(y_k^T s_k \right) + \|g_{k+1}\|^2 + (-\delta_k - \gamma_k) \|g_{k+1}\|^2,$$

$$-g_k^T g_{k+1} \|g_k\|^2 = \delta_k \left(\|g_{k+1}\|^2 \left(y_k^T s_k \right) - \|g_{k+1}\|^2 \|g_k\|^2 \right) + \gamma_k \left(\left(g_{k+1}^T y_k \right) \left(y_k^T s_k \right) - \|g_{k+1}\|^2 \|g_k\|^2 \right)$$

$$-g_k^T g_{k+1} \|g_k\|^2 = \delta_k \left(y_k^T s_k - \|g_k\|^2 \right) \|g_{k+1}\|^2 + \gamma_k \left(\left(g_{k+1}^T y_k \right) \left(y_k^T s_k \right) - \|g_{k+1}\|^2 \|g_k\|^2 \right).$$

Finally,

$$\gamma_k = -\frac{g_k^T g_{k+1} \|g_k\|^2 + \delta_k \left(y_k^T s_k - \|g_k\|^2 \right) \|g_{k+1}\|^2}{\left(g_{k+1}^T y_k \right) \left(y_k^T s_k \right) - \|g_{k+1}\|^2 \|g_k\|^2}, \quad \text{where, } 0 < \delta_k < 1. \tag{2.7}$$

Supposing that d_k is a descent direction $(d_0 = -g_0)$, then for the algorithm given by (1.2) and (2.6) we can prove the following result.

Theorem 2.2. *Assume that α_k in algorithm (1.2) and (2.6) is determined by the Wolfe line search (1.3) and (1.4), if $0 < \delta_k, \gamma_k < 1$, and*

$$\left| \frac{\left(\|g_k\|^2 + \left(y_k^T s_k \right) \right) g_k^T s_k}{\left(y_k^T s_k \right) \|g_k\|^2} \right| \|g_{k+1}\|^2 \geq 2 \frac{\left(g_{k+1}^T y_k \right) \left(g_{k+1}^T s_k \right)}{\|g_k\|^2}. \tag{2.8}$$

Then direction d_{k+1}^{hyb} given by (2.6) is a descent direction.

Proof. We have

$$y_k^T d_k = (g_{k+1} - g_k)^T d_k = g_{k+1}^T d_k - g_k^T d_k, \tag{2.9}$$

from the strong Wolfe condition, we get

$$y_k^T d_k = g_{k+1}^T d_k - g_k^T d_k \geq \sigma g_k^T d_k - g_k^T d_k \geq -(1-\sigma)g_k^T d_k > 0. \tag{2.10}$$

Then,

$$y_k^T d_k > 0, \tag{2.11}$$

multiplying by α_k, we obtain

$$y_k^T \alpha_k d_k = y_k^T s_k > 0. \tag{2.12}$$

Since $0 < \delta_k, \gamma < 1$, $|1 - \delta_k - \gamma_k| < 1$, from (2.6) we get

$$
\begin{aligned}
g_{k+1}^T d_{k+1}^{hyb} &= -g_{k+1}^T g_{k+1} + \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \left(g_{k+1}^T s_k \right) + \gamma_k \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(g_{k+1}^T s_k \right) + (1 - \delta_k - \gamma_k) \frac{\|g_{k+1}\|^2}{y_k^T s_k} \left(g_{k+1}^T s_k \right), \\
&\leq -\|g_{k+1}\|^2 + \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \left(g_{k+1}^T s_k \right) + \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(g_{k+1}^T s_k \right) + \frac{\|g_{k+1}\|^2}{y_k^T s_k} \left(g_{k+1}^T s_k \right), \\
&\leq \left(-1 + \frac{g_{k+1}^T s_k}{\|g_k\|^2} + \frac{g_{k+1}^T s_k}{y_k^T s_k} \right) \|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(g_{k+1}^T s_k \right).
\end{aligned}
\tag{2.13}
$$

