Stud. Univ. Babeş-Bolyai Math. 69(2024), No. 2, 457–468
DOI: 10.24193/subbmath.2024.2.14

New hybrid conjugate gradient method as a convex combination of PRP and RMIL+ methods

Ghania Hadji, Yamina Laskri, Tahar Bechouat and Rachid Benzine

Abstract. The Conjugate Gradient (CG) method is a powerful iterative approach for solving large-scale minimization problems, characterized by its simplicity, low computation cost and good convergence. In this paper, a new hybrid conjugate gradient HLB method (HLB: Hadji-Laskri-Bechouat) is proposed and analysed for unconstrained optimization. We compute the parameter β_k^{HLB} as a convex combination of the Polak-Ribière-Polyak $\left(\beta_k^{PRP}\right)$ and the Mohd Rivaie-Mustafa Mamat and Abdelrhaman Abashar $\left(\beta_k^{RMIL+}\right)$ i.e. $\beta_k^{HLB} = (1-\theta_k)\,\beta_k^{PRP} + \theta_k \beta_k^{RMIL+}$. By comparing numerically CGHLB with PRP and RMIL+ and by using the Dolan and More CPU performance, we deduce that CGHLB is more efficient.

Mathematics Subject Classification (2010): 90C26, 65H10, 65K05, 90C26, 90C06.

Keywords: Unconstrained optimization, hybrid conjugate gradient method, line search, descent property, global convergence.

1. Introduction

Consider the nonlinear unconstrained optimization problem

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

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function, bounded from below. The gradient of f is denoted by $g(x)$. To solve this problem, we start from an initial

Received 08 October 2021; Accepted 12 September 2022.

point $x_0 \in \mathbb{R}^n$. Nonlinear conjugate gradient methods generate sequences $\{x_k\}$ of the following form:

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

where x_k is the current iterate point and $\alpha_k > 0$ is the step size which is obtained by line search [7].

The iterative formula of the conjugate gradient method is given by (1.2), where d_k is the search direction defined by

$$d_{k+1} = \begin{cases} -g_k & \text{si } k = 1 \\ -g_{k+1} + \beta_k d_k & \text{si } k \geq 2 \end{cases} \tag{1.3}$$

where β_k is a scalar and $g(x)$ denotes $\nabla f(x)$ [10]. If f is a strictly convex quadratic function, namely,

$$f(x) = \frac{1}{2} x^T H x + b^T x, \tag{1.3bis}$$

where H is a positive definite matrix and if α_k is the exact one-dimensional minimizer along the direction d_k, i.e.

$$\alpha_k = \arg\min_{\alpha > 0} \{f(x + \alpha d_k)\} \tag{1.3tris}$$

then (1.2), (1.3), (1.3bis), (1.3tris) is called the linear conjugate gradient method. Otherwise, (1.2), (1.3) is called the nonlinear conjugate gradient method. Conjugate gradient methods can broadly be classified based on the used strategies of the way in which the search direction is updated and the algorithms dealing with the step size minimization along a direction [6]. In [12], a convex combination of LS and FR ([1]) is proposed with a newton descent direction.

The line search in the non linear conjugate gradient methods is often based on the standard Wolfe conditions [23]:

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \rho \alpha_k g_k^t d_k \tag{1.4}$$

$$g_{k+1}^t d_k \geq \delta g_k^t d_k \tag{1.5}$$

where $0 < \rho \leq \delta < 1$.

Conjugate gradient methods differ in their way of defining the scalar parameter β_k. In the literature, there have been proposed several choices for β_k which give rise to distinct conjugate gradient methods [16], [27]. The most well known conjugate gradient methods are the Hestenes–Stiefel (HS) method [17], the Fletcher-Reeves (FR) method [1], [13], the Polak-Ribière-Polyak (PRP) method [20], [19], the Conjugate Descent method(CD) [13], the Liu-Storey (LS) method [18], the Dai-Yuan (DY) method [08], [09], Hager and Zhang (HZ) method [15] and the RMIL+ method [21], [22]. The update parameters of these methods are respectively specified as follows:

