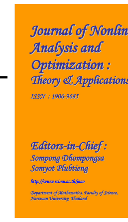

Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 5, No. 2, (2014), 127-137
ISSN : 1906-9605
http://www.math.sci.nu.ac.th

ANOTHER HYBRID CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION PROBLEMS

M. KOONTSE AND P. KAELO*

University of Botswana, Department of Mathematics, Private Bag UB00704, Gaborone, Botswana

ABSTRACT. Conjugate gradient method is one of the most useful method for solving large scale unconstrained optimization problems. In this article a new hybrid conjugate gradient method that satisfies the descent condition independently of the line searches is proposed. In particular, it is a hybrid of the Fletcher-Reeves (β_k^{FR}) and Polak-Ribiere-Polyak (β_k^{PRP}) methods. Convergence analysis of the new method is presented. Numerical results of the method show that the proposed hybrid algorithm is just as competitive.

KEYWORDS : hybrid Conjugate Gradient, line search, Convergence analysis.
AMS Subject Classification: 90C30, 90C06, 65K05

1. INTRODUCTION

Conjugate gradient methods (CG) are very useful in finding the optimal solution to the unconstrained optimization problem

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

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function and is continuously differentiable. Conjugate gradient methods are the most preferred methods for solving large scale unconstrained problems because, unlike Newton and Quasi-Newton methods [4, 13, 23], they only need the first derivatives and hence less storage capacity is needed. They are also relatively simple to program.

Given an initial guess $x_0 \in \mathbb{R}^n$, the CG method generates a sequence $\{x_k\}$ for problem (1.1) as

$$x_{k+1} = x_k + \alpha_k d_k, \qquad k = 0, 1, 2, \ldots, \tag{1.2}$$

where α_k is a step length which is determined by a line search and d_k is a descent direction of f at x_k. The step length α_k is obtained by carrying out an exact or

* *Corresponding author.*
Email address : modunco@yahoo.com (M. Koontse), kaelop@mopipi.ub.bw (P. Kaelo).
Article history : Received November 12, 2013 Accepted January 28, 2015.

inexact one dimensional line search. If exact line search is used, then α_k is such that

$$f(x_k + \alpha_k d_k) = \min_{\alpha \geq 0} f(x_k + \alpha d_k). \tag{1.3}$$

As for inexact line searches, we have the Amirjo condition [4, 23], which requires α_k to satisfy

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \mu \alpha_k \nabla f(x_k)^T d_k, \tag{1.4}$$

and the standard Wolfe conditions [4, 23], which require α_k to satisfy (1.4) and the curvature condition

$$\nabla f(x_k + \alpha_k d_k)^T d_k \geq \sigma \nabla f(x_k)^T d_k, \tag{1.5}$$

where $0 < \mu < \sigma < 1$. Strong Wolfe conditions have also been used in a number of papers and are given by (1.4) and

$$|\nabla f(x_k + \alpha_k d_k)^T d_k| \leq -\sigma \nabla f(x_k)^T d_k, \tag{1.6}$$

again with $0 < \mu < \sigma < 1$. The search direction d_k for CG methods is generated as

$$d_k = \begin{cases} -g_k, & \text{if} \quad k = 0 \\ -g_k + \beta_k d_{k-1}, & \text{if} \quad k \geq 1, \end{cases} \tag{1.7}$$

where $g_k = \nabla f(x_k)$ is the gradient of f at x_k and β_k is a scalar, known as the conjugate gradient coefficient. Different choices of the conjugate gradient coefficient β_k lead to different CG methods. Some of the well-known CG methods include the Hestenes-Stiefel (β_k^{HS}) method [3, 8, 18], the Polak-Ribière-Polyak (β_k^{PRP}) method [11, 15, 16, 23, 24, 26], the Fletcher-Reeves (β_k^{FR}) method [11, 13, 14, 23, 25, 28], the Liu-Storey (β_k^{LS}) method [3, 19], the conjugate descent (β_k^{CD}) method [3, 13] and the Dai-Yuan (β_k^{DY}) method[6, 8, 10, 11]. The conjugate gradient coefficient β_k for these mentioned CG methods are, respectively,

$$\beta_k^{HS} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})}, \tag{1.8}$$

$$\beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{||g_{k-1}||^2}, \tag{1.9}$$

$$\beta_k^{FR} = \frac{g_k^T g_k}{||g_{k-1}||^2}, \tag{1.10}$$

$$\beta_k^{LS} = -\frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T g_{k-1}}, \tag{1.11}$$

$$\beta_k^{CD} = -\frac{g_k^T g_k}{d_{k-1}^T g_{k-1}}, \tag{1.12}$$

$$\beta_k^{DY} = \frac{g_k^T g_k}{d_{k-1}^T g_{k-1}}, \tag{1.13}$$

where $|| \cdot ||$ represent the norm of vectors.