On the other hand, we have

$$-1 + \frac{g_{k+1}^T s_k}{y_k^T s_k} = \frac{-y_k^T s_k + y_k^T s_k + g_k^T s_k}{y_k^T s_k} = \frac{g_k^T s_k}{y_k^T s_k}, \tag{2.14}$$

according to the relation (2.14), the relation (2.13) becomes

$$g_{k+1}^T d_{k+1}^{hyb} \leq \left(\frac{g_k^T s_k}{y_k^T s_k} + \frac{g_{k+1}^T s_k}{\|g_k\|^2} \right) \|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{\|g_k\|^2} \left(g_{k+1}^T s_k \right). \tag{2.15}$$

Denoting $A = \frac{g_k^T s_k}{y_k^T s_k} + \frac{g_{k+1}^T s_k}{\|g_k\|^2}$, to facilitate the theoretical proof, we must simplify A as shown.

$$
\begin{aligned}
A &= \frac{g_k^T s_k}{y_k^T s_k} + \frac{g_{k+1}^T s_k}{\|g_k\|^2}, \\
&= \frac{\left(g_k^T s_k \right) \|g_k\|^2 + \left(g_{k+1}^T s_k \right) \left(y_k^T s_k \right)}{\left(y_k^T s_k \right) \|g_k\|^2} \\
&= \frac{\left(g_k^T s_k \right) \|g_k\|^2 + \left(y_k^T s_k \right)^2 + \left(g_k^T s_k \right) \left(y_k^T s_k \right)}{\left(y_k^T s_k \right) \|g_k\|^2}.
\end{aligned}
$$

Finally, we get

$$A = \frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2} + \frac{y_k^T s_k}{\|g_k\|^2}, \tag{2.16}$$

using (2.16), then we obtain

$$g_{k+1}^T d_{k+1}^{hyb} \leq \left(\frac{g_k^T s_k}{y_k^T s_k} + \frac{g_{k+1}^T s_k}{\|g_k\|^2}\right)\|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{\|g_k\|^2}\left(g_{k+1}^T s_k\right),$$

$$\leq \left(\frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2}\right)\|g_{k+1}\|^2 + \frac{y_k^T s_k}{\|g_k\|^2}\|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{\|g_k\|^2}\left(g_{k+1}^T s_k\right),$$

$$\leq \left(\frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2}\right)\|g_{k+1}\|^2 + \frac{y_k^T s_k}{\|g_k\|^2}g_{k+1}^T g_{k+1} + \frac{g_{k+1}^T y_k}{\|g_k\|^2}\left(g_{k+1}^T s_k\right),$$

$$\leq \left(\frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2}\right)\|g_{k+1}\|^2 + \frac{\left(g_{k+1}^T y_k\right)}{\|g_k\|^2}\left(g_{k+1}^T s_k\right) + \frac{\left(g_{k+1}^T y_k\right)}{\|g_k\|^2}\left(g_{k+1}^T s_k\right).$$

Finally, we get

$$g_{k+1}^T d_{k+1}^{hyb} \leq \left(\frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2}\right)\|g_{k+1}\|^2 + 2\frac{\left(g_{k+1}^T y_k\right)}{\|g_k\|^2}\left(g_{k+1}^T s_k\right). \tag{2.17}$$

Then, since $y_k^T s_k > 0$ by (2.12) and since $g_k^T s_k \leq 0$, it follows that

$$\left(\frac{\left(\|g_k\|^2 + y_k^T s_k\right)\left(g_k^T s_k\right)}{\left(y_k^T s_k\right)\|g_k\|^2}\right)\|g_{k+1}\|^2 \leq 0. \tag{2.18}$$

Therefore, from (2.8) it follows that $g_{k+1}^T d_{k+1}^{hyb} \leq 0$ *i.e.* the direction d_{k+1}^{hyb} is descent one. $\qquad\square$

Now we can prove that d_{k+1}^{hyb} satisfies the sufficient descent condition in the following theorem.