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k},\ \beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2},\ \beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2},\ \beta_k^{CD} = -\frac{\|g_{k+1}\|^2}{d_k^T g_k}$$

$$\beta_k^{LS} = -\frac{g_{k+1}^T y_k}{d_k^T g_k},\ \beta_k^{DY} = \frac{\|g_{k+1}\|^2}{d_k^T y_k},\ \beta_k^{HZ} = \left(y_k - 2 d_k \frac{\|y_k\|^2}{d_k^T y_k} \right)^T \frac{g_{k+1}}{d_k^T y_k},$$

$$\beta_k^{RMIL+} = \frac{g_{k+1}^T(g_{k+1} - g_k - d_k)}{\|d_k\|^2}.$$

Some of these methods, such as Fletcher and Reeves (FR) [13], Dai and Yuan (DY) [8] and Conjugate Descent (CD) [13] have strong convergence properties, but they may have modest practical performance due to jamming. On the other hand, the methods of Polak and Ribière and Polyak (PRP) [20], Hestenes and Stiefel (HS) [17] or Liu and Story (LS) [18] may not generally be convergent, but they often have better computational performance.

In the process of obtaining more robust and efficient conjugate gradient methods, some researchers suggested the hybrid conjugate gradient algorithm which combined the good features of the methods involve in the hybridization. Even though conjugate gradient improvement using hybridization is a classic deeply investigated problem; it still an attractive topic for the research community due to its contemporary use in numerous prominent disciplines [25].

The first hybrid conjugate gradient method was given by Touati-Ahmed and Storey (1990) [24] to avoid jamming phenomenon.

The researchers were motivated by the works of Andrei [5], [4]; Dai and Yuan [9]; Zhang and Zhou [26]. Their parameter β_k^N is computed as a convex combination of β_k^{FR} and β_k^* other algorithms, i.e.

$$\beta_k^N = (1 - \theta_k)\,\beta_k^{FR} + \theta_k \beta_k^*$$

The Wolfe line search was employed to determine the step length $\alpha_k > 0$ and the new method proved to be more robust numerical wise as compared to FR and other methods. The global convergence was established under some suitable conditions.

In [4] Andrei has proposed a new hybrid conjugate gradient algorithm where the parameter β_k^A is computed as a convex combination of the Polak-Ribière-Polyak and the Dai-Yuan conjugate gradient algorithms i.e.

$$\beta_k^A = (1 - \theta_k)\,\beta_k^{PRP} + \theta_k \beta_k^{DY}$$

and θ_k is presented to satisfy the conjugacy condition

$$\theta_k = \theta_k^{CCOMB} = \frac{(y_k^t g_{k+1})\,(y_k^t s_k) - (y_k^t g_{k+1})\,(g_k^t g_k)}{(y_k^t g_{k+1})\,(y_k^t s_k) - \|g_{k+1}\|^2 \|g_k\|^2}$$

where $s_k = x_{k+1} - x_k$. To satisfy Newton direction he takes

$$\theta_k = \theta_k^{NDOMB} = \frac{(y_k^t g_{k+1} - s_k^t g_{k+1})\,\|g_k\|^2 - (y_k^t g_{k+1})\,(y_k^t s_k)}{\|g_{k+1}\|^2 \|g_k\|^2 - (y_k^t g_{k+1})\,(y_k^t s_k)}$$

but in the combination of HS and DY from Newton direction, he puts

$$\theta_k = \frac{-s_k^t g_{k+1}}{g_k^t g_{k+1}}.$$

On the other hand, from Newton direction with modified secant condition (Hybrid M-Andrei), Andrei has proposed another method

$$\beta_k^{HYBRIDM} = (1 - \theta_k)\,\beta_k^{HS} + \theta_k \beta_k^{DY}$$

where

$$\theta_k = \frac{\left(\frac{\delta\eta_k}{s_k^t s_k} - 1\right) s_k^t g_{k+1} - \frac{y_k^t g_{k+1}}{y_k^t s_k}\delta\eta_k}{g_k^t g_{k+1} + \frac{g_k^t g_{k+1}}{y_k^t s_k}\delta\eta_k}$$

δ is parameter. In [14] Salah Gazi Shareef and Hussein Ageel Khatab have introduced a new hybrid CG method

$$\beta_k^{New} = (1-\theta_k)\,\beta_k^{PRP} + \theta_k\beta_k^{BA}$$

where β_k^{BA} is selected in [2].

Recently Delladji *et al.* [11] proposed a hybridazation of PRP and HZ schemes using the congugacy condition.