It has been shown in the literature that although the above formulae are equivalent for the quadratic functions, their performance strongly depends on the coefficient β_k. The CG methods β_k^{FR}, β_k^{CD} and β_k^{DY} possess strong global convergence properties [1, 5, 8, 7, 17, 23], but have less computational performance. On the other hand, the β_k^{PRP}, β_k^{HS} and β_k^{LS} methods have been shown that although they may not always converge, they often offer better computational performance [5, 15, 16, 17, 23].

In this paper, we suggest another approach to get a hybrid conjugate gradient method that combines the strengths of β_k^{PRP} and β_k^{FR} methods. This proposed method is presented in section 2. In Section 3 we present the convergence analysis of the new algorithm. Section 4 presents some numerical experiments and conclusion is given in Section 5.

2. New algorithm

In this section a hybrid of the β_k^{PRP} and β_k^{FR} methods is presented. As already mentioned, β_k^{FR} method has an attractive property as far as convergence is concerned. Its strength, that is, the global convergence property usually happens under strong Wolfe conditions. On the other hand, the β_k^{PRP} method has good computational properties and often performs better compared to other conjugate gradient methods. This method has been proved that when the function is strongly convex and the line search is exact, then the method is globally convergent. However, for general nonlinear functions, the convergence of the β_k^{PRP} method is uncertain. It appeared, after several failed attempts to prove global convergence of the β_k^{PRP} algorithm, that positiveness of β_k is crucial as far as convergence is concerned. This lead Gilbert and Nocedal [15] to modify β_k^{PRP} method as

$$\beta_k^{PRP+} = \max\{0, \beta_k^{PRP}\}$$

and proved that it is globally convergent with the standard Wolfe conditions.

There are a number of other β_k^{PRP} and β_k^{FR} hybrid conjugate gradient methods that have been proposed in the literature. One of the first hybrid conjugate gradient method of this form was introduced by Touati-Ahmed and Storey [27] where the parameter β_k is computed as

$$\beta_k^{TS} = \max\{0, \min\{\beta_k^{PRP}, \beta_k^{FR}\}\}.$$

This motivated other researchers to come up with more and improved β_k hybrids involving β_k^{PRP} and β_k^{FR}. For instance, Mo, Gu and Wei [21] proposed a β_k method which is a modification of the hybrid method proposed by Touati-Ahmed and Storey [27]. Their hybrid method computes β_k as

$$\beta_k^{MGW} = \max\{0, \min\{\beta_k^{PRP}, \beta_k^{FR}, \beta_k^{+}\}\},$$

where

$$\beta_k^{+} = \beta_k^{PRP} + \frac{2g_k^T g_{k-1}}{\| g_{k-1} \|^2}.$$

They proved that their hybrid method is globally convergent when the step size satisfies the strong Wolfe conditions. Another hybrid was that of Gilbert and Nocedal [15] who suggested a combination between the β_k^{PRP} and β_k^{FR} method as

$$\beta_k^{GN} = \max\{-\beta_k^{FR}, \min\{\beta_k^{PRP}, \beta_k^{FR}\}\}, \tag{2.1}$$

which is an extension of the Touati-Ahmed and Storey [27] method.

In this article, we propose yet another hybrid of β_k^{PRP} and β_k^{FR} that is based on the ideas of Gilbert and Nocedal [15], where their β_k given by (2.1), and those of Dai and Yuan [8], where they suggested the hybrid method

$$\beta_k^{HS-DY} = \max\{-c\beta_k^{DY}, \min\{\beta_k^{HS}, \beta_k^{DY}\}\}, \tag{2.2}$$

where $c = \frac{1-\gamma}{1+\gamma} > 0$. In particular, we propose a hybrid method which computes the parameter β_k as

$$\beta_k^{*} = \max\{\min\{-c\beta_k^{PRP}, \beta_k^{FR}\}, \min\{\beta_k^{FR}, \beta_k^{PRP}\}\}, \tag{2.3}$$

