STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **16**, December 2026, pp 5572–5590.
Published online in International Academic Press (www.IAPress.org)

A New Efficient Conjugate Gradient Scheme for Solving Unconstrained Nonlinear Problems

Ibtissam Kermiche[1], Amina Boumediene[1], Alaa Luqman Ibrahim[2], Amira Hamdi[1,*]

[1] *Laboratory Informatics and Mathematics (LIM), University of Souk Ahras, B.P. 1553, Souk Ahras, 41000, Algeria*
[2] *Department of Mathematics, College of Science, University of Zakho, Zakho, Kurdistan Region, Iraq*

Abstract This paper introduces a new hybrid Conjugate Gradient (CG) algorithm, referred to as KBH, for large-scale unconstrained optimization. The proposed method constructs the search direction using an adaptively weighted convex combination of the conjugacy parameters from Conjugate Descent (CD), Polak–Ribière–Polyak (PRP), and Fletcher–Reeves (FR) method. This formulation seeks to merge the reliable descent characteristics of CD and FR with the strong practical efficiency often observed for PRP, while maintaining numerical stability. The adaptive weights are obtained analytically from the classical conjugacy condition, yielding a true three-term hybrid approach with data-dependent coefficients. Assuming standard smoothness conditions and employing a strong Wolfe line search, the sufficient descent property of the resulting directions is shown, and global convergence of the method is established. Moreover, safeguarding, truncation, and restart strategies are added to enhance robustness in practical use. Comprehensive numerical tests on a wide set of large-scale benchmark problems show that KBH achieves competitive results relative to several classical CG methods in terms of iteration numbers, function evaluations, and CPU time.

Keywords Hybrid conjugate gradient; Adaptive convex combination; Sufficient descent condition; Strong Wolfe line search; Large-scale unconstrained optimization; Global convergence

AMS 2010 subject classifications 90C52, 65K05, 49M37

DOI: 10.19139/soic-2310-5070-4207

1. Introduction

Unconstrained optimization problems of the form

$$(P) : \min_{x \in \mathbb{R}^n} f(x), \tag{1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable. The Conjugate Gradient (CG) methods are widely used for solving large-scale unconstrained optimization problems due to their low memory requirements [7]and good convergence properties. The general form of the nonlinear CG iterative method generates a sequence $\{x_k\}$ according to

$$x_{k+1} = x_k + \alpha_k d_k, \tag{2}$$

where x_k is the current iterate, $\alpha_k > 0$ is the step size which is obtained by some line search [30] we usually use the inexact one [33] [37]. The exact line search is given as follows:

$$f(x_k + \alpha_k d_k) = \min_{\alpha > 0} f(x_k + \alpha d_k). \tag{3}$$

*Correspondence to: Amira Hamdi (Email: amira.hamdi@univ-soukahras.dz). Laboratory Informatics and Mathematics (LIM), University of Souk Ahras, Algeria.

ISSN 2310-5070 (online) ISSN 2311-004X (print)
Copyright © 2026 International Academic Press

Since the exact line search is expensive in practice, in this paper the method is implemented and analysed under the strong Wolfe line search [34, 35] defined by

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \sigma_1 \alpha_k g_k^T d_k \tag{4}$$

$$|g_{k+1}^T d_k| \leq -\sigma_2 g_k^T d_k \tag{5}$$

where $0 < \sigma_1 < \sigma_2 < 1$ (the convergence analysis of Sections 2–3 requires $\sigma_2 < \frac{5}{12}$). The search direction $\{d_k\}$ is defined by:

$$d_0 = -g_0, \quad d_{k+1} = -g_{k+1} + \beta_k d_k, \quad k \geq 0,$$

With $g_k = \nabla f(x_k)$, $x_k \in \mathbb{R}^n$, where β_k is a scalar parameter that specifies the particular CG method. Several classical CG variants are distinguished by their choice of β_k. Table 1 summarizes the formulas for the HS, HZ, DY, and LS methods, together with the FR, PRP and CD formulas used in this paper:

Method	Formula for β_k
Hestenes–Stiefel (HS)	$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}$ [23]
Hager–Zhang (HZ)	$\beta_k^{HZ} = \frac{1}{d_k^T y_k}\left(y_k - 2d_k \frac{\|y_k\|^2}{d_k^T y_k}\right)^T g_{k+1}$ [16]
Dai–Yuan (DY)	$\beta_k^{DY} = \frac{\|g_{k+1}\|^2}{y_k^T d_k}$ [9]
Liu–Storey (LS)	$\beta_k^{LS} = -\frac{g_{k+1}^T y_k}{g_k^T d_k}$ [25]
Fletcher–Reeves (FR)	$\beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}$ [14]
Polak–Ribière–Polyak (PRP)	$\beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}$ [28, 29]
Conjugate Descent (CD)	$\beta_k^{CD} = -\frac{\|g_{k+1}\|^2}{g_k^T d_k}$ [13]

Table 1. Classical Conjugate Gradient Formulas for β_k

Here $g_k = \nabla f(x_k)$,

$$y_k = g_{k+1} - g_k,$$

and let $\|\cdot\|$ represent the Euclidean norm. Conjugate Gradient methods have a rich history of development and application. Classical methods such as Fletcher–Reeves (FR) [14], Polak–Ribière–Polyak (PRP) [28], Conjugate Descent (CD) [13], Dai–Yuan (DY) [9], and Hestenes–Stiefel (HS) [23] have been extensively studied to understand their convergence properties and numerical behavior. Foundational works by Zoutendijk (1970) [40], Al-Baali (1985) [2], and Touati-Ahmed & Storey (1990) [31] established theoretical frameworks for analyzing convergence under line search conditions, particularly the strong Wolfe conditions [34] [35], highlighting the strengths and limitations of each classical CG formula.

Despite their effectiveness, each classical method has limitations: FR is theoretically stable but slow in practice; PRP often converges faster but may fail to ensure descent; and CD is theoretically strong but sensitive to line search. These limitations motivated the development of **hybrid CG methods**, which combine the advantages of multiple classical methods in a single framework. Touati-Ahmed & Storey (1990) [31] developed early hybrid formulas to improve both stability and convergence. Later, Andrei (2009) [5] introduced a two-term hybrid combining FR and PRP with a weighting factor θ_k:

$$\beta_k^{hyb} = \theta_k \beta_k^{FR} + (1 - \theta_k)\beta_k^{PRP}, \quad 0 \leq \theta_k \leq 1,$$

With growing interest in hybridization, three-term hybrid formulas emerged. Mohammed Zaki et al. (2025) [38] combined DY–HS–HZ:

$$\beta_k^{hyb} = \alpha_k \beta_k^{DY} + \theta_k \beta_k^{HS} + (1 - \alpha_k - \theta_k)\beta_k^{HZ}, \quad 0 \le \alpha_k, \theta_k \le 1,$$

And Wang & Gao (2025) [32]proposed FR–CD–DY to balance theoretical stability with numerical speed:

$$\beta_k^{hyb} = \alpha_k \beta_k^{FR} + \theta_k \beta_k^{CD} + (1 - \alpha_k - \theta_k)\beta_k^{DY}, \quad 0 \le \alpha_k, \theta_k \le 1.$$

In this journal, Adeleke et al. [43] constructed a family of hybrid CG methods from ten modified CG methods, with descent and global convergence under the strong Wolfe line search. Mellal and Sellami [44] combined the Al-Bayati–Al-Assady and Wei–Yao–Liu methods through a convex parameter determined from the conjugacy condition. Mehamdia et al. [45] proposed an improved hybrid nonlinear CG method with an application to image restoration.

More recently, Hamdi and Ibrahim (2026) [22] introduced a DY–HS–LS hybrid conjugate gradient coefficient constructed through a convex combination strategy. The proposed method combines the robustness of the Dai–Yuan formula, the practical efficiency of the Hestenes–Stiefel parameter, and the favorable numerical behavior of the Liu–Storey method. Theoretical analysis established sufficient descent and global convergence, while numerical results showed its efficiency on a collection of unconstrained optimization test problems.

This rapid evolution of hybrid formulas reflects the ongoing need to improve CG algorithms beyond the limitations of classical methods. Recent studies have shown that hybrid strategies achieve better convergence, provide a balance between theoretical stability and numerical performance, and enhance the efficiency of solutions for large-scale unconstrained optimization problems.

In this context, our proposed approach focuses on a novel three-term hybrid method that combines **Conjugate Descent (CD), Polak–Ribière–Polyak (PRP), and Fletcher–Reeves (FR)**. The method is designed to leverage the complementary strengths of these classical techniques: FR contributes theoretical stability and global convergence properties, PRP provides fast practical convergence, and CD enhances the descent property. The hybrid conjugate gradient coefficient is defined as:

$$\beta_k^{hyb} = \delta_k \beta_k^{CD} + \gamma_k \beta_k^{PRP} + (1 - \delta_k - \gamma_k)\beta_k^{FR}, \quad 0 \le \delta_k, \gamma_k \le 1,$$

where δ_k and γ_k are parameters that control the contribution of each component at every iteration. By appropriately tuning these weights, the method aims to achieve a **balanced performance**, combining the descent guarantee, global convergence, and accelerated numerical efficiency in solving large-scale unconstrained optimization problems.