In this paper, we present another hybrid CG algorithm noted CGHLB (HLB is an abbreviation to Hadji; Laskri and Bechouat), witch is a convex combination of the PRP ([20]) and RMIL+ ([21]) conjugate gradient algorithms.We are interested to combine these two methods in a hybrid CG algorithm because PRP has good computational properties and RMIL+ has strong convergence properties. In section 2, we introduce our hybrid CG method and prove that it generates descent directions. In Section 3 we present and prove global convergence results. Numerical results and a conclusion are presented in section 4. By comparing numerically CGHLB with PRP and RMIL+ and by using the Dolan and More CPU performance, we deduce that CGHLB is more efficient.

2. HLB conjugate gradient method

The iterates $x_0, x_1, ...$ of the proposed HLB algorithm are computed by means of the recurrence (1.2) where the step size $\alpha_k > 0$ is determined according to the wolfe line search conditions (1.4), (1.5). The directions d_k are generated by the rule:

$$d_k = \begin{cases} -g_0 & \text{if } k = 0 \\ -g_k + \beta_{K-1}^{HLB} d_{k-1} & \text{if } k \geq 1 \end{cases} \tag{2.1}$$

where

$$\beta_k^{HLB} = (1-\theta_k)\,\beta_k^{PRP} + \theta_k\beta_k^{RMIL+}$$

i.e.

$$\beta_k^{HLB} = (1-\theta_k)\,\frac{g_{k+1}^t y_k}{\|g_k\|^2} + \theta_k\frac{g_{k+1}^t\,(g_{k+1} - g_k - d_k)}{\|d_k\|^2} \tag{2.2}$$

HLB is an abbreviation to Hadji; Laskri and Bechouat; θ_k is a scalar parameter which will be determined in a specific way to be described in the following section. Observe that if $\theta_k = 0$ then $\beta_k^{HLB} = \beta_k^{PRP}$ and if $\theta_k = 1$, then $\beta_k^{HLB} = \beta_k^{RMIL+}$. On the other hand if $0 < \theta_k < 1$, then β_k^{HLB} is a convex combination of $\beta_k^{PRP}$and β_k^{RMIL+}. The parameter θ_k is selected in such away that at every iteration the conjugacy condition is satisfied. It can be noted that,

$$d_{k+1} = -g_{k+1} + (1-\theta_k)\,\frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k + \theta_k\frac{g_{k+1}^t\,(g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k \tag{2.3}$$

so multiply both sides of above equation by y_k and by using the conjugacy condition $\left(d_{k+1}^t y_k = 0\right)$ we have:

$$0 = -g_{k+1}^t y_k + (1-\theta_k)\frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t y_k + \theta_k \frac{g_{k+1}^t\left(g_{k+1} - g_k - d_k\right)}{\|d_k\|^2} d_k^t y_k \tag{2,4}$$

After a simple calculation we get

$$\theta_k = \frac{g_{k+1}^t y_k \|g_k\|^2 \|d_k\|^2 - \left(g_{k+1}^t y_k\right)\left(d_k^t y_k\right)\|d_k\|^2}{\left(\left(g_{k+1}^t\left(y_k - d_k\right)\right)\|g_k\|^2 - \left(g_{k+1}^t y_k\right)\|d_k\|^2\right)\left(d_k^t y_k\right)} \tag{2.5}$$

So, to ensure the convergence of this method when the parameter θ_k goes out of interval $]0,1[$, i.e. when $\theta_k \le 0$ or $\theta_k \ge 1$, we prefer to take β_k^{HLB} as following:

$$\beta_k^{HLB} = \begin{cases} (1-\theta_k)\,\beta_k^{PRP} + \theta_k \beta_k^{RMIL+} & \text{if } 0 < \theta_k < 1 \\ \beta_k^{PRP} & \text{if } \theta_k \le 0 \\ \beta_k^{RMIL+} & \text{if } \theta_k \ge 1 \end{cases} \tag{2.5(bis)}$$

We are now able to present our new algorithm, the Conjugate Gradient CGHLB Algorithm:

CGHLB Algorithm

Step 1: Initialization:

Set $k = 0$, select the initial point $x_o \in \mathbb{R}^n$.select the parameters $0 < \rho \le \delta < 1$, and $\varepsilon > 0$.

Compute $f(x_0)$, and $g_0 = \nabla\ f(x_0)$. Consider $d_0 = -g_0$.

Step 2: Test for continuation of iterations:

If $\|g_k\| \le \varepsilon$ then stop else set. $d_k = -g_k$

Step 3: Line search:

Compute $\alpha_k > 0$ satisfying the Wolfe line search condition (1,4) and (1,5) and update the variables, $x_{k+1} = x_k + \alpha_k d_k$; compute $f(x_{k+1})$, g_{k+1} and $s_k = x_{k+1} - x_k$; $y_k = g_{k+1} - g_k$.