with $c = \frac{1-\gamma}{1+\gamma}$, $\gamma \in [\frac{1}{2}, 1]$ and the direction d_k defined as

$$d_k = \begin{cases} -g_k & k = 0 \\ -\theta_k g_k + \beta_k^* d_{k-1} & k \geq 1 \end{cases} \tag{2.4}$$

where $\theta_k = 1 + \beta_k^* \frac{d_{k-1}^T g_k}{\|g_k\|^2}$. The parameter θ_k, as defined, makes the direction d_k satisfy the descent condition independently of any line search. Also, from the above definition of β_k^* and the range of γ, we see that $0 < \beta_k^* \leq \beta_k^{FR}$ for all k. Now, with β_k^* and d_k defined as above, we present our new β_k^* algorithm.

Algorithm 2.1. The New β_k^* algorithm

Step 1 Given $x_0 \in \mathbb{R}^n$ and the parameters $\epsilon > 0$, $0 < \mu < \sigma < 1$, $\gamma \in [\frac{1}{2}, 1]$
set $k = 0$
compute $f(x_0)$ and $g_0 = \nabla f(x_0)$
set $d_0 = -g_0$,
if $\| g_0 \| \leq \epsilon$ then stop.

Step 2 Compute $\alpha_k > 0$ using any line search and find the next iterate

$$x_{k+1} = x_k + \alpha_k d_k.$$

compute $f(x_{k+1})$, $g_{k+1} = \nabla f(x_{k+1})$
if $||g_{k+1}|| \leq \epsilon$ then stop.

Step 3 compute β_k^* from (2.3) and generate d_k from (2.4)

Step 4 let $k = k + 1$ and go to step 2.

3. Convergence analysis

To establish the convergence of our method, we make the following basic assumptions on the objective function which have been widely used in the literature to analyze the global convergence of conjugate gradient methods.

Assumptions

(i) f is bounded below on the level set $S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$, where x_0 is the starting point.

(ii) In some neighborhood N of S the function f is continuously differentiable and its gradient, $g(x) = \nabla f(x)$, is Lipschitz continuous, i.e. there exist a constant $L > 0$ such that $\| g(x) - g(y) \| \leq L \| x - y \|$ for all $x, y \in N$

Under Assumptions (i) and (ii) on f, we have the following lemma.

Lemma 3.1. *(**Zoutendijk**). Suppose that Assumptions (i) and (ii) hold. Consider a CG method in the form $x_{k+1} = x_k + \alpha_k d_k$ and (1.7), where d_k is a descent direction and the step length α_k satisfies the standard Wolfe conditions (1.4) and (1.5). Then we have that*

$$\sum_{k=0}^{\infty} \frac{(\nabla f(x_k)^T d_k)^2}{\| d_k \|^2} < +\infty. \tag{3.1}$$

Proof. From the Lipschitz continuity and (1.5) we have that

$$(\sigma - 1) d_k^T \nabla f(x_k) \leq d_k^T (\nabla f(x_k + \alpha_k d_k) - \nabla f(x_k)) \tag{3.2}$$

$$\leq \| \nabla f(x_k + \alpha_k d_k) - \nabla f(x_k) \| \| d_k \| \tag{3.3}$$

$$= L\alpha_k \| d_k \|^2 \tag{3.4}$$

Thus,

$$\alpha_k \geq \frac{(\sigma - 1) d_k^T \nabla f(x_k)}{L \| d_k \|^2}. \tag{3.5}$$

It follows from (1.4) that

$$f(x_k) - f(x_k + \alpha_k d_k) \geq -\mu \left(\frac{(\sigma - 1) d_k^T \nabla f(x_k)}{L \parallel d_k \parallel^2} \right) \nabla f(x_k)^T d_k, \tag{3.6}$$

which implies

$$f(x_k) - f(x_k + \alpha_k d_k) \geq C_1 \frac{(\nabla f(x_k)^T d_k)^2}{\parallel d_k \parallel^2}, \tag{3.7}$$

where $C_1 = \frac{\mu(1-\sigma)}{L} > 0$. Now,

$$f(x_0) - f(x_0 + \alpha d_0) \geq C_1 \frac{(\nabla f(x_0)^T d_0)^2}{\parallel d_0 \parallel^2} \tag{3.8}$$

$$f(x_1) - f(x_2) \geq C_1 \frac{(\nabla f(x_1)^T d_1)^2}{\parallel d_1 \parallel^2} \tag{3.9}$$

$$f(x_2) - f(x_3) \geq C_1 \frac{(\nabla f(x_2)^T d_2)^2}{\parallel d_2 \parallel^2} \tag{3.10}$$

$$\vdots \tag{3.11}$$

$$\tag{3.12}$$

$$f(x_{k-1}) - f(x_k) \geq C_1 \frac{(\nabla f(x_{k-1})^T d_{k-1})^2}{\parallel d_{k-1} \parallel^2} \tag{3.13}$$