The choice of the three components is motivated by their complementary theoretical and numerical properties. The FR method has strong global convergence properties under the strong Wolfe line search [2], but it may take very small steps and perform poorly in practice. The PRP method has an automatic restart property: when a small step is generated, $g_{k+1} \approx g_k$ and $\beta_k^{PRP} \approx 0$, so that the search direction is close to the steepest descent direction and jamming is avoided [17]. The CD method generates descent directions under the strong Wolfe line search for any $\sigma_2 < 1$ [13, 17]. Existing three-term hybrids combine other triples of classical parameters, for example DY–HS–HZ [38], FR–CD–DY [32], FR–PRP–DY [20], LS–FR–PRP [21], DY–CD–HS [18] and DY–HS–LS [22]. To the best of our knowledge, the combination of CD, PRP and FR has not been studied, although it gathers in a single parameter the descent property of CD, the restart property of PRP and the convergence properties of FR. Moreover, several convergence analyses of convex-combination hybrids assume a positive lower bound on the step size or the descent property of each component method under the strong Wolfe line search.

The main contributions of this paper are as follows.

- A new three-term hybrid CG parameter is proposed as a convex combination of the CD, PRP and FR parameters. The CD weight δ_k is computed from the conjugacy condition, with an explicit adjustment rule that keeps the combination convex.
- The PRP component is controlled by the Powell restart criterion, and the resulting parameter is kept in the Fletcher–Reeves envelope. With these safeguards, the sufficient descent property is proved under the strong Wolfe line search with $\sigma_2 < \frac{5}{12}$, without assuming the descent of the PRP component.

- Global convergence is proved for general (not necessarily convex) smooth functions, without assuming a positive lower bound on the step size.
- The numerical performance of the method is compared with classical CG methods on a set of large-scale test problems.

2. Proposed Method and Algorithm

In this work, we introduce a new combination of the CD, PRP, and FR conjugate gradient algorithms. The conjugate gradient parameter is defined as follows:

$$\beta_k^{\text{New}} = \delta_k \beta_k^{\text{CD}} + \gamma_k \beta_k^{\text{PRP}} + (1 - \delta_k - \gamma_k)\beta_k^{\text{FR}}, \qquad \delta_k, \gamma_k \geq 0, \;\; \delta_k + \gamma_k \leq 1, \tag{6}$$

where

$$\beta_k^{CD} = \frac{\|g_{k+1}\|^2}{-g_k^T d_k}, \qquad \beta_k^{PRP} = \frac{g_{k+1}^T y_k}{\|g_k\|^2}, \qquad \beta_k^{FR} = \frac{\|g_{k+1}\|^2}{\|g_k\|^2}, \qquad y_k = g_{k+1} - g_k. \tag{7}$$

Consequently, the search direction is given by:

$$d_0 = -g_0, \qquad d_{k+1} = -g_{k+1} + \beta_k^{\text{New}} d_k, \quad k \geq 0. \tag{8}$$

The parameters δ_k and γ_k in (6) will be chosen in a manner that will be specified later. In particular:

- When $\delta_k = 1$ and $\gamma_k = 0$, then $\beta_k^{\text{New}} = \beta_k^{CD}$.
- When $\delta_k = 0$ and $\gamma_k = 1$, then $\beta_k^{\text{New}} = \beta_k^{PRP}$.
- When $\delta_k = 0$ and $\gamma_k = 0$, then $\beta_k^{\text{New}} = \beta_k^{FR}$.
- When $\delta_k = 0$ and $0 < \gamma_k < 1$, then $\beta_k^{\text{New}} = \gamma_k \beta_k^{PRP} + (1 - \gamma_k)\beta_k^{FR}$, i.e., a convex combination of β_k^{PRP} and β_k^{FR}.
- When $\gamma_k = 0$ and $0 < \delta_k < 1$, then $\beta_k^{\text{New}} = \delta_k \beta_k^{CD} + (1 - \delta_k)\beta_k^{FR}$, i.e., a convex combination of β_k^{CD} and β_k^{FR}.
- When $1 - \delta_k - \gamma_k = 0$, with $0 < \delta_k, \gamma_k < 1$, then $\beta_k^{\text{New}} = \delta_k \beta_k^{CD} + (1 - \delta_k)\beta_k^{PRP}$, i.e., a convex combination of β_k^{CD} and β_k^{PRP}.

Finally, when $\delta_k \in (0, 1)$, $\gamma_k \in (0, 1)$, and $0 < \delta_k + \gamma_k < 1$, we obtain a new hybrid CG method as a convex combination of CD, PRP, and FR. From (6) and (8), the search direction can be explicitly written as:

$$d_{k+1} = -g_{k+1} + \delta_k \frac{\|g_{k+1}\|^2}{-g_k^T d_k} d_k + \gamma_k \frac{g_{k+1}^T y_k}{\|g_k\|^2} d_k + (1 - \delta_k - \gamma_k) \frac{\|g_{k+1}\|^2}{\|g_k\|^2} d_k, \qquad k \geq 0. \tag{9}$$

As in [20], where one hybridization parameter is fixed and the other one is determined from the conjugacy condition, the parameter γ_k is kept fixed:

$$\gamma_k \equiv \gamma \in [0, 1). \tag{10}$$

To determine the parameter δ_k, we employ the classical conjugacy condition, i.e., $d_{k+1}^T y_k = 0$ (see, e.g., [44]). This leads to the following lemma.

Lemma 2.1
Consider the hybrid conjugate gradient method defined by $d_{k+1} = -g_{k+1} + \beta_k^{\text{New}} d_k$, where β_k^{New} is given by (6) with γ_k given by (10). Assume that $g_k^T d_k < 0$, $d_k^T y_k \neq 0$ and $\|g_k\|^2 + g_k^T d_k \neq 0$. If the conjugacy condition $d_{k+1}^T y_k = 0$ is satisfied, then the parameter δ_k is given by:

$$\delta_k = -\frac{\left(g_{k+1}^T y_k \|g_k\|^2 - \|g_{k+1}\|^2 d_k^T y_k - \gamma_k \left(g_{k+1}^T y_k - \|g_{k+1}\|^2\right) d_k^T y_k\right) d_k^T g_k}{\left(\|g_{k+1}\|^2 d_k^T g_k + \|g_{k+1}\|^2 \|g_k\|^2\right) d_k^T y_k}. \tag{11}$$

Proof
Multiplying the search direction formula $d_{k+1} = -g_{k+1} + \beta_k^{\text{New}} d_k$ by y_k^T from the left, we obtain

$$d_{k+1}^T y_k = -g_{k+1}^T y_k + \beta_k^{\text{New}}\, d_k^T y_k. \tag{12}$$

Using the conjugacy condition $d_{k+1}^T y_k = 0$, we get

$$\beta_k^{\text{New}} = \frac{g_{k+1}^T y_k}{d_k^T y_k}. \tag{13}$$

Writing (6) as $\beta_k^{\text{New}} = \beta_k^{FR} + \gamma_k(\beta_k^{PRP} - \beta_k^{FR}) + \delta_k(\beta_k^{CD} - \beta_k^{FR})$, relation (13) becomes

$$\delta_k \left(\beta_k^{CD} - \beta_k^{FR}\right) = \frac{g_{k+1}^T y_k}{d_k^T y_k} - \beta_k^{FR} - \gamma_k \left(\beta_k^{PRP} - \beta_k^{FR}\right). \tag{14}$$

From (7),

$$\beta_k^{CD} - \beta_k^{FR} = -\frac{\|g_{k+1}\|^2 \left(\|g_k\|^2 + d_k^T g_k\right)}{\|g_k\|^2\, d_k^T g_k}, \qquad \beta_k^{PRP} - \beta_k^{FR} = \frac{g_{k+1}^T y_k - \|g_{k+1}\|^2}{\|g_k\|^2}. \tag{15}$$

Substituting (15) into (14) and multiplying both sides by $-\|g_k\|^2\, d_k^T g_k\, d_k^T y_k$, we get

$$\delta_k \|g_{k+1}\|^2 \left(\|g_k\|^2 + d_k^T g_k\right) d_k^T y_k = -\left(g_{k+1}^T y_k \|g_k\|^2 - \|g_{k+1}\|^2 d_k^T y_k - \gamma_k \left(g_{k+1}^T y_k - \|g_{k+1}\|^2\right) d_k^T y_k\right) d_k^T g_k. \tag{16}$$

Dividing (16) by $\|g_{k+1}\|^2 \left(\|g_k\|^2 + d_k^T g_k\right) d_k^T y_k \neq 0$ gives (11), which completes the proof. □

In (11), by the strong Wolfe condition (5), $d_k^T y_k = g_{k+1}^T d_k - g_k^T d_k \geq \sigma_2 g_k^T d_k - g_k^T d_k = (1-\sigma_2)\left(-g_k^T d_k\right) > 0$, and if $\|g_k\|^2 + g_k^T d_k = 0$, then $\beta_k^{CD} = \beta_k^{FR}$ by (15), so β_k^{New} does not depend on δ_k and we set $\delta_k = 0$.