Theorem 2.3. *Suppose that* $\left(g_{k+1}^T y_k\right)\left(g_{k+1}^T s_k\right) \leq 0$, *if* $0 < \delta_k, \gamma_k < 1$. *Then the direction* d_{k+1}^{hyb} *given by* (2.2) *satisfies the sufficient descent condition*

$$g_{k+1}^T d_{k+1}^{hyb} \leq -\left(1 - (1 - \delta_k - \gamma_k)\frac{g_{k+1}^T s_k}{y_k^T s_k}\right)\|g_{k+1}\|^2. \tag{2.19}$$

Proof. For $k = 0$, then $d_0 = -g_0$, it holds $g_0^T d_0 = -\|g_0\|^2$, we can see that (2.19) holds for $k = 0$. Next we assume that (2.19) holds for some $k \geq 1$. Multiplying (2.6) by g_{k+1}^T from the left, we get

$$g_{k+1}^T d_{k+1}^{hyb} = -\|g_{k+1}\|^2 + \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2}\left(g_{k+1}^T s_k\right) + \gamma_k \frac{\left(g_{k+1}^T y_k\right)}{\|g_k\|^2}\left(g_{k+1}^T s_k\right) + (1 - \delta_k - \gamma_k)\frac{\|g_{k+1}\|^2}{y_k^T s_k}\left(g_{k+1}^T s_k\right).$$

$$\leq -\|g_{k+1}\|^2 + \delta_k \frac{\|g_{k+1}\|^2}{\|g_k\|^2}\left(g_{k+1}^T s_k\right) + (1 - \delta_k - \gamma_k)\frac{\|g_{k+1}\|^2}{y_k^T s_k}\left(g_{k+1}^T s_k\right)$$

$$\leq -\left(1 - \delta_k \frac{g_{k+1}^T s_k}{\|g_k\|^2} - (1 - \delta_k - \gamma_k)\frac{g_{k+1}^T s_k}{y_k^T s_k}\right)\|g_{k+1}\|^2.$$

Let $0 < \delta_k < 1$, so we can write $0 \geq -\delta_k \geq -1$, now we get

$$g_{k+1}^T d_{k+1}^{hyb} \leq -\left(1 - (1 - \delta_k - \gamma_k)\frac{g_{k+1}^T s_k}{y_k^T s_k}\right)\|g_{k+1}\|^2 \leq 0.$$

Observe that, since $y_k^T s_k > 0$ by (2.12) and since $g_{k+1}^T s_k = y_k^T s_k + g_k^T s_k < y_k^T s_k$, then $g_{k+1}^T s_k < y_k^T s_k$ implies $\frac{y_k^T s_k}{g_{k+1}^T s_k} > 1$. Therefore $1 - \delta_k - \gamma_k < 1$, it follows that

$$1 - \delta_k - \gamma_k < \frac{y_k^T s_k}{g_{k+1}^T s_k},$$

then

$$-(1 - \delta_k - \gamma_k) > -\frac{y_k^T s_k}{g_{k+1}^T s_k},$$

we obtain

$$\left(1 - (1 - \delta_k - \gamma_k)\frac{g_{k+1}^T s_k}{y_k^T s_k}\right) > 1 - \left(\frac{y_k^T s_k}{g_{k+1}^T s_k}\right)\left(\frac{g_{k+1}^T s_k}{y_k^T s_k}\right) = 1 - 1 = 0.$$

Therefore

$$\left(1 - (1 - \delta_k - \gamma_k)\frac{g_{k+1}^T s_k}{y_k^T s_k}\right) > 0,$$

proving the theorem. □

3. ALGORITHM OF NEW HYBRID CONJUGATE GRADIENT COEFFICIENT

Step 1. *Initialization.* Select $x_0 \in \mathbb{R}^n$ and the parameters $0 < \delta \leq \sigma < 1$. Let $k = 0$. Compute $f(x_0)$, $g_0 = \nabla f(x_0)$. Consider $d_0 = -g_0$, set the initial guess: $\alpha_0 = 0$ and $\delta_k = 0.8$.