Step 4: θ_k Parameter computation:

If $\left(\left(g_{k+1}^t\left(y_k - d_k\right)\right)\|g_k\|^2 - \left(g_{k+1}^t y_k\right)\|d_k\|^2\right)\left(d_k^t y_k\right) = 0$;

then set $\theta_k = 0$, otherwise, compute θ_k as in (2.5).

Step 5: β_k^{HLB}Conjugate gradient parameter computation:

If $0 < \theta_k < 1$, then compute β_k^{HLB} as in (2.2).

If $\theta_k \ge 1$, then set $\beta_k^{HLB} = \beta_k^{RMIL+}$.

If $\theta_k \le 0$, then set $\beta_k^{HLB} = \beta_k^{PRP}$.

Step 6: Direction computation:

Compute $d_{k+1} = -g_{k+1} + \beta_k^{HLB} d_k$.

Set k=k+1 and go to step 3.

The following theorem shows that our method assures the descent condition, when $0 < \theta_k < 1$.

Theorem 2.1. *In the algorithm (1.2), (1.3) and (2.5) assume that d_k is a descent direction $(g_k^t d_k < 0)$, and α_k is determined by the Wolfe line search (1.4); (1.5). If $0 < \theta_k < 1$ then the direction d_{k+1}given by (2.3) is a descent direction.*

Proof. Multiply both sides of (2,3) by g_{k+1} we have:

$$\begin{aligned} g_{k+1}^T d_{k+1} &= -\|g_{k+1}\|^2 + (1-\theta_k)\frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t g_{k+1} \\ &\quad + \theta_k \frac{g_{k+1}^t (g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k^t g_{k+1} \end{aligned}$$

$$\begin{aligned} g_{k+1}^T d_{k+1} &= -(1-\theta_k+\theta_k)\|g_{k+1}\|^2 + (1-\theta_k)\frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t g_{k+1} \\ &\quad + \theta_k \frac{g_{k+1}^t (g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k^t g_{k+1} \end{aligned}$$

$$\begin{aligned} g_{k+1}^T d_{k+1} &= \left[-(1-\theta_k)\|g_{k+1}\|^2 + (1-\theta_k)\frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t g_{k+1}\right] \\ &\quad + \left[-(\theta_k)\|g_{k+1}\|^2 + \theta_k \frac{g_{k+1}^t (g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k^t g_{k+1}\right] \end{aligned}$$

$$\begin{aligned} g_{k+1}^T d_{k+1} &= (1-\theta_k)\left[-\|g_{k+1}\|^2 + \frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t g_{k+1}\right] \\ &\quad + (\theta_k)\left[-\|g_{k+1}\|^2 + \frac{g_{k+1}^t (g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k^t g_{k+1}\right] \end{aligned}$$

since $0 < \theta_k < 1$ then

$$\begin{aligned} g_{k+1}^T d_{k+1} \leq &\left[-\|g_{k+1}\|^2 + \frac{g_{k+1}^t y_k}{\|g_k\|^2} d_k^t g_{k+1}\right] \\ &+ \left[-\|g_{k+1}\|^2 + \frac{g_{k+1}^t (g_{k+1} - g_k - d_k)}{\|d_k\|^2} d_k^t g_{k+1}\right] \end{aligned} \tag{2.6}$$

If the step length α_k is chosen by an exact line search. Then $g_{k+1}^T d_k = 0$.
If the step length α_k is chosen by an inexact line search $\left(g_{k+1}^T d_k \neq 0\right)$ then we have:

$$g_{k+1}^T d_{k+1} < 0$$

because the algorithms of (PRP) and $(RMIL+)$ satisfied the descent property.
The proof is completed. □

3. Global convergence properties

The following assumptions are often needed to prove the convergence of the nonlinear CG:

Assumption 1

The level set $\Omega = \{x \in \mathbb{R}^n / f(x) \leq f(x_0)\}$ is bounded, where x_0 is the starting point.

Assumption 2

In some neighborhood N of Ω, the objective function is continuously differentiable and its gradient is Lipschitz continuous, namely, there exists a constant $l > 0$ such that:

$$\|g(x) - g(y)\| \leq l\,\|x - y\| \quad for\ any\ x, y \in N$$

Under these assumptions on f there exists a constant μ such that $\|g(x)\| \leq \mu$, for all $x \in \Omega$.