Adding up we get

$$f(x_0) - f(x_k) \geq C_1 \sum_{i=0}^{k-1} \frac{(\nabla f(x_i)^T d_i)^2}{\parallel d_i \parallel^2} \tag{3.14}$$

Noting that f is bounded from below as $k \to \infty$, we have

$$f(x_0) - f^* \geq C_1 \sum_{k=0}^{\infty} \frac{(\nabla f(x_k)^T d_k)^2}{\parallel d_k \parallel^2}, \tag{3.15}$$

where

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

Hence

$$\sum_{k=0}^{\infty} \frac{(\nabla f(x_k)^T d_k)^2}{\parallel d_k \parallel^2} < +\infty. \tag{3.16}$$

□

Lemma 3.2. *Let $x_{k+1} = x_k + \alpha_k d_k$ be given by Algorithm (2.1). Then the direction d_k given by (2.4) satisfies the descent condition*

$$d_k^T g_k = - \parallel g_k \parallel^2, \qquad \forall k \geq 0. \tag{3.17}$$

Proof. Let $\beta_k = \beta_k^*$. For $d_0 = -g_0$, we have

$$g_0^T d_0 = -g_0^T g_0 \tag{3.18}$$

$$= - \parallel g_0 \parallel^2 . \tag{3.19}$$

Therefore the result holds for $k = 0$.
For $k \geq 1$, we have that

$$d_k = -\theta_k g_k + \beta_k^* d_{k-1}. \tag{3.20}$$

Now, for $\beta_k = \beta_k^*$, we have

$$d_k = -(1 + \beta_k^* \frac{g_k^T d_{k-1}}{\| g_k \|^2}) g_k + \beta_k^* d_{k-1} \tag{3.21}$$

$$= \beta_k^* d_{k-1} - (1 + \beta_k^* \frac{g_k^T d_{k-1}}{\| g_k \|^2}) g_k. \tag{3.22}$$

Multiplying both sides by g_k^T we get

$$g_k^T d_k = \beta_k^* g_k^T d_{k-1} - (1 + \beta_k^* \frac{g_k^T d_{k-1}}{\| g_k \|^2}) g_k^T g_k \tag{3.23}$$

$$= \beta_k^* g_k^T d_{k-1} - \| g_k \|^2 - \beta_k^* \frac{g_k^T d_{k-1}}{\| g_k \|^2} \| g_k \|^2 \tag{3.24}$$

$$\Rightarrow \; g_k^T d_k = - \| g_k \|^2 . \tag{3.25}$$

Thus (3.17) holds for all $k \geq 1$, which concludes the proof. □

Theorem 3.3. *Suppose that Assumptions (i) and (ii) hold. Consider the conjugate gradient method of the form $x_{k+1} = x_k + \alpha_k d_k$ and d_k is given by (2.4) with α_k satisfying any line search. Then either $g_k = 0$ for some k or*

$$\lim_{k \to \infty} \inf \| g_k \| = 0. \tag{3.26}$$

Proof. If $g_k = 0$ then the statement holds. Suppose that (3.26) is not true, then there exist a constant $\varepsilon > 0$ such that

$$\| g_k \| \geq \varepsilon \; \forall k. \tag{3.27}$$

From (2.4), we have

$$d_k + \theta_k g_k = \beta_k^* d_{k-1} \tag{3.28}$$

By squaring both sides of (3.28) and applying the descent condition (3.17), we get

$$\| d_k \|^2 = (\beta_k^*)^2 \| d_{k-1} \|^2 - 2\theta_k d_k^T g_k - \theta_k^2 \| g_k \|^2 .$$