Step 2. *Test a criterion for stopping iterations,* if $\frac{\|g_k\|}{\|g_0\|} < 10^{-6}$, then stop. Else continue with Step 3.

Step 3. *Line search.* Compute $\alpha_k > 0$ by the strong Wolfe line search, *i.e.* α_k satisfies (1.3), (1.4).

Step 4. *Generate.* $x_{k+1} = x_k + \alpha_k d_k$. Compute $f(x_{k+1})$, $g_{k+1} = \nabla f(x_{k+1})$ and $s_k = x_{k+1} - x_k$, $y_k = g_{k+1} - g_k$.

Step 5. *Compute γ_k as in equation (2.7).*

Step 6. Calculate β_k^{hyb} by equation (2.1).

Step 7. *Computation of the search direction.* Compute $d = -g_{k+1} + \beta_k^{hyb} s_k$. If the restart criterion of Powell

$$\left|g_{k+1}^T g_k\right| \geq 0.2\|g_{k+1}\|^2,$$

is satisfied, then $d_{k+1} = -g_{k+1}$. Otherwise define $d_{k+1} = d$.

Step 8. Put $k = k + 1$ and continue with Step 2.

4. CONVERGENCE PROPERTIES

In this section, we study the global convergence properties of the proposed conjugate gradient method. For further considerations we need the following assumptions and lemmas.

Assumption 4.1. *The level set $S = \{x \in \mathbb{R}^n \ f(x) \leq f(x_0)\}$ is bounded, i.e. there exists a constant $B > 0$, such that*

$$\|x\| \leq B, \text{ for all } x \in S. \tag{4.1}$$

Assumption 4.2. *In a neighborhood $\mathcal{N}$ of S the function f is continuously differentiable and its gradient $\nabla f(x)$ is Lipschitz continuous, i.e. there exists a constant $0 < L < \infty$ such that*

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \text{for all} \quad x, y \in \mathcal{N}. \tag{4.2}$$

Under Assumptions 4.1 and 4.2 on f, there exists a constant $\Gamma \geq 0$, such that

$$\|\nabla f(x)\| \leq \Gamma, \tag{4.3}$$

for all $x \in S$.

Lemma 4.3. *Let Assumptions 4.1 and 4.2 hold. Consider the method (1.2), (1.5), where d_k is a descent direction and α_k obtained by the strong Wolfe line search, if*

$$\sum_{k \geq 1} \frac{1}{\|d_k\|^2} = \infty. \tag{4.4}$$

Then

$$\lim_{k \to \infty} \inf \|g_k\| = 0. \tag{4.5}$$

Lemma 4.4 ([16])**.** *Suppose that assumptions (2.1) and (2.2) holds, if d_k is a descent direction and α_k satisfies*

$$g_{k+1}^T d_k \geq \sigma g_k^T d_k, \quad \sigma < 1. \tag{4.6}$$

Then,

$$\alpha_k \geq \frac{(1 - \sigma)}{L} \frac{|d_k^T g_k|}{\|d_k\|^2}. \tag{4.7}$$

Proof. It follows (4.6), the Lipschitz condition, the Cauchy–Bunyakovsky–Schwartz inequality, it holds that

$$-(1 - \sigma)d_k^T g_k \leq \sigma d_k^T g_k - d_k^T g_k \leq d_k^T(g_{k+1} - g_k) \leq d_k^T L\alpha_k d_k \leq L\alpha_k \|d_k\|^2. \tag{4.8}$$

Hence, the assertion (4.7) holds.   □

According to the Lemma 4.4, assumptions (2.1) and (2.2), the strong Wolfe condition and (2.19), we conclude that α_k which obtained in our new method is not equal to zero *i.e.* there exists a constant $\lambda > 0$ such that

$$\alpha_k \geq \lambda, \quad \text{for all } k \geq 0. \tag{4.9}$$

Now, by Lemma 4.3, we can prove the following convergence result.