Lemma 3.1. [28] *Suppose Assumption 1 and 2 hold, and consider any conjugate gradient method (1.2) and (1.3), where d_k is a descent direction and α_k is obtained by the strong Wolfe line search. If*

$$\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} = +\infty \tag{3.1}$$

then

$$\liminf_{k\to\infty} \|g_k\| = 0 \tag{3.2}$$

Assume that the function f is uniformly convex function, i.e. there exists a constant $\Gamma \geq 0$ such that,

$$for\ all\ x, y \in \Omega : (\nabla f(x) - \nabla f(y))^t (x - y) \geq \Gamma \|x - y\|^2 \tag{3.3}$$

and the steplength α_k is given by the strong Wolfe line search.

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \sigma_1 \alpha_k g_k^t d_k \tag{3.4}$$

$$\left|g_{k+1}^t d_k\right| \leq -\sigma_2 g_k^t d_k \tag{3.5}$$

For uniformly convex function which satisfies the above assumptions, we can prove that the norm of d_{k+1} given by (2.3) is bounded above.

Using the above lemma, we obtain the following theorem.

Theorem 3.2. *Suppose that Assumption 1 and 2 hold. Consider the algorithm (1.2), (2.3) and (2.5), where $0 \leq \theta_k \leq 1$ and α_k is obtained by the strong Wolfe line search (3.4) and (3.5).*

If d_k tends to zero and there exists non negative constants η_1 and η_2 such that:

$$\|g_k\|^2 \geq \eta_1 \|s_k\|^2 \ \ and\ \ \|g_{k+1}\|^2 \leq \eta_2 \|s_k\| \tag{3.6}$$

and f is uniformly convex function, then

$$\lim_{k\to\infty} g_k = 0 \tag{3.7}$$

Proof. From (3,3) it follows that

$$y_k^t s_k \geq \Gamma \|s_k\|^2$$

since $0 \leq \theta_k \leq 1$, from uniform convexity and (3.6) we have

$$\left|\beta_k^{HLB}\right| \leq \left|\frac{g_{k+1}^t y_k}{\|g_k\|^2}\right| + \left|\frac{g_{k+1}^t\left(g_{k+1} - g_k - d_k\right)}{\|d_k\|^2}\right|$$

$$\leq \frac{\left|g_{k+1}^t y_k\right|}{\|g_k\|^2} + \frac{\left|g_{k+1}^t y_k\right|}{\|d_k\|^2} + \frac{\left|g_{k+1}^t d_k\right|}{\|d_k\|^2}$$

$$\leq \frac{\|g_{k+1}\|\,\|y_k\|}{\|g_k\|^2} + \frac{\|g_{k+1}\|\,\|y_k\|}{\|d_k\|^2} + \frac{\|g_{k+1}\|\,\|d_k\|}{\|d_k\|^2}$$

from Lipschitz condition

$$\|y_k\| \leq l\,\|s_k\|$$

$$\left|\beta_k^{HLB}\right| \leq \frac{\|g_{k+1}\|\,\|y_k\|}{\eta_1\,\|s_k\|^2} + \frac{\|g_{k+1}\|\,\|y_k\|}{\|d_k\|^2} + \frac{\|g_{k+1}\|}{\|d_k\|}$$

$$\leq \frac{\mu l\,\|s_k\|}{\eta_1\,\|s_k\|^2} + \frac{\mu l\,\|s_k\|\,\alpha_k^2}{\|s_k\|^2} + \frac{\mu\alpha_k}{\|s_k\|}$$

$$= \frac{\mu l}{\eta_1\,\|s_k\|} + \frac{\mu l\alpha_k^2}{\|s_k\|} + \frac{\mu\alpha_k}{\|s_k\|}$$

Hence

$$\|d_{k+1}\| \leq \|g_{k+1}\| + \left|\beta_k^{HLB}\right|\,\|d_k\|$$

$$\leq \mu + \frac{\mu l\,\|s_k\|}{\eta_1\alpha_k\,\|s_k\|} + \frac{\mu l\,\|s_k\|\,\alpha_k^2}{\alpha_k\,\|s_k\|} + \frac{\mu\alpha_k\,\|s_k\|}{\alpha_k\,\|s_k\|}$$

$$= 2\mu + \mu l\alpha_k + \frac{\mu l}{\eta_1\alpha_k}$$

which implies that (3.1) is true. Therefore, by Lemma 1 we have (3.2), which for uniformly convex functions is equivalent to (3.7). □