Dividing both sides by $(g_k^T d_k)^2$, and noting that $g_k^T d_k = - \| g_k \|^2$, we have

$$\frac{\| d_k \|^2}{(g_k^T d_k)^2} = (\beta_k^*)^2 \frac{\| d_{k-1} \|^2}{(g_k^T d_k)^2} + \frac{2\theta_k}{\| g_k \|^2} - \frac{\theta_k^2}{\| g_k \|^2}. \tag{3.29}$$

Since $0 < \beta_k^* \leq \beta_k^{FR}$, we have that

$$\frac{\| d_k \|^2}{(g_k^T d_k)^2} \leq (\beta_k^{FR})^2 \frac{\| d_{k-1} \|^2}{\| g_k \|^4} + \frac{2\theta_k}{\| g_k \|^2} - \frac{\theta_k^2}{\| g_k \|^2} \tag{3.30}$$

$$= \left(\frac{\| g_k \|^2}{\| g_{k-1} \|^2} \right)^2 \frac{\| d_{k-1} \|^2}{\| g_k \|^4} + \frac{2\theta_k}{\| g_k \|^2} - \frac{\theta_k^2}{\| g_k \|^2} \tag{3.31}$$

$$= \frac{\| d_{k-1} \|^2}{\| g_{k-1} \|^4} + \frac{2\theta_k}{\| g_k \|^2} - \frac{\theta_k^2}{\| g_k \|^2} \tag{3.32}$$

$$= \frac{\| d_{k-1} \|^2}{\| g_{k-1} \|^4} - \frac{1}{\| g_k \|^2} (\theta_k^2 - 2\theta_k + 1 - 1) \tag{3.33}$$

$$= \frac{\| d_{k-1} \|^2}{\| g_{k-1} \|^4} - \frac{(\theta_k - 1)^2}{\| g_k \|^2} + \frac{1}{\| g_k \|^2} \tag{3.34}$$

$$\leq \frac{\| d_{k-1} \|^2}{\| g_{k-1} \|^4} + \frac{1}{\| g_k \|^2} \tag{3.35}$$

$$= \frac{\| d_{k-1} \|^2}{(g_{k-1}^T d_{k-1})^2} + \frac{1}{\| g_k \|^2}. \tag{3.36}$$

From the above, and the fact that $g_0^T d_0 = - \| g_0 \|^2$, it follows that

$$\frac{\| d_1 \|^2}{(g_1^T d_1)^2} \leq \frac{\| d_0 \|^2}{(g_0^T d_0)^2} + \frac{1}{\| g_1 \|^2} \tag{3.37}$$

$$= \frac{1}{\| g_0 \|^2} + \frac{1}{\| g_1 \|^2} \tag{3.38}$$

$$= \sum_{i=0}^{1} \frac{1}{\| g_i \|^2}. \tag{3.39}$$

$$\frac{\| d_2 \|^2}{(g_2^T d_2)^2} \leq \frac{\| d_1 \|^2}{(g_1^T d_1)^2} + \frac{1}{\| g_2 \|^2} \tag{3.40}$$

$$\leq \frac{1}{\| g_0 \|^2} + \frac{1}{\| g_1 \|^2} + \frac{1}{\| g_2 \|^2} \tag{3.41}$$

$$= \sum_{i=0}^{2} \frac{1}{\| g_i \|^2}. \tag{3.42}$$

$$\vdots \tag{3.43}$$

$$\frac{\| d_k \|^2}{(g_k^T d_k)^2} \leq \frac{\| d_{k-1} \|^2}{(g_{k-1}^T d_{k-1})^2} + \frac{1}{\| g_k \|^2} \tag{3.44}$$

$$\leq \frac{1}{\| g_0 \|^2} + \frac{1}{\| g_1 \|^2} + \frac{1}{\| g_2 \|^2} + \cdots + \frac{1}{\| g_k \|^2} \tag{3.45}$$

$$= \sum_{i=0}^{k} \frac{1}{\| g_i \|^2}. \tag{3.46}$$

Thus,

$$\frac{\| d_k \|^2}{(g_k^T d_k)^2} \leq \sum_{i=0}^{k} \frac{1}{\| g_i \|^2}. \tag{3.47}$$