Theorem 4.5. *Consider the iterative method, defined by algorithm (3) and let all conditions of Theorem 2.3 hold then either $g_k = 0$, for some k, or*

$$\lim_{k \to \infty} \inf \|g_k\| = 0. \tag{4.10}$$

Proof. Suppose that $g_k \neq 0$ for all k. Then, we are going to prove (4.10).

Suppose by contradiction that (4.10) is false. Then there exists $r > 0$ such that

$$\|g_k\| \geq r, \quad \forall k \text{ sufficiently large}. \tag{4.11}$$

From the above Theorem 2.3, we have

$$g_k^T d_k \leq -K\|g_k\|^2, \quad \text{for all} \quad K. \tag{4.12}$$

From the strong Wolfe conditions, we get

$$y_k^T d_k = g_{k+1}^T d_k - g_k^T d_k \geq \delta g_k^T d_k - g_k^T d_k \geq -(1-\delta)g_k^T d_k \geq K(1-\delta)\|g_k\|^2, \tag{4.13}$$

multiplying (4.13) by $\alpha_k > 0$ and using (4.9), (4.12), we obtain

$$y_k^T s_k \geq K(1-\delta)\alpha_k\|g_k\|^2 \geq K(1-\delta)\lambda r^2.$$

On the other side, we have

$$\|y_k\| = \|g_{k+1} - g_k\| \leq L\|x_{k+1} - x_k\| \leq LD,$$

where D is the diameter of the level set S.

We have

$$d_{k+1}^{hyb} = -g_{k+1} + \beta_K^{hyb} s_k.$$
$$\left\|d_{k+1}^{hyb}\right\| \leq \|g_{k+1}\| + \left|\beta_K^{hyb}\right|\|s_k\|. \tag{4.14}$$

While

$$\beta_K^{hyb} = \delta_k \beta_K^{\mathrm{FR}} + \gamma_k \beta_K^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k)\beta_K^{\mathrm{DY}}.$$
$$\left|\beta_K^{hyb}\right| \leq |\beta_K^{\mathrm{FR}}| + |\beta_K^{\mathrm{PRP}}| + |\beta_K^{\mathrm{DY}}|.$$

Further

$$|\beta_K^{\mathrm{FR}}| = \frac{\|g_{k+1}\|^2}{\|g_k\|^2} \leq \frac{\Gamma^2}{r^2}. \tag{4.15}$$

Also

$$|\beta_K^{\mathrm{PRP}}| = \left|\frac{g_{k+1}^T y_k}{g_k^T g_k}\right| \leq \frac{\|g_{k+1}\|\,\|y_k\|}{\|g_k\|^2} \leq \frac{\Gamma LD}{r^2}.$$

Finally, we get

$$|\beta_K^{\mathrm{PRP}}| \leq \frac{\Gamma LD}{r^2}. \tag{4.16}$$

Also

$$|\beta_K^{\mathrm{DY}}| = \frac{\|g_{k+1}\|^2}{|y_k^T s_k|} \leq \frac{\Gamma^2}{K(1-\delta)\lambda r^2}. \tag{4.17}$$

From (4.15) to (4.17), we can write

$$\left|\beta_K^{hyb}\right| \leq \frac{\Gamma^2}{r^2} + \frac{\Gamma LD}{r^2} + \frac{\Gamma^2}{K(1-\delta)\lambda r^2} = M. \tag{4.18}$$

According to the relation (4.18), the relation (4.14) becomes

$$\left\|d_{k+1}^{hyb}\right\| \leq \|g_{k+1}\| + M\|s_k\| \leq \Gamma + MD$$

wherefrom

$$\sum_{k \geq 1} \frac{1}{\|d_k\|^2} = \infty. \tag{4.19}$$

So, applying Lemma 4.3, we conclude that

$$\lim_{k \to \infty} \inf\|g_k\| = 0. \tag{4.20}$$

This is a contradiction with (4.11), so we have finished the proof of (4.10). $\qquad\square$