The parameter δ_k given by (11) can take values outside the interval $[0, 1-\gamma_k]$. To ensure a valid convex combination in (6), the following adjustment rule is applied, with γ_k kept fixed:

$$\delta_k := \min\left\{\max\left\{\delta_k,\, 0\right\},\, 1-\gamma_k\right\}. \tag{17}$$

Hence $\delta_k < 0$ is replaced by $\delta_k = 0$, $\delta_k > 1-\gamma_k$ is replaced by $\delta_k = 1-\gamma_k$, and in all cases $\delta_k, \gamma_k \geq 0$ and $\delta_k + \gamma_k \leq 1$.

Finally, as in the hybrid methods of Touati-Ahmed and Storey [31] and Gilbert and Nocedal [41], the parameter is kept in the Fletcher–Reeves envelope:

$$\beta_k^{\text{KBH}} = \max\left\{-\beta_k^{FR},\, \min\left\{\beta_k^{\text{New}},\, \beta_k^{FR}\right\}\right\}, \qquad \text{so that} \qquad \left|\beta_k^{\text{KBH}}\right| \leq \beta_k^{FR}. \tag{18}$$

Algorithm 1 KBH Algorithm

Step 0: Initialization. Select $x_0 \in \mathbb{R}^n$, $k = 0$, $0 < \sigma_1 < \sigma_2 < \frac{5}{12}$, ϵ and $\gamma \in [0, 1)$.
Compute $f(x_0)$ and g_0. Consider $d_0 = -g_0$. Set the initial guess $\alpha_0 = 1/\|g_0\|$.
Step 1: If $\|g_k\| < \epsilon$, then Stop. Otherwise, go to the next step.
Step 2: Compute α_k by the strong Wolfe line search conditions (4) and (5).
Step 3: Let $x_{k+1} = x_k + \alpha_k d_k$, $g_{k+1} = g(x_{k+1})$. Compute $y_k = g_{k+1} - g_k$.
Step 4: Compute δ_k as in (11) with $\gamma_k = \gamma$, and apply the adjustment rule (17).
Step 5: Compute β_k^{New} as in (6) and β_k^{KBH} as in (18).
Step 6: Generate $d = -g_{k+1} + \beta_k^{\mathrm{KBH}} d_k$.
If the restart criterion of the Powell condition

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

is satisfied, then set $d_{k+1} = -g_{k+1}$, otherwise define $d_{k+1} = d$.
Compute the initial guess

$$\alpha_{k+1} = \alpha_k \frac{\|d_k\|}{\|d_{k+1}\|}.$$

Step 7: Set $k = k + 1$ and go to **Step 1**.

Computational cost. Algorithm 1 only stores a few vectors of $\mathbb{R}^n$, namely x_k, g_k, g_{k+1}, d_k and y_k, so its memory requirement is $O(n)$, as for the classical CG methods. At each iteration, apart from the function and gradient evaluations required by the line search, the method computes the inner products $\|g_{k+1}\|^2$, $g_{k+1}^T g_k$, $g_{k+1}^T y_k$, $g_{k+1}^T d_k$ and $d_k^T y_k$, while $\|g_k\|^2$ and $g_k^T d_k$ are available from the previous iteration. Each inner product costs $O(n)$ operations. The quantities β_k^{CD}, β_k^{PRP}, β_k^{FR}, δ_k in (11), the adjustment (17) and the bound (18) are then obtained with $O(1)$ scalar operations, and the update of d_{k+1} costs $O(n)$ operations. Hence the cost per iteration of the KBH method is $O(n)$, of the same order as that of the CD, PRP and FR methods, which makes it suitable for large-scale problems.

2.1. Sufficient Descent Condition

In this study, we aim to establish that the proposed method satisfies the sufficient descent condition, which is a key property for proving global convergence. To do so, we adopt the following assumptions:

Assumption 1
The level set

$$S = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$$

is bounded. There exists a constant $B > 0$ such that

$$\|x\| \leq B, \quad \forall x \in S.$$

Assumption 2
In a neighborhood $\mathcal{N}$ of S, the function f is continuously differentiable, and its gradient $\nabla f(x)$ is Lipschitz continuous. There exists a constant $L > 0$ such that

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \forall x, y \in \mathcal{N}.$$

Under these assumptions, there exists a constant $\Gamma \geq 0$ such that

$$\|\nabla f(x)\| \leq \Gamma, \quad \forall x \in S.$$

Throughout, $\sigma := \sigma_2 < \frac{5}{12}$ and

$$c := \frac{1 - 2.4\,\sigma}{1 - 1.2\,\sigma} > 0, \qquad M := \frac{1}{1 - 1.2\,\sigma} > 1. \tag{19}$$

Note that $1 - 1.2\sigma M = c$ and $1 + 1.2\sigma M = M$. The following theorem establishes that the search directions generated by the proposed method satisfy the sufficient descent condition.

Theorem 2.2
Assume that the sequences $\{g_k\}$, $\{d_k\}$ are produced by the KBH algorithm. Then, the generated search direction d_k fulfils the sufficient descent property:

$$g_k^T d_k \leq -c\|g_k\|^2, \quad \forall k \geq 0, \tag{20}$$

and moreover

$$-g_k^T d_k \leq M\,\|g_k\|^2, \quad \forall k \geq 0. \tag{21}$$

Proof
The argument proceeds by mathematical induction. First, we verify the result for $k = 0$. Since $d_0 = -g_0$, it immediately follows that

$$g_0^T d_0 = -\|g_0\|^2,$$

so (20) and (21) hold because $c < 1 < M$. Next, suppose that (20) and (21) hold for a certain index $k \geq 0$. We shall prove that they remain valid for the subsequent iterate. By (20), $g_k^T d_k < 0$, and the strong Wolfe condition (5) gives

$$\left|g_{k+1}^T d_k\right| \leq \sigma\left(-g_k^T d_k\right). \tag{22}$$

If the Powell restart is used, then $g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2$ and both inequalities hold. Otherwise $|g_{k+1}^T g_k| < 0.2\|g_{k+1}\|^2$ by the Powell restart criterion [42] (Step 6 of Algorithm 1), and since $g_{k+1}^T y_k = \|g_{k+1}\|^2 - g_{k+1}^T g_k$, we get

$$\left|\beta_k^{PRP}\right| = \frac{\left|\|g_{k+1}\|^2 - g_{k+1}^T g_k\right|}{\|g_k\|^2} \leq 1.2\,\frac{\|g_{k+1}\|^2}{\|g_k\|^2} = 1.2\,\beta_k^{FR}. \tag{23}$$

If the truncation (18) is active, then $\beta_k^{\mathrm{KBH}} = \pm\beta_k^{FR}$ and the result follows exactly as in Case 3 below. Otherwise $\beta_k^{\mathrm{KBH}} = \beta_k^{\mathrm{New}}$.

From the KBH update, the new search direction is given by

$$d_{k+1} = -g_{k+1} + \beta_k^{\mathrm{New}} d_k, \qquad \beta_k^{\mathrm{New}} = \delta_k\beta_k^{\mathrm{CD}} + \gamma_k\beta_k^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k)\beta_k^{\mathrm{FR}}.$$

We can rewrite d_{k+1} as

$$d_{k+1} = \delta_k\big(-g_{k+1} + \beta_k^{\mathrm{CD}} d_k\big) + \gamma_k\big(-g_{k+1} + \beta_k^{\mathrm{PRP}} d_k\big) + (1 - \delta_k - \gamma_k)\big(-g_{k+1} + \beta_k^{\mathrm{FR}} d_k\big).$$

Defining

$$d_{k+1}^{\mathrm{CD}} = -g_{k+1} + \beta_k^{\mathrm{CD}} d_k, \quad d_{k+1}^{\mathrm{PRP}} = -g_{k+1} + \beta_k^{\mathrm{PRP}} d_k, \quad d_{k+1}^{\mathrm{FR}} = -g_{k+1} + \beta_k^{\mathrm{FR}} d_k,$$

we obtain the convex combination form

$$d_{k+1} = \delta_k d_{k+1}^{\mathrm{CD}} + \gamma_k d_{k+1}^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) d_{k+1}^{\mathrm{FR}}. \tag{24}$$

Multiplying (24) by g_{k+1}^T from the left yields

$$g_{k+1}^T d_{k+1} = \delta_k g_{k+1}^T d_{k+1}^{\mathrm{CD}} + \gamma_k g_{k+1}^T d_{k+1}^{\mathrm{PRP}} + (1 - \delta_k - \gamma_k) g_{k+1}^T d_{k+1}^{\mathrm{FR}}. \tag{25}$$

This decomposition allows analysis of the sufficient descent property by considering the contributions of each component separately. Seven cases have been analyzed and proven accordingly.