From (3.27), we have

$$\sum_{i=0}^{k} \frac{1}{\| g_i \|^2} \leq \frac{k+1}{\varepsilon^2}.$$

Therefore, the last inequality implies

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\| d_k \|^2} \geq \varepsilon^2 \sum_{k=0}^{\infty} \frac{1}{k+1} = +\infty,$$

which contradicts (3.1). Thus the proof is complete. □

4. Numerical results

In this chapter, we present numerical results of the new β_k^* algorithm. We also do a comparison of our method with other methods in the literature. These methods include the β_k^{GN} hybrid by Gilbert and Nocedal [15], the β_k^{TS} conjugate gradient hybrid method by Touati-Ahmed and Storey [27] and the β_k^{HS-DY} (Hestenes and Stiefel and Dai and Yuan) [8] hybrid conjugate gradient method. A total of 14

test problems are used to test our algorithms and have been taken from different sources, that is, Luksan and Vlcek [20], Neculai Andrei [2] and More, Garbow and Hillstrom [22].
A number of parameters used are defined. These are the tolerance, ϵ, the constants μ and σ and the step length α_k. The tolerance has been set to $\epsilon = 10^{-6}$, the constants μ and σ are set to $\sigma = 0.7$ and $\mu = 0.3$. The step length $\alpha_k > 0$ is calculated using the strong Wolfe line search. All the parameters used in testing our algorithms have been set to the same values for each algorithm. Our new algorithm is coded in MATLAB R2010a.

We first of all present our numerical results in the form of a table, Table 1, where the methods are presented as follows:

- M1: The new β_k^* hybrid method;
- M2: The Gilbert and Nocedal β_k^{GN} Hybrid method [15];
- M3: The Touti Ahmed and Storey β_k^{TS} hybrid method [27];
- M4: The Dai and Yuan β^{HS-DY} hybrid method [8].

The columns 'Problem' and 'Dim' represent the name of the test problem and the dimension of the problems, respectively. The results are denoted by $'iter/fe'$, where $iter$ and fe are the number of iterations and function evaluations, respectively. The highlighted results show the best out of the 4 methods.

		M1	M2	M3	M4
Problem	Dim	$iter/fe$	$iter/fe$	$iter/fe$	$iter/fe$
Rosenbrock	2	72/1476	68/1314	76/1486	**47/894**
Freud n Roth	2	**36/723**	71/1434	60/1205	50/1033
Beale	2	69/741	**28/288**	**28/288**	45/453
Himmelblau	2	**16/167**	18/179	21/212	23/195
White	6	52/1140	**50/1008**	66/1389	82/1760
Wood	4	165/3593	148/3004	**94/1922**	106/2134
PQuad	7	35/289	31/221	34/236	**29/205**
Power	6	**41/526**	50/596	50/596	**41/492**
Fletcher	5	42/1035	**33/798**	36/873	39/942
Trig	3	17/249	19/254	20/265	**16/220**
Powell	2	19/793	**15/632**	17/708	18/727
ExPowell	4	266/3954	**150/2165**	199/2862	253/3530
Penalty I	5	**10/31**	13/25	13/25	12/26
Broyden tri	10	**29/463**	33/514	33/514	29/449

TABLE 1. Numerical results for all the four methods

From Table 1, we see that the new β_k^* hybrid method (M1) and the Gilbert and Nocedal hybrid method (M2) requires fewer function evaluations and number of iterations for 5 problems. On the other hand Touti-Ahmed and Storey hybrid method (M3) and Dai and Yuan hybrid (M4) have fewer function evaluations and number of iterations for 2 problems and 4 problems, respectively. We also see that although M2 and M3 had the same number of iterations and function evaluations for Beale problem, it is not always the case as we see that for the Power problem, M1 and M4 have the same number of iterations but the number function evaluations is higher for M1. Generally, we can say that our new hybrid conjugate gradient method is promising and competitive.