5. Numerical experiments

In this section we present the computational performance of a Fortran implementation of the new algorithm on a set of 450 unconstrained optimization test problems. The test problems are the unconstrained problems in the CUTE [7] library, along with other large-scale optimization problems presented in [4]. We selected 30 large-scale unconstrained optimization problems in extended or generalized form. For each function we have considered numerical experiments with the increasing number of variables $n = 2, 4, 10, \ldots, 25\,000$. In order to assess the reliability of our new proposed method, we have tested them against FR, PRP and DY algorithms using the same test problems. All these algorithms implement the Wolfe line search conditions with $\delta = 10^{-3}$ and $\sigma = 10^{-4}$, the same stopping criterion $\frac{\|g_k\|_\infty}{\|g_0\|_\infty} < 10^{-6}$, where $\|.\|_\infty$ is the maximum absolute component of a vector, and the hybridization parameter $\delta_k = 0.8$. The comparisons of algorithms are given in the following context. Let $f_i^{\mathrm{ALG1}}(x^*)$ and $f_i^{\mathrm{ALG2}}(x^*)$ be the optimal value found by ALG1 and ALG2, for problem $i = 1, \ldots, 450$, respectively. We say that, in the particular problem i, the performance of ALG1 was better than the performance of ALG2 if

$$\frac{\left|f_i^{\mathrm{ALG1}}(x^*) - f_i^{\mathrm{ALG2}}(x^*)\right|}{\left|f_i^{\mathrm{ALG1}}(x_0) - f_i^{\mathrm{ALG2}}(x_0)\right|} < 10^{-3}, \tag{5.1}$$

and the number of iterations, or the number function-gradient evaluations, or the CPU time of ALG1 was less than the number of iterations, or the number of function-gradient evaluations, or the CPU time corresponding to ALG2, respectively.

All codes are written in Matlab and compiler settings on the PC machine with Intel(R) Core(TM) i3-4030U CPU @ 1.90 GHz processor and 4 GB RAM memory and windows 7 professional system.

Figures 1 and 2 show the performance of these methods relative to the number of iterations (iter) and CPU time (time), which were evaluated using the profiles of Dolan and Moré [11]. Benchmark results are generated by running a solver on a set P of problems and recording information of interest the number of iterations and CPU time and using parallel processing running a different CG method in each processor choosing in every step the result giving the least value of the function [18]. Let S be the set of solvers in comparison. Assume that S consists of n_s solvers, P consists of n_p problems. For each problem $p \in P$ and solver $s \in S$, denote $t_{p,s}$ be the computing time (or the number of iterations) required to solve problem $p \in P$ by solver $s \in S$, and the comparison between different solvers is based on the performance ratio defined by

$$r_{p,s} = \frac{t_{p,s}}{\min\{t_{p,s} : s \in S\}}.$$

Assume that a large enough parameter $r_M \geq r_{p,s}$ for all p, s is chosen, and $r_{p,s} = r_M$ if and only if solvers s does not solver problem p. Define

$$\rho_s(\tau) = \frac{1}{n_p}\mathrm{size}\{p \in P : \log r_{p,s} \leq \tau\},$$

where size A means the number of elements in set A, then $\rho_s(\tau)$ is the probability for solver $s \in S$ that a performance ratio $r_{p,s}$ is within a factor $\tau \in \mathbb{R}^n$. The ρ_s is the (cumulative) distribution function for the performance ratio. The value of $\rho_s(1)$ is the probability that the solver will win over the rest of the solvers.

That is, for each method, we plot the fraction P of problems for which the method is within a factor of the best time. The left side of the figure gives the percentage of the test problems for which a method is the fastest, the right side gives the percentage of the test problems that are successfully solved by each of the methods. The top curve is the method that solved the most problems in a time that was within a factor of the best time.