Case 1. When $\delta_k = 1$ and $\gamma_k = 0$, then

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

Multiplying the CD relation by g_{k+1}^T from the left and substituting the expression of β_k^{CD} from (7) gives

$$g_{k+1}^T d_{k+1}^{CD} = -\|g_{k+1}\|^2 \Big(1 + \frac{g_{k+1}^T d_k}{g_k^T d_k}\Big). \tag{26}$$

Since $g_k^T d_k < 0$, dividing (22) by $-g_k^T d_k > 0$ gives

$$-\sigma \le \frac{g_{k+1}^T d_k}{g_k^T d_k} \le \sigma. \tag{27}$$

Substituting (27) into (26) yields

$$-(1+\sigma)\|g_{k+1}\|^2 \;\le\; g_{k+1}^T d_{k+1}^{CD} \;\le\; -(1-\sigma)\|g_{k+1}\|^2. \tag{28}$$

Therefore, the CD direction satisfies the sufficient descent condition:

$$g_{k+1}^T d_{k+1}^{CD} \le -C_1 \|g_{k+1}\|^2, \qquad C_1 = 1 - \sigma > 0.$$

Case 2. When $\delta_k = 0$ and $\gamma_k = 1$, relation (25) reduces to

$$g_{k+1}^T d_{k+1} = g_{k+1}^T d_{k+1}^{PRP}.$$

In Algorithm 1, β_k^{PRP} is used only when the Powell restart criterion [42] is not satisfied, so that the bound (23) holds. By definition, the PRP direction satisfies

$$g_{k+1}^T d_{k+1}^{PRP} = -\|g_{k+1}\|^2 + \beta_k^{PRP}\, g_{k+1}^T d_k. \tag{29}$$

Using (23), (22) and the induction hypothesis (21), we obtain

$$\big|\beta_k^{PRP}\, g_{k+1}^T d_k\big| \le 1.2\, \frac{\|g_{k+1}\|^2}{\|g_k\|^2}\, \sigma \left(-g_k^T d_k\right) \le 1.2\, \sigma M\, \|g_{k+1}\|^2. \tag{30}$$

Substituting (30) into (29), and using $1 - 1.2\sigma M = c$ and $1 + 1.2\sigma M = M$, it follows that

$$-M\|g_{k+1}\|^2 \;\le\; g_{k+1}^T d_{k+1}^{PRP} \;\le\; -c\,\|g_{k+1}\|^2. \tag{31}$$

Hence, the sufficient descent condition holds with

$$C_2 = c > 0.$$

Case 3. When $\delta_k = 0$ and $\gamma_k = 0$, relation (25) simplifies to

$$g_{k+1}^T d_{k+1} = g_{k+1}^T d_{k+1}^{FR}.$$

Following Al-Baali [2], we have $g_{k+1}^T d_{k+1}^{FR} = -\|g_{k+1}\|^2 + \beta_k^{FR} g_{k+1}^T d_k$ and, by (22) and (21),

$$\big|\beta_k^{FR}\, g_{k+1}^T d_k\big| \le \frac{\|g_{k+1}\|^2}{\|g_k\|^2}\, \sigma \left(-g_k^T d_k\right) \le \sigma M\, \|g_{k+1}\|^2. \tag{32}$$

The same bound holds with β_k^{FR} replaced by $-\beta_k^{FR}$, which covers the case where the truncation (18) is active. Hence, since $1 + \sigma M \le M$ and $1 - \sigma M \ge c$,

$$-M\|g_{k+1}\|^2 \;\le\; g_{k+1}^T d_{k+1}^{FR} \;\le\; -C_3\,\|g_{k+1}\|^2, \qquad C_3 = 1 - \sigma M \ge c > 0. \tag{33}$$

Note that $C_1 = 1-\sigma \geq c$ and $1+\sigma \leq M$, because $(1-\sigma)(1-1.2\sigma) \geq 1-2.4\sigma$ and $(1+\sigma)(1-1.2\sigma) \leq 1$. Hence, by (28), (31) and (33), each of the three components satisfies

$$-M\|g_{k+1}\|^2 \;\leq\; g_{k+1}^T d_{k+1}^i \;\leq\; -c\,\|g_{k+1}\|^2, \qquad i \in \{CD,\ PRP,\ FR\}. \tag{34}$$

Case 4. Assume that $\delta_k = 0$ and $0 < \gamma_k < 1$. In this situation (the convex combination of FR and PRP, see also Djordjević [10]),

$$g_{k+1}^T d_{k+1} = \gamma_k g_{k+1}^T d_{k+1}^{\text{PRP}} + (1-\gamma_k) g_{k+1}^T d_{k+1}^{\text{FR}} \leq -(C_2\gamma_k + C_3(1-\gamma_k))\|g_{k+1}\|^2 \leq -c\,\|g_{k+1}\|^2,$$

and, by (34), $g_{k+1}^T d_{k+1} \geq -M\|g_{k+1}\|^2$.

Case 5. Assume that $\gamma_k = 0$ and $0 < \delta_k < 1$. By convexity of the combination, we have

$$g_{k+1}^\top d_{k+1} = \delta_k\, g_{k+1}^\top d_{k+1}^{\text{CD}} + (1-\delta_k)\, g_{k+1}^\top d_{k+1}^{\text{FR}} \leq -\left(C_1\delta_k + C_3(1-\delta_k)\right)\|g_{k+1}\|^2 \leq -c\,\|g_{k+1}\|^2,$$

since $C_1, C_3 \geq c$; and, by (34), $g_{k+1}^T d_{k+1} \geq -M\|g_{k+1}\|^2$.

Case 6. When $1-\delta_k-\gamma_k = 0$, with $0 < \delta_k, \gamma_k < 1$, then $\gamma_k = 1-\delta_k$ and, by the convexity of the combination,

$$g_{k+1}^\top d_{k+1} = \delta_k\, g_{k+1}^\top d_{k+1}^{\text{CD}} + (1-\delta_k)\, g_{k+1}^\top d_{k+1}^{\text{PRP}} \leq -\left(C_1\delta_k + C_2(1-\delta_k)\right)\|g_{k+1}\|^2 \leq -c\,\|g_{k+1}\|^2,$$

and, by (34), $g_{k+1}^T d_{k+1} \geq -M\|g_{k+1}\|^2$.

Case 7. Finally, assume $0 < \delta_k < 1$, $0 < \gamma_k < 1$, and $0 < \delta_k + \gamma_k < 1$. From (25) and (34),

$$\begin{aligned} g_{k+1}^T d_{k+1} &= \delta_k g_{k+1}^T d_{k+1}^{CD} + \gamma_k g_{k+1}^T d_{k+1}^{PRP} + (1-\delta_k-\gamma_k) g_{k+1}^T d_{k+1}^{FR} \\ &\leq -\Big(\delta_k C_1 + \gamma_k C_2 + (1-\delta_k-\gamma_k)C_3\Big)\|g_{k+1}\|^2 \;\leq\; -c\,\|g_{k+1}\|^2, \end{aligned} \tag{35}$$

because $C_1, C_2, C_3 \geq c$ and the weights are nonnegative with sum one. In the same way, $g_{k+1}^T d_{k+1} \geq -M\|g_{k+1}\|^2$.

In all cases, (20) and (21) hold for $k+1$, which completes the induction. □

3. Convergence Analysis

We analyze the global convergence of the proposed hybrid **KBH** by using Assumptions 1 and 2.

3.1. Zoutendijk Condition

The Zoutendijk condition [40] is widely used as a fundamental tool to prove the global convergence of conjugate gradient (CG) methods. In addition, the following lemma confirms that this condition holds for the proposed method under the strong Wolfe line search criteria given in equations (4) and (5).

Lemma 3.1
Suppose Assumptions 1 and 2 hold. Consider the iterate sequence (2) where d_k is a descent direction and α_k is determined by the strong Wolfe line search (4)–(5). Then the Zoutendijk condition

$$\sum_{k\geq 0} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty \tag{36}$$

holds.

The following theorem establishes the global convergence of the **KBH** method.