4. Numerical results and discussion

In the present numerical experiments, we analyze the efficiency of β^{HLB}, as compared to the classic methods: β^{PRP} and β^{RMIL+}. These comparisons are based on the number of iterations and CPU time per second to reach the optimum. All the comparisons are done with two or three different initial points and different number of variables ranging from 2 to 20000. All variables have been experimented to each function test [3]. For the numerical tests, the strong Wolfe line searches parameters have been experimentally fixed to $\rho = 10^{-3}$ and $\delta = 10^{-4}$. All tests were terminated when the stopping criteria $\|g_k\| \leq \varepsilon$ is fulfilled, where $\varepsilon = 10^{-6}$. When the iteration number exceeds 2000 or the CPU execution time exceeded 500 seconds, the test is considered as failed.

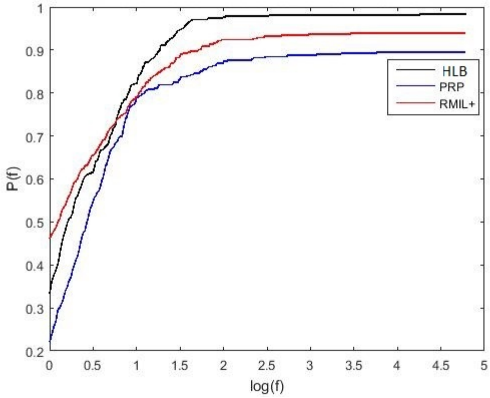

FIGURE 1. Performance Profile based on the CPU time

Figures 1 and 2 show that the method of β^{HLB} is superior when compared to β^{PRP} and β^{RMIL+} with the least duration of CPU time. The highest percentage of successful comparison is with β^{HLB} at 98.34%, followed by β^{RMIL+} with 93.72%. However, the successful rate comparison for β^{PRP} is low at 90.05%. Hence, our method (β^{HLB}) successfully solves the test problems, and it is competitive with the well-known conjugate gradient methods for unconstrained optimization.