From the two figures, we can see that the new method is superior to the other conjugate gradient methods on the testing problems.
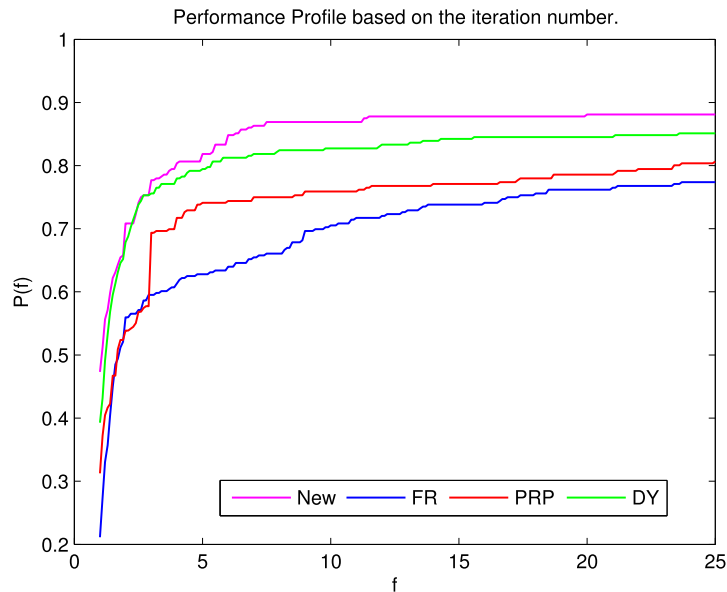
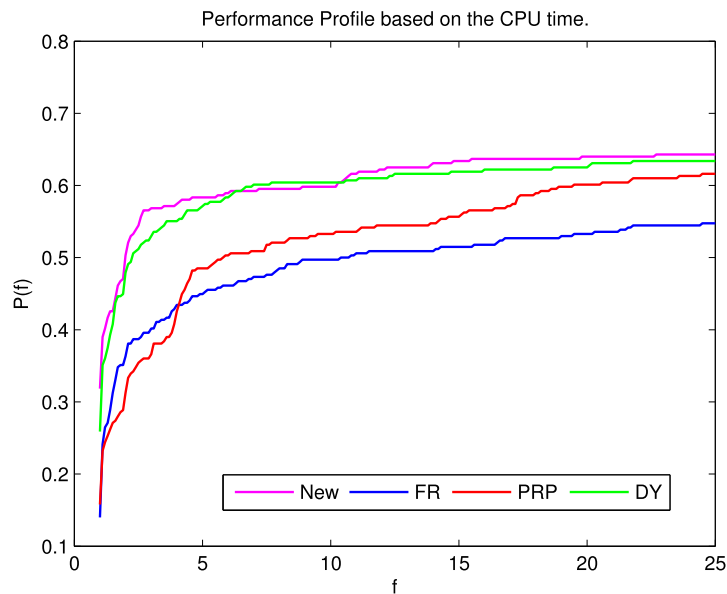FIGURE 1. Performance Profile based on the iteration number.

FIGURE 2. Performance Profile based on the CPU time.

6. CONCLUSION

There are many conjugate gradient methods for solving unconstrained problems, especially large scale ones. One of the most useful CG methods is the hybrid method, which combines the classical CG methods in order to create a new method that performs well. In this paper, we have proposed a new hybrid method where the parameter β_k is computed as a convex combination of three parameters β_k^{FR}, β_k^{PRP} and β_k^{DY}, the sufficient

descent and global convergence have been proved and the practical results shown that the selected method is faster and more efficient compared to the methods used.

References

[1] A. Abubakar, P. Kumam, M. Malik, P. Chaipunya and A. Ibrahim, A hybrid FR-DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection. *AIMS Math.* **66** (2021) 126.

[2] M. Al-Baali, Descent property and global convergence of the Fletcher-Reeves method with inexact line search. *IMA J. Numer. Anal.* **5** (1985) 121–124.

[3] N. Andrei, Another nonlinear conjugate gradient algorithm for unconstrained optimization. *Optim. Methods Softw.* **24** (2008) 89–104.

[4] N. Andrei, An unconstrained optimization test functions collection. *Adv. Model. Optim.* **10** (2008) 147–161.