Theorem 3.2
Suppose Assumptions 1 and 2 hold. Let $\{x_k\}$ be generated by the **KBH** algorithm. Then

$$\liminf_{k\to\infty} \|g_k\| = 0. \tag{37}$$

Proof
We prove this theorem by contradiction. Assume, for the sake of contradiction, that formula (37) is false. Then there exists a constant $\varepsilon > 0$ such that

$$\|g_k\| \geq \varepsilon, \quad \forall k \geq 0. \tag{38}$$

From the sufficient descent property of **KBH** (Theorem 2.2), we have

$$d_k^T g_k \leq -c\|g_k\|^2, \quad \forall k \geq 0. \tag{39}$$

Moreover, using the Lipschitz continuity of ∇f, we obtain

$$\|y_k\| = \|g_{k+1} - g_k\| \leq L\|s_k\| \leq LD, \tag{40}$$

where $D = \max\{\|x - y\| : x, y \in S\}$ is the diameter of S and $s_k = x_{k+1} - x_k$. The hybrid direction update gives

$$d_{k+1} = -g_{k+1} + \beta_k^{\mathrm{KBH}} d_k,$$

$$\|d_{k+1}\| \leq \|g_{k+1}\| + |\beta_k^{\mathrm{KBH}}|\,\|d_k\|. \tag{41}$$

To obtain a precise estimate of the growth of $\|d_k\|$, following the argument of Al-Baali [2] and Gilbert and Nocedal [41], we write $d_{k+1} + g_{k+1} = \beta_k^{\mathrm{KBH}} d_k$ (with $\beta_k^{\mathrm{KBH}} = 0$ at a restart). Squaring both sides, we get

$$\|d_{k+1}\|^2 = \left(\beta_k^{\mathrm{KBH}}\right)^2 \|d_k\|^2 - 2g_{k+1}^T d_{k+1} - \|g_{k+1}\|^2. \tag{42}$$

By (21), $-2g_{k+1}^T d_{k+1} \leq 2M\|g_{k+1}\|^2$, hence

$$\|d_{k+1}\|^2 \leq \left(\beta_k^{\mathrm{KBH}}\right)^2 \|d_k\|^2 + \bar{c}\,\|g_{k+1}\|^2, \qquad \bar{c} := 2M - 1 \geq 1. \tag{43}$$

Dividing (43) by $\|g_{k+1}\|^4$ and using $\left(\beta_k^{\mathrm{KBH}}\right)^2 \leq \left(\beta_k^{FR}\right)^2 = \|g_{k+1}\|^4/\|g_k\|^4$ from (18), we obtain

$$\frac{\|d_{k+1}\|^2}{\|g_{k+1}\|^4} \leq \frac{\|d_k\|^2}{\|g_k\|^4} + \frac{\bar{c}}{\|g_{k+1}\|^2}. \tag{44}$$

Applying (44) recursively, and using $\|d_0\|^2/\|g_0\|^4 = 1/\|g_0\|^2$, (38) and $\|g_k\| \leq \Gamma$, we get

$$\|d_k\|^2 \leq \|g_k\|^4\, \bar{c} \sum_{j=0}^{k} \frac{1}{\|g_j\|^2} \leq \frac{\Gamma^4\, \bar{c}}{\varepsilon^2}\,(k+1), \qquad \forall k \geq 0. \tag{45}$$

Which implies that

$$\sum_{k\geq 0} \frac{1}{\|d_k\|^2} \geq \frac{\varepsilon^2}{\Gamma^4\, \bar{c}} \sum_{k\geq 0} \frac{1}{k+1} = +\infty.$$

On the other hand, from equations (36), (38) and (39), we obtain

$$c^2\varepsilon^4 \sum_{k\geq 0} \frac{1}{\|d_k\|^2} \leq \sum_{k\geq 0} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < +\infty.$$

This contradicts the previous relation. Therefore, equation (38) cannot hold, and the claim in equation (37) is established. □

4. Numerical Experiments

This section reports a comprehensive set of numerical experiments conducted to assess the efficiency, robustness, and practical performance of the proposed KBH (CG) method. The test problems are selected from the CUTE library [15] as well as other widely adopted benchmark collections for unconstrained optimization [4, 26]. These problems span a wide range of dimensions and structural properties, ensuring a fair and representative evaluation.

All algorithms were implemented in MATLAB R2017 and executed on a personal computer running Windows 11 with the following specifications: Intel(R) Core(TM) 5 210H processor at 2.20 GHz, 16 GB of RAM, and a 64-bit operating system. To guarantee a consistent comparison, all competing methods were implemented using identical coding practices and stopping criteria.

The proposed KBH method is compared with the classical CD, PRP and FR conjugate gradient algorithms, which are widely regarded as standard benchmarks in unconstrained optimization. It is worth noting that the primary focus of this paper is to establish the theoretical framework and foundational convergence properties of the KBH scheme. While several advanced hybrid methods (such as those by Andrei, Djordjevic, Hallal et al., and Hamdi and Ibrahim [5, 11, 19, 22]) exist in the literature, comparing against standard classical benchmarks (CD, PRP, and FR) provides a clear, transparent, and direct baseline to evaluate the specific impact of the proposed convex combination parameters without the confounding effects of disparate algorithmic implementations or distinct line search variants.

For all methods, the strong Wolfe line search conditions were employed with parameters $\delta = 0.01$ and $\sigma = 0.1$, which are commonly adopted in the literature to ensure global convergence.

The iterative process was terminated when at least one of the following stopping criteria was satisfied:

(i) $\|g_k\| < 10^{-6}$,

(ii) the number of iterations exceeded 2000,

(iii) the CPU time exceeded 500 seconds.

The detailed numerical outcomes are reported in Table 2, where the symbol `NaN` indicates that the corresponding algorithm failed to converge for a given test problem. A summarized comparison of the overall performance is provided in Tables 3, 4 and 5. The evaluation metrics include the number of iterations (NOI), the number of function evaluations (NOF), and the CPU time measured in seconds (TCPU).

To provide a quantitative and visual comparison among the competing solvers, we employ the performance profile methodology introduced by Dolan and Morè [12]. Let S denote the set of solvers and P the set of test problems, with $n_s = |S|$ and $n_p = |P|$. For each problem $p \in P$ and solver $s \in S$, let $\tau_{p,s}$ represent the computational cost associated with solving problem p using solver s, where $\tau_{p,s}$ may correspond to NOI, NOF, or TCPU.

The performance ratio is defined as

$$r_{p,s} = \frac{\tau_{p,s}}{\min\{\tau_{p,i} : 1 \le i \le n_s\}}.$$

Let r_M be a sufficiently large constant such that $r_{p,s} \le r_M$ for all $p \in P$ and $s \in S$, where $r_{p,s} = r_M$ indicates that solver s failed to solve problem p. The performance profile of solver s is then given by

$$P_s(t) = \frac{\mathrm{card}\{p \in P : r_{p,s} \le t\}}{n_p}, \qquad t \ge 1,$$

where $\mathrm{card}(\cdot)$ denotes the cardinality of a set.

The function $P_s(t)$ is non-decreasing, piecewise constant, and right-continuous. The value $P_s(t)$ represents the probability that solver s performs within a factor t of the best solver for a randomly selected problem. In particular, $P_s(1)$ corresponds to the probability that solver s achieves the best performance among all competitors.

Figures 1a–1c illustrate the performance profiles of the KBH method compared with the CD, PRP and FR methods in terms of NOI, NOF, and TCPU, respectively. The results indicate that the proposed KBH method is competitive with the classical CG methods on a part of the considered benchmark problems.

Table 2. Numerical performance comparison of the proposed KBH conjugate gradient method with the CD, PRP and FR methods on a set of unconstrained optimization test problems.