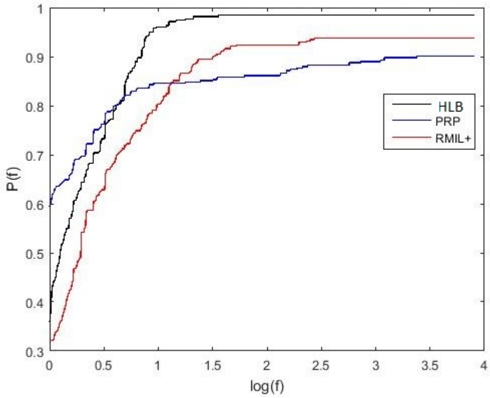

FIGURE 2. Performance Profile based on the iteration number

Table 1. A list of test problems

No.	Function	Dimension	Initial points
01	Alpine 1	4, 5, 7, 10, 12, 30, 100	(1, ..., 1)
02	Beale	2	(−1, −1) ; (0, 0) ; (1, 1)
03	Booth	2	(−1, −1) ; (1, 1) ; (3, 3)
04	Branin	2	(−1, −1) ; (0, 0) ; (1, 1)
05	Diagonal 1	2, 4, 6, 8, 10, 20, 100, 200	(1, ..., 1) ; (2, ..., 2) ; (3, ..., 3)
06	Diagonal 2	2, 4, 10, 100, 200, 400, 500, 600, 1000	(−1, ..., −1) ; (0, ..., 0) ; (1, ..., 1)
07	Diagonal 4	1000, 5000, 8000, 10000, 14000, 16000, 20000	(2, ..., 2) ; (5, ..., 5) ; (10, ..., 10)
08	Exponential	2, 4, 6, 8, 10, 12, 14, 15, 16, 20	(1, ..., 1)
09	Griewank	10, 100, 500, 1000, 2000, 5000, 10000	(−2, ..., −2) ; (2, ..., 2)
10	Hager	2, 4, 10, 100, 200, 500, 800, 1000	(−1, ..., −1) ; (0, ..., 0)
11	Himmelblau	2, 4, 10, 100, 1000, 5000, 10000, 20000	(−5, ..., −5) ; (5, ..., 5)
12	Leon	2	(−0.5, −0.5) ; (0, 0) ; (0.5, 0.5)
13	Matyas	2	(1, 1) ; (2, 2) ; (5, 5)
14	Penalty	2, 10, 100, 500, 1000, 2500, 4000, 5000, 10000	(−1, ..., −1) ; (0, ..., 0) ; (1, ..., 1)
15	Perquadratic	2, 4, 8, 10, 20, 50, 200	(−5, ..., −5) ; (3, ..., 3) ; (5, ..., 5)
16	Power	2, 4, 8, 10, 20, 50, 100, 500	(−2, ..., −2) ; (2, ..., 2)
17	Qing	2, 10, 100, 200, 300, 400, 500, 1000, 2000	(−2, ..., −2) ; (2, ..., 2)
18	Quadratic	2, 10, 100, 200, 500, 750, 1000	(2, ..., 2) ; (4, ..., 4)
19	Quartic	2, 4, 10, 100, 200, 500	(1, ..., 1) ; (2, ..., 2)
20	Rastrigin	2, 10, 100, 200, 500	(−5, ..., −5) ; (5, ..., 5)
21	Raydan 1	2, 4, 10, 20, 50, 80, 90, 100	(−2, ..., −2) ; (2, ..., 2)
22	Raydan 2	2, 10, 100, 500, 1000, 2000, 3000	(−2, ..., −2) ; (2, ..., 2)
23	Rosenbrock	2, 10, 10, 50, 100, 200, 1000, 2000, 5000, 10000	(0, ..., 0)
24	Schwefel 2. 20	2, 4, 10, 20	(−1, ..., −1) ; (2, ..., 2)
25	Schwefel 2. 21	5, 10, 15, 20	(1, ..., 1) ; (2, ..., 2)
26	Schwefel 2. 23	2, 5, 10, 20	(−1, ..., −1) ; (1, ..., 1)
27	Sphere	2, 10, 20, 100, 1000, 5000, 20000	(−4, ..., −4) ; (4, ..., 4)
28	Styblinski	2, 10, 100, 500, 1000, 2000, 5000	(0, ..., 0) ; (2, ..., 2)
29	Sumsquares	2, 10, 20, 100, 300, 500, 1000	(5, ..., 5) ; (10, ..., 10)

5. Conclusion

Numerous studies have been devoted to develop and improve hybrid conjugate gradient methods. In this paper we have presented a new convex hybridation of the PRP and the RMIL+ conjugate gradient algorithms; HLB. The global convergence of our method is demonstrated for $0 < \theta < 1$. Numerical experiments reveal that our method is reaching the optimum in less iteration number and CPU time comparing to RMIL+ and PRP.

References

[1] Al-Baali, M., *Descent property and global convergence of Fletcher-Reeves method with inexact line search*, IMA J. Numer. Anal., **5**(1985), no. 1, 121-124.

[2] Al-Bayati, A.Y., Al-Assady, N.H., *Conjugate Gradient Method*, Technical Research report, Technical Research, School of Computer Studies, Leeds University, 1986.

[3] Andrei, N., *An unconstrained optimization test functions collection*, Adv. Model. Optim., **10**(2008), no. 1, 147-161.

[4] Andrei, N., *Another hybrid conjugate gradient algorithm for unconstrained optimization*, Numer. Algorithms, **47**(2008), no. 2, 143-156.

[5] Andrei, N., *Hybrid Conjugate Gradient Algorithm for Unconstrained Optimization*, J. Optim. Theory Appl., **141**(2009), no. 2, 249-264.

[6] Andrei, N., *Nonlinear Conjugate Gradient Methods for Unconstrained Optimization*, Springer International Publishing, 2020.

[7] Bongartz, I., Conn, A.R., Gould, N.I.M., Toint, P.L., *CUTE: Constrained and unconstrained testing environments*, ACM Trans. Math. Softw., **21**(1995), no. 1, 123-160.

[8] Dai, Y.H., Yuan, Y., *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM J. Optim., **10**(1999), no. 1, 177-182.

[9] Dai, Y.H., Yuan, Y., *An efficient hybrid conjugate gradient method for unconstrained optimization*, Ann. Oper. Res., **103**(2001), no. 1, 33-47.

[10] Daniel, J.W., *The conjugate gradient method for linear and nonlinear operator equations*, SIAM J. Optim., **10**(1967), no. 1, 10-26.