[5] N. Andrei, Accelerated hybrid conjugate gradient algorithm with modified secant condition for unconstrained optimization. *Numer. Algorithms* **54** (2010) 23–46.

[6] B. Balaram, M. Narayanan and P. Rajendrakumar, Optimal design of multi-parametric nonlinear systems using a parametric continuation based genetic algorithm approach. *Nonlinear Dyn.* **67** (2012) 2759–2777.

[7] I. Bongartz, A.R. Conn, N. Gould and P.L. Toint, CUTE: constrained and unconstrained testing environments. *ACM Trans. Math. Softw. (TOMS).* **21** (1995) 123–160.

[8] Y.H. Dai and Y. Yuan, A nonlinear conjugate gradient method with a strong global convergence property. *SIAM J. Optim.* **10** (1999) 177–182.

[9] S.S. Djordjević, New hybrid conjugate gradient method as a convex combination of FR and PRP methods. *Filomat* **30** (2016) 3083–3100.

[10] S.S. Djordjević, New hybrid conjugate gradient method as a convex combination of LS and FR conjugate gradient methods. *Acta Math. Sci.* **39** (2019) 214–228.

[11] E.D. Dolan and J.J. Moré, Benchmarking optimization software with performance profiles. *Math. Program.* **91** (2002) 201–213.

[12] R. Fletcher, Practical Methods of Optimization, Unconstrained Optimization. Vol. 1. John Wiley and Son, New York (1980).

[13] R. Fletcher and C. Reeves, Function minimization by conjugate gradients. *Comput. J.* **7** (1964) 149–154.

[14] W.W. Hager and H. Zhang, A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J. Optim.* **16** (2005) 170–192.

[15] M.R. Hestenes and E.L. Stiefel, Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Standards* **49** (1952) 409–436.

[16] J.K. Liu and S.J. Li, New hybrid conjugate gradient method for unconstrained optimization. *Appl. Math. Comput.* **245** (2014) 36–43.

[17] D. Liu and C. Story, Efficient generalized conjugate gradient algorithms, Part 1: theory. *J. Optim. Theory App.* **69** (1991) 129–137.

[18] P.K.-H. Phua, W. Fan and Y. Zeng, Parallel algorithms for large-scale nonlinear optimization. *Int. Trans. Oper. Res.* **5** (1998) 67–77.

[19] E. Polak and G. Ribiere, Note sur la convergence de méthodes de directions conjuguées. *Revue Française D'informatique et De Recherche Opérationnelle, Série Rouge* **3** (1969) 35–43.

[20] B.T. Polyak, The conjugate gradient method in extremal problems. *USSR Comput. Math. Math. Phys.* **9** (1969) 94–112.

[21] Z. Salleh and A. Alhawarat, An efficient modification of the Hestenes-Stiefel nonlinear conjugate gradient method with restart property. *J. Inequalities App.* **2016** (2016) 1–14.

[22] Z. Wei, G. Li and L. Qi, New nonlinear conjugate gradient formulas for large-scale unconstrained optimization problems. *Appl. Math. Comput.* **179** (2006) 407–430.

[23] P. Wolfe, Convergence conditions for ascent methods. *SIAM Rev.* **11** (1969) 226–235.

[24] P. Wolfe, Convergence conditions for ascent methods. II: some corrections. *SIAM Rev.* **13** (1971) 185–188.

[25] H. Yan, L. Chen and B. Jiao, HS-LS-CD hybrid conjugate gradient algorithm for unconstrained optimization. In: Second International Workshop on Computer Science and Engineering. Vol. 1. IEEE (2009) 264–268.

[26] G. Yuan, Z. Wei and Q. Zhao, A modified Polak–Ribiere–Polyak conjugate gradient algorithm for large scale optimization problems. *IIE Trans.* **46** (2014) 397–413.

[27] X.Y. Zheng, X.L. Dong, J.R. Shi and W. Yang, Further comment on another hybrid conjugate gradient algorithm for unconstrained optimization by Andrei. *Numer. Algorithm* **84** (2019) 603–608.