Test Function	*N*	*CD*			*KBH*			*PRP*			*FR*		
		NOI	*NOF*	*TCPU*	*NOI*	*NOF*	*TCPU*	*NOI*	*NOF*	*TCPU*	*NOI*	*NOF*	*TCPU*
'cosine'	***500***	25	83	0.009673	53	319	0.043	42	123	0.010	20	74	0.008
'cosine'	***1000***	31	116	0.00675	53	346	0.031	NaN	NaN	NaN	17	77	0.008
'cosine'	***5000***	23	90	0.015363	95	571	0.150	NaN	NaN	NaN	31	96	0.028
'cosine'	***10000***	26	102	0.034944	NaN	NaN	NaN	NaN	NaN	NaN	20	85	0.038
'dixmaana'	***1500***	20	98	0.067248	41	249	0.143	23	85	0.041	14	77	0.038
'dixmaana'	***3000***	36	161	0.109516	27	153	0.161	19	82	0.077	14	67	0.069
'dixmaana'	***15000***	22	114	0.351073	36	196	0.853	28	94	0.387	19	84	0.358
'dixmaana'	***30000***	39	178	1.178964	29	185	1.750	26	94	0.880	17	83	0.788
'dixmaanb'	***1500***	24	101	0.055414	28	164	0.111	32	96	0.048	20	83	0.043
'dixmaanb'	***3000***	39	152	0.110058	27	149	0.154	18	82	0.090	20	82	0.085
'dixmaanb'	***15000***	27	115	0.355311	41	246	1.041	18	84	0.354	17	84	0.365
'dixmaanb'	***30000***	40	195	1.296176	41	224	2.099	25	94	0.881	16	91	0.863
'dixmaanc'	***1500***	22	90	0.048516	21	130	0.091	16	79	0.040	18	79	0.040
'dixmaanc'	***3000***	27	135	0.093477	35	187	0.185	19	81	0.077	21	85	0.080
'dixmaanc'	***15000***	51	177	0.531199	37	221	0.936	50	175	0.737	19	83	0.344
'dixmaand'	***30000***	27	129	0.903357	45	290	2.779	84	209	1.991	25	106	1.012
'dixmaand'	***1500***	29	121	0.044027	39	232	0.116	34	105	0.054	16	81	0.043
'dixmaand'	***1500***	29	121	0.040536	39	232	0.122	34	105	0.053	16	81	0.040
'dixmaand'	***3000***	38	148	0.103961	29	155	0.156	79	221	0.235	20	86	0.084
'dqrtic'	***500***	801	1137	0.091367	42	196	0.024	37	139	0.015	22	101	0.011
'dqrtic'	***1000***	936	1279	0.113304	39	185	0.037	38	179	0.036	32	118	0.025
'dqrtic'	***5000***	701	997	0.233283	52	218	0.218	65	264	0.258	32	136	0.129
'dqrtic'	***10000***	823	1148	0.485805	78	333	0.660	105	371	0.727	56	183	0.362
'edensch'	***500***	38	112	0.050438	69	438	0.050	146	1010	0.104	47	130	0.011
'edensch'	***1000***	46	128	0.063391	73	470	0.089	NaN	NaN	NaN	51	128	0.018
'edensch'	***5000***	54	178	0.240528	100	778	0.722	348	2451	2.219	84	414	0.097
'edensch'	***10000***	50	169	0.502357	116	847	1.538	NaN	NaN	NaN	59	183	0.148
'fletchcr'	***1000***	162	252	0.032028	189	1673	0.038	NaN	NaN	NaN	94	178	0.005
'fletchcr'	***5000***	211	349	0.088364	110	854	0.047	NaN	NaN	NaN	127	222	0.015
'fletchcr'	***10000***	173	314	0.137817	1027	9940	1.852	NaN	NaN	NaN	139	222	0.033
'fletchcr'	***12000***	188	345	0.170844	213	1864	0.231	NaN	NaN	NaN	130	248	0.038
'himmelbg'	***500***	2	9	0.012901	2	9	0.005	2	9	0.000	2	9	0.000
'himmelbg'	***1000***	2	13	0.001838	2	13	0.001	2	13	0.001	2	13	0.001

'himmelbg'	***5000***	2	13	0.005815	2	13	0.003	2	13	0.002	2	13	0.002
'himmelbg'	***10000***	2	17	0.011614	2	17	0.004	2	17	0.004	2	17	0.006
'penalty1'	***500***	24	131	0.130442	19	128	0.045	22	90	0.024	19	89	0.024
'penalty1'	***1000***	38	251	0.601778	58	408	0.322	NaN	NaN	NaN	31	214	0.168
'penalty1'	***4000***	53	354	14.64699	NaN	NaN	NaN	NaN	NaN	NaN	48	321	3.630
'penalty1'	***10000***	19	113	33.37649	57	381	36.242	20	95	8.983	19	94	8.912
'quartc'	***500***	596	3141	0.738118	42	196	0.024	37	139	0.015	22	101	0.011
'quartc'	***1000***	105	578	0.230977	39	185	0.037	38	179	0.036	32	118	0.024
'quartc'	***5000***	689	3645	8.573192	52	218	0.210	65	264	0.248	32	136	0.132
'quartc'	***10000***	436	2216	10.42115	78	333	0.631	105	371	0.698	56	183	0.345
'bdexp'	***500***	2	7	0.023477	2	7	0.008	2	7	0.000	2	7	0.000
'bdexp'	***1000***	2	11	0.003295	NaN	NaN	NaN	2	11	0.001	2	11	0.000
'bdexp'	***5000***	2	11	0.016135	2	11	0.003	2	11	0.002	2	11	0.002
'bdexp'	***10000***	2	17	0.016425	NaN	NaN	NaN	2	17	0.007	2	17	0.008
'exdenschnf'	***500***	22	92	0.026249	42	261	0.017	59	152	0.006	23	90	0.003
'exdenschnf'	***1000***	28	100	0.01133	36	204	0.004	36	114	0.002	26	95	0.002
'exdenschnf'	***5000***	24	100	0.021291	39	243	0.011	60	174	0.008	27	100	0.005
'exdenschnf'	***10000***	26	102	0.039058	47	226	0.024	62	174	0.018	29	105	0.012
'exdenschnb'	***500***	17	72	0.016681	27	151	0.006	23	91	0.002	17	75	0.001
'exdenschnb'	***1000***	28	98	0.008641	42	239	0.005	44	116	0.002	19	83	0.002
'exdenschnb'	***5000***	16	78	0.015086	25	153	0.006	84	227	0.009	22	86	0.004
'exdenschnb'	***10000***	18	85	0.022691	38	215	0.020	29	95	0.009	14	74	0.006
'genquartic'	***500***	25	102	0.020157	42	267	0.012	153	374	0.008	23	83	0.002
'genquartic'	***1000***	25	101	0.009979	37	212	0.004	206	443	0.008	21	83	0.002
'genquartic'	***5000***	18	81	0.018	42	198	0.010	21	84	0.004	14	74	0.005
'genquartic'	***10000***	32	97	0.040027	44	259	0.031	184	426	0.050	16	74	0.009
'sine'	***500***	228	375	0.046975	NaN	NaN	NaN	NaN	NaN	NaN	175	328	0.010
'sine'	***1000***	20	86	0.010988	151	601	0.028	158	433	0.018	22	80	0.004
'sine'	***3000***	353	551	0.215754	NaN	NaN	NaN	NaN	NaN	NaN	81	152	0.017
'sine'	***4000***	26	102	0.051517	NaN	NaN	NaN	156	579	0.070	33	114	0.015
'nonscomp'	***1000***	75	164	0.012845	59	260	0.006	NaN	NaN	NaN	69	153	0.004
'nonscomp'	***5000***	65	143	0.025133	96	476	0.025	NaN	NaN	NaN	77	150	0.010
'raydan2'	***500***	12	79	0.015819	42	273	0.009	10	74	0.001	13	66	0.001
'raydan2'	***1000***	16	97	0.010426	29	203	0.005	31	85	0.002	18	83	0.002
'raydan2'	***5000***	14	90	0.023654	17	112	0.008	26	90	0.006	12	81	0.006
'raydan2'	***10000***	36	183	0.082624	45	291	0.049	31	90	0.014	13	90	0.014

'diagonal1'	***4***	19	63	0.011428	32	152	0.005	34	92	0.001	19	62	0.001
'diagonal1'	***10***	29	79	0.005895	41	215	0.003	81	198	0.003	42	95	0.002
'diagonal1'	***50***	158	331	0.021962	129	775	0.010	NaN	NaN	NaN	143	256	0.005
'diagonal1'	***100***	235	530	0.036083	186	1224	0.017	NaN	NaN	NaN	230	659	0.011
'diagonal3'	***4***	23	71	0.010666	35	163	0.005	52	129	0.002	27	69	0.001
'diagonal3'	***10***	43	90	0.005466	58	313	0.004	185	400	0.005	59	105	0.002
'diagonal3'	***50***	250	378	0.030508	98	447	0.006	NaN	NaN	NaN	167	280	0.006
'diagonal3'	***100***	389	729	0.051	201	1375	0.021	NaN	NaN	NaN	377	584	0.013
'ie'	***10***	22	78	0.010711	28	184	0.007	12	51	0.002	11	48	0.002
'ie'	***100***	31	99	0.361055	35	199	0.375	33	87	0.167	14	57	0.108
'ie'	***500***	21	76	7.486317	41	242	11.626	16	55	2.633	19	80	3.842
'lin'	***10***	12	70	0.03195	15	97	0.014	70	117	0.006	12	71	0.004
'lin'	***100***	15	84	0.096382	15	102	0.055	13	63	0.039	13	63	0.028
'lin'	***500***	18	110	0.724974	15	104	0.330	17	84	0.289	18	84	0.251
'lin'	***1000***	19	80	39.8182	19	123	21.755	19	80	13.738	19	80	14.135
'trid'	***500***	67	127	0.153878	123	498	0.303	479	1029	0.554	59	118	0.061
'trid'	***1000***	65	126	0.600251	71	362	0.787	400	861	1.859	73	134	0.283

A closer inspection of Table 2 reveals notable numerical patterns for specific test functions, notably 'himmelbg' and 'quartc'. For the extended Himmelblau function ('himmelbg'), all evaluated algorithms consistently record exactly two iterations (NOI $= 2$) across dimensions $N \in \{500, 1000, 5000, 10000\}$. This behavior is attributed to the block-separable structure of the function and the standard initial point configuration, which positions the initial iterate within the immediate quadratic basin of the global minimizer, allowing the gradient stopping criterion to be satisfied rapidly regardless of problem dimension. Conversely, for the quartic function ('quartc'), the classical CD method exhibits significantly higher iteration counts and sensitivity to dimensionality. This is due to steep gradient variations and temporary loss of conjugacy in standard formulations, whereas the proposed KBH method successfully mitigates these issues through curvature-aware hybridization parameters.