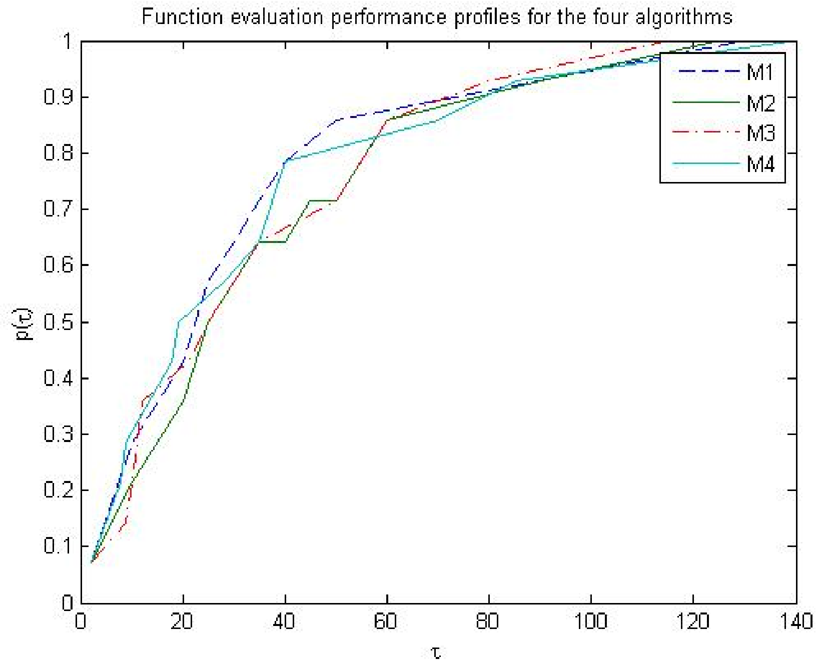

FIGURE 1. Performance Profile for Function Evaluations

To better compare the numerical performance of the 4 methods, we use performance profiles, introduced in [12]. This is reflected in Figure 1 where the performance profile for function evaluations is plotted. Letting $P = \{p_1, p_2, \ldots, p_{14}\}$ be the set of problems and $S = \{s_1, s_2, s_3, s_4\}$ be the set of the solvers M1, M2, M3, M4, respectively, we compare the performance of the solvers in S on the problems in P. Let $a_{p,s}$ denote the performance measure (e.g. function evaluations) required by solver $s \in S$ to solve problem $p \in P$. Then the performance ratio is given by

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

We assume that a parameter $r_M \geq r_{p,s}$ is chosen, for all p, s, and $r_{p,s} = r_M$ if and only if solver s does not solve problem p. To obtain an overall assessment of the performance of the solver, we define performance profile as,

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

where $\rho_s(\tau)$ is the probability for solver $s \in S$ that a performance ratio $r_{p,s}$ is within a factor $\tau \in \mathbb{R}$ of the best possible ratio and n_p is the number of problems. The function ρ_s is the cumulative distribution function for the performance ratio. Note that we always have $r_{p,s} \geq 1$. When $r_{p,s} = 1$ we have

$$a_{p,s} = \min\{a_{p,s} : s \in S\},$$

meaning that solver $s \in S$ was best for a certain problem p of all the problems.

Figure 1 shows the performance of the four methods relative to the function evaluations. We can see that all the methods successfully solved all the problems. From the figure, we see that the new method β_k^* is very much competitive with the other hybrid methods. Thus, the new method adds to the already available collection of hybrid methods that can be useful both to other researchers and people looking for solutions to optimization problems in the industries.

5. Conclusion

In this research, we have presented a new hybrid conjugate gradient algorithm in which the parameter β_k is a combination of the ideas of Dai and Yuan [8] and Gilbert and Nocedal [15]. Our new computational scheme takes advantage of the attractive features of the Fletcher Reeves (β_k^{FR}) and Polak-Ribiere-Polyak (β_k^{PRP}) methods. The direction d_k generated by our algorithm satisfies the descent condition independently of the line search used. A convergence analysis of the proposed algorithm was also carried out and we showed that the algorithm is globally convergent independently of any line search.

Furthermore, our new algorithm was compared with three other hybrid conjugate gradient methods that have been proposed in the literature. Using a set of 14 test unconstrained optimization problems, a numerical study concerning the behavior of our new algorithm has been presented. The numerical results show that our algorithm is very competitive with these other methods.

Further research will be done on developing more hybrid conjugate gradient methods for large scale unconstrained optimization problems. Although test problems of lower dimension were mainly used for testing our algorithm, we intend to extend the algorithm in future to problems with much higher dimension. Another direction would be to extend this conjugate gradient methods to constrained optimization problems, as well as optimal control problems.