[11] Delladji, S., Belloufi, M., Sellami, B., *Behavior of the combination of PRP and HZ methods for unconstrained optimization*, Numer. Algebra Control Optim., **11**(2021), no. 3, 377-389.

[12] Djordjević, S.S., *New hybrid conjugate gradient method as a convex combination of LS and FR methods*, Acta Math. Sci. Ser. B (Engl. Ed.), **39**(2019), no 1, 214-228.

[13] Fletcher, R., *Practical Methods of Optimization, vol. 1: Unconstrained Optimization*, John Wiley & Sons, New York, 1987.

[14] Gazi, S., Khatab, H., *New iterative conjugate gradient method for nonlinear unconstrained optimization using homptopy technique*, IOSR Journal of Mathmatics, (2014), 78-82.

[15] Hager, W.W., Zhang, H., *A new conjugate gradient method with guaranteed descent and an effcient line search*, SIAM J. Optim., **16**(2005), no. 1, 170-192.

[16] Hager, W.W., Zhang, H., *A survey of nonlinear conjugate gradient methods*, Pac. J. Optim., **2**(2006), no. 1, 35-58.

[17] Hestenes, M., *Methods of conjugate gradients for solving linear systems*, Research Journal of the National Bureau of Standards, **49**(1952), no. 22, 409-436.

[18] Liu, Y., Storey, C., *Efficient generalized conjugate gradient algorithms, Part 1*, J. Optim. Theory Appl., **69**(1991), no. 1, 129-137.

[19] Polak, E., Ribiere, G., *Note sur la convergence des méthodes de directions conjugées*, ESAIM: Math. Model. Numer. Anal., **3**(1969), no. R1, 35-43.

[20] Polyak, B.T., *The conjugate gradient method in extremal problems*, Comput. Math. Math. Phys., **9**(1969), no. 4, 94-112.

[21] Rivaie, M., Mustafa, M., Abashar, A., *A new class of nonlinear conjugate gradient coefficients with exact and inexact line searches*, Appl. Math. Comput., **268**(2015), 1152-1163.

[22] Rivaie, M., Mustafa, M., June, L.W., Mohd, I., *A new class of nonlinear conjugate gradient coefficient with global convergence properties*, Appl. Math. Comput., **218**(2012), no. 22, 11323-11332.

[23] Shanno, D.F., *Conjugate gradient methods with inexact searches*, Math. Oper. Res., **3**(1978), no. 3, 244-256.

[24] Touati-Ahmed, D., Storey, C., *Efficient hybrid conjugate gradient technique*, J. Optim. Theory Appl., **64**(1990), no. 2, 379-397.

[25] Wang, L.J., Xu, L., Xie, Y.X., Du, Y.X., Han, X., *A new hybrid conjugate gradient method for dynamic force reconstruction*, Advances in Mechanical Engineering, **11**(2019), no. 1, 1-21.

[26] Zhang, L., Zhou, W., *Two descent hybrid conjugate gradient methodq for optimization*, J. Comput. Appl. Math., **216**(2008), no. 1, 251-264.

[27] Zhifeng, D., *Comments on hybrid conjugate gradient algorithm for unconstrained optimization*, J. Optim. Theory Appl., **175**(2017), no. 1, 286-291.

[28] Zoutendijk, G., *Nonlinear programming, computational methods*, Integer and Nonlinear Programming (J. Abadie, ed.), North-Holland, Amsterdam, (1970), 37-86.

Ghania Hadji
Faculty of Sciences, Department of Mathematics,
Badji Mokhtar University, B.P. 12 Annaba 23000, Algeria
Faculty of Science and Technology,
Department of Mathematics and Informatics,
Mohamed Cherif Messaadia University, P.O.Box 1553,
Souk Ahras 41000, Algeria
e-mail: g.hadji@univ-soukahras.dz

Yamina Laskri
ESTI, Faculty of Sciences, Department of Mathematics,
Badji Mokhtar University, B.P. 12 Annaba 23000, Algeria
e-mail: yamina.laskri@univ-annaba.org

Tahar Bechouat
Faculty of Science and Technology,
Department of Mathematics and Informatics,
Mohamed Cherif Messaadia University, P.O.Box 1553,
Souk Ahras 41000, Algeria
e-mail: t.bachaouette@univ-soukahras.dz

Rachid Benzine
Laboratory LANOS, Faculty of Sciences,
Department of Mathematics, Badji Mokhtar University,
B.P. 12 Annaba 23000, Algeria
e-mail: rabenzine@yahoo.fr