To provide a balanced and transparent evaluation, it is important to address the operational limitations and non-convergent cases (denoted as `NaN` in Table 2). These occurrences are primarily concentrated in highly non-convex or severely ill-conditioned test instances (such as high-dimensional extensions of the "cosine" and "penalty1" functions), where rapid gradient fluctuations cause line-search backtracking failures or violations of strong Wolfe conditions. While the proposed KBH method solves most of the test problems, these instances highlight the inherent challenges of unconstrained optimization in complex landscapes. Furthermore, rather than claiming absolute global supremacy, our findings establish that the KBH framework provides a competitive and theoretically sound alternative to standard classical benchmarks, with parameter tuning remaining flexible within empirically validated bounds.

Table 3. Relative computational performance of the proposed KBH method compared with the classical conjugate descent (CD) method. All values are reported as percentages, with CD normalized to 100%.

Method	**NOI (%)**	**NOF (%)**	**TCPU (%)**
CD (reference)	100.000	100.000	100.000
KBH	38.538	40.711	29.530

Table 4. Relative computational performance of the proposed KBH method compared with the Polak–Ribière–Polyak (PRP) conjugate gradient method. All values are reported as percentages, with PRP normalized to 100%.

Method	**NOI (%)**	**NOF (%)**	**TCPU (%)**
PRP (reference)	100.000	100.000	100.000
KBH	60.118	27.685	38.536

Table 5. Relative computational performance of the proposed KBH method compared with the Fletcher–Reeves (FR) conjugate gradient method. All values are reported as percentages, with FR normalized to 100%.

Method	**NOI (%)**	**NOF (%)**	**TCPU (%)**
FR (reference)	100.000	100.000	100.000
KBH	42.022	43.923	79.058

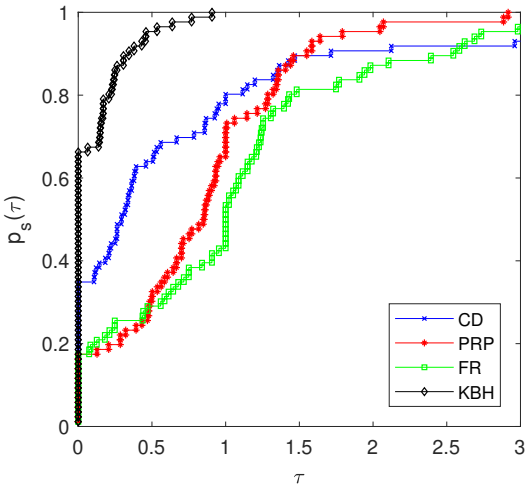
As reported in Tables 3–5, the proposed KBH method requires, in total, fewer iterations, function evaluations and CPU time than the classical CG algorithms on the problems summarized in these tables.

In comparison with the classical CD method (Table 3), the KBH algorithm reduces the computational cost. Specifically, the NOI is reduced by approximately 61.5%, while the NOF and TCPU are reduced by about 59.3% and 70.5%, respectively. These results suggest that, on these problems, the KBH search directions are more effective than those of the CD method.

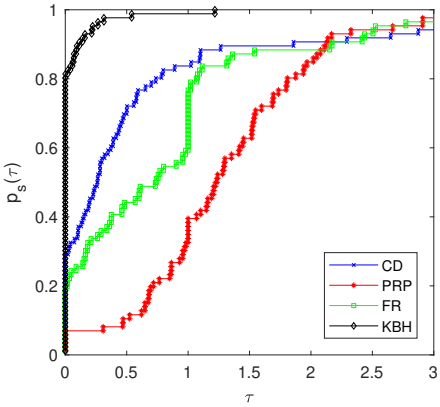
When compared with the PRP method, as shown in Table 4, the proposed KBH algorithm also reduces the computational cost. The KBH method reduces the NOI by approximately 39.9%, the NOF by 72.3%, and the TCPU by 61.5%. The decrease in NOF suggests that the proposed method requires fewer function evaluations on these problems.

A similar trend is observed in the comparison with the FR method (Table 5). In this case, the KBH algorithm reduces the NOI and NOF by approximately 58.0% and 56.1%, respectively. Although the reduction in CPU time is more moderate (about 20.9%), the performance remains competitive. This relatively smaller improvement in TCPU can be attributed to implementation-dependent overheads and system-related factors, which commonly influence wall-clock time in numerical experiments.

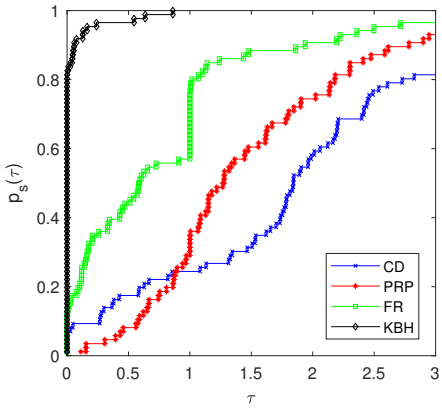
Overall, these results indicate that the proposed KBH method is competitive with the CD, PRP and FR methods on a part of the test problems. However, Table 2 also shows that KBH fails on some problems solved by CD and FR, and that FR remains more efficient on many instances. Therefore, these results do not establish a general superiority of KBH, and a comparison with recent hybrid methods is left for future work.



(a) Performance profiles based on NOI.



(b) Performance profiles based on the NOF.



(c) Performance profiles based on TCPU.

Figure 1. Performance profiles of the KBH method compared to the CD, PRP and FR methods.

Although the primary focus of this paper is to establish a rigorous theoretical foundation and comprehensive benchmark validation for the proposed KBH method, we fully recognize the importance of demonstrating its practical utility. Due to the theoretical and algorithmic scope of the current manuscript, real-world implementations are designated as part of our ongoing and immediate future work. Specifically, we intend to deploy the KBH method in solving large-scale machine learning training tasks, robotic manipulator trajectory optimization, and signal/image restoration and reconstruction problems.

5. Discussion on Limitations and Practical Considerations

Despite the theoretical guarantees established for the proposed KBH method, several potential limitations and practical considerations must be addressed for comprehensive applicability:

1. **Behavior Near Zero Denominator ($d_k^T y_k \to 0$):** Similar to most conjugate gradient algorithms, the scaling or conjugacy denominator $d_k^T y_k$ may occasionally approach zero or become negative in severely non-convex regions. If left unhandled, this can lead to ill-conditioned search directions or numerical instability. To mitigate this, a safeguard mechanism—such as Powell's restarting condition or an adaptive lower bound (e.g., $d_k^T y_k \geq \epsilon \|g_k\|^2$)—is recommended in practical implementations.

2. **Effect of Rounding Errors:** In high-dimensional large-scale optimization problems, finite-precision floating-point arithmetic introduces cumulative rounding errors. These errors can gradually degrade the orthogonality between successive gradients and exact conjugacy properties. Employing periodic restarts or high-precision data types can alleviate this accumulation.

3. **Guidelines for Parameter Selection in Practice:** The performance of the KBH method depends on the appropriate selection of the convexity parameters. Empirical tuning suggests that restricting these parameters within a closed interval (such as $[\epsilon, 1-\epsilon]$ for a small $\epsilon > 0$) ensures that the sufficient descent condition is consistently satisfied without overly constraining the flexibility of the search direction.

6. Conclusion

We have introduced the KBH algorithm, a three-term hybrid conjugate gradient method for large-scale smooth unconstrained optimization. The search direction is built as an adaptively weighted convex combination of the CD, PRP, and FR directions, with the weighting coefficient δ_k determined at each step by enforcing the classical conjugacy condition. This construction allows the method to simultaneously exploit the descent guarantee of CD, the practical efficiency of PRP, and the global stability of FR.

Two key theoretical properties were established, under the strong Wolfe line search with $\sigma_2 < \frac{5}{12}$, using the Powell restart criterion and the Fletcher–Reeves envelope (18): the sufficient descent condition holds for every generated direction, and the algorithm converges globally in the sense that $\liminf_{k\to\infty} \|g_k\| = 0$. The proofs rely only on standard smoothness assumptions and do not require convexity of f or a positive lower bound on the step size.