References

1. M. Al-Baali, *Descent property and global convergence of the Fletcher-Reeves method with inexact line search*, IMA Journal of Numerical Analysis 5 (1985), 121–124.
2. N. Andrei, *An unconstrained Optimization Test Functions Collection*, Advanced Modeling and Optimization 10 Issue 1 (2008), 147-161.
3. N. Andrei, *Hybrid conjugate gradient algorithm for unconstrained optimization*, Journal of Optimization Theory and Applications 141 (2009), 249–264.
4. A. Antoniou and W. S. Lu, *Practical Optimization, Algorithms and Engeneering Applications*, Springer, New York (2007).
5. S. Baaie-Kafaki, *A hybrid conjugate gradient method based on a quadratic relaxation of the Dai-Yuan Hybrid conjugate gradient parameter*, Optimization 62 Issue 7 (2013), 929–941.
6. Y. H. Dai, *New properties of a nonlinear conjugate gradient method*, Numerische Mathematik 89 Issue 1 (2001), 83–98.
7. Y. H. Dai, *Convergence of conjugate gradient methods with constant step sizes*, Optimization Methods and Software 26 Issue 6 (2011), 895–909.
8. Y. H. Dai and Y. Yuan, *An Efficient Hybrid Conjugate Gradient Method for Unconstrained Optimization* , Annals of Operations Research 103 (2001), 33–47.
9. Y. H. Dai and Y. Yuan, *A note on the nonlinear conjugate gradient method*, Journal of Computational Mathematics 20 Issue 6 2002, 575–582.
10. Y. H. Dai, *A family of hybrid conjugate gradient methods for unconstrained optimization*, Mathematics of Computation 72 Issue 243 (2003), 1317–1328.
11. Y. H. Dai and Y. Yuan, *Some properties of a new conjugate gradient method*, in Advances in Nonlinear Programming, Y. Yuan ed., Kluwer Publications, Boston (1998), 251-262.
12. E. D. Dolan and J. J. More, *Benchmarking Optimization Software with performance profiles*, Mathematical Programming, Series A. 91 (2002), 201-213.
13. R. Fletcher, *Practical methods of optimization*, John Wiley, New York, (1981).
14. R. Fletcher, and C. Reeves, *Function minimization by conjugate gradients*, Computer Journal 7 (1964), 149–154.
15. J.C. Gilbet and J. Nocedal, *Global convergence properties of conjugate gradient methods for optimization*, SIAM Journal on Optimization, 2 (1992), 21-42.

16. L. Grippo and S. Lucidi, *Convergence conditions, line search algorithms and trust region implementations for the Polak-Ribiere conjugate gradient method*, Optimization Methods and Software, 20(1) (2005), 71-78.
17. W.W. Hager and H. Zhang, *A survey of nonlinear conjugate gradient methods*, Pacific Journal of Optimization 2 (2006), 35–58.
18. M. R. Hestenes and E. L. Stiefel, *Methods of conjugate gradients for solving linear systems* , Journal of Research of the National Bureau of Standards, 49(6) (1952), 409-436.
19. X. Li and X. Zhao, *A hybrid conjugate gradient method for optimization problems*, Natural Science, 3(1) (2011), 85-90.
20. L. Lukšan and J. Vlček, *Test problems for Unconstrained Optimisation*, Institute of Computer Science, Technical report, 897 (2003), 1-24.
21. J. Mo, N. Gu and Z. Wei, *Hybrid conjugate gradient methods for unconstrained optimization*, Optimization Methods and Software, 22(2) (2007), 297-307.
22. J. J. More, B. S. Garbow and K. E. Hillstrom, *Testing unconstrained Optimization Software*, ACM Transactions on Mathematical Software, 7(1) (1981), 17-41.
23. J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd edition. Springer, New York (2006).
24. E. Polak, and G. Ribiére, *Note sur la convergence de méthodes de directions conjuguées*, Revue française d'infomatique et de recherche opérationnelle, série rouge, 3(1) (1969), 35–43, 1969.
25. M. J. D. Powell, *Nonconvex minimization calculations and the conjugate gradient method*, Lecture notes in Mathematics, 1066 (1984), 121-141.
26. Z. J. Shi and J. Shen, *Convergence of the Polak-Ribiere-Polyak conjugate gradient method*, Nonlinear Analysis, 66 (2007), 1428-1441.
27. D. Touti-Ahmed and C. Storey, *Efficient Hybrid Conjugate Gradient Techniques*, Journal of Optimization Theory and Applications, 64(2) (1990), 379-397.
28. C. Wang and S. Lian, *Global convergence properties of the two new dependent Fletcher-Reeves conjugate gradient methods*, Applied Mathematics and Computation, 181 (2006), 920-931.
29. A. Zhou, Z. Zhu, H. Fan and Q. Qing, *Three New Hybrid Conjugate Gradient Methods for Optimization*, Applied Mathematics, 2 (2011), 303-308.