Numerical experiments on a collection of large-scale benchmark problems indicate that KBH is competitive with the CD, PRP and FR methods in terms of iteration count, function evaluations and CPU time on a part of the test problems. These results are limited to the classical CG methods and to the test set considered; a comparison with recent hybrid CG methods and an application to real problems are left for future work.

Several directions for future research present themselves naturally. These include extending the adaptive weighting strategy to handle stochastic or noisy gradients (as encountered in machine learning), incorporating second-order curvature estimates to further accelerate convergence, and adapting the framework to constrained or non-smooth optimization settings. In addition, we plan to apply the KBH method to real-world problems, such as the training of artificial neural networks, image restoration and reconstruction, parameter estimation and robotic manipulator trajectory optimization, and to compare it with recent hybrid CG methods, together with an ablation and sensitivity study of its components and parameters.

Acknowledgements

The authors thank the staff and colleagues of the Laboratory Informatics and Mathematics (LIM), University of Mohamed Cherif Messaadia Souk Ahras, for their support during this research. A.L.I. gratefully acknowledges the hospitality of the Department of Mathematics, University of Zakho.

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.* 6 (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, *New Hybrid Conjugate Gradient Algorithms for Unconstrained Optimization*, in *Encyclopedia of Optimization*, 2009, pp. 2560–2571.
6. N. Andrei, *Accelerated Hybrid Conjugate Gradient Algorithm with Modified Secant Condition for Unconstrained Optimization*, *Numer. Algorithms* 54 (2010) 23–46.
7. 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.
8. I. Bongartz, A. R. Conn, N. Gould and P. L. Toint, *CUTE: Constrained and Unconstrained Testing Environments*, *ACM Trans. Math. Softw.* 21 (1995) 123–160.
9. Y. H. Dai and Y. Yuan, *A Nonlinear Conjugate Gradient Method with a Strong Global Convergence Property*, *SIAM J. Optim.* 10 (1999) 177–182.
10. S. S. Djordjević, *New Hybrid Conjugate Gradient Method as a Convex Combination of FR and PRP Methods*, *Filomat* 30 (2016) 3083–3100.
11. S. S. Djordjević, *New Hybrid Conjugate Gradient Method as a Convex Combination of LS and FR Methods*, *Acta Math. Sci.* 39 (2019) 214–228.
12. E. D. Dolan and J. J. Moré, *Benchmarking Optimization Software with Performance Profiles*, *Math. Program.* 91 (2002) 201–213.
13. R. Fletcher, *Practical Methods of Optimization: Unconstrained Optimization*, Vol. 1, John Wiley & Sons, New York, 1980.
14. R. Fletcher and C. Reeves, *Function Minimization by Conjugate Gradients*, *Comput. J.* 7 (1964) 149–154.
15. N. I. M. Gould, D. Orban and P. L. Toint, *CUTEr and Sifdec: A Constrained and Unconstrained Testing Environment, Revisited*, *ACM Trans. Math. Softw.* 29 (2003) 373–394.
16. 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.
17. W. W. Hager and H. Zhang, *A Survey of Nonlinear Conjugate Gradient Methods*, *Pac. J. Optim.* 2 (2006) 35–58.
18. A. Hallal, M. Belloufi and B. Sellami, *An Efficient New Hybrid CG-Method as Convex Combination of DY, CD, and HS Algorithms*, *J. Comput. Appl. Math.* 402 (2022) 113949.
19. A. Hallal, M. Belloufi, and B. Sellami, *A New Hybrid Conjugate Gradient Method as Convex Combination*, *Mathematical Foundations of Computing*, vol. 7, no. 4, pp. 522–530, 2024.
20. S. B. Hanachi, B. Sellami, and M. Belloufi, *New Iterative Conjugate Gradient Method for Nonlinear Unconstrained Optimization*, *RAIRO–Operations Research*, vol. 56, no. 3, pp. 2315–2327, 2022.
21. S. B. Hanachi, B. Sellami, and M. Belloufi, *A New Family of Hybrid Conjugate Gradient Method for Unconstrained Optimization and Its Application to Regression Analysis*, *RAIRO–Operations Research*, vol. 58, pp. 613–627, 2024.
22. A. Hamdi and A. L. Ibrahim, *A New Hybrid Iterative Conjugate Gradient Method for Nonlinear Unconstrained Optimization*, *Indian Journal of Pure and Applied Mathematics*, 2026. https://doi.org/10.1007/s13226-026-00937-3.
23. M. R. Hestenes and E. L. Stiefel, *Methods of Conjugate Gradients for Solving Linear Systems*, *J. Res. Nat. Bur. Standards* 49 (1952) 409–436.
24. J. K. Liu and S. J. Li, *New Hybrid Conjugate Gradient Method for Unconstrained Optimization*, *Appl. Math. Comput.* 245 (2014) 36–43.
25. D. Liu and C. Storey, *Efficient Generalized Conjugate Gradient Algorithms. Part 1: Theory*, *J. Optim. Theory Appl.* 69 (1991) 129–137.
26. J. J. Moré, B. S. Garbow and K. E. Hillstrom, *Testing Unconstrained Optimization Software*, *ACM Trans. Math. Softw.* 7 (1981) 17–41.
27. P. K.-H. Phua, W. Fan and Y. Zeng, *Parallel Algorithms for Large-Scale Nonlinear Optimization*, *Int. Trans. Oper. Res.* 5 (1998) 67–77.
28. E. Polak and G. Ribière, *Note sur la Convergence de Méthodes de Directions Conjuguées*, *Rev. Française Inform. Rech. Opér.* 3 (1969) 35–43.
29. B. T. Polyak, *The Conjugate Gradient Method in Extremal Problems*, *USSR Comput. Math. Math. Phys.* 9 (1969) 94–112.
30. Z. Salleh and A. Alhawarat, *An Efficient Modification of the Hestenes–Stiefel Nonlinear Conjugate Gradient Method with Restart Property*, *J. Inequal. Appl.* 2016 (2016) 1–14.
31. D. Touati-Ahmed and C. Storey, *Efficient Hybrid Conjugate Gradient Techniques*, *J. Optim. Theory Appl.* 64 (1990) 379–397.
32. W. Wang and J. Gao, *An Efficient Three-Term Conjugate Gradient Algorithm with Restart Strategy*, *Sci. Rep.* 15 (2025) 1–14.
33. 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.
34. P. Wolfe, *Convergence Conditions for Ascent Methods*, *SIAM Rev.* 11 (1969) 226–235.
35. P. Wolfe, *Convergence Conditions for Ascent Methods. II: Some Corrections*, *SIAM Rev.* 13 (1971) 185–188.
36. H. Yan, L. Chen and B. Jiao, *HS–LS–CD Hybrid Conjugate Gradient Algorithm for Unconstrained Optimization*, in *Proc. Second Int. Workshop on Computer Science and Engineering*, IEEE, 2009, pp. 264–268.
37. G. Yuan, Z. Wei and Q. Zhao, *A Modified Polak–Ribière–Polyak Conjugate Gradient Algorithm for Large Scale Optimization Problems*, *IIE Trans.* 46 (2014) 397–413.

38. S. S. Mohammed Zaki, H. N. Jabbar, and S. S. Haider, *HZA Novel Conjugate Gradient Algorithm as a Convex Combination of Classical Conjugate Gradient Methods*, *Journal of Applied Mathematics*, 2025.
39. 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. Algorithms* 84 (2019) 603–608.
40. G. Zoutendijk, *Nonlinear Programming, Computational Methods*, in *Integer and Nonlinear Programming*, edited by J. Abadie, North-Holland, Amsterdam, 1970, pp. 37–86.
41. J. C. Gilbert and J. Nocedal, *Global Convergence Properties of Conjugate Gradient Methods for Optimization*, *SIAM J. Optim.* 2 (1992) 21–42.
42. M. J. D. Powell, *Restart Procedures for the Conjugate Gradient Method*, *Math. Program.* 12 (1977) 241–254.
43. O. J. Adeleke, A. E. Ezugwu and I. A. Osinuga, *A New Family of Hybrid Conjugate Gradient Methods for Unconstrained Optimization*, *Stat. Optim. Inf. Comput.* 9 (2021) 399–417. https://doi.org/10.19139/soic-2310-5070-480
44. R. Mellal and N. Sellami, *A Novel Hybrid Conjugate Gradient Algorithm for Solving Unconstrained Optimization Problems*, *Stat. Optim. Inf. Comput.* 15 (2025) 380–395. https://doi.org/10.19139/soic-2310-5070-2807
45. A. E. Mehamdia, I. A. R. Moghrabi, B. A. Hassan and N. Guler, *An Improved Hybrid Nonlinear Conjugate Gradient Method and Application to Image Restoration Problems*, *Stat. Optim. Inf. Comput.* 16 (2026) 1–14. https://doi.org/10.19139/soic-2310-5070